Jianhua Ma
Hai Jin
Laurence T. Yang
Jeffrey J.-P. Tsai (Eds.)

LNCS 4159

Ubiquitous Intelligence and Computing

Third International Conference, UIC 2006
Wuhan, China, September 2006
Proceedings

Springer

Lecture Notes in Computer Science 4159

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Jianhua Ma Hai Jin
Laurence T. Yang Jeffrey J.-P. Tsai (Eds.)

Ubiquitous Intelligence and Computing

Third International Conference, UIC 2006
Wuhan, China, September 3-6, 2006
Proceedings

Springer

Volume Editors

Jianhua Ma
Hosei University, Faculty of Computer and Information Sciences
3-7-2, Kajino-cho, Koganei-shi, Tokyo 184-8584, Japan
E-mail: jianhua@k.hosei.ac.jp

Hai Jin
Huazhong University of Science and Technology
School of Computer Science and Technology
Wuhan, 430074, China
E-mail: hjin@hust.edu.cn

Laurence T. Yang
St. Francis Xavier University, Department of Computer Science
Antigonish, NS, B2G 2W5, Canada
E-mail: lyang@stfx.ca

Jeffrey J.-P. Tsai
University of Illinois, Department of Computer Science
851 S. Morgan St., Chicago, IL 60607, USA
E-mail: tsai@cs.uic.edu

Library of Congress Control Number: 2006931260

CR Subject Classification (1998): H.4, C.2, D.4.6, H.5, I.2, K.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-38091-4 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-38091-7 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11833529 06/3142 5 4 3 2 1 0

Preface

Welcome to the proceedings of the Third International Conference on Ubiquitous Intelligence and Computing (UIC 2006), Building Smart Worlds on Real and Cyber Spaces, which was held in Wuhan and Three Gorges, China, September 3-6, 2006.

Following ubiquitous computers, networks, information, services, etc., is a road towards a smart world (SW) created on both real and cyber spaces. A SW is mainly characterized by ubiquitous intelligence (UI) or computational intelligence pervasive in the physical world, filled with ubiquitous intelligent or smart things that are capable of computing, communicating, and behaving smartly with some intelligence. One of the profound implications of such ubiquitous smart things is that various kinds and levels of intelligence will exist ubiquitously in everyday objects, environments, systems and even ourselves, and possibly be extended from man-made to natural things. "Ubicomp" or "percomp" can be regarded as the computing of all these intelligent/smart things/u-things, that are essential elements and components of the SW.

A smart thing can be endowed with different levels of intelligence, and may be context-aware, active, interactive, reactive, proactive, assistive, adaptive, automated, sentient, perceptual, cognitive, autonomic and/or thinking. Intelligent/smart things is an emerging research field covering many disciplines. A series of grand challenges exist to move from the ubiquitous world with universal services of any means/place/time to the SW of trustworthy services with the right means/place/time. UIC 2006 was a successor of the Second International Symposium on Ubiquitous Intelligence and Smart Worlds (UISW 2005) held in Japan, December, 2005, which succeeded the First International Workshop on Ubiquitous Smart Worlds (USW 2005) held in Taiwan, March, 2005.

The UIC 2006 conference provided a forum for engineers and scientists in academia, industry, and government to exchange ideas and experiences in developing intelligent/smart objects, environments, and systems as well as to discuss various personal/social/physical issues faced by UI and SWs.

There was a very large number of paper submissions (382), representing 25 countries and regions, not only from Asia and the Pacific, but also from Europe, and North and South America. All submissions were reviewed by at least three Program or Technical Committee members or external reviewers. It was extremely difficult to select the presentations for the conference because there were so many excellent and interesting submissions. In order to allocate as many papers as possible and keep the high quality of the conference, we finally decided to accept 117 papers for presentations, reflecting a 30% acceptance rate. We believe that all of these papers and topics not only provided novel ideas, new results, work in progress and state-of-the-art techniques in this field, but also

stimulated the future research activities in the area of ubiquitous intelligence and computing.

The exciting program for this conference was the result of the hard and excellent work of many others, such as Program and Technical Committee members, external reviewers and Publication Chairs under a very tight schedule. We are also grateful to the members of the Local Organizing Committee for supporting us in handling so many organizational tasks, and to the keynote speakers for accepting to come to the conference with enthusiasm. Last but not least, we hope you enjoy the conference program, and the beautiful attractions of Three Gorges, China.

August 2006

Jianhua Ma, Hai Jin, Laurence T. Yang
Jeffrey J.P. Tsai, Victor Callaghan
Zhaohui Wu, Albert Zomaya
UIC 2006 Steering, General and Program Chairs

Organization

UIC 2006 was organized and sponsored by Huazhong University of Science & Technology (HUST), co-sponsored by the National Science Foundation of China, 863, ChinaGrid, and International Federation for Information Processing (IFIP). It was held in cooperation with the IEEE Computer Society and *Lecture Notes in Computer Science* (LNCS) of Springer.

Executive Committee

Honorary Chairs:	Norio Shiratori, Tohoku University, Japan
	Yaoxue Zhang, Tsinghua University, China
General Chairs:	Jeffrey J.P. Tsai, University of Illinois at Chicago, USA
	Zhaohui Wu, Zhejiang University, China
	Albert Zomaya, University of Sydney, Australia
Program Chairs:	Laurence T. Yang, St. Francis Xavier University, Canada
	Hai Jin, Huazhong University of Science & Technology, China
	Victor Callaghan, University of Essex, UK
International Advisory Committee:	Makoto Amamiya, Kyushu University, Japan
	Marios C. Angelides, Brunel University, UK
	Leonard Barolli, Fukuoka Institute of Technology, Japan
	Jingde Cheng, Saitama University, Japan
	Sumi Helal, University of Florida, USA
	Ali R. Hurson, Pennsylvania State University, USA
	Haruhisa Ichikawa, NTT Network Innovation Lab, Japan
	Janusz Kacprzyk, Polish Academy of Sciences, Poland
	Moon Hae Kim, Konkuk University, Korea
	Gabriele Kotsis, Johannes Kepler University of Linz, Austria
	Beniamino Di Martino, Second University of Naples, Italy
	Ivan Stojmenovic, Ottawa University, Canada
	Makoto Takizawa, Tokyo Denki University, Japan
	Jhing-Fa Wang, National Cheng Kung University, Taiwan

	Stephen S. Yau, Arizona State University, USA
	Xingshe Zhou, Northwestern Polytechnical University, China
Steering Chairs:	Jianhua Ma, Hosei University, Japan
	Laurence T. Yang, St. Francis Xavier University, Canada
Publicity Chairs:	Mieso Denko, University of Guelph, Canada
	Hani A.K. Hagras, University of Essex, UK
	Qun Jin, Waseda University, Japan
International Liaison Chairs:	Vipin Chaudhary, Wayne State University, USA
	Ismail K. Ibrahim, Johannes Kepler University Linz, Austria
	Jadwiga Indulska, University of Queensland, Australia
Publication Chairs:	Wenbin Jiang, Huazhong University of Science & Technology, China
	Deqing Zou, Huazhong University of Science & Technology, China
	Thomas Noel, University Louis Pasteur, France
	Jon(Jong-Hoon) Youn, University of Nebraska at Omaha, USA
Award Chairs:	Arjan Durresi, Louisiana State University, USA
	Antonio Puliafito, University of Messina, Italy
	Timothy K. Shih, Tamkang University, Taiwan
Panel Chair:	Jiannong Cao, Hong Kong Polytechnic University, China
Financial Chair:	Xia Xie, Huazhong University of Science & Technology, China
Web Chairs:	Wenbin Jiang, Huazhong University of Science & Technology, China
	Tony Li Xu, St. Francis Xavier University, Canada
Local Organizing Chair:	Xia Xie, Huazhong University of Science & Technology, China

Program Committee

Waleed Abdullah	University of Auckland, New Zealand
Borhanuddin Ali	Universiti Putra, Malaysia
Michael Amberg	University of Erlangen-Nuernberg, Germany
Giuseppe Anastasi	University of Pisa, Italy
Bernady O. Apduhan	Kyushu Sangyo University, Japan
Juan Carlos Augusto	University of Ulster at Jordanstown, UK
Rocco Aversa	Second University of Naples, Italy
Irfan Awan	University of Bradford, UK

Ruth Aylett	University of Heriott-Watt, UK
Stuart J. Barnes	University of East Anglia, UK
Christian Becker	University of Stuttgart, Germany
Azzedine Boukerche	University of Ottawa, Canada
Rafael Capurro	University of Applied Sciences, Germany
Valentina Casola	University "Federico II" of Naples, Italy
Chih-Yung Chang	Tamkang University, Taiwan
Han-Chieh Chao	National Dong Hwa University, Taiwan
Kuo-Ming Chao	Coventry University, UK
Barbara Chapman	University of Houston, USA
Xiaowu Chen	Beihang University, China
Yuh-Shyan Chen	National Chung Cheng University, Taiwan
Zixue Cheng	The University of Aizu, Japan
Jeannette Chin	University of Essex, UK
Paul Davidsson	Blekinge Institute of Technology, Sweden
Petre Dini	Cisco Systems, USA
Michael Ditze	University of Paderborn, Germany
Monica Divitini	Norwegian University of Science and Technology, Norway
Hakan Duman	University of Essex, UK
Tomoya Enokido	Rissho University, Japan
Thierry Ernst	Keio University, Japan
Alois Ferscha	University of Linz, Austria
Elgar Fleisch	University of St. Gallen, Switzerland
Michael Gardener	Chimera, UK
Frank Golatowski	University of Rostock, Germany
Antonio Mana Gomez	University of Malaga, Spain
Jinhua Guo	University of Michigan at Dearborn, USA
J. Felix Hampe	University of Koblenz-Landau, Germany
Sung-kook Han	Wonkwang University, Korea
Sunyoung Han	Konkuk University, Korea
Takahiro Hara	Osaka University, Japan
Guenter Haring	University of Vienna, Austria
Xubin He	Tennessee Technological University, USA
Karen Henricksen	University of Queensland, Australia
Jiman Hong	Kwangwoon University, Korea
Hui-Huang Hsu	Tamkang University, Taiwan
Chung-Ming Huang	National Cheng Kung University, Taiwan
Runhe Huang	Hosei University, Japan
Tsung-Chuan Huang	National Sun Yat-Sen University, Taiwan
Jason C. Hung	Northern Taiwan Institute of Science and Technology, Taiwan
Ren-Hung Hwang	National Chung Cheng University, Taiwan
Christophe Jelger	FOKUS, Germany
Brendan Jennings	The Waterford Institute, Ireland

Dongwon Jeong	Kunsan National University, Korea
Young-sik Jeong	Wonkwang University, Korea
Tao Jiang	Brunel University, UK
Yu (Cathy) Jiao	Oak Ridge National Lab., USA
Achilles Kameas	Hellenic Open University, Greece
Daeyoung Kim	Information and Communications University, Korea
Doohyun Kim	Konkuk University, Korea
Chung-Ta King	National Tsing Hua University, Taiwan
Tetsuo Kinoshita	Tohoku University, Japan
Dieter Kranzlmueller	University of Linz, Austria
Stan Kurkovsky	Connecticut State University, USA
Choonhwa Lee	Hanyang University, Korea
Wonjun Lee	Korea University, Korea
Jae Yeol Lee	Chonnam National University, Korea
Hong-Va Leong	Hong Kong Polytechnic University, China
Jiandong Li	Xidian University, China
Jiang (Leo) Li	Howard University, USA
Kuan-Ching Li	Providence University, Taiwan
Yinsheng Li	Fudan University, China
Weifa Liang	The Australian National University, Australia
Shih-Wei (Steve) Liao	INTEL, USA
Seng Loke	La Trobe University, Australia
Antonio Lopez	University of Oviedo, Spain
Paul Lukowicz	UMIT, Austria
Mary Lou Maher	University of Sydney, Australia
Pedro Jose Marron	University of Stuttgart, Germany
Ian Marshall	University of Kent, UK
Andreas Meissner	Fraunhofer IPSI, Germany
Geyong Min	University of Bradford, UK
Wolfgang Minker	University of Ulm, Germany
Vojislav B. Misic	University of Manitoba, Canada
Nicolas Montavont	ENST Bretagne, France
Francesco Moscato	Second University of Naples, Italy
Soraya Kouadri Mostefaoui	Oxford Brookes University, UK
Yi Mu	University of Wollongong, Australia
Max Muhlhauser	Darmstadt University of Technology, Germany
Maurice Mulvenna	University of Ulster, UK
Amiya Nayak	University of Ottawa, Canada
Wolfgang Nejdl	University of Hannover, Germany
Tom Pfeifer	Waterford Institute of Technology, Ireland
Marius Portmann	University of Queensland, Australia

Rosa Preziosi	University of Sannio, Italy
Aaron J. Quigley	University College Dublin, Ireland
Massimiliano Rak	Second University of Naples, Italy
Carlos Ramos	Polytechnic of Porto, Portugal
Matthias Rauterberg	Technical University of Eindhoven, The Netherlands
Angelica Reyes	Technical University of Catalonia, Spain
Kouichi Sakurai	Kyushu University, Japan
Albrecht Schmidt	University of Munich, Germany
Ali Shahrabi	Glasgow Caledonian University, UK
Elhadi Shakshuki	Acadia University, Canada
Yuanchun Shi	Tsinghua University, China
Behrooz Shirazi	Washington State University, USA
David Simplot-Ryl	University Lille 1, France
Carsten Sorensen	London School of Economics, UK
Alexei Sourin	Nanyang Technological University, Singapore
Bala (Srini) Srinivasan	Monash University, Australia
Willy Susilo	University of Wollongong, Australia
Evi Syukur	Monash University, Australia
David Taniar	Monash University, Australia
Tsutomu Terada	Osaka University, Japan
Anand Tripathi	University of Minnesota, USA
Yu-Chee Tseng	National Chiao-Tung University, Taiwan
Klaus Turowski	University of Augsburg, Germany
Salvatore Venticinque	Second University of Naples, Italy
Javier Garcia Villalba	Complutense University of Madrid, Spain
Umberto Villano	University of Sannio, Italy
Natalija Vlajic	York University, Canada
Agustinus Borgy Waluyo	Monash University, Australia
Cho-li Wang	Hong Kong University, Hong Kong
Guojun Wang	Central South University, China
Sheng-De Wang	National Taiwan University, Taiwan
Ying-Hong Wang	Tamkang University, Taiwan
Hongyi Wu	University of Louisiana at Lafayette, USA
Bin Xiao	Hong Kong Polytechnic University, China
Naixue Xiong	JAIST, Japan
Zhiyong Xu	Suffolk University, USA
Lu Yan	Turku Centre for Computer Science, Finland
Chu-Sing Yang	National Sun Yat-Sen University, Taiwan
Stephen Yang	National Central University, Taiwan
George Yee	National Research Council, Canada
Masao Yokota	Fukuoka Institute of Technology, Japan
Takaichi Yoshida	Kyushu Institute of Technology, Japan
Muhammed Younas	Oxford Brookes University, UK

Mohamed Younis	University of Maryland Baltimore County, USA
Ming Yu	State University of New York at Binghamton, USA
Zhiwen Yu	Northwestern Polytechnical University, China
Arkady Zaslavsky	Monash University, Australia
Daqing Zhang	Institute for Infocomm Research, Singapore
Jingyuan (Alex) Zhang	University of Alabama, USA
Qiangfu Zhao	The University of Aizu, Japan
Xiaobo Zhou	University of Colorado at Colorado Springs, USA
Yian Zhu	Northwestern Polytechnical University, China

Additional Reviewers

Gian-Franco Dalla Betta	Antoine Gallais	Stefano Marrone
Damiano Carra	Mark Halpern	Danilo Severina
Oliver Diessel	Mauro Iacono	Wei Wang

Table of Contents

Keynote Speech

Track 1: Smart Objects and Embedded Systems

Track 2: Smart Spaces/Environments/Platforms

Track 3: Ad Hoc and Intelligent Networks

Track 4: Sensor Networks

Track 5: Pervasive Communications and Mobile Systems

Track 6: Context-Aware Computing and Systems

Track 7: Security, Safety and Privacy

Track 8: Services, Models, Personal/Social Factors

Transparent Computing: A New Paradigm for Pervasive Computing

Yaoxue Zhang and Yuezhi Zhou

Department of Computer Science and Technology, Tsinghua University,
Beijing 100084, P.R. China
{cxzyx, zhouyz}@mail.tsinghua.edu.cn

Abstract. Due to the research and technological advances, ubiquitous or pervasive computing is emerging rapidly as an exciting new discipline to provide computing and communication services all the time and everywhere. While with many ongoing initiatives, it is too far to achieve the vision that Mark Weiser described. After a comprehensive analysis on traditional paradigms, we argue that, not users-friendly, i.e., users can not get services from computer easily, is one of the main reasons. In this paper, a new computing paradigm, i.e., Transparent Computing will be presented to solve this problem partially. Accordingly, we propose and develop a pilot system, which runs in a network environment and operates at the assembler instruction level. This system lets users demand heterogeneous OSes and applications upon them from centered simple servers, similar to choose different TV channels in daily life. We also present some primitive real and experimental results to show that it is a feasible and efficient solution for future computing infrastructure.

1 Introduction

In the last two decades, the technologies have made rapid improvements. For example, Intel CPU has evolved from X286 in 1982 to Pentium IV; the related operating systems such as Windows from DOS in 1981 to Windows 2000/XP. As we all know, computing paradigms will evolve with the rapid advances in hardware, software and networking technologies. The computing paradigms have shifted from centralized mainframe computing towards distributed personal desktop computing, and now to ubiquitous or pervasive computing [1], which aims at that users can only focus on their tasks without distraction in a way of using traditional appliances. Thus, we are moving away from a machine-centric view of computing to a more service-oriented perspective, i.e. users just simply want to accomplish their tasks by using computing services on their data, and don't care about the machine specifics and management details. We believe that the future computing infrastructure will provide ubiquitous access to computing services globally through distributed resources. In this context, the computing systems and environments must be hassle-free for end users and much easier to maintain and manage.

However, current computer systems are usually constructed to consist of a complete set of hardware and software platforms. The tight coupling of software and

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1 – 11, 2006.

hardware means that each individual machine must be installed with OSes and applications, constantly upgraded or patched. With the systems are becoming more complex, the maintenance and management tasks are becoming more and more unmanageable for common users, even professionals. We argue that the problems of traditional computing systems can be concluded as follows.

- Complexity: As stated above, due to the combination and tight coupling of hardware and software, the evolvement of each part will lead to a more complex structure, which is inclined to be unreliable. For example, there are more bugs in Windows operating systems.
- High TCO: The total cost of ownership (TCO), including hardware and software cost, especially regarding to maintenance and management cost that is increasing sharply. As estimated in [2], about 40 percent of the total cost of a software system is spent on maintenance.
- Weak security: There are main two security risks with all software and data are stored on the local disks of individual machines. The first is associated with malicious attacks, such as viruses, worms and spy wares. The other is the risk of information leakage and data theft. If sensitive data are fetched and cached on local disks, they will be potentially available to the public or to the attackers.
- Not user-friendly: Although the GUI interface of modern OSes makes common users access computing easier than past, it is too far from in a service-oriented and hassle-free way, like that with a traditional appliance. With personal computing devices, users have to install software, update or patch the system, troubleshoot errors, fight against virus, worms or spy ware, perform backups and even re-install system in some cases. Unfortunately, these tasks of administration, maintenance and management are very tedious and frustrating for most users, even professionals.

Pervasive computing promises a computing environment that ubiquitously supports users in accomplishing their tasks in a way that the actual computing devices and technologies are largely invisible. In order to achieve this goal, various types of devices, such as pads, wearable, smart phones are invented and widely available, as well as the wireless networking standards, such as IEEE 802.11 or Bluetooth [11]. At the same time, due to the limitation of existing distributed application technologies, a number of research projects have been exploring how to build systems and applications in such a global computing infrastructure. For example, projects such as Globus [12] and Smart Classroom [13] have been explored to extend the traditional operating systems to support service-oriented computing or provide seamless and convenient computer and human interfaces. Other efforts focus on managing autonomous mobile computing environments, such as Agent [14]. While these technologies are clearly useful, they lack an integrated framework for building applications and thus may require additional services from operating systems. In addition, the technical issues such as mobility, invisibility or scalability have not been addressed to achieve the vision described by Mark Weiser [1].

We argue that, not users-friendly, i.e., users can not get services from computer easily, is one of the main reasons why such a vision as "technology that disappears" can not be realized. Thus, current computer systems are not suitable for pervasive computing. In order to address these challenges, in this paper, we propose a new paradigm that will make computer system more cost-effective, more reliable, more

flexible, more efficient and easier to use. Transparent Computing paradigm decouples software (including different OS), data and state from the underlying hardware. Specifically, all the required software and data are centralized on central servers and streamed to the clients on demand to carry out the computing tasks leveraging the local CPU and memory resources. In such a way, we can satisfy the above requirements of a new paradigm, as described in details later.

The rest of this paper is organized as follows. Section 2 introduces the concept of Transparent Computing. Section 3 presents the design and implementation of a pilot system based on the principle of Transparent Computing. Section 4 evaluates the pilot system with primitive real world and experimental results to show the feasibility and effectiveness. Finally, Section 5 concludes this paper.

2 Transparent Computing

Transparent Computing paradigm is aiming at the vision advocated by ubiquitous/pervasive computing, in which users can demand computing services in a hassle-free way. This paradigm is working in a network environment which may be wireless or not. Some centralized simple servers where the software and data, including heterogeneous OSes and applications are stored upon them and some bare client machines which may be thin clients, PDAs or mobile phones are connected together through the networks. The servers in this environment are working as warehouses which only store the software resources such as OSes, application software, Webpages, or database materials. The bare clients are the tools interacting with users and calculating. A user can choose heterogeneous OSes and applications upon them on demand through these bare clients, similar to choose different TV channels in daily life. A selected OS and its applications will be allocated to the memories of the clients by paging in and the calculation of the OSes and the applications are performed with the client CPUs and memories, not on the servers.

The brief description of the Transparent Computing paradigm can be illustrated in Figure 1. As shown in Figure 1, the servers are considered as the resource provider. They provide with OSes, application software and data to the bare clients, who need these resources. This paradigm is different from any current computing paradigm such as diskless workstation or client/server paradigm, since it can share different OSes, applications and data stored on centralized servers via the same bare client.

The Transparent Computing paradigm can achieve the following desired features:

- User and application transparency: The use of Transparent Computing technologies is transparent to users and applications, and requires no application modification. From the perspective of users and applications, there is no difference from their accustomed usage of tradition paradigm.
- Heterogeneous OS support: Users can flexibly choose to boot and use the desired operating systems and applications via the same bare hardware platform.
- Streaming delivery: The software and data are delivered in a steaming way to clients instead of residing on the client permanently, achieving a computing on demand.

- Supports of various devices: Because of recent proliferation of special-purpose computing devices in ubiquitous/pervasive computing, Transparent Computing aims to provide a common software delivery mechanism to various types of computing devices.
- Enhanced security: The system can be protected at the servers due to the centralized management. In addition, since no data or state can be preserved at the clients, it can reduce the risks of information leakage and data theft.

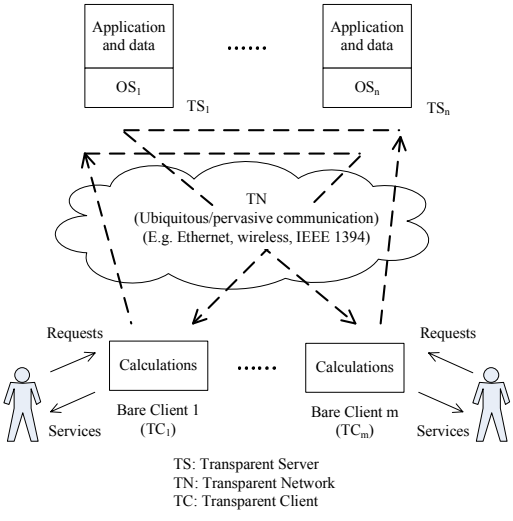


Fig. 1. A brief description of the Transparent Computing paradigm

3 A Pilot System

The Transparent Computing paradigm are proposed to support various kinds of smart devices, wired or wireless networking environments that can provide an appropriate latency performance. However, in this paper, we only implemented and evaluated the Transparent Paradigm with low end PC hardware and wired common Ethernet LAN to investigate the issues of implementing Transparent Computing on various devices and networking environments. We believe that the Transparent Computing paradigm can be extended to support a pervasive computing infrastructure.

3.1 System Overview

To investigate the idea described above, we designed and developed a pilot system, named as TransCom system. TransCom system adopts the conventional client and server architecture. Each transparent client machine is a bare-hardware without any local hard disks. The transparent server can be a regular PC or a high-end dedicated machine that stores all the needed software and data required for completing tasks at clients. The transparent network is a local area network (LAN). To use a TransCom client, users just need to power it on and boot it remotely and load the selected OS and

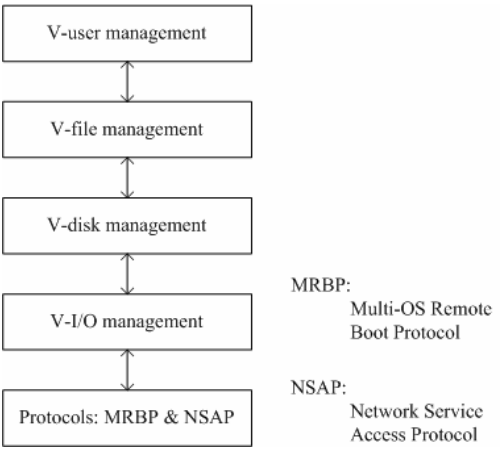


Fig. 2. The 4VP+ software platform

software from a server. After this process, users can access the client in the same way as a regular PC with local storage devices. In this sense, the client is operating like a service transceiver, and the server is like a service repository and provider by delivering software and data to clients in a way similar to an audio or video streaming.

The implementation of TransCom is based on a 4VP+ (shortly for four virtual layers and two protocols) software platform which partly operates at the assembler instruction level. The structure of 4VP+ is shown in Figure 2. This 4VP+ software platform is mostly installed in a management server, except a part of MRBP (Multi-OS Remote Boot Protocol) which is burned into the BIOS EEPROM in the bare clients. However, the other programs of 4VP+ platform run in both client and server according to their functions.

The MRBP is used to boot the TransCom clients remotely from servers. It can let users select their desired OSes and applications. The MRBP then installs a NSAP (Network Service Access Protocol) module which is written in an assembler instruction to enable a virtual I/O for clients. Through this virtual I/O interface, clients can access the OS images located at servers and then load them as with a regular PC. After the needed OS is started up, the OS-specific in kernel modules for Virtual I/O (V-IO), Virtual disk (V-disk), Virtual file (V-file) and Virtual users (V-user) will function to help the users to access the software and carry out their tasks seamlessly.

The V-user module is used to manage users. Before users setting up a connection to a server to access their private information or data, the system has to authenticate them through a network authentication protocol, such as Kerberos [3] with the V-user manager in the server. If it successes, the server will serve the clients through the NSAP. Note that users can access their information via any clients in the system for convenience. In the following sections, we will discuss other technologies in more details.

3.2 Multi-OS Remote Boot

In lack of local storage of software and data, TransCom clients have to support loading the desired OS environment from the server. In order to support heterogeneous OS environments, we can not adopt traditional methods for booting client machines without local hard disks, for example, burning a specific OS kernel into a client ROM [4], or a more flexible way by downloading the OS kernel directly from the server [5] that can not support OSes like Windows that haven't a clear kernel.

Our key idea is to first initialize a virtual I/O device for clients, whose requests will be sent through networks and executed by a server instead of a local hard disk. In such a way, any OS can be loaded according to its normal procedure as if a real hard disk existed. Therefore, we used a universal OS loader to load the desired OS with the virtual I/O device. The universal OS loader will be downloaded from the server and loaded by the client. Figure 6 lists the main steps involved in the remote boot process.

Step 1: Discovery. Each client sends a discovery message to the server and then obtains an IP address for subsequent network connections. Given the boot discovery message, the server will send back the corresponding IP address to the client. This process is similar to a DHCP process [6].

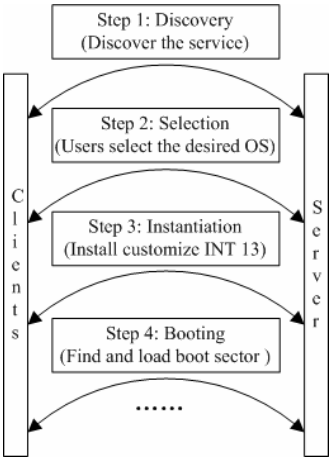


Fig. 3. The main steps involved in MRBP

Step 2: Selection. After setting up an IP connection to the server, the TransCom client sends a request to obtain the available OS environment lists provided by the system. The list will be displayed at the screen and users can then select one of them to boot up.

Step 3: Instantiation. After users' selection, the client then downloads from the server a universal OS loader. The universal OS loader is to instantiate a BIOS-enabled virtual I/O device. The approach is to replace the BIOS hard disk access function (For example, INT 13 on X86 machine) with a customized one that redirects all I/O access requests to the server.

Step 4: Booting. The immediate next step is to find the boot sector and load the OS-specific loader provided by different OS providers. Usually, the boot sector is the Master Block Record in the virtual disk, which is the first step in a regular OS boot process. After this point, the OS takes control and continues to boot up as normal.

However, most modern OSes today access memory in protected mode for performance considerations, thus the BIOS-enabled V-I/O will no longer function due to its real mode memory management with the regular OS boot processes. To avoid this, we install several OS-specific modules to supersede BIOS-enabled V-I/O functions before they were disabled. These OS-specific modules are implemented as in-kernel OS modules, which will continue to deliver the services for users with OS-enabled virtual I/O and other functions.

3.3 Virtual Disks Accessing

The core idea of TransCom system is the virtualization of devices. For transparency for OSes and users, the virtual I/O is triggered with virtual disks. Each virtual disk is mapped to a virtual disk image in the server repositories. The virtual image holds the actual disk contents and is the basic management unit. The contents of virtual images are organized in blocks corresponding to virtual disk blocks seen by client machines. While a real disk's parameters will be stored in the machine CMOS memory, the parameter of a virtual disk image is stored in the image's added block number, which can be queried by the client. The block number of an image contains the following information about the corresponding virtual disk: total size, block length, and CHS (Cylinder/Head/Sectors) parameters. Note that a V-disk seen by users can be mapped to different images. This feature provides flexibility for sharing virtual images among different users.

We implemented a virtual disk management system to manage virtual disks and images as illustrated in Figure 4. If the data requested by the client's CPU is not found in its main memory, there is a page fault error, which will trigger a trap through the file system. This trap will be captured by the VDMS client, which will encapsulated it into a NSAP packet and send it to the server. Upon receiving a request from the client, the VDMS server will first look up its cache to see if the requested data exists. If it is true, the contents will be replied to the client with NSAP. However, if the data doesn't exist in the cache, the VDMS server will fetch the contents from the virtual disk images located on the server disks and then reply it to the clients.

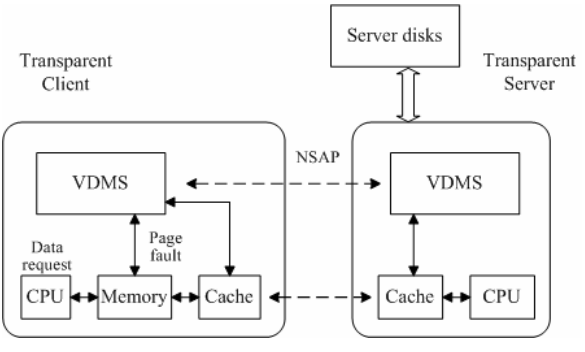


Fig. 4. The virtual disk management system

3.4 Protection and Anti-virus

Because the software is centralized in servers in TransCom system, they can be shared among users. Thus it is very important to protect the system from attacks from user errors, virus, worms and spy wares. There are two main approaches in TransCom to protect the system: One is to rewrite the BIOS and protect the BIOS from outside threatens; the other is to protect the system images from being destroyed by users through a V-file mechanism.

TransCom uses a file system level agent called File System Filter (FSF) to provide a virtual view of file system for users to protect the system files from being modified. Specifically, each client is configured with three types of V-disks.

- Mono disk: It is used to store operating system and application files that are shared across all clients in a TransCom system. Different clients' mono disks are mapped to a single V-disk image at the server. Thus mono disk is for read only.
- Shadow disk: Each client has a shadow disk that is used to store customized system files. The shadow disk is mapped to a client specific V-disk image at the server. The corresponding data are private and will not be shared by other client hosts. Therefore, the shadow disk is in essence a copy-on-write disk for isolating user-specific configuration files and system files.
- Private disk: Each client has one or more private disks that are used to store private user data. Similar to shadow disk, each client's private disk is mapped to a client-specific V-disk images at the server, and will not be shared.

The FSF translates the file access requests on user-perceived V-disks into those on server-perceived V-disks by intercepting all file system calls. If the file to be accessed locates on the user disk (user-perceived), the FSF simply maps the request to the same file on the server-perceived private disk. If the file to be accessed is on the system V-disk, the FSF will redirect the request to the shadow disk in the following two cases: (1) a read request to a system file that already has a customized copy on the shadow disk, and (2) a write request to a system file. Otherwise, the FSF will redirect the request to the mono disk. The FSF therefore supports dynamic redirection of system files for enabling file system level copy-on-write semantics.

3.5 Implementation

We have implemented a prototype of TransCom that supports both Windows 2000 Professional and RedFlag 4.1 Desktop (Linux kernel 2.4.26-1) [7]. Our implementation uses the Intel PXE [8] as the remote boot protocol for sending boot requests. Because device drivers are platform dependent, we implemented two different V-disk drivers, customized for Windows and Linux, respectively. The implementations are in C++. Since Windows 2000 is a modified microkernel, we modified the corresponding Windows Registry files for the OS to load the V-disk driver. Thus there is no need to change or recompile the kernel. However, Linux is monolithic kernel. Thus we compiled the V-disk driver into the kernel by modifying the related kernel source code before recompilation.

4 Evaluation

In this section, we evaluated the pilot system in test bed and real world environments which consist of low end cheap PC hardware, common PC servers and LANs to show that Computing Paradigm may also be implemented with other types of smart devices and wireless networking environments in an acceptable performance in cases that the network latency is less enough.

Our initial experimental results focus on the Windows implementation. In addition, due to the space limited, we only present primitive performance results. In all experiments, TransCom clients are Intel Celeron 1GHz machines, each with 128 MB DDR 133 RAM and 100 Mps onboard network card. The server is an AMD Athlon64 3000+ machine, with 2 GB Dual DDR 400 RAM, two 80 GB Seagate Barracuda 7200rpm soft RAID0 hard disks, and 1 Gbps onboard network card. The clients and the server are connected by an Ethernet switch with 48 100Mbps interfaces (used for clients) and two 1 Gbps interfaces (used for the server). The server OS is Windows 2003 Server (SP1) edition. TransCom clients use Windows 2000 professional (SP2).

4.1 Virtual Disks Through

In order to examine V-disk performance, we evaluated the V-disk access throughput using the Microsoft I/O performance tool SQLIO [9] to summit random disk access requests of different size to the client. We just show the result of random, unbuffered case in Figure 5 and 6. The unbuffered means that the client's file system caches are disabled. We varied the server's memory size and disabled the server's cache to see the impact on the performance. We also compared the throughput with that of the regular PC's hard disk.

For read access, V-disk throughput increases with the request size and is higher than the local disk, but decreasing when the request size is larger than 32KB, which is the maximum size delivered. So, this may be due to that the network communication time dominates the latency, for that a large request size will result in several service requests. We also can see that the V-disk throughput drops sharply with the decreasing size of the server's memory, alike when the server's cache was disabled. Thus the

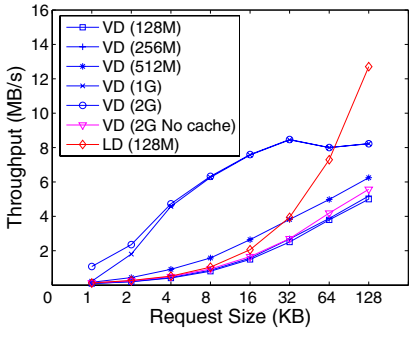


Fig. 5. Random V-disk read throughput

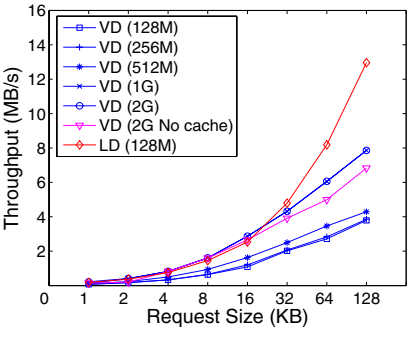


Fig. 6. Random V-disk write throughput

server's memory cache is the key factor for the performance and can explain why the V-disk performance is higher than the local disk. The write throughput is similar to the read, but with a smaller amount. In addition, we have tested the throughput in the buffer case, the result is also similar.

4.2 Application Performance

Our Windows based system has been deployed in a university e-learning classroom for daily usage for 14 months. In our real experiment, we varied the number of clients supported by a TransCom server, and compared the performance with the performance of a regular PC (with the same hardware configuration, and has an additional local hard disk of 80GB Seagate Barracuda 7200rpm).

Table 1. TransCom performance in four different categories (latency: seconds)

	PC	TransCom client			
		1 client	10 clients	20 clients	30 clients
OS boot	53.13	48.73	70.62	92.79	142.57
MS Word 2003 (start up)	2.23	1.26	2.28	6.35	11.50
MS Word 2003 (open a 1MB file)	3.07	2.13	3.57	7.27	11.57
MS PPT 2003 (start up)	5.21	3.04	6.58	9.98	18.18
Photopshop v7.0 (start up)	13.29	11.08	16.48	27.51	60.51
Flash V6.0 (start up)	18.62	7.16	31.41	74.30	76.56
3D MAX V4.0 (start up)	29.71	25.68	34.24	54.18	76.56
Copy a file (20MB)	11.59	8.95	19.75	37.51	56.13
Copy a file (50MB)	28.24	24.33	49.48	109.52	246.56

Table 1 lists the four categories of performance in terms of latency: OS booting, office applications, image applications, and file copy. The OS boot latency refers to the time elapsed from powering on the client to the login window displayed on screen. For all four categories of performance, we observe that TransCom outperforms the regular PC in the single client scenario. The latency increases with the numbers of clients within our range and is on the order of tens of seconds, with the exception of the cases of OS boot and copying a 50 MB files with 20 to 30 clients. During our initial deployment, TransCom has been running stably most of time. There were a few system crashes, mostly due to software errors, and can be recovered by system rebooting. After deploying TransCom, the total client service downtime in the e-learning classroom is far shorter compared with past. Furthermore, the tasks of installing and patching software of a classroom of 30 clients can now be finished within one hour, as opposed to one or two days in the past. We expect the scale of deployment to increase in the near future.

5 Summary and Future Work

We proposed a new paradigm for pervasive computing. In Transparent Computing, software and data are decoupled from the underlying hardware, and users can demand software services on demand from a centralized server. We also presented a pilot system TransCom to show the effectiveness. By using virtualization of devices and files, TransCom can flexibly support running heterogeneous OSes including Windows and can also share and protect the system from attacks. The initial experimental results and real world deployment of our prototype system have suggested that TransCom is a feasible and cost-effective solution for future computing infrastructure.

Future work includes supporting a broader range of devices (PDA, Intelligent Mobile Phone, Digital Appliance, etc.), OSes (e.g., Open Solaris [10]) and more applications, further optimizing TransCom performance, increasing the system robustness, and enhancing the system security with data encryption and user-level access control.

References

1. Weiser, M.: The computer for the twenty-first century. Scientific American, 265(3) 94-104, Sept. 1991
2. Turver, R. J., Malcolm, M.: An early impact analysis technique for software maintenance. Journal of Software Maintenance: Research and Practice, 6(1) 35–52, 1994
3. Neuman, B. C., Ts'o, T.: Kerberos: An Authentication Service for Computer Networks, IEEE Communications, 32(9)33-38. September 1994
4. Sun Microsystems Inc., Sun Ray Overview, White Paper, Version 2, December 2004. Available at: http://www.sun.com/sunray/whitepapers.html
5. Cheriton, D. R., Zwaenepoel, W.: The Distributed V Kernel and its Performance for Diskless Workstations. In Proceedings of the 9th ACM Symposium on Operating Systems Principles, pages: 128-140, Bretton Woods, N.H., Oct. 1983
6. Droms, R.: Dynamic Host Configuration Protocol. RFC 2131, March 1997
7. RedFlag Linux. http://www.redflag-linux.com/eindex.html
8. Preboot Execution Environment (PXE) Specification. ftp://download.intel.com/labs/manage/wfm/download/pxespec.pdf, 1999.
9. Chung, L., Gray, J., Worthington, B., Host, R.: Windows 2000 Disk IO Performance. Technical Report MSTR-2000-55, Microsoft Research, 2000
10. http://www.opensolaris.org/os/
11. Bluetooth SIG. http://www.bluetooth.com
12. Foster, I., Kesselman, C.: Globus: A metacomputing infrastructure toolkit. International Journal of Supercomputer Applications and High Performance Computing, 11(2) 115-128, 1997
13. Shi, Y., Xie, W., Xu, G., et al: The smart classroom: merging technologies for seamless tele-eduction, IEEE Pervasive Computing, 2(2) 47-55, April-June 2003.
14. Milojicic, D., Douglis, F., Wheeler, R (eds): Mobility: Processes, Computers, and Agents, chapter 14, pages 457-641. ACM Press, Addison-Wesley (1999)

Drag and Drop by Laser Pointer: Seamless Interaction with Multiple Large Displays

Liang Zhang[1], Yuanchun Shi[1], and Jichun Chen[2]

[1] Key Laboratory of Pervasive Computing,
Department of Computer Science and Technology,
Tsinghua University, Beijing 100084, China
blinkzhangpaul@gmail.com, shiyc@tsinghua.edu.cn
[2] Institute of Software,
Chinese Academy of Sciences,Beijing 100084, China
chen1937@sohu.com

Abstract. This paper presents a novel interaction technique with multiple large displays in smart space. We call it D2LP, short for Drag and Drop by Laser Pointer, where a specially designed laser pointer uPen is a handheld interactive device for users. By D2LP, large displays driven by different computers can be made as one seamless integrated uniform system. With a uPen in hand users can directly point and draw on the surface of displays, drag and drop digital objects among multiple displays continuously and smoothly. Report on evaluation experiments shows that D2LP can effectively support the freely interacting with multiple wall-sized displays and it is preferred over the conventional mouse-driven cursor based system for the application in smart spaces with rich information displays.

1 Introduction

In smart spaces, especially those used in command centers or meeting rooms, large displays are needed for practical reasons. With the rapid development in large display manufacturing, multiple wall-sized displays are increasingly common in universities and corporate conference rooms. Large displays on walls can be easily seen by a group of users and multiple large displays can show more to the audience simultaneously. However, input devices, keyboards and mice, are designed for a single computer and a single user. When dealing with multiple large displays driven by different computers, users need to switch between various devices to manipulate different computers and they have to interrupt their presentations to physically move to the front of the computers. Furthermore, in traditional input devices based systems, digital objects transfer among multiple displays is rather difficult to carry out, but the transfer occurs quite often in smart spaces.

To overcome this cumbersome discomfort, we develop a new interaction technique based on laser pointer for environment with multiple large displays: D2LP which is short for drag and drop by laser pointer. Common laser pointers can not serve the purpose, so we also invent a special laser pointer named uPen. A user can interact on multiple large displays more freely and simultaneously with

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 12–20, 2006.

a uPen[1]. All of these large displays, driven by different computers, are treated as a coordinated system. Moreover, digital objects transfer-ring from one display to another can be done by drag and drop act smoothly and easily.

2 Related Works

Studies on interaction with laser pointers on large displays at a distance have been conducted by many researchers. Kirstein and Muller [2] first presented a system that uses a laser pointer as a pointing device. Nielsen [3] proposed an inexpensive interaction technique by introducing a set of window events for the laser pointer such as laser-on/off, and laser-move/dwell. Winograd and Guimbretiere [4] proposed a new kind of interaction techniques for large displays, which are based on "gesture and sweep" paradigm instead of the usual "point and click [5]". Chen and Davis [6] described a system that can provide multiple laser pointer inputs with multiple cameras. Ji-Yong [7] used different blink patterns to distinguish laser pointers.

None of them focuses on the interaction between different large displays using laser pointers while some other researchers explore design issues for coordinated device activity, for example: Rekimoto's pick and drop [5], GeiBler's throwing [8], Chiu's ModSlideShow [9] and Khan's Frisbee [10]. But their ideas either need high cost touch boards or need some additional input devices such as tablets to deal with multiple large displays. Low cost laser pointer used as an input device is not much researched yet.

The rest of this paper will discuss the principles and use study of D2LP system.

3 Interaction Principles

In smart space, D2LP system augments multiple large displays to serve as extended information kiosks, where users can easily access, share and transfer digital objects. These processes are done by D2LP's functions of Distant Manipulation and Drag and Drop by Laser Pointer:

3.1 Distant Manipulation

People often stand away to use large displays. For instance, in a conference hall, large title displays are always set at a high position where is out of users' reach and even in a small meeting room, a speaker is not used to just sit in front a computer reading slides. He/She prefers to face the audience and displays and even walk around sometimes. Accordingly, there is a need to allow the user to interact with the displays at a distance with no cable connected. In systems supported with D2LP, this problem is well solved. One can use a uPen [1], which will be introduced in detail in Section 4.2, as an augmented mouse to manipulate the computer at a distance.

The red laser spot emitted by the uPen on the large screen is grabbed by a video camera, and its position is interpreted as the location of the cursor. The

Left and Right buttons on the uPen are collected by the receiver and emulated as a standard mouse's left and right button events respectively. Additionally, as the On/Off button simultaneously emits a laser beam and the uPen's ID. The ID information can be added to the mouse event. As a result, the system has the ability of recognizing who is interacting currently, which is potential to support personalized service.

3.2 Drag and Drop by Laser Pointer

Motivation: Multiple large displays are often adopted in smart space, and digital objects transferring occurs quite often, shown as a table in Rekimoto's paper in CHI'98 [5]. As mentioned above, there are several researches on the issue, but all focus on touch based displays or common computers. Interaction on normal wall sized large displays with distant input devices such as laser pointers is overlooked. We focus on D2LP interaction because there are several advances over the touch based display when transferring data using a laser pointer:

- All the manipulation is done at one position at a distance facing the displays, so it will make the physical movement as limited as possible, and it is quite user friendly while a user is making a speech or giving a presentation.
- Some areas of the display are out of reach for the user, so touch based display system has to split the display and make a specified region for the manipulation area out of reach or adopt other devices such as tablets severed as additional input devices. But in D2LP system, this is done easily and smoothly: One just uses a uPen to drag a digital object and drop at any position in the displays as one wishes.

Implementation: D2LP is designed and implemented to facilitate interaction across multiple large displays driven by different computers, especially for the

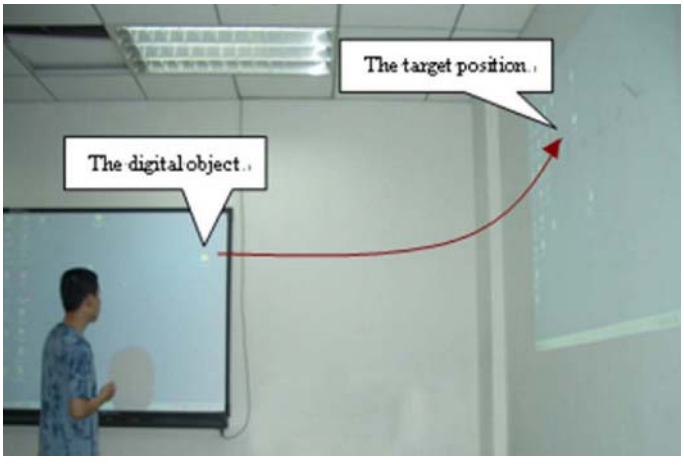


Fig. 1. One D2LP scene in smart space

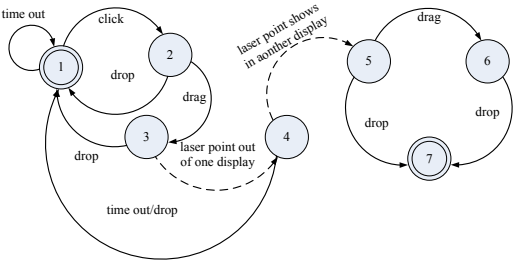


Fig. 2. The state transition diagram of D2LP

digital objects transferring between displays. D2LP allows users to seamlessly drag digital objects via a laser pointer from a display and across different displays then drop at a position on another display. It works as the following:

A user selects the object on one large display by a uPen (pointing the laser beam onto the digital object and pressing down the Left Button). The selected object will move along with the laser spot on the display. When the laser beam reaches the edge of the display, it continues its motion in the same direction, crosses over the gap between displays, and then moves on to the edge of the adjacent display. Following the track of the laser spot, the selected object is also transferred onto the machine which drives the adjacent display and continues its motion along with the laser spot until the user drops it by releasing the Left button. For a user, it seems that he/she is dragging the object on a large virtual display rather than manipulating on several separate screens driven by different computers, which makes the interaction more user-centered and facilitating. An actual D2LP scene set up in our laboratory is shown in Fig.1.

D2LP system works according to the state transition shown in Fig.2. State 1 is the start state which is the time when the system has started up or finishes one object transfer already. The end states are State 1 and 7. If processes end at state 1, it means the digital object is still on the original display: the user either completes a drag and drop act on one display or fails when transferring the object to another display. If processes end at state 7, it indicates that the drag and drop act transferring between different displays is successful. Transfers between different displays go along the route 1-2-3-4-5-6-7 or 1-2-3-4-5-6 while normal drag and drop acts take the transition loop 1-2-3. The descriptions on the lines are the state transfer conditions. For example, Starting at state 1, only if the user clicks the Left button on a uPen, the state will transfer to state 2, which means the uPen's Left Button is pressed and the object is "adhered" to it.

4 System Design

4.1 System Configuration

Fig.3 shows the framework set up in smart classroom [11], which is a typical configuration of D2LP system. In this system, when standing or sitting in front

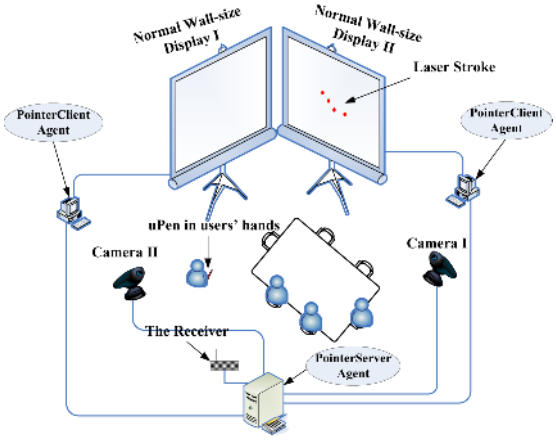


Fig. 3. A typical configuration of D2LP system

of the multiple large displays at a distance, a user can use a uPen, which is a handhold augmented laser pointer to complete all the jobs done by mice. Furthermore, he/she can interact with multiple large displays just as with one uniform large display screen. In order to make it clear, we give some detailed description on the devices shown in Fig.3:

- Pointer Server Agent is the key server computer of D2LP system. It collects the video data from the cameras, searches the laser point from every frame and simulates the mouse move events. At the same time, it also listens to the receiver collecting button click events and simulates the mouse click and release events. Then it sends out the simulated mouse events to the other agents through Smart Platform [12], which is a software infrastructure for smart space application in our smart classroom [11].
- Camera I and Camera II take the charge of video data collecting. In order to decrease the search complexity in practice, we add on wave-filter screens in front of the camera lens.
- The Receiver is a device of the task to collect the button click events emitted by uPen.
- Pointer Client Agents are computers which drive the displays by projects. Each display is driven by one pointer client agent, and it results in the difficulty in interacting between displays in traditional input devices and systems. Our solution is that every PCA receives the information from the smart platform and judges whether it is the target computer. If not, it ignores the information, or else it responds by stimulating the mouse act or filing transferring and so on.
- Wall-sized displays are normal displays and in our smart classroom, one of the displays is just part of one wall as shown in Fig.1.

4.2 uPen: The Interaction Tool

uPen [1] is an augmented laser emitter, with three press buttons (On/Off Button, Left Button, Right Button), and a Contact-Activated Button and a wireless communication module. Fig.4 illustrates the structure of a uPen. To receive the signal emitted by the uPen, a receiver is connected to the computer, which is also illustrated in Fig.4.

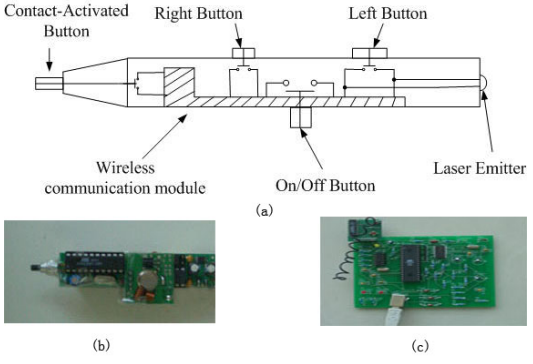


Fig. 4. (a) is the structure of uPen and (b) is the inside of uPen. (c) is the inside of the Receiver.

Functions of additional components on uPen are detailed as follows:

- Laser emitter: Emitting a laser beam
- On/Off Button: Emitting the uPen's ID
- Right Button : Emulating a mouse's right button
- Left Button : Emulating a mouse's left button
- Contact-Activated Button: Emitting uPen's ID when being pressed directly on the surface of the display. Note: this is designed for touch-base board in this paper it is not concerned.
- Wireless-communication module: Communicating with computer.

5 Experimental Evaluation

5.1 Goal

In order to know better how D2LP performs in smart spaces, the experimental evaluation is undertaken. The two research questions are:

- Whether D2LP system is a potential substitute for the traditional input system in the manipulation in smart spaces with large displays driven by different computers.
- Whether D2LP system can facilitate users interactively while dealing with digital objects transferring.

5.2 Design

Two experiments have been made and 6 subjects (5 male and 1 female), experienced computer users between the ages of 22 and 28 are recruited.

Our configuration is shown in Fig.3. The subjects are asked to use the traditional input way: input with mice and keyboards and our D2LP system to complete two jobs:

- job 1: Open one PowerPoint file on the desktop of one display and make it full screen; then switch to another display and open another PowerPoint file with full screen model.
- job 2: Transfer a given digital object from one display to another.

We request the subjects to do each job twice and three of them are asked to use mice and keyboards first while the others use D2LP first. The time each task cost is recorded and questionnaires are sent to subjects after they finish the experiments to get feedback.

5.3 Results

Our data (shown in Fig.5) confirms that users perform much faster while interacting on multiple large displays with D2LP system compared with the traditional way of input.

Fig.5a shows that the average time cost to finish job1 with D2LP is two seconds faster than that with a mouse. This is not notable and we should notice that when manipulating on one computer, the D2LP is a little slower than the mouse input system if the users are allowed to sit in front of the computer. But the reason why D2LP is still faster to finish job1 is that in smart space users do not just interact with one display, they interact with multiple displays simultaneously. Although each task may be simple on one separate display, the real work consists of a series of them with different displays and switches between them. The switch time depresses the performance of the traditional input system. D2LP makes it easy: switch time is lowered to nearly none.

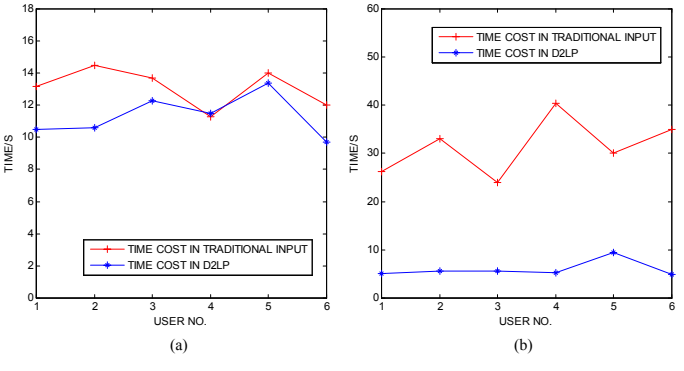


Fig. 5. (a) is Job 1 time cost data and (b) is Job 2 time cost data

From Fig.5b, the advantage of D2LP is obvious. In digital objects transferring between different large displays, D2LP facilitates interaction on one uniform screen and the time cost is quite brief, which is, on average, 20 seconds faster. Moreover, in traditional way users have to switch on different computers even with some physical move and they have to think of a way to transfer the object: three of the subjects use FTP to finish job2, and one uses floppy disk and the other two use share folder. But in D2LP system, this job is quite natural and simple. A user just needs to click, drag and drop with a laser pointer.

From Fig.5b, it is evident that one subject costs much more time than the other five with some 4 seconds more (user no.5 in Fig.5b). It is because when he drags the objects through the gap between two large displays, he drops the object, which means he releases the Left button on uPen heedlessly and he has to click on the digital object and drag to another display again. It costs additional time but, still, it is much faster than the traditional way.

Some important information from the questionnaires indicate that most subjects enjoy the use of uPen with D2LP in smart spaces. They admit that D2LP is nature and acceptable. They acclaim that it is amazing when dragging an object from one screen to another. They also propose issues which need to be further studied such as jittering and latency which hinder the manipulation; the confusion some users may have while the laser point and cursor are shown at the same time.

6 Conclusion and Future Work

In this paper, we introduced D2LP - a new technique for interacting with multiple large title displays in smart spaces. D2LP is designed to support and facilitate the free interaction between multiple wall-sized displays. A user in smart space equipped with D2LP can use a uPen to manipulate different large displays driven by different computers as on a large virtual display, including transferring the digital objects between displays in a nature and simple way. We also show the value of augmenting the traditional input system with D2LP by running an experiment and we report a significant performance improvement in the file transfer job.

Future research work is needed in the following aspects. First, clicking is really a difficult act for interaction based on the laser pointer, because jitter is so distinct that users can not orientate the cursor at a distance at ease. We want to discover a gesture input function to substitute for clicking. Second, we plan to extend the D2LP to support multiple users with personalized service. uPen is designed for multiple users to employ but the problem of simultaneity confusion remains to be solved now. Also we hope to find out whether other techniques can be added in the D2LP, replacing the laser pointing or augmenting it.

Acknowledgment. This work is supported by the National Development Project for the Next Generation Internet , CNGI-04-15-3A.

References

1. Xiaojun Bi, Yuanchun Shi, X.C.: upen: A smart pen-liked device for facilitating interaction on large displays. In: Proceedings of the First IEEE International Workshop on Horizontal Interactive Human Computer Systems. (2006) 160–168
2. Carsten Kirstein, H.M.: Interaction with a projection screen using a camera-tracked laser pointer. In: Proceedings of Conference on Multimedia Modeling. (1998) 191–192
3. Nielsen, D.R.T.N.: Laser pointer interaction. In: Proceedings of CHI01. (2001) 17–22
4. Winograd, T., Guimbretiere, F.: Visual instruments for an interactive mural. In: CHI99 Abstracts. (1999) 234–235
5. Rekimoto, J.: Pickcandcdrop: A direct manipulation technique for multiple computer environments. In: Proceedings of UIST97. (1997) 31–39
6. Xing Chen, J.D.: Lumipoint: Multi-user laser-based interaction on large tiled displays. In: Stanford CS Technical Report. (2001)
7. Ji-Young Oh, W.S.: Laser pointers as collaborative pointing devices. In: Graphics Interfaces. (2002) 141–149
8. GeiBler, J.: Shuffle, throw or take it! working efficiently with an interactive wall. In: CHI '98 LateCBreaking Results. (1998) 265–266
9. Pederson, E., M.K.M.T.H.F.T.: An electronic whiteboard for informal workgroup meetings. In: Proceedings of CHI93. (1993) 391–398
10. Khan, A., F.G.A.D.B.N.K.G.: A remote control interface for large displays. In: Proceedings of UIST04. (2004) 127–136
11. Yuanchun Shi, Weikai Xie., G.X.R.S.E.C.Y.M., Liu, F.: The smart classroom: Merging technologies for seamless tele-education. In: IEEE Pervasive Computing. (2003) 47–55
12. Xie, W.: Smart platform: A software infrastructure for smart space (siss). In: ICMI2002,IEEE CS Press. (2002) 429–434

A Flexible Display by Integrating a Wall-Size Display and Steerable Projectors

Li-Wei Chan[1], Wei-Shian Ye [2], Shou-Chun Liao[2], Yu-Pao Tsai[3,4], Jane Hsu[1,2], and Yi-Ping Hung[1,2,3]

[1] Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan
[2] Dept. of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan
[3] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[4] Dept. of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan
{hung, yjhsu}@csie.ntu.edu.tw

Abstract. Many wall-size display systems are built to provide large-scale visualization. These systems may be quite successful for some limited applications, but are very inflexible, since these systems only have fixed display regions. This paper integrates steerable projectors whose beam can be moved under computer control onto a wall-size display system to strengthen its display ability. With the steerable projectors, the integrated display system, named *Flexible Display*, provide an extendable display region. This consists of a large-scale display region and several movable display regions, such that the integrated display system has great potential in the area of human-computer interaction and information visualization. This paper applies the Flexible Display to a virtual museum application to give the users fluent navigation experience. For the application, the Flexible Display provides the following functions: 1) intensity and resolution enhancement of sub-region of display wall, 2) information augmentation, and 3) "stepping user interfaces" for its viewers interacting with display wall.

1 Introduction

Large displays, with the large-scale, high-resolution displaying capability, are highly applicable in many applications and public places. For example, information visualization in transport centers such as in airports and train stations; advertisement, bargaining, and information retrieval of purchasing in shopping center or markets; virtual reality, visual effect demonstration of artifacts or environments in museum or in an exhibition room. With its applicability and being increasingly found in public places, large displays have shown its importance as a ubiquitous technology in our future life.

Although benefits from embedding a large display in the environment are attractive, building such a display system is not easy. Numerous of attempts have been made by scholars to show how tiled projectors, forming projector arrays, to provide single large display. However, this kind of projector system is fixed and always requiring time-consuming calibration process. Besides, occupying large floor space also

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 21 – 31, 2006.

makes it impractical in a normal environment setting. Raskar et al. [5] proposed that an ad-hoc cluster of projectors can create a self-configuring display on even a non-planar wall. Their approach has greatly reduced the process of calibration. Claudio Pinhanez [6, 7] makes projector-enabled displays a big step toward ubiquity by introducing steerable projectors into the environment. The steerable projector is composed of a projector and a mirror mounted on the pan-tile unit. Rotating the mirror can freely direct the beams onto almost everywhere in the environment. In summary, a fixed wall-size display is capable of presenting large scale and high resolution imagery, while a steerable projector is charactering on its ubiquity. Determining which projector system to be deployed depends on the application's needs.

In this paper, we are investigating how a wall-size display and steerable projectors coexisted in the environment can gain benefits from each other. We also show that how our system enriches interactions between the display wall and the viewers by leveraging strengths of fixed and steerable projector settings. The system, we called *Flexible Display*, offers advantages towering previous wall-size display systems on both its presentation and interaction abilities. We propose following functions of our system: 1) *intensity and resolution enhancement* of sub-region of display wall, 2) *information augmentation*, and 3) *stepping user interfaces* interacting with display wall. The details of these functions are described in section 2. We also demonstrated the system in a museum application.

This research investigates the impacts and its applicability of joining both fixed and steerable projectors in the environment. This paper starts by describing several systems offering projective display to environment and their limitations. We then propose that the Flexible Display, leveraging strengths from the fixed and steerable projector settings, transcends previous systems on both its presentation and interaction abilities. Section 2 details the scenario and functions provided by the Flexible Display. Section 3 describes the core techniques required of the system. Section 4 presents a museum application, followed by the conclusion and future work in Section 5.

2 Scenarios and Functions

Our system (Flexible Display) can be extensively used in variety of applications in public places. Here is a sample scenario in a museum: Visiting to museum, Susan enters the intelligent exhibition room, where a projective wall-size display is on a particular wall. Knowing Susan's entrance, the display wall shows a question asking whether it is Susan's first time to attend the exhibition. At the same time, the steerable projector moves its projection, showing two buttons, labeled "Y/N", on the floor nearby Susan's foots. Susan steps on the "Y" button. The display then starts a movie clip introducing several very exhibits of the exhibition. The steerable projection, in the meantime, highlights the sub-regions of display wall to give guidance to the viewers' focus. Following the steerable projection, Susan finds it is an easier way to perceive the information delivered by the clip.

While the movie ends, the steerable projector displays several icons of artifacts on the floor. Susan steps on one of them. The following is the display wall shows the selected artifact and the steerable projector augments some description over the display wall. After that, she moves the steerable projection by dragging a stylus on her PDA. She then can appreciate every detail of that artifact.

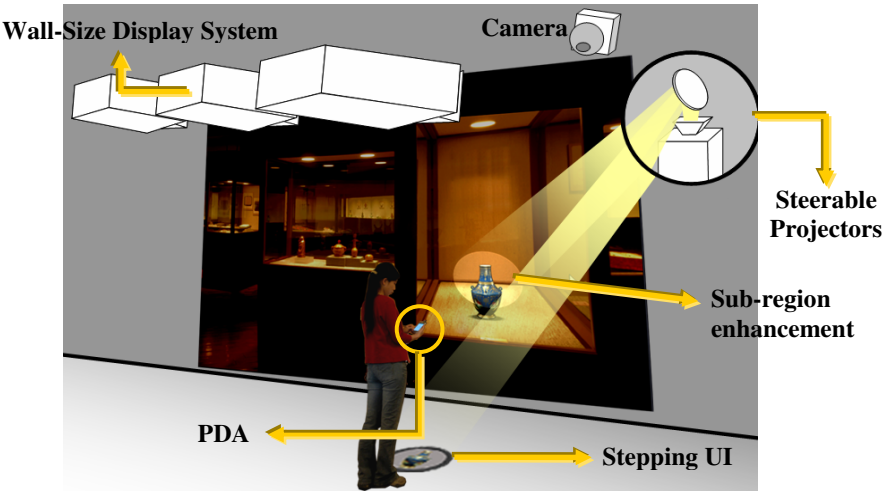


Fig. 1. A Diagram Depicting the Museum Scenario

The Flexible Display consists of following components:

1. A few projectors fixed on the ceiling, cooperatively projecting large scale display on a particular wall. In the implementation, we use three projectors to form a wall-size display system.
2. At least one steerable projector, offering moveable projection for polishing portion of display wall, augmenting information over the display or the environment, and delivering projective user interfaces which are easier accessed by the viewers.
3. A camera mounting on upper-side of display wall, for detecting the viewers' interaction with the projective user interface.
4. PDAs, that the viewers can move the steerable projection by dragging a stylus, detailing a part of the display.

We can now summarize three functions provided by our system.

1) *intensity and resolution enhancement* of sub-region of display wall,
2) *information augmentation*,
3) *stepping user interface* interacting with display wall,

2.1 Intensity and Resolution Enhancement of Sub-region of Display Wall

Current researches aim to create large, high-resolution displays by presenting improved methods of creating a "projector mosaic" [4] which is a collection of projected images combined to form one large display. These approaches usually uses numerous of projectors, causing high cost and troublesome calibration. In our previous work [8], we proposed a multi-resolution approach in the sense that the audience only focuses on a part of the projected area, so only this area requires higher resolution. Therefore, the multi-resolution approach requires only a few projectors, reducing the cost. In this approach, we assume the user always looks at the center of the display, where we

called Fovea Region, is to be displayed in high resolution. The other region, called Peripheral Region, only provides the user an overview of the displayed content, and thus only requires lower resolution. Therefore only two projectors, each serving for one region, are used to build a two-level display.

Based on the previous approach, we further replace the projector served for Fovea Region with steerable projectors. Since the steerable projectors can freely move their projections, our system can detail every part of the display where the high resolution is required. To create large scale high resolution coverage, we combine only a few conventional projectors and at least one steerable projector. The conventional projectors which fixed somewhere form a projector cluster are to provide large coverage projective display on the wall, while the steerable ones are to provide smaller but higher resolution projections. Before the fixed and steerable projectors cooperatively projecting one seamless display, we need to apply calibrations both on the wall-size display system and the steerable projectors. We leave the technique details in section 3. Compared to previous projector-based large display, our system is low cost for its large scale high resolution display enacted by only a few projectors.

2.2 Information Augmentation

The wall-size display system and the steerable projectors though are to display information on the surface, in our system, they serve with different purposes. Being partners of the wall-size display system, the steerable projectors are not only to polish portions of the display, but can also be used to augment additional information of the content. For example, while the large display presents artifacts, the steerable projections can augment related information around the artifacts. Since the steerable projectors have higher intensity, their projection can help the viewers easily perceive the information. Besides, with the mobility of the steerable projections, the steerable projectors can extend the display wall by projecting augmentations outside the coverage of the display wall, making the displaying more flexible. Figure 7 shows that an artifact and its description are displayed on the wall. In this figure, the description occupies space too large to be contained in the display wall. The steerable projection extends the display wall to show the full description.

2.3 Stepping User Interface Interacting with Display Wall

As we show in the scenario, the display wall and the steerable projections work together to provide stepping interfaces to the viewers. So that Susan could answer a question from the display wall by simply moving her foot. To provide everywhere stepping user interface, a steerable projector coupled with a camera can create an active region around the viewers. The viewers interact with the large display by stepping on widgets within the active region. Claudio Pinhanez [6] had demonstrated that the steerable projection to the environment can be versatile, creating ubiquitous interfaces for its users interacting with the environment. In this case, we show such interactions can be applied to offer direct and intuitive interactions between the display wall and the viewers.

3 System Design

The Flexible Display consists of a wall-size display system, steerable projectors, cameras, and personal handhelds. We may divide our system in three components: calibration of wall-size display system, calibration of steerable projectors regarding to the wall-size display system, i.e., the display wall, and the environment, and the interaction approaches. In the following, we describe each component and the required techniques.

3.1 Calibration of the Wall-Size Display System

While creating one large multi-projector display, we primarily have to solve two problems: geometric misalignment and photometric variations.

For the geometric problem, in the past, most relative research uses a fixed-lens camera to do geometric calibration [1,2]. However, their method requires that the entire projected area has to be completely visible by the fixed-lens cameras. The calibration may further degrade the accuracy of the measurements when the projected area becomes large. In order to increase calibration accuracy, here we utilized a technique similar to the technique adopted by Chen et al. [3] to increase the measurement resolution by combining several zoom-in images acquired by a pan-tilt-zoom camera. In addition, the zoom-in camera views have lens distortion that will increase the calibration error so lens correction is needed for all the zoom-in images.

For the photometric variation problems, readers can refer to the survey paper [11], where color variation in multi-projector displays has been classified clearly. Moreover, the projectors used to build a multi-projector display are generally video projectors because video projectors can provide more-accurate color representation. Therefore all projectors are considered to be video projectors here.

When we create the multi-projector display, there are obvious seams caused by the different chromatic response among projectors. For example, the display shows severe color differences when simply displaying pure red, green or blue due to the non-identical color gamut of each projector. Therefore in order to guarantee that any input-RGB color looks exactly the same over the display region, the common color gamut of the display should be determined. In addition, the non-linear behavior of the display and projectors also has to be considered and all input-RGB values have to be adjusted accordingly.

Another photometric problem is the non-uniformity of luminance over the display region. For example, the luminance on overlapped region is very noticeable obvious and luminance variation spatially inside a projector itself can also distract an audience from watching the display. Thus, in order to seamlessly merge the images without any obvious luminance non-uniformity, a luminance adjustment technique similar to [10] is used to smoothly combine the images into one large picture.

Therefore with regard to the two photometric variation problems described above, we utilized a two-phase photometric calibration method to not only solve chrominance variation problem among projectors but also to reduce perceptually the luminance variation over the tiled display. Figure 2 shows an image projected by the wall-size display system after geometric and photometric calibrations. More details can be found in our previous work [12].

(a) without calibration

(b) with geometric and photometric calibration

Fig. 2. Virtual Museum Projected from the Fixed Projector System

3.2 Calibration of Steerable Projectors Regarding to the Display Wall and the Environment

The Flexible Display includes at least one steerable projector to assist in both presentation and interaction. We show the steerable projector prototype in figure 3. For the presentation, the steerable projectors act on sub-region enhancement, and information augmentation. For the interaction, they provide stepping user interfaces for the viewers. Both of the two purposes require calibration of the steerable projectors regarding to the display wall or to the environment.

After completing the calibration of the wall-size display system described above, we now have large display coverage on the wall. In case of the steerable projectors overlaying their projections onto the display wall, the projected imageries needs to fine match the display wall. For this purpose, we need to find out the relationship between the steerable projectors and the display wall. Once the relationship is defined, we let the steerable projectors pre-warp the images before the images are projected.

Since the display wall and wall-size display system are calibrated, we can simplify the problem as illustrated in figure 4. Our goal now is to obtain the homographies between the image planes of the steerable projectors and display wall. To minimize the human intervention, we use a pan-tilt-zoom camera to help automate the calibration process. For each steerable projector projecting to a portion of the display wall, we describe four steps required to obtain undistorted projections: First, we let the wall-size display system project a grid pattern and use the camera to capture the pattern for computing the homography H_{CD} between the camera view and the wall-size display. We then identify the cross points of the grid patters in the camera view. These cross points are considered as the correspondences between camera view and

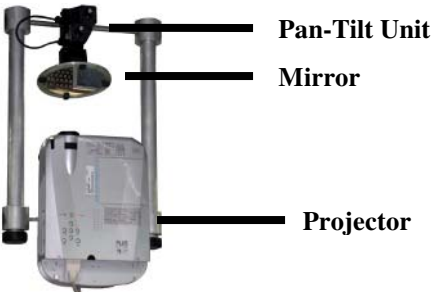


Fig. 3. Steerable Projector (This is a device composed of a projector and a pan-tilt unit mounted a mirror in front of the projector. We can change the projection area by rotating the mirror.)

image plane of the wall-size display system. Notice that at least four correspondences are required to determine the homography. In order to obtain robust results, as many as possible correspondences are used in our implementation. Next, the steerable projector displays another grid pattern. The same process is carried out to find the homography H_{CS} between the camera and the steerable projector. Therefore we can derive the homography H_{SD} between the steerable projector and the wall-size display system from $H_{SD} = H_{CS}^{-1} H_{CD}$.

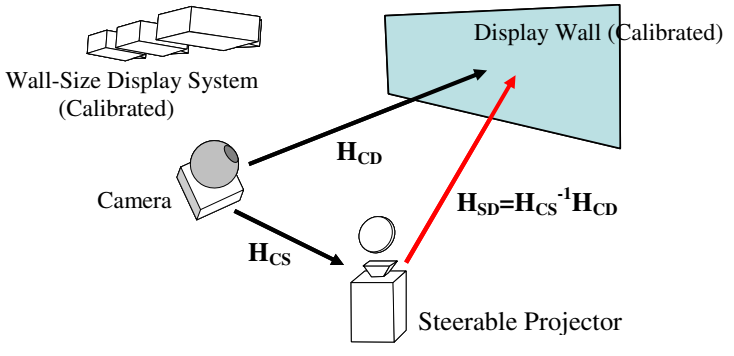


Fig. 4. The transformations among the steerable projector, the camera, and the fixed projector system

Calibration process described above is aimed at a dedicated pose of the mirror said a dedicated position of the projection of the steerable projector. In order that the steerable projection is able to correctly project over the whole display wall, we repeat the process to determine different homographies for different display regions in the calibration stage.

For the steerable projectors projecting onto the environment, the calibration of the steerable projectors regarding to the environment is required in order to deliver undistorted images for the users.

For the steerable projector which is required to project onto the environment, we first move its steerable projection to cover the area, say a target surface of the floor,

where requires undistorted projection. Here we develop an interactive tool to help the users performing calibration. First, a rectangle pattern is displayed by the steerable projector. Since the steerable projector is not calibrated with regard to the target surface, the projected pattern appears distorted, may be any quadrangle. Next, through the tool, the user iteratively drags the corners of the quadrangle to reshape it as a rectangle. The homography between image plane of the steerable projector and the target surface is then determined, according to the transformation between the origin pattern and the modified pattern.

3.3 Interactions of the Flexible Display

We provide the users two kinds of interaction mentioned in system functions section. The first interaction is stepping user interface. While the steerable projector brings widgets around the user, a pan-tilt-zoom camera mounted on upper-side of the display wall then starts monitoring whether the user is stepping on the widget, thus to launch reactions to the display wall. The detection algorithm is mainly differencing operations on image sequences followed by morphological operations to identify a moving object over the widget.

Another interaction, the user accesses the display through the handhelds. We use the PDA, iPAQ hx4700, in the current implementation. During visiting in the environment, the PDA constantly searches nearby Bluetooth access points offered by the display wall. Once the connection built, the PDA offers two modes to interact with the display wall. The first mode, the PDA is a program selector of the display wall. The PDA receives the content program from the display wall and shows it on the screen, the user can drag and throw a certain object to the display wall. The second mode, the PDA can be a touch pad for moving steerable projection over the display wall. The user drags by a stylus on the PDA to freely move the projection, facilitating the high-resolution quality provided by it.

4 Experimental Results

In the work of [9], we presented a stereoscopic kiosk for virtual museum, which combines techniques of stereoscopic displays, image-based rendering, virtual reality, and augmented reality. The virtual museum contains 3D scenes implemented by using the image-based technique and by using the model-based technique. For the 3D scenes constructed by geometric models, viewers can interactively view the virtual world from arbitrary viewing directions, but for that built as panoramas, viewers can only watch from some specific viewpoints. In general, the exhibition space implemented by the image-based approach appears to be more realistic. The artifacts are presented as object movies in the virtual exhibition — the image of the artifact shown on screen is selected according to the viewing direction of the user. In this way, artifacts can be rotated and moved in 3D.

In the implementation, we redistribute the virtual museum to the Flexible Display in our laboratory. Flexible Display not only gives its viewers wall-size imagery

perception that a normal kiosk can not do, but richer interactions that the viewers can better experience with the virtual artifacts. Figure 5 shows a shot of the wall-size display system and the steerable projector in our laboratory. The wall-size display system composing of three projectors projects large scale display coverage on the wall. The steerable projection, at the same time, highlights the artifact placed in the center of the display wall.

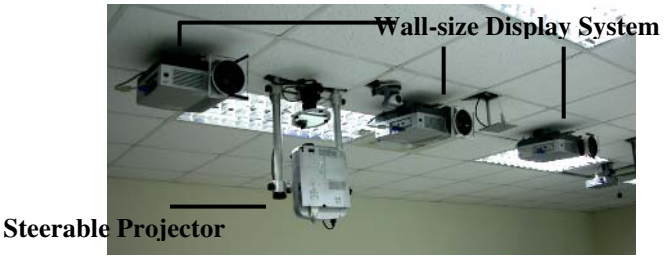


Fig. 5. The Experimental Devices

Figure 6(a) shows the five icons, projected by the steerable projector, around the viewer's foot. Each icon stands for an artifact in this scene. The viewer steps on one of them, the display wall then present the corresponding one on the wall. The viewer steps again to perceive a enlarge mode of the same artifact, as shown in figure 6(b).

(a) (b)

Fig. 6. The viewer Steps on "Stepping User Interface" to perceive an artifact on the display wall

While in the enlarge mode, the viewer sees two icons standing for instructions "Return" and "Information" projected around the viewer's foot. The steerable projector will augment information over the artifact if the viewer steps on the information icon. Figure 7 shows the full description of the artifact augmented over the display.

To enhance portions of the artifact, the viewer uses the PDA to move the steerable projection. Figure 8 shows the enhanced area presents more details to its viewers. The words in the steerable projection area are clearer the those outside the area.

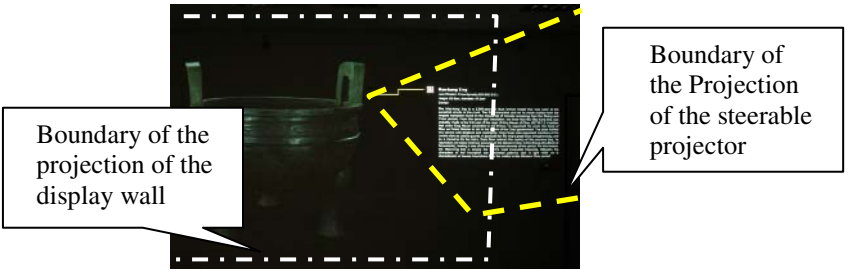


Fig. 7. Extending the Display (The steerable projector extends the display by augmenting the description of the artifact partially outside the display wall. The dash lines on the figure indicate the projection boundaries of the fixed projector system and the steerable projector).

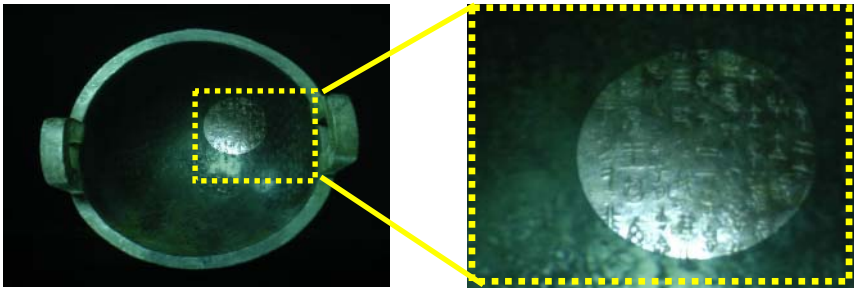


Fig. 8. Enhancing the Display (a)The steerable projection enhances a part of the artifact on the display wall. (b) The zoom-in view shows that the area enhanced by steerable projection are clearer than those outside the enhanced area.

5 Conclusion

In this paper, we are investigating how a wall-size display wall and one steerable projector coexisted in the environment can gain benefits from each other. We have developed Flexible Display composing of a wall-size display system, a steerable projector, a camera and PDAs, as well as its three basic functions: sub-region enhancement, information augmentation and stepping user interface. A virtual museum applying Flexible Display is carried out in our laboratory to demonstrate the better capabilities on both the presentation and interaction.

Acknowledgement

This work was supported in part by the grants of NSC 94-2422-H-002-019 and NSC 94-2752-E-002-007-PAE.

References

1. R. Raskar, M. S. Brown, R. Yang, W.-C. Chen, G. Welch, H. Towles, B. Seales, and H. Fuch.: Multi-Projector Displays Using Camera-Based Registration. *Proc. of IEEE Visualization*, 1999, pp. 161-68.

2. R. Sukthankar, R. G. Stockton, and M. D. Mullin.: Smarter Presentations: Exploiting Homography in Camera-Projector Systems. *Proc. of ICCV 2001*, 2001.
3. H. Chen, R. Sukthankar, G. Wallace, T.-J. Cham.: Calibrating Scalable Multi-Projector Displays Using Camera Homography Trees. *CVPR Technical Sketch*, 2001.
4. R. Raskar, J. V. Baar, and J. X. Chai.: A Low-Cost Project Mosaic with Fast Registration. *Proc. of Asian Conference on Computer Vision* (ACCV2002), 2002.
5. R. Raskar, J. V. Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines.: iLamps: Geometrically Aware and Self-Configuring Projectors. *ACM Transactions on Graphics*, Vol.22 No.3, p.809-818, July 2003.
6. C. Pinhanez, R. Kjeldsen, A. Levas, G. Pingali, M. Podlaseck, N. Sukaviriya.: Applications of Steerable Projector-Camera Systems. *IEEE Workshop on Projector-Camera Systems* (Procams 2003), 2003.
7. G. Pingali, C. Pinhanez, A. Levas, R. Kjeldsen, M. Podlaseck, H. Chen, N. Sukaviriya.: Steerable Interfaces for Pervasive Computing Spaces. *IEEE Conference on Pervasive Computing and Communications* (Percom 2003), pp. 315.
8. Y. P. Tsai, Y. N. Wu, and Y. P. Hung.: Generating a Multiresolution Display by Integrating Multiple Projectors. *IEEE Workshop on Projector-Camera Systems* (Procams 2003), 2003.
9. W. Y. Lo, Y. P. Tsai, C. W. Chen, Y. P. Hung.: Stereoscopic Kiosk for Virtual Museum. *Proceedings International Computer Symposium* (ICS 2004), December 2004.
10. A. Majumder, R. Stevens.: Perceptual photometric seamlessness in projector-based tiled displays. *ACM Transactions on Graphics*, Vol.24 No.1, p.118-139, January 2005
11. M. Brown, A. Majumder, and R. Yang.: Camera-Based Calibration Techniques for Seamless Multiprojector Displays. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 11, No. 2, March-April, 2005.
12. Y. P. Tsai, S. C. Liao, Y. P. Hung and Z. C. Shih.: Two-Phase photometric calibration for multi-projector displays. *Third Taiwanese-French Conference on Information Technology*, 2006.

Design and Implementation of a Smart Tag System for IT-Based Port Logistics

Hyuntae Cho, Hoon Choi, Woonghyun Lee,
Yeonsu Jung, and Yunju Baek

Department of Computer Science and Engineering,
Pusan National University,
Busan, Republic of Korea
marine@pnu.edu,
{hara, woonga, rookie}@juno.cs.pusan.ac.kr,
yunju@pnu.edu

Abstract. Logistics has become a fast growing industry in recent years. In particular, the large hub ports have heavy congestions in getting the containers in and out of the ports. Therefore the ports are regarded as the bottleneck in the logistics. In this paper, we analyze the new requirements of port logistics to figure out heavy congestions and then present the system prototype for identification and positioning or tracking in the port environments. Our solution is composed of three parts: RFID, electronic container seal, and RTLS system. The system design focuses on three parts: 1) the standard compliance with the ISO, 2) the energy saving mechanism to maintain the longevity of tags as long as possible, and 3) the high identification rate in the presence of multiple tags. The performance evaluation of the system is also included in the paper. These systems will provide a complete range of port logistics services.

1 Introduction

Ports have been the gateway to extended markets and have in that sense always been an important part of the global logistics chain. Furthermore, the large hub ports have heavy congestions in getting the containers in and out of the ports. Therefore the ports are currently regarded as the bottleneck in the total supply chain. By enhancing the efficiency of the port, we can mitigate an accumulation of goods. The traditional approach to enhance the efficiency of ports has been to make large capital investments, either new machines or hiring more labor. Another approach is to use pervasive computing which is applauded from a lot of experts in recent. Pervasive computing systems aim to provide service specific to the user's context or location. Especially, RFID devices are often used to link a user with a location or a context for proximity. In order to improve the efficiency of ports, we can extend and apply an active RFID tag[1] [2], which has a great transmission range, to port environments. To apply the RFID system to the port environment, some basic challenges have to be overcome. In the port, the most important issue is to manage containers which are deployed in the yard.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 32–43, 2006.

The port manager or management system can easily collect the information of container that includes container ID, status, location, trajectory, and whether container is opened or not. In this paper, we designed and developed a smart system to take into account these requirements. In case research on container logistics fuses into the active smart tag technology, studies need to cope with next generation active smart tags such as electronic container seal and RTLS(real time locating system)[3]. Our system includes three parts: the active RFID tag, the electronic container seal, and the RTLS system. This smart tag we develop has a great advantage for low-power, light weight, and fault tolerance. So, various applications can be developed by adding basic requirement such security and localization for port logistics

In order to enhance the efficiency of ports, this paper will concentrate on the design and implementation of the systems we named LITeTag(Logistics Information Technology electronic Tag). The organization of the paper is as following. We present international standards and related works in the next section. Then, we describe the development of our systems operating at 433MHz in section 3. Before concluding this paper, we verify our systems and describe the performance evaluation of the systems in section 4 and 5.

2 Related Work

Standards for RFID systems are defined by ISO/IEC. 15961[4] addresses the tag commands, 15962[5] depicts the data syntax, and 19799 demonstrates about API. In addition, 18000 is intended to address air interface. 18000 consists of the following parts: part 1, for reference architecture and definition of parameters to be standardized, part 2, for below 135kHz, part 3, for 13.56MHz, part 4, for 2.45GHz, part 6, for 860∼960MHz and part 7, for 433MHz[6]. These standards specify the operating frequency, modulation and coding schemes, anti-collision routines and communication protocols.

ISO/IEC 18185 international standard provides a system and a reference for the identification and presentation of information about freight container electronic seals. The identification system provides an unambiguous unique identification of the container seal, its status, and related information. The presentation of this information is provided through a radio-communication interface providing seal identification and a method to determine whether a freight container's seal has been opened. The physical layer, ISO/IEC 18185-7, of an electronic seal compliant with ISO/IEC 18185 standard shall be in accordance with ISO/IEC 18000, Part 7. ISO/IEC 18185-1[7] standard specifies the communication protocol a freight container seal identification system, with an associated system for verifying the accuracy of use, having a seal status identification system, a battery status indicator, and a unique seal identifier including the identification of the manufacturer.

ISO/IEC 24730-3[8], being enacted as an international standard, defines RTLS system operating at 433MHz radio frequency. Also, ANSI 371 dictates RTLS system, where part 1[9] defines 2.4GHz, part 2[10] intends to address 433MHz, and part 3[11] demonstrates API. These standards define the air interface protocol

for a location system that provides zone location information utilizing a device that generates a radio frequency transmission that can be received at a minimum of 100 meters open-field separation between the transmitter and receiver and is fully compliant with radio frequency regulatory requirements. The 433MHz systems utilizing the air interface protocol defined by ANSI 371-2 document we focus on will allow the user to manage assets within the infrastructure of the system on a real-time basis. The average location data provided by the system are within 10 meters of the actual location of the transmitting device. RTLS system consists of servers, tags, and readers. For RTLS system, location ranging technique can be performed by a TDOA(Time Difference of Arrival)/ROA(RSSI on Arrival) system.

3 Smart Tag for IT-Based Logistics

In this section, we design and implement a new system platform compliant with international standards for port logistics. Our system is composed of three parts: LITeTag, called the active RFID, LITeSeal, called the electronic container seal, and LITeRTLS, called the RTLS system. Basically, our systems operate at 433MHz frequency radio and have a great communication range. LITeTags can be attached to containers and pallets, and comply with ISO/IEC 18000-7 international standard. LITeSeal can guarantee status and security of objects or container and complies with ISO/IEC 18185-1/7 international standard. And, LITeRTLS system is included for an extended application.

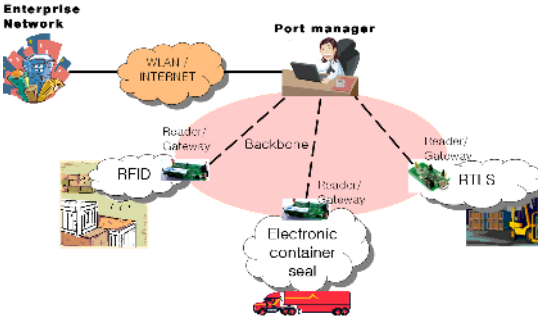


Fig. 1. Architecture of our system that is composed of three parts: RFID, ECS, RTLS. Port manager or user can collect lots of information from each system.

3.1 LITeTag for Port Logistics

LITeTag is an active RFID technology to identify containers or pallets in port logistics. In this section, we present the design and implementation of hardware and software for an active RFID, which complies with the part 7 of ISO/IEC 18000, operating at 433.92MHz.

Hardware implementation of LITeTag. The RFID platform is composed of two principal components: a tag, referred to as a transponder, and a reader, referred to as an interrogator. These two components are powered by 3V AA batteries attached to the bottom of the board. The tag consists of a processor, an RF transceiver, and an RTC(real time clock)[14]. The processor can completely control the radio through the SPI interface, and transmit/receive data using the processor's port D. To make it easy to change the operating mode, we use a RF switch through port B. In addition, to visualize the RF communication between the reader and the tag, three LED are added. The reader basically contains a radio frequency module(transceiver), a microcontroller unit, an antenna and the RS-232 interface to the host system. To supply higher voltage to the LCD module, LT1302[15], that converts 3V to 5V DC supply, is used. In addition, an expansion connector is added for interfacing with sensors.

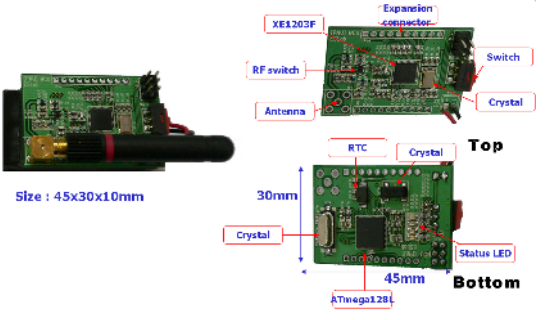


Fig. 2. Hardware components that shows the hardware components of the LITeTag and module names. The system is commercial off-the-shelf products.

Atmel's Atmega128L[12] was chosen as the processing unit of our platform. It is able to operate at a maximum frequency of 8MHz, providing reasonable processing power to explore a wide variety of applications. The Atmega128L processor provides sufficient memory resources for a wide range of experiments. The on-chip memory includes 4KB of RAM, 4KB of EEPROM, and 128KB of flash memory. 53 general purpose I/O pins and serial ports, such as RS-232 and SPI, are provided by the CPU.

The communication subsystem of our LITeTag is based on the XEMICS's XE1203F radio[13], which is connected to the processor through an SPI interface and data bus. The XE1203F is 433, 868, 915MHz compliant single-chip RF transceiver, which is designed to provide a fully functional multi-channel FSK communication. It has an effective data rate of 153.2kbps, making it ideally suited for applications where high data rates are required. The processor can completely turn off the radio or simply put it in sleep mode through the SPI interface, while the XE1203F can wake up the processor when an RF message has been successfully received. In addition, the radio chip provides a variety of 4 different output power levels that can be used for transmission. Output power is applicable according to the application. The power consumption of the radio during transmission heavily depends on the output power level used.

Software implementation of LITeTag. For ISO 18000-7 implementation, we need to solve some basic problems. Firstly, the standard requires Manchester encoding but XE1203F provide NRZ. So, we solve this difficulty by using asynchronous mode in terms of software. This is completed in transmitter. Second, there are timing difficulties in the receiver. A receiver can detect encoded data sent by transmitter. The receiver needs to detect exact timing of data. For example, the preamble has 20 pulses. Each pulse is 30us high and 30us low signal. But, checking the exact timing in the signal of the receiver is difficult because our system use the asynchronous technique. To solve the problem, we use a sampling mechanism. In detail, the receiver waits for until the microcontroller detects a high signal demodulated by RFIC. The microcontroller samples at inter-val of every 30us. If the preamble detection is successfully completed, it will check sync bit. If sync bit would also be identified successfully, data format detection will be started. All the detected data will be transfer to upper layer. If the detection is failed, the collision is occurred and the tag discards the packet.

Software for the active RFID system is separated into three types: a host program, a reader program, and a tag program. The host system controls the data flow between the reader and tags. It can be as simple as a personal computer connected to the reader by an RS-232 serial cable. More complex systems are possible where there are many readers in different locations and data is transferred to host servers through LANs or even over the Internet. We will deal with this problem in the near future. The host receives information from user, analyzes the command, generates packets, and then sends them to the reader system.

The communication between the reader and the tag is of master-slave type, where the reader always initiates communication and listens for response from a tag. Upon receiving data from the host program, the reader classifies data into commands and data, and then generates a new packet. Subsequently, CRC-CCITT[16] 2 bytes are appended to verify the data in tags. The complete packet is sent through radio frequency after wake-up signal is sent. As the RF subsystem transmits data by FSK modulating, we encode each bit with the Manchester code. The UART has to be initialized before any communication take place. The initialization process normally consists of setting the baud rate, setting frame format and enabling the Tx or Rx depending on the usage. Although baud rate is possible up to 250kbps using double speed operation, we just use 9.6kbps baud rate. Then the reader waits for a response message from the tags. The received message will be transmitted to the host.

Figure 3 depicts the operation of the tag. When the tags placed within the reader RF communication range receive a wake-up signal that is broadcast by the reader, they will move into the active state. The selection is determined by a random number generator. On receiving a collection command, tags will detect by checking invalid CRC. If a tag received with valid CRC, the tag will select their slot and respond with messages to reader. In addition, in order to store a tag's unique ID, we use 4K bytes of data EEPROM memory. We also add a simple security mechanism to prevent access-ing by malicious users.

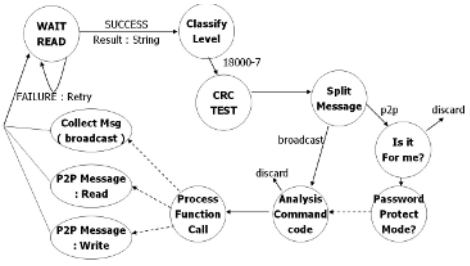


Fig. 3. State diagram for the tag system, LITeTag

Basically, the message type between the reader and the tag has two different formats: broadcast message and point-to-point message. P2P message format, which includes all commands except collection commands, requires a tag ID in order to access a particular tag. P2P message is one of sleep, status, user ID length, user ID, owner ID, firmware revision, model number, read/write memory, set password, password protect, and unlock commands. The broadcast commands are used to collect tag IDs, user IDs or short blocks of data from the selected group of tags using a batch collection algorithm. Broadcast commands are used for tag ID collection within reader RF communication range.

There are a number of tags in an RFID field. If some tags try to transmit their signal at the same time, collision can occur in the reader. So, the reader can not complete their collection command. We need to consider these collisions. To figure out the collision problem, we used a dynamic framed slotted ALOHA[17]. In the slotted ALOHA, response is performed in a frame. In the frame, each tag randomly selects their time slot where data transmission is performed. Then, the number of slots is usually predefined by administrator. However, this solution is bad when two or more heterogeneous systems co-exist. So, we thought that each node compute their time slot to transmit data. Each time slot is the time of preamble signal plus the time of data to be transferred. Preamble signal is twenty thirty high pulse and thirty low pulses, and forty two high and fifty four low signals. The data type and total time are determined according to reader command. In each frame, if data is successfully delivered to reader, the reader send sleep command to tags, in which transmission is completed, using point to point command after each frame. Otherwise, if data transmission is failed, the reader re-broadcast collection without wake-up signal in the next round. So, only residual tags by collision can response to reader. This action is performed until 3 consequent rounds are empty.

Finally, because the active RFID tag is powered by internal energy, the lifetime of battery is critical. So the tag requires some power saving mechanism. The tag is in the sleep mode initially. The tag will enter the active state, when the tags placed within the reader RF communication range receive wake-up signal that is broadcast by the reader. After data processing, the tag that completed the data processing enters sleep mode.

3.2 LITeSeal for IT-Based Port Logistics

The physical layer of an electronic seal compliant with ISO/IEC 18185 standard shall be in accordance with ISO/IEC 18000, Part 7. So, we reuse LITeTag system for LITeSeal system. LITeSeal consists of three parts: a core board included a MCU and a RFM, a mechanical seal, and a sensor board. The core board is the same as the LITeTag, which consists of a microprocessor and a RF module. The mechanical seal is designed for detecting the seal condition. If seal is broken, LITeSeal detects as broken state. Also, CdS (cadmium sulfide cell) is attached at the end of the mechanical seal. If seal is exposure to light, LITeSeal detects as seal is broken. The outward and the block diagram of LITeSeal are depicted in Figure 4. Furthermore, a humidity and temperature sensor, a Piezzo ceramic buzzer, and an expansion connector are added for ex-tended applications. Especially, the buzzer will alert the broken condition of seal as well as real time RF reporting. LITeSeal maintains its history of the event using RTC, and some sensors since sealed. The mandatory data of the electronic seal includes seal

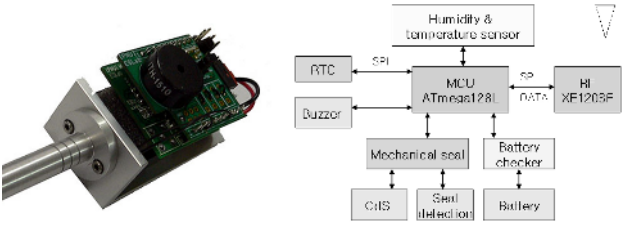


Fig. 4. Architecture of electronic container seal, LITeSeal

status, seal tag ID and manufacturer ID, data/time for sealing and opening, seal status, low battery status, protocol ID, model ID, product version, and protocol version. Seal status is stored in EEPROM with date and time information. The message type between the reader and the seal has three different formats: broadcast message, point-to-point message, and alert message. P2P message is one of sleep, product version, model ID, read RTC, read seal product parameter, standby, read event records, or get the status of the seal. Broadcast message is one of collection, collect seal IDs with event record, or sleep all but message. LITeSeal system is always not Master/Slave communication. In order to alert the broken condition of the seal to the reader, the electronic container seal can initiate communication using the alert message. The advantage of the use of electronic container seal system is multi-fold. It would enhance productivity by reducing manpower needed to check the movement of containers and inspection of seals. It helps to enhance security as with electronic container seal, any tampering could be easily detected.

3.3 LITeRTLS System for IT-Based Logistics

RTLS is applicable for both TDOA and ROA location methodologies. In particular, a ROA system is a very simple, very low-cost RTLS system that will provide location with a range of resolution within the covered space varying from 10 meters to 100 meters, depending on the location of the tag with respect to the readers. In many situations, a simple technique such as this can offer what most applications truly need without the complexity and cost of much more sophisticated techniques.

LITeRTLS is composed of a reader having USB interface to connect with RTLS server and a tag, called the RTLS transmitter. LITeTag is used for the RTLS tag without any modification. However, a problem exists in the RTLS reader. XE1203F RFIC we used only provides 3bits RSSI information in the receiver, which is the lack of resolution to estimate the tag's location. To improve tag location accuracy, we used an external RSSI approach[18], using IAMP and QAMP. The algorithm is as following.

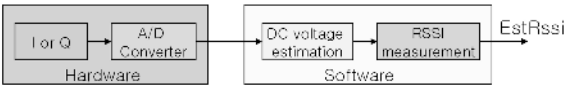


Fig. 5. Algorithm for an external RSSI

EstRssi relates to the level of the signal at the antenna, so depending on the application; one can calculate the real power of the signal by applying the formula or works in relative. The basic data format, implemented in software, is the same as 18000-7. RTLS broadcast blink messages using this data format periodically. Our system will depend on blink interval. If blink interval is long, the longevity of RTLS tag is prolonged. Otherwise, the real time tracking of the objects is possible. This is a trade-off. That is it depends on the application.

Our RTLS system is basically composed of 4 readers in short range. The reader receive the signal from transmitter and calculate RSS(receive signal strength), and then send that to the RTLS server. The server can estimate the location of the transmitter using RSS received from RTLS readers. These basic RTLS systems have the short coverage to detect many transmitters that distributed through a wide of the port. For extending the coverage of RTLS system in the port, many RTLS reader can be added in the reader network, which attempts to form a grid type. Then, the higher the scalability of readers is, the further the reader is from the server. Also the number of readers connected to the server is limited. In order to figure out some limitation each reader can communicate with its neighbor using radio and delivery the received transmitter information to server by multi-hop manner. The intermediate reader collects and aggregates data from its neighbors, and then forwards the compressed data to the RTLS server. Because simple broadcast are used for communication between readers, sub-problems, such as the slot allocation and time synchronization, we have to figure out exist. These sub-problems are considered, when realistic RTLS system, our future work, system implement.

4 Performance Evaluation

In this section, we report on some results from a preliminary performance evaluation of our system, LITeTag. Our goals in conducting this evaluation study were two-fold. First, because tags can use up their limited supply of energy simply performing computations and transmitting information in a wireless environment, energy conserving forms of communication and computation are essential. The other is multi-tag collection. Collecting multiple co-located tags within communication range is difficult and often unreliable because collisions may be detected.

In this paper, for performance evaluation we made a testbed in the port, South Korea and experiment the realistic application for the IT-based port logistics. The port we tried to test is enhanced as the hub for supply chain in recent years. Annual processing capability of the port is increasing. Thus, the port manger has to consider the method to process containers efficiently.

4.1 Energy Budget

The sleep mode enables the application to shut down the processor of unused modules, thereby saving power. In the RF part, possible states are one of transmitter, receiver, or sleep mode. Additionally, an extra state, called standby state, is added for state transitions and the initial state. Table 1 shows the different power consumptions according to the operation mode. Our system runs on a pair of AAA batteries, with a typical capacity of 2.5 ampere hour (Ah). We make a conservative estimate that the batteries will be able to supply 2.5Ah at 3V. Assuming the system will operate uniformly over the deployment period, each tag has 0.303876mA per day available for use. 2 AAA(2500mAh) battery allows us to use for 1 year

Table 1. Power consumption in the system

	State	Condition	Typical	Max.	Unit
MCU	Active	all		5.5	mA
	Sleep	WDT	<15	25	mA
RF	Tx	5dBm	33	40	mA
	Rx		14	17	mA
	Standby	Osc.	0.85	1.1	mA
	Sleep		0.2	1	mA

4.2 Identifying Multiple Tags and Measuring RSS

Collecting multiple tags is separate into two groups. One is to identify only one tag periodically. The other is to collect many tags at the same time. For identifying a tag periodically, on the one hand, we have to mention several assumptions. First, the time interval is a period between a point of time when the previous message was sent completely and the start time of next message. Second, the

length of message is 50bytes except for the preamble message, and the number of messages is limited to be thousand. For this experiment, the time intervals are 51.125usec, 101.125usec, 1msec, 10msec and 30msec. The result of performance evaluation shows great performance shown as Figure 6. Figure 6 (a) shows that successful transmission rate is sufficiently high(99% or more) whichever time interval is selected.

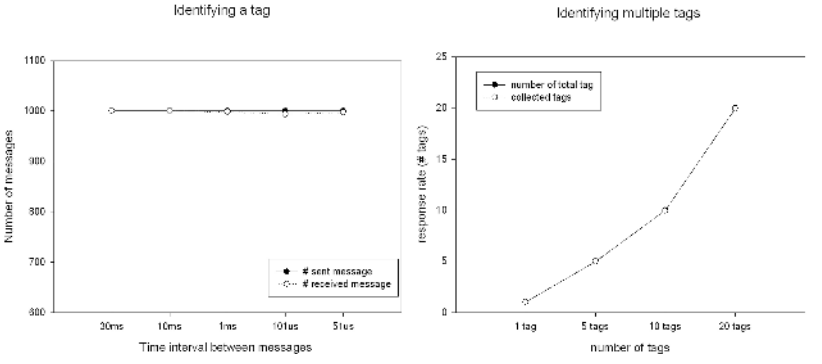


Fig. 6. Success rate all tags collect. (a) Identifying one tag (b) Identifying multiple tags.

In addition, we performed the other evaluation for identifying multiple tags, which are deployed to RFID fields shown as Figure 6(b). To collect multiple tags, the reader broadcast a collection command, which is read by the tags. Then, the collision arbitration sequence during tag collection is performed for an efficient and orderly collection of the tags placed within the reader communication range and to receive information on the tag. We use DFSA(Dynamic Frame Slotted ALOHA) mechanism for collision arbitration. The windows size is set to be 57.3ms. A collection round consists of a number of slots we set as 6. Each slot is long enough for the reader to receive a tag response message. The actual duration of a slot is determined by the reader collection command type and is a function of the tag transmission time. All experiments are repeated 100 times. Each experiment is performed by separating the test model into 1, 5, 10, and 20 in terms of the number of tags. As earlier mentioned, miss rate in identifying items is important because that is directly related with owner's property. Figure 6 (b) illustrates the success rate to collect all the tags. In the figure the response rates are 99.5% or more. This figure shows that it is sufficient to deploy the realistic application for the port logistics. Every tag should select their slot in the frame or round. The number of response tags collected per each round is shown in Figure 7 (a). In case 10 tags are deployed in the RFID field, 3 or more tags are collected. In case of 20 tags, 4 or more tags are collected in each round.

Finally, for RTLS system using RSSI, basically RSS data we measured by our system are required. In order to measure the received signal strength, we test and get much information. Figure 7 (b) shows that RSS values are diminished according to the distance. This result can be useful for the locating system

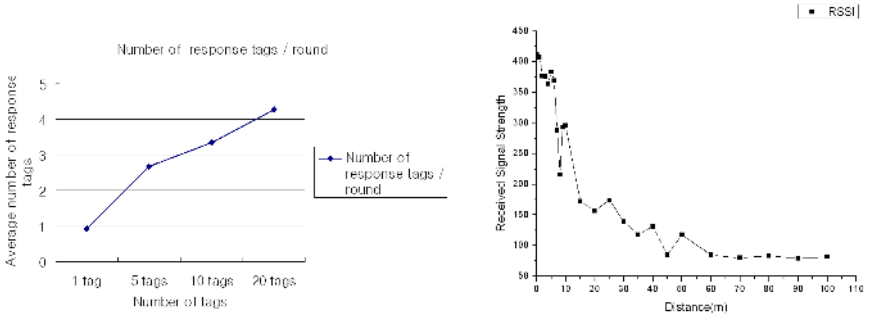


Fig. 7. (a) Response tags per each round (b) The RSS distribution in terms of dist

using 433MHz. Also, wire-less communication environment can be impacted by many external factors. Ensuring connectivity using RSS can provide the powerful communication link.

5 Conclusion

In this paper, we presented the LITeTag for the active RFID system, the LITeSeal for the electronic container seal, and the LITeRTLS tag for RTLS system. We describe the detailed design and implementation of our system platform. Our system has features such as high identification rate of multiple tags, reliable energy budget, and standard compliance with ISO/IEC or ANSI. Our systems for port logistics are effective for development and evaluation of prototype applications because of the flexibility of the design of both hardware and software. So, our platform will be suitable for versatile port management applications. The future works include as follows: 1) the investigation of cost reduction of the platform, and 2) the sophisticated design of the collision arbitration for multiple tag or interaction between different systems.

Acknowledgment. This work was supported by "Research Center for Logistics Information Technology (LIT)" hosted by the Ministry of Education & Human Resources Development in Korea.

References

1. Klaus Finkenzeller: RFID Handbook: fundamentals and applications in contactless smart cards and identification, Wiley press (2003)
2. Michael, K., McCathie, L.: The Pros and Cons of RFID in Supply Chain Management, International Conference on Mobile Business (2005)
3. John scott: WhereNet RTLS in Marine TerminalsRTLS, WhereNet (2001)
4. ISO/IEC 15961: Information technology - radio frequency identification (RFID) for item management - data protocol: application interface, ISO/IEC (2004)
5. ISO/IEC 15962: Information technology - radio frequency identification (RFID) for item management - data protocol: data encoding rules and logical memory functions, ISO/IEC (2004)

6. ISO/IEC 18000-7: Information technology - radio frequency identification for item management - Part 7: parameters for active air interface communications at 433 MHz, ISO/IEC (2004)
7. ISO/IEC 18185-1: Freight containers - Electronic seals - Part 1: Communication protocol, ISO/IEC (2004)
8. ISO/IEC 24730-3: Information technology - Automatic identification and data capture techniques - Real Time Locating Systems (RTLS) - Part 3: 433 MHz, ISO/IEC (2005)
9. ANSI INCITS 371-1: American National Standard for Information Technology - Real Time Locating Systems (RTLS) - Part 1: 2.4 GHz-Air Interface Protocol, INCITS (2003)
10. ANSI INCITS 371-2: American National Standard for Information Technology - Real Time Locating Systems (RTLS) - Part 2: 433-MHz Air Interface Protocol, INCITS (2003)
11. ANSI INCITS 371-3: American National Standard for Information Technology - Real Time Locating Systems (RTLS) - Part 3: Application Programming Interface, INCITS (2003)
12. Atmel: Atmega128(L) datasheet, http://www.atmel.com (2005)
13. XEMICS: XE1203F datasheet, http://www.xemics.com (2005)
14. RICOH: RTC Rx5C348A datasheet, http://www.ricoh.com (2003)
15. Linear technology: LT1302 datasheet, http://www.linear.com (2004)
16. B. Schneier: Applied Cryptography, John Wiley & sons, New York (1996)
17. Jae-Ryong Cha, Jae-Hyun Kim: Novel Anti-collision Algorithms for Fast Object Identification in RFID System, Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on Volume 2, 20-22 (2005) Page(s):63 - 67
18. SEMTECH: AN1200.01 Application note, http://www.semtech.com (2005)

A Smart Schoolbag System for Reminding Pupils of the Forgotten Items

Lei Jing[1], Noriko Yamamoto[1], Zixue Cheng[1],
Hui-Huang Hsu[2], and Tongjun Huang[1]

[1] Aizu-Wakamatsu, Fukushima-ken 965-8580 Japan
{d8071202, s1100225, z-cheng, t-huang}@u-aizu.ac.jp
[2] 151 Ying-chuan Road Tamsui,
Taipei County Taiwan 25137, Republic of China
hhsu@cs.tku.edu.tw

Abstract. In this research, a ubiquitous learning support system making use of the schoolbag is presented to assist elementary school pupils in their personal items management. Through some sensors, the micro-computer embedded in the schoolbag can monitor what is getting in or taking out of the schoolbag, and maintain a schoolbag item list. Moreover, the teacher can make up a schedule that specifies required items on a given day. The micro-computer then compares the schedule with the maintained item list and provides the reminding service for the pupils.

1 Introduction

To substitute the explicit input such as keyboard and mouse, ubiquitous computing research has been trying to enable the computer system to get information from the implicit input such as location, time and so on, so called context-aware. The focus of the ubiquitous learning is to provide valuable service to the learner in a future ubiquitous computing environment. It can be deemed as the combination of the ubiquitous computing and the e-learning. Since learners are human and no two learners are identical, except the usual context such as location, time mentioned above the ubiquitous learning emphasizes on providing personalized service according to the learner behavior.

In the paper, a ubiquitous learning support system making use of the schoolbag is developed. It is an illustrational application of the learner's context-aware. By using the RFID tag to identify each item and the infrared sensors to judge the item's in/out status, the micro-computer embedded in the schoolbag can record what is getting in or taking out of the schoolbag, and maintain a schoolbag items list. Moreover, the teacher can make up a schedule that specifies the required items on a given day. Then the micro-computer receives the schedule and provides the reminding service for the pupils according to the comparing result of the received schedule with the maintained item list.

"Reminding about Tagged Object using Passive RFIDs" is a system that can monitor personal items by using RFID, and it tells a user about missing items if a user forgets something when he/she arrives at a location [1]. The area that the system

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 44 – 50, 2006.

manages is a large range such as inside a house, but the system is not suited for managing narrow scope like inside a bag. "Build Your Own Bag" is a system that can monitor information of objects in the bag by RFID and some sensors, and in case that anything is missing, it can inform the user [2]. The system has good management of personal items because it is created in mind for general bags, but it has not taken learning environment into consideration. "Kids in Feel" is a system that manages information of pupil's commutation [3]. The system monitors pupil's commutation information by RFID and sends a message to pupil's parents by e-mail or web. This system is only for safety management, and assistance to learning is not supported. Lastly, "SHABERANDOSERU" is a system that monitors personal items information and manages personal items using a school bag with an RFID reader [4]. The system is necessary to improve how to attach an RFID reader and tags and to raise detection accuracy. The system of this research can monitor personal items information in a school bag.

The purpose of the paper is to introduce a schoolbag system which can manage pupil's personal items such as textbook, note, pencil box and so on. The following problems have to be solved. The first one is how to monitor what is in/out the schoolbag. The second one is how to know which items are forgotten. And the third one is how to remind the pupil.

The rest of the paper is organized as follows. In Section 2, a model of the system is shown, and the problems are defined. In Section 3, our method is presented in details. In Section 4, implementation of our method is described and the effect of our system is shown by experiment. Finally, in Section 5, conclusion and future work are presented.

2 Model and the Problems

Figure 1 has a schematic for the model of the schoolbag system. In this research, we consider the system that can monitor the pupil's personal items and in case there are any forgotten items the system can manage to remind the pupil. The judgment on whether any items are forgotten is made on a time period set by user called Warning Time. For example, we can point morning 7:00 to 8:00 is the Warning Time, then during each day's 7:00 to 8:00, if any items status on the current schedule are "out" which means the required item is out of the schoolbag, then the system will try to notice the student by displaying some warning message on the LCD displayer and send email to parents' cell phone. A micro-computer in the schoolbag with a RF-ID reader and infrared sensors reads information of a personal item with a tag which will be discussed concretely in section 3.

There are three key points to realize this model. The first one is how to attach an RFID reader and infrared sensors into a schoolbag and tags onto each personal item to monitor information of personal items correctly. If an RFID reader reads a personal item with a tag after it is put into the schoolbag, accuracy of monitoring information of personal items decrease because the tag's radio wave is interfered with some books. Therefore, in this research, a reader reads tag's information when pupil takes personal items in and out of the schoolbag. The second one is how to know which items have been forgotten. It pre-requests that the system know what need for tomorrow's classes. So we need a schedule specify a list of items.

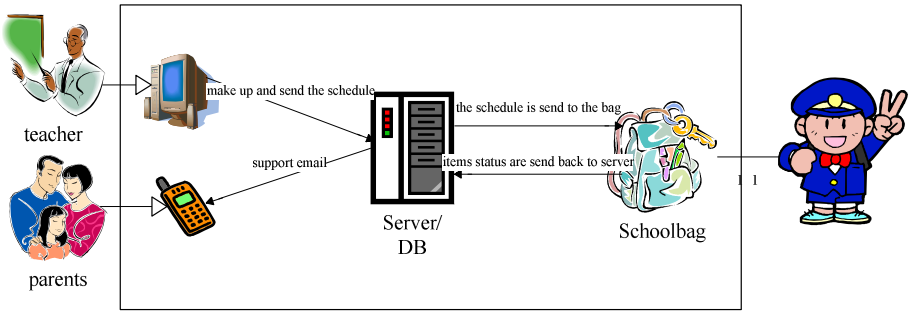


Fig. 1. The Model of the Ubiquitous Learning Support System

This schedule can be made up by the teacher or the parents. To simplify the problem, we consider that only the teacher can make the schedule. Then the items list maintained by the micro-computer will be compared with the schedule and conclude about what have been forgotten. The third point is how to remind the pupils. In fact, only by making using of the schoolbag, it proves difficult to remind the pupils at proper time in a proper way. If in a ubiquitous computing environment, we can provide the reminding service in different ways based on different situations. In this paper, three simple methods are adopted to remind the pupils such as during the Warning Time, the schoolbag can display a warning message on the LCD, send a buzzer call, or send an email to parents mobile phone.

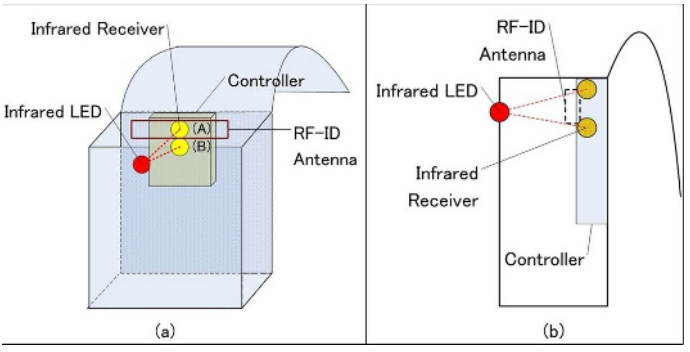
3 A System Can Remind the Forgotten Items

3.1 Placement of the RFID Antenna and the Sensors

An antenna of an RFID reader and infrared sensors are attached to top of inner surface of a school bag. Two infrared receivers are attached to the central back of the school bag, and infrared LEDs are attached to the other side of infrared receivers. One RFID antenna is attached next to infrared receivers. A controller of them is put in the corner of the school bag. Figure 2 shows how to attach them. Furthermore, an RFID tag is attached to the middle of each personal item.

The reason that the infrared sensor is adopted in the system is based on the following two points of consideration. First, although the RFID is good at understanding what the item is, it is not good at making judge on the in/out status of the item. Secondly, it is difficult to confine the RFID reader by only reading the RFID tag in the schoolbag. In case the book with RFID tag is out of the bag but in the range of the RFID reader, it will make a mistake. So the infrared sensors are adopted to assist the RFID reader.

The system can make judgment on whether a personal item is taken in or out of the school bag by reactions of an RFID reader and infrared sensors. The system detects identification of a personal item by RFID tags and detects a personal item is in or out



(a) Front view (b) Side view

Fig. 2. How to attach antenna of an RFID reader and infrared sensors

by the order of two infrared receivers' reactions. Table 1 shows detection by infrared sensors' reaction. A is used to denote the top infrared receiver, and B is used to denote the bottom infrared receiver. ¬A or ¬B means that an infrared sensor is not react, in other words, an object is in front of an infrared receiver, an object is detected. Information of personal items, which is read by sensors, is sent to the server through the controller in the school bag, and it is pushed and managed by the server.

Table 1. Detection by infrared sensor reaction

(¬A) → (¬B) → (A) → (B)	In
(¬B) → (¬A) → (B) → (A)	Out

3.2 Personal Items Monitor Module

The function of this module is to grasp the physical object's identity and IN/OUT (Table 1) status. The object's identity is gotten by the RFID reader. The IN/OUT statuses of the object is monitored by the two infrared sensors. Procedure 1 shows the module of process for monitor of personal items.

```
Procedure 1
If the top infrared sensor is in no signal status or
the bottom infrared sensor is in no signal status
          To make a judgment on the pattern based on the
Table 1
          If the pattern is IN
            Get RFID tag's ID
            If the ID is on the items list
              Set the status of the item to IN
          Else if the pattern is OUT
```

```
Get RFID tag's ID
If the ID is on the items list
  Set the status of the item to OUT
```

3.3 Forgotten Items Check Module

The module mainly functions as telling the system about the timing to provide reminding service (Procedure 2).

```
Procedure 2
If current time has arrived at any schedule's Warning
threshold
          Compare the items list in the schedule with
the items list maintained by the schoolbag
          If any items status in the schedule are OUT
           Do reminding service
          Else
           Show encourage message
```

4 Implementation and Verification of the System

First we explain the hardware and software tools which have been adopted by the system. The pupils side subsystem's deploy diagram is shown in Figure 3. AT89S52 is selected as the micro-processor. WiPort is used for wireless communication so that the schoolbag can be portable. 24LC64 connected with AT89S52 through I2C bus was used as ROM to store the schedule received from the server. The LCD display adopts a graphical LCD chip. Two infrared sensors are used to grasp the in/out status

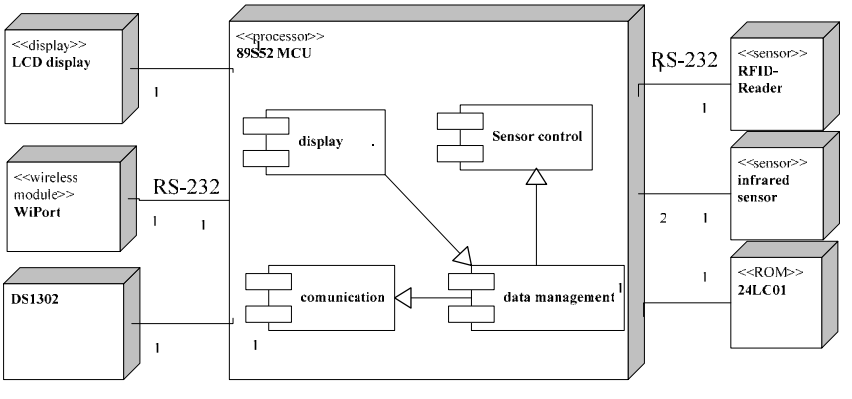


Fig. 3. A deploy diagram of the pupil side subsystem in the schoolbag

of an item. RF-MOD10H2 of Microtechnica is chosen as the RFID reader to read the data of RFID tag which attached to each item. In the system, two RS-232 ports are needed, but the AT89S52 only has one RS-232 port. So another RS-232 port is simulated by using software methods.

Two kinds of languages Java and C have been selected to develop the system. The server side program and the teacher's client side program were developed in SUN J2SDK1.4.2. The pupils side program which runs on the embedded hardware was accomplished in Keil C51. The Microsoft Access was used to keep a history of the schedules and items status.

When items are taking out or putting into the schoolbag, the precision of monitoring the in/out status is statistically experimented. Three kind of object is tested in the experiment. They are a B5 size book, a A5 size notebook and an RFID tag card of 13x6 size. Each object is tested in three kinds of speed, slow about 15cm/s, normal about 30cm/s and fast about 50cm/s. Each repeated for twenty times. The RFID reader, which we use, is non-anticollision, so it is prerequisite for this experiment that we take objects in and out one by one. The result of this experiment is shown in Table 2.

Table 2. Result of accuracy of taking out and putting in to the school bag each pattern

	B5	A5	Tag card
Slow 15cm/s	80%	90%	90%
normal 30cm/s	85%	95%	75%
Fast 50cm/s	70%	85%	55%

From Table 2, the size of the object has a relatively big influence on the accuracy only when the object is moving in the fast speed. The reason is an RFID reader need 200-300ms to get the tag's data. So when the speed is too fast, it is difficult to judge on the item's identity.

5 Conclusion

A ubiquitous learning support system is developed which can get valuable information from the human behavior in a implicit way and based on the collected data to some extent the personalize service can be provided. Specifically, we developed a schoolbag with a personal management system using an RFID and infrared sensors. The system can monitor whether a pupil puts/takes personal items in/out a schoolbag by attaching a RFID reader and infrared sensors on the schoolbag. The system can communicate with the server via the wireless networks. The system can manage information of personal items and support checking lost items.

Pupils can understand necessary things for classes certainly and prevent forgotten items. In doing so, pupils can have the necessary items for school work. Pupils can get information from teachers in real-time.

As shown in Table 2, when the item moving speed increases, the accuracy of the judgment is decreased. To solve this problem, on one hand faster RFID readers should be adopted as mentioned in Section 4. On the other hand, higher performance processor

should be used. AT89S52 which is an 8-bit processor has been used. In the future, 16-bit micro-processor like H8 or 32-bit micro-processor like ARM7 should be adopted to substitute AT89S52.

So far, the system can provide reminding service based on the data from the put in or take out behavior. Since the history data have been collected in the database, further support can be provided for the pupil. For example, additional module can be plug into the existing system to help the pupil to give up some bad habit and culture good habit.

Acknowledgement

This research is partially supported by a Subsidy from Fukushima Prefectural Foundation for Advancement of Science and Education, and a Grants-in-Aid from Ministry of Education, Culture, Sports, Science, and Technology (MEXT).

References

1. Borriello, G., Brunette, W., Hall, M., Hartung, C., Tangney, C.: Reminding about Tagged Objects using Passive RFIDs, UbiComp 2004, LNCS 3205, pp.36-53, 2004
2. Object-Based Media group at the MIT: build your own bag, http://alumni.media.mit.edu/~nanda/design/electronics/byob/byob.html
3. DNP and DOCOMO-SYSTEMS: Kids in Feel, http://www.docomo-sys.co.jp/products/kidsinfeel/index.html
4. Ito, R.: SYABERANDOSERU: A Satchel with Reminding Functions using Ubiquitous, Thesis the Graduate School of Computer science and Engineering, University of Aizu, 2004

Passive Radio Frequency Exteroception in Robot Assisted Shopping for the Blind

Chaitanya Gharpure, Vladimir Kulyukin,
Minghui Jiang, and Aliasgar Kutiyanawala

Computer Science Assistive Technology Laboratory,
Utah State University, Logan, UT 84321, USA
cpg@cc.usu.edu, vladimir.kulyukin@usu.edu, mjiang@cc.usu.edu,
aliasgar@cc.usu.edu
http://www.cs.usu.edu/ vkulyukin/vkweb/research/sandee.html

Abstract. In 2004, the Computer Science Assistive Technology Laboratory (CSATL) of Utah State University (USU) started a project whose objective is to develop RoboCart, a robotic shopping assistant for the visually impaired. RoboCart is a continuation of our previous work on RG, a robotic guide for the visually impaired in structured indoor environments. The determinism provided by exteroception of passive RFID-enabled surfaces is desirable when dealing with dynamic and uncertain environments where probabilistic approaches like Monte Carlo Markov localization (MCL) may fail. We present the results of a pilot feasibility study with two visually impaired shoppers in Lee's MarketPlace, a supermarket in Logan, Utah.

1 Introduction

There are 11.4 visually impaired users living in the U.S. [1]. Grocery shopping is an activity that presents a barrier to independence for many visually impaired people who either do not go grocery shopping at all or rely on sighted guides, e.g., friends, spouses, and partners [2]. While some visually impaired people currently rely on the store personnel to help them shop, they express two common concerns [3]: 1) such personnel may not be immediately available, which results in the shopper having to wait for assistance for a lengthy period of time, and 2) the shopper may not be comfortable shopping for gender sensitive items with a stranger, e.g. purchasing items related to personal hygiene.

In our previous work, we investigated several technical aspects of robot-assisted navigation for the blind, such as RFID-based localization, greedy free space selection, and topological knowledge representation in [4,5,2]. In this paper, we focus on how passive radio frequency (PRF) sufaces can assist a robotic shopping assistant in a grocery store. Many systems that operate in smart environments utilize *proprioception* (action is determined relative to an internal frame of reference) or *exteroception* (action is determined from a stimulus originating in the environment itself). RFID has become an exteroceptive technology of choice due to low power requirements , low cost, and ease of installation.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 51–60, 2006.

This paper is organized as follows. In section 2, we present related work. In section 3, we explain the hardware and navigation algorithms of RoboCart. In section 4, we desscribe two proof-of-concept experiments which demonstrate the advantages of using RFID mats and the practicality of a smart device like RoboCart. In section 5, we give our conclusions.

2 Related Work

Smart environments have become a major focus of assistive technology research [6]. The researchers at the Smith-Kettlewell Eye Research Institute developed Talking Signs©, audio signage IR sensors for the visually impaired that associate audio signals with various signs in the environment [7]. Willis and Helal [8] propose an assisted navigation system where an RFID reader is embedded into a blind navigator's shoe and passive RFID sensors are placed in the floor. Vorwerk [9], a German company, manufactures carpets containing integrated RFID technology for the intelligent navigation of service robots. However, they place RFID tags strictly in a rectangular grid format.

Several research efforts in mobile robotics are similar to the research described in this paper insomuch as they use RFID technology for robot navigation. Kantor and Singh [10] use RFID tags for robot localization and mapping. They utilize time-of-arrival signals from known RFID tags to estimate distance from detected tags and localize the robot. Hahnel et. al. [11] propose a probabilistic measurement model for using RFID signals to to analyze whether RFID can be used to improve the localization of mobile robots in office environments. They demonstrate how RFID can be used to improve the performance of laser based localization.

3 Robot-Assisted Shopping

3.1 RoboCart's Hardware

The RoboCart hardware design is a modification of RG, our indoor robotic guide for the blind that we built in 2003-2004 on top of another Pioneer 2DX base [12]. RoboCart is built on top of a Pioneer 2DX robotic platform from ActivMedia, Inc. RoboCart's wayfinding toolkit resides in a polyvinyl chloride (PVC) pipe structure securely attached to the platform (See figure 1). The wayfinding toolkit consists of a Dell ™ Ultralight X300 laptop connected to the platform's microcontroller, a SICK laser range finder, a TI-Series 2000 RFID reader from Texas Instruments, and a Logitech © camera facing vertically down. The RFID reader is attached to a 200mm x 200mm antenna. Unlike in RG which had its RFID antenna on the right side of the PVC structure approximately a meter and a half from the floor, in RoboCart, as seen in Figure 1, the RFID antenna resides close to the floor in front of the robot for reasons that will be explained later.

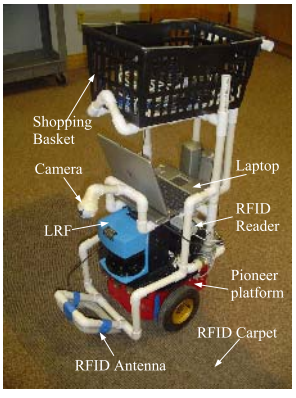


Fig. 1. RoboCart Hardware

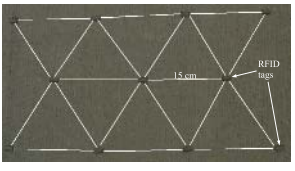


Fig. 2. RFID mat

3.2 Navigation

Navigation in RoboCart is based on Kuipers' Spatial Semantic Hierarchy (SSH) [13]. The SSH is a model to represent spatial knowledge. According to SSH, spatial knowledge can be represented in five levels: sensory, control, causal, topological and metric. Sensory level is the interface to the robot's sensory system. RoboCart's primary sensors are a laser range finder, a camera, and an RFID reader. The control level represents the environment in terms of *control laws* which have trigger and termination conditions associated with them. The causal level describes the environment in terms of *views* and *action*. Views specify triggers; actions specify control laws. For example, *follow-hall* can be a control law triggered by *start-of-hall* and terminated by *end-of-hall*. The topological level of the SSH is a higher level of abstraction, consisting of places, paths and regions, and their connectivity, order and containment relationships. The metrical level describes a global metric map of the environment within a single frame of reference.

To deal with large open spaces, we decided to use laser-based Monte Carlo Markov localization (MCL) [14], as it was already implemented in ActivMedia's Laser Mapping and Navigation software. After several field tests, we discovered some problems with MCL localization. First, the robot's ability to accurately localize rapidly deteriorated in the presence of heavy shopper traffic. Second, MCL sometimes failed due to wheel slippage on a wet floor or due to the blind shopper inadvertently pulling on the handle. Third, since MCL relies exclusively on odometery to localize itself along a long uniform hallway that lacks unique laser range signatures, it would frequently get lost in an aisle. Fourth, MCL localization frequently failed in the store lobby, because the lobby constantly changed its layout due to promotion displays, flower stands, product boxes. Finally, once MCL fails, it either never recovers, or recovers after a long drift.

3.3 RFID-Based Recalibration

We conjectured that MCL was a viable option if the robot could somehow recalibrate, periodically and reliably, its position on the global map. To allow for periodic and reliable MCL recalibration, we decided to turn the floor of the store into an RFID-enabled surface, where each RFID tag had its 2D coordinates. The literature search showed that some ubiquitous computing researchers had started thinking along the same lines [8]. The concept of the RFID-enabled surface was refined into the concept of *recalibration areas*, i.e., areas of the floor with embedded RFID tags. In our current implementation, recalibration areas are RFID mats which are small carpets with embedded RFID tags. The mats are placed at specified locations in the store without causing any disruption to the indigeneous business processes.

The literature search showed that RFID has been used to assist laser-based localization. For example, in [15], the authors demonstrate how RFID can be used to improve the performance of laser-based localization through a probabilistic measurement model for RFID readers. While this is certainly a valid approach, we think that one advantage of recalibration areas is deterministic localization: when the robot reaches a recalibration area, its location is known with certainty. We built several RFID mats with RFID tags embedded in a hexagonal fashion. An RFID mat is shown in Figure 2. Every recalibration area is mapped to a corresponding rectangular region in the store's metric global map constructed using ActivMedia's metric map building software, Mapper3©. In the future, as larger recalibration areas are deployed, every RFID tag may have a unique ID so that a recalibration area may act as a topological region with its own co-ordinate system and a frame of reference.

3.4 Semi-automatic Acquistion of Topology and Causality

A principal limiation of RG was the fact that the topological and causal levels of the SSH had to be manually created for a given environment [5]. In RoboCart's, several aspects of acquiring topological and causal knowledge were automated. The problem here is to have the robot itself acquire the connectivity of landmarks and maneuvers that can be executed at each landmark.

There are two types of representations of the environment that must be acquired before RoboCart can navigate a grocery store. First, the metrical level representation in form of an occupancy grid map, which is used in the MCL algorithm, and second, the control/causal level representation, which is an abstraction of the metric map (used for path planning).

In RoboCart, the acquistion process has four steps. First, the robot is manually driven through the environment to acquire a global metric map with ActivMedia's Mapper3 laser-based software. Figure 3 shows the metric map for the area of Lee's MarketPlace used in the experiments. Second, a dark blue masking tape is placed on on the floor. In Figure 3, the tape goes north from the robot's home location and turns west between the cash registers and the grocery aisles. Third, the robot follows the tape to acquire the topological and causal knowlege. Fourth, the tape is removed.

The robot uses a Logitech © web cam to capture floor images. Four actions are used: follow-tape, turn-left-90, turn-right-90, turn-180. Two action triggers are tape intersections and ends of turns. An edge detection algorithm is used to follow the tape and to recognize three tape fiducials: straight-tape, intersection, and, horizontal-tape. When a tape intersection is detected, the robot stops and presents a confirmation dialogue to the operator. The operator accepts the landmark if it is a true positive, and rejects it if it is a false positive. Thus, the causal schemas $< View, Action, View >$ are obtained through user interaction, where $Action \in$ {follow-tape, turn-left-90, turn-right-90, turn-180} and $View \in$ {tape-intersection, horizontal-tape, end-of-turn}. If a visual landmark is accepted, a fuzzy metric landmark is created. If the global position is $< x, y >$, the fuzzy landmark is a rectangular region from x_i to x_j, and from y_k to y_m, where $x_i = x - \delta$, $x_j = x + \delta$, $y_k = y - \delta$, and $y_m = y + \delta$, where δ is an integer constant.

It took us 40 minutes to acquire the topological and causal knowledge of the area of Lee's MarketPlace shown in Figure 3: 10 minutes for the metric map acquisition, 10 minutes for deploying the tape, 15 minutes for running the robot, and 5 minutes for removing the tape. Thus, the acquired knowledge of the environment consists of three files: the global metric map file, a file with fuzzy metric landmarks, and a file with a fuzzy metric landmark connectivity graph that also contains the actions that can be executed at each landmark described in the next section.

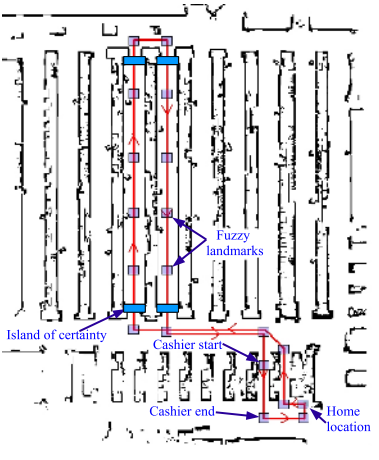


Fig. 3. Fuzzy areas in the grocery store environment

3.5 Actions

Path planning in RoboCart is done through a breadth first search. A path plan is a sequence of fuzzy metric landmarks, connected by actions. Whenever a landmark is reached, the appropriate action is triggered. There are five

actions: turn-into-left-aisle, turn-into-right-aisle, follow-maximum-empty-space and track-target. The first three rely on finding and choosing the maximum empty space as the direction to travel. The track-target *action* relies on current pose P_c<x-y-θ> and the destination pose P_d<x-y-θ>, to compute direction of travel. Other actions which are not a part of the pre-planned path are stop-and-recalibrate, beep-and-stop, and inform-and-stop. If there is an obstacle in the path, RoboCart emits a beep and stops for 8 seconds before starting to avoid the obstacle. When RoboCart reaches a destination, it informs the user about the destination and stops. Different actions and their trigger and termination views are listed in Table 1.

Table 1. Actions and Views

Trigger	Action	Termination
fuzzy-area	track-target	fuzzy-area
fuzzy-area	turn-into-right-aisle	fuzzy-area
fuzzy-area	turn-into-left-aisle	fuzzy-area
fuzzy-area	maximum-empty-space	fuzzy-area
RFID-tag	stop-and-recalibrate	pose=tag-x-y
Obstacle	beep-and-stop	no-obstacle
Obstacle	beep-and-stop	time-out
destination	inform-and-stop	none
cashier	inform-and-stop	resume

3.6 Human-Robot Interaction

The shopper communicates with RoboCart through a 10-key numeric keypad attached on the handle of the cart. A speech enabled menu allows the shopper to perform tasks like browse through the hierarchical database, select products, navigate, pause, and resume. A wireless IT2020 barcode reader from Hand Held Products Inc., is attached to the laptop. When the shopper reaches the desired product in the aisle, he/she picks up the barcode and scans the barcodes on the edge of the shelf. When a barcode is scanned the reader beeps. If the barcode scanned is that of the desired item, the shopper hears the product title in the Bluetooth (R) headphones. The shopper can then carefully reach for the product above the scanned barcode and place it in the shopping basket installed on RoboCart.

4 Proof-of-Concept Experiments

4.1 Experiment 1

Localization error samples were collected from the two populations: P_{nomat} and P_{mat}, where, P_{nomat} is the population of localization-errors when no recalibration is done at RFID mats, and P_{mat} is the population of localization-errors when recalibration is done. Localization error in centimeters was calculated from true

and calculated poses as follows. A white masking tape was placed on a dark brown office floor forming a rectangle with a 30-meter perimeter. At 24 selected locations, the tape was crossed by perpendicular stretches of the same white masking tape. The x and y coordinates of each intersection were recorded.

Four new PRF mats were developed. Each mat consisted of a carpet surface, 1.2 meters long and 0.6 meters wide, instrumented with 12 RFID tags. The x-y regions of each mat were supplied to the robot. The mats were placed in the middle of each side of the rectangle. The robot used the vision-based tape following and tape intersection recognition algorithms to determine its true pose (ground truth). Thus, whenever a tape intersection was recognized, the robot recorded two readings: its true posed determined from vision and its estimated pose determined from MCL.

The first 16 runs without recalibration produced 384 (24 landmarks x 16 runs) samples from P_{nomat}. Another 16 runs with recalibration produced 384 samples from P_{mat}. Let $H_0 : \mu_1 - \mu_2 = 0$, be the null hypothesis, where μ_1 and μ_2 are the means of P_{nomat} and P_{mat}, respectively. The paired *t-test* at $\alpha = 0.001$ was used to compute the *t-statistic* as $t = (M_1 - M_2)/\sqrt[2]{(\sigma_1^2/n_1) + (\sigma_2^2/n_2)}$, where M_1 and M_2 are the sample means. From the data obtained in the experiments, the value of the *t-statistic* was calculated to be 6.67, which was sufficient to reject H_0 at selected α.

The use of PRF mats as recalibration areas showed a 20.23 % reduction in localization error: from a mean localization error of 16.8cm without recalibration, to 13.4cm with recalibration. Since the test was conducted in a simple office environment, the errors are small. It is expected that in a larger environment, e.g. a supermarket, the errors will be significantly larger. Negotiations with the store for conducting recalibration experiments are underway as this paper is being written.

4.2 Experiment 2

Upon entering the store, a visually impaired shopper must complete the following tasks: find RoboCart, use RoboCart to navigate to shelf sections with needed grocery items, find those items on the shelves, place them into RoboCart, navigate to the cash register, place the items on the conveyer belt, pay for the items, navigate to the exit, remove the shopping bags from RoboCart, and leave the store.

The purpose of the second experiment was to test the feasibility with respect to these tasks. In particular, we focused on two questions: 1) Can the shopper successfully retrieve a given set of products?; and 2) Does the repeated use of RoboCart result in the reduction of the overall shopping time?

The sample product database consisted of products from aisles 9 and 10, with 8 products on the top shelf, 8 products on the third shelf from the bottom, and 8 products on the bottom shelf. The trials were run with two visually impaired shoppers from the local community over a period of six days. A single shopping trial consisted of the user picking up RoboCart from the docking area, navigating to three pre-selected products, and navigating back to the docking area through

the cash register. Before the actual trials, the shopper was given 15 minutes of training on using the barcode reader to scan barcodes on the shelves.

We ran 7 trials for three different sets of products. To make the shopping task realistic, for each trial, one product was chosen from the top shelf, one from the third shelf, and one from the bottom shelf. Split timings for each of the ten tasks were recorded and graphed. Figure 3 shows the path taken by RoboCart during each trial. The RFID mats were placed at both ends of the aisles as shown in Figure 3.

The shoppers successfully retrieved all products. The times for the seven shopping iterations for product sets 1, 2 and 3 were graphed. The time taken by the different navigation tasks remained fairly constant over all shopping trials. From the graph in figure 4 it can be seen that the time to find a product reduces after a few trials. The initial longer time in finding the product is due the fact that the user is not aware of the exact location of the product on the shelf. Eventually the user learns where to look for the barcode, and the product retrieval time reduces. The product retrieval time stabilized at an average of 20 to 30 seconds.

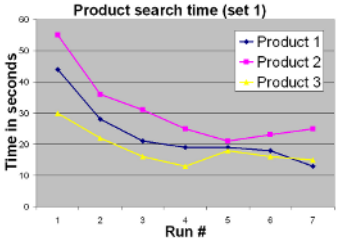


Fig. 4. Product Retrieval Performance for Participant 1

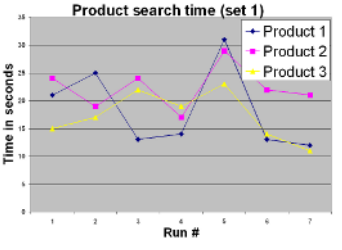


Fig. 5. Product Retrieval Performance for Participant 2

After conducting the experiments with the first participant, we felt the need to modify the structure of the barcode reader, so that the user could easily scan barcodes on the shelves. This led to a minor ergonomic modification to the barcode which enabled the user to rest the barcode on the shelves and scan the barcode with ease. This modification greatly improved the performance, which can be seen from the results in figure 5.

5 Conclusions

We presented a proof-of-concept prototype of RoboCart, a robotic shopping assistant for the visually impaired. We described how we approach the navigation problem in a grocery store. We also presented our approach to semi-automatic acquisition of two levels of the SSH. Our use of RFID mats to recalibrate MCL was described. We experimentally discovered that RFID-based recalibration reduced the MCL localization error by 20.23%.

In the pilot experiments, we observed that the two visually impaired shoppers successfully retrieved all products and that the repeated use of RoboCart

resulted in the reduction of the overall shopping time. The overall shopping time appears to be inversely related to the number of shopping trials and eventually stabilized.

While the pilot feasilibity study presented in this paper confirmed the practicality of a device like a smart shopping cart and gave valuable insights into the design of future experiments, our approach has limitations. Recalibration areas are currently placed in an ad-hoc fashion. The system would greatly benefit if the placement of recalibration areas on the global metric map was done algorithmically. The automatic or semi-automatic construction of the product database with useful descriptions and handling instructions for all products has not been attempted.

Acknowledgements

This research has been supported, in part, through NSF CAREER grant (IIS-0346880) and two Community University Research Initiative (CURI) grants (CURI-04 and CURI-05) from the State of Utah awarded to Vladimir Kulyukin.

References

1. LaPlante, M.P., Carlson, D.: Disability in the united states: Prevalence and causes. In: U.S. Department of Education, National Institute of Disability and Rehabilitation Research, Washington, DC (2000)
2. Kulyukin, V., Gharpure, C., Nicholson, J.: Robocart: Toward robot-assisted navigation of grocery stores by the visually impaired. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE/RSJ (2005)
3. Burrell, A.: Robot lends a seeing eye for blind shoppers. USA Today, Monday, Jul 11, 2005 (2005)
4. Kulyukin, V., Gharpure, C.P., De Graw., N.: Human-computer interaction in a robotic guide for visually impaired. In: the Proceedings of AAAI Spring Symposium, Palo Alto, CA (2004)
5. Kulyukin, V., Gharpure, C.P., Nicholson, J., Pavithran, S.: Rfid in robot-assisted indoor navigation for the visually impaired. In: Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004), Sendai, Japan (2004)
6. Zita Haigh, K., Kiff, L., Myers, J., Guralnik, V., Gieb, C., Phelps, J., Wagner, T.: The intependent life style assistant: Ai lessons learned. In: Proceedings of the 2004 IAAI Conference, San Jose, CA, AAAI (2004)
7. Marston, J., Golledge, R.: Towards an accessible city: Removing functional barriers for the blind and visually impaired: A case for auditory signs. Technical Report, Department of Geography, University of California at Santa Barbara (2000)
8. Scooter, S., Helal, S.: A passive rfid information grid for location and proximity sensing for the blind user. University of Florida Technical Report number TR04-009 (2004)
9. Smart carpet, vorwerk and co. In: http://www.vorwerk-teppich.de. (2006)

10. Kantor, G., Singh, S.: Priliminary results in range-only localization and mapping. In: IEEE Conference on Robotics and Automation, Washington, D.C. (2002)
11. Hahnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M.: Mapping and localization with rfid technology. In: Technical Report, IRS-TR-03-014, Intel Research Institute, Seattle, Washington (2003)
12. Kulyukin, V., Gharpure, C.P., Sute, P., DeGraw, N., Nicholson, J.: A robotic wayfinding system for the visually impaired. In: Proceedings of the Sixteenth Innovative Applications of Artificial Intelligence Conference, San Jose, CA (2004)
13. Kupiers, B.: The spatial semantic hierarchy. Artificial Intelligence **119** (2000) 191–233
14. Fox, D.: Markov Localization: A Probabilistic Framework for Mobile Robot Localization and Navigation. PhD thesis, University of Bonn, Germany (1998)
15. Hahnel, D., Burgard, W., Fox, D., Fishkin, K., Philipose, M.: Mapping and localization with rfid technology. In: Intel Research Institute. Tech. Rep. IRS-TR-03-014, Seattle, WA (2003)

A Smart Identification Card System Using Facial Biometric: From Architecture to Application

Kun Peng[1], Liming Chen[1], and Su Ruan[2]

[1] Laboratoire d'InfoRmatique en Images et Systems d'information (LIRIS), Département MI, Ecole centrale de Lyon, BP 163, 36 avenue Guy de Collongue, 69131 Ecully Cedex, France
{Kun.Peng, Liming.Chen}@ec-lyon.fr
[2] Equipe Image, CReSTIC, Département GE&II, IUT de Troyes, 9 rue de Quebec, 10026 Troyes, France
s.ruan@iut-troyes.univ-reims.fr

Abstract. This paper presents a smart identification card system using facial biometric information for identity authentification. For a trade-off between the security and the cost, this system utilizes an architecture containing three security levels of identity authentification, "manual face verification offline", "manual face verification online" and "automatic (biometric) face verification", which satisfy the different security requirements of various applications of identification cards. For the function of "manual face verification online", we bring out an idea based on decomposing the face image into two parts which are stocked into the card and the database of system respectively. And for the function of "automatic face verification", we proposed a novel face verification scheme based on class-specific face models. The technique Active Appearance Model is applied, as the way of face modelling, to realize the proposed scheme. A prototype application of such system, which contains a fix version for PC and a mobile version for Pocket PC, is also introduced in this paper.

1 Introduction

1.1 Background

In practical life, identification cards play a very important role to identify and authenticate its holder in various applications. Unfortunately, the current paper-based identification card is often the subject of falsification, leading to insecurity for critical applications. The idea of applying the smart card to the field of identity composes the principal motivation of the conception of "smart identification card".

Recently, the context of the terrorist threat caused the world to accelerate the works on the documents of voyage and identity, in which an increased security could be seen by using smart card as well as biometrics. In this paper, we focus our attention on using the face of the card holder for biometric authentification. And the principal objective of this paper aims at designing a smart identification card system which uses facial biometric information of card holder to verify or authenticate his identity.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 61 – 70, 2006.

1.2 State-of-the-Art

Among various biometric methods, face recognition is the most important means used by the human beings to recognize themselves. A general statement of the face recognition problem (in computer vision) can be formulated as follows: Given still or video images of a scene, identify or verify one or more persons in the scene using a stored database of faces. In fact, face recognition scenarios can be classified into two types [1]: (i) face verification (or authentification) and (ii) face identification (or recognition). Face verification ("Am who I say I am?") is a one-to-one match that compares a query of face image against a template face image whose identity is being claimed. Face identification ("Who am I?") is a one-to-many matching process that compares a query face image against all the template images in a face database to determine the identity of the query face. The identification of the test image is performed by locating the image in the database that has the highest similarity with the test image. Obviously, the application of face recognition to smart identification card system falls into face verification.

Currently, image-based face recognition techniques can be mainly categorized into two groups based on the face representation which they use [1]: (i) appearance-based which uses holistic texture features; (ii) model-based which employs shape and texture of the face.

Appearance-based approaches, including linear appearance-based approaches (PCA [2], ICA [3], LDA [4] …) and non-linear appearance-based approaches (KPCA [5] …), represent an object in terms of several object views. An image is considered as a high-dimensional vector, i.e., a point in a high-dimensional vector space. Many view-based approaches use statistical techniques to analyze the distribution of the object image vectors in the vector space, and derive an efficient and effective representation (feature space) according to different applications. Given a test image, the similarity between the stored prototypes and the test view is then carried out in the feature space. The image vector representation allows the use of learning techniques for the analysis and for the synthesis of images. Face recognition can be treated as a spacing-searching problem combined with a machine-learning problem. Regardless of the success of some of these methods in constrained scenarios, all these approaches have two main limitations. Firstly, they don't utilize the prior (expert) knowledge of the human faces. Secondly, all these methods are sensitive to facial variations, such as 3D pose, illumination and face expression.

The model-based face recognition scheme is aimed at constructing a model of the human face, which is able to capture the facial variations. The prior knowledge of human face is highly utilized to design the model. For example, feature-based matching derives distance and relative position features from the placement of internal facial elements (e.g., eyes, etc.). Kanade [6] developed one of the earliest face recognition algorithms based on automatic feature detection. By localizing the corners of the eyes, nostrils, etc. in frontal views, his system computed parameters for each face, which were compared (using a Euclidean metric) against the parameters of known faces. A more recent feature-based system, based on elastic bunch graph matching, was developed by Wiskott et al. [7] as an extension to their original graph matching system [8]. By integrating both shape and texture, Cootes et al. [9] developed a 2D morphable face model (active appearance model), through which the

face variations are learned. A more advanced 3D morphable face model [10] is explored to capture the true 3D structure of human face surface. Both morphable model methods come under the framework of "interpretation through synthesis". The main advantage of model-based methods is that the face model, which integrates prior human knowledge, has intrinsic physical relationship with real faces.

2 Architecture

The security problem is very important to information systems, especially for the identification card system which refers to a base of security for many other applications. We always wish to realize an information system which is completely secure, but that is only a myth. So, it's necessary to find a point of balance between an acceptable risk and a reasonable cost.

In our system, identity verification using facial information is the most important and useful function. However, the security needs of identity verification are variable to the different real applications. Thus, we use a strategy consisting of several security levels. A lower level of security, which logically leads less cost, will be applied to the cases which have less requirements of security; on the other hand, for the cases where more security needs exists, a higher level of security with more expensive cost will be applied. We proposed here a strategy consisting of three security levers for our system:

- Manual face verification offline: Almost as same as the traditional solution which is largely applied now, this level of verification is performed manually without needs to connect to remote server. This manual verification is based on the photography visible with the naked eye on surface of the cart, or done by an electronic reading of the digitized photography stored in the chip.
- Manual face verification online: The idea here is to decompose a face image of the holder into two parts: one part, which is a very compact signature of the face image, will be stored in the chip of the card; the other part will be stored in the database of verification system. Only the conjunction of these two parts of data non-modified gives the possibility to reconstruct the original face image, otherwise a disturbed image will be obtained. It is based on this face image reconstructed that a manual verification is performed. Clearly, this solution carries out to a security reinforced against falsification, thanks to the need for connecting to a server of high security. Moreover, this method also economizes the limited and expensive memory space in the smart card. In fact, a very small occupation of the memory space in the chip gives the possibility to install there more other complex applications, for example biometric verification applications.
- Automatic (biometric) face verification: It's in this security level that the facial biometric information is used to realize automatic identity verification. Despite the advantages of the biometric authentification, this solution is the most expensive one due to additional hardware and software. So it will be only applied to the cases which have rigorous requirements for security.

According to the analyses above, we illustrate the architecture of the system in Fig.1. This figure mainly explains the flowchart of the last two security levels: the

arrows with very dense broken line mean the path of "manual face verification online"; the arrows with normal broken line for the path of "automatic face verification". And the arrows with continuous line mean the common path.

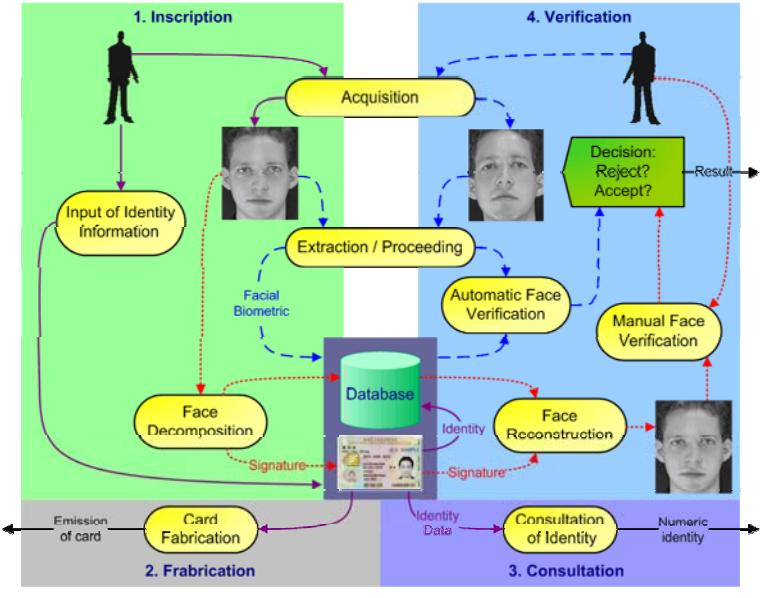


Fig. 1. Architecture of the system

3 Kernel Algorithms

To realize the design obtained in last section, there lack two kernel algorithms. One is the algorithm of face decomposition and reconstruction for manual face verification offline [11]; the other is the algorithm of automatic face verification [12].

3.1 Face Decomposition and Reconstruction

As we have said, the idea here is simple: decompose a face image of the holder into two parts which are respectively stocked in the smart card and a database; only the conjunction of these two parts of data non-modified gives the possibility to reconstruct an original face image for authentification use. Here, we present an algorithm base on Singular Value Decomposition (SVD) of to realize this idea.

3.1.1 Singular Value Decomposition

A matrix of dimension $M \times N$ ($M \geq N$) can be described by:

$$A = U \Lambda V^T , \tag{1}$$

where U and V are matrices of dimension $M \times N$ and $N \times N$, respectively, with orthogonal columns, and Λ is a diagonal matrix of dimension $N \times N$ whose elements

are the singular values of the original matrix A. More precisely, the columns of U are eigenvectors of AA^T, and the columns of V are eigenvectors of A^TA. As the U and V are unitary matrices ($U^{-1}=U^T$, $V^{-1}=V^T$),

$$\Lambda = U^T A V \ . \tag{2}$$

Singular value decomposition of a matrix can be used to obtain lower rank approximations of the matrix. If we take the first r columns of U, V and the leading $r \times r$ submatrix of Λ, and define:

$$A_r = U_r \Lambda_r V_r^T = \sum_{j=1}^{r} \Lambda_{j,j} u_j v_j^T \ , \tag{3}$$

then, A_r is the best rank r approximation to A. Here u_j and v_j are j^{th} column of U and V, respectively.

3.1.2 Face Decomposition and Reconstruction Using SVD

Given an image A of dimension $M \times N$ ($M \geq N$), we calculate firstly two matrices AA^T and A^TA. Then we extract the eigenvalue and eigenvector of these two matrices. For each matrix, on choosing n vectors corresponding to n grandest eigenvalues ($n << M$, $n << N$), and we calculate the matrix of singular value according to the Eq. (2):

$$\Lambda_{opt} = U_{opt}^T A V_{opt} \ . \tag{4}$$

Here U_{opt} and V_{opt} are the matrices of n eigenvectors chosen in AA^T and A^TA, and Λ_{opt} is the diagonal matrix of the first n singular values. We extract the diagonal elements of Λ_{opt}, which is just n principle singular values. These n singular values will act as the compact signature which will be stored on the smart ID card, and the two matrices of n eigenvectors will be stored in the database.

When we want to verify the card holder's identity, manually online, a face image of the card holder must be reconstructed. By sending a request with the ID card number to the server, we will get two matrices of eigenvectors U_{opt} and V_{opt} from the database. Then with the singular values Λ_{opt} extracted from the card, the face image can be reconstructed (see Fig. 2) according to the Eq. (3):

$$A_{re} = U_{opt} \Lambda_{opt} V_{opt}^T \ . \tag{5}$$

If the matrices from the database and the singular values form the ID card come from the same photo, the face image will be reconstructed correctly. But, if someone has changed the data from card or the data from database, the image reconstructed will be perturbed.

3.1.3 Experimental Results

Fig. 2 shows the results of decomposition and reconstruction experiments using SVD-based algorithm for the face images of six different persons in ORL database. The images in the first line are original face images in the database. The second line shows the reconstructed face images with unmodified data from the card and database. In the process of encryption, we choose first 30 singular values (120 octets) as the compact signature stored in the card. We can see that a reconstruction with 30 singular values

has already a satisfactory result of reconstruction. The last line shows the reconstructed images, in which there is not any legal human face appeared, with modified data from the card or from the database.

Fig. 2. Face reconstruction results

3.2 Face Verification

In the two groups of face recognition techniques, we choose model-based methods because they use a more scientific way, in theory, to represent faces. Not using traditional model-based scheme due to its limitations, we propose here a novel scheme for face verification. And we apply the technique Active Appearance Model (AAM), which models the face by shape and texture components, to realize the proposed scheme.

3.2.1 Scheme of Face Verification

The traditional model-based scheme usually contains three steps: (i) Constructing the model; (ii) Fitting the model to the given face image; (iii) Using the parameters of the fitted model as the feature vector to calculate the similarity between the query face and prototype faces in the database to perform the recognition. However, in spite of its superiority in theory, this traditional scheme has two main limitations which lead to a relatively low recognition performance. On the one hand, the generic face model must be capable to interpret all the face images, including the unseen faces in various unknown conditions, which is very difficult to realize. On the other hand, using the parameters of the generic model to represent face would ignore the local facial features which are same important as the global facial features for face recognition.

In order to overcome these limitations, we propose here a novel scheme base on class-specific models for face verification, which also contains three steps: (i) Constructing a class-specific model for each subject; (ii) Fitting the corresponding model to the face image of probe; (iii) Using the fitting result to make the final judgment of verification. Fig. 3 illustrates the flowchart of these three steps. Although

these steps seem to be same as those of the traditional scheme, there are two main differences between them. Firstly, a class-specific face model is constructed for each subject in the proposed scheme. This class-specific face model can be seen as a binary classifier to the corresponding subject. Secondly, this novel scheme can be only applied to face verification, because the procedure of model fitting is time-consuming.

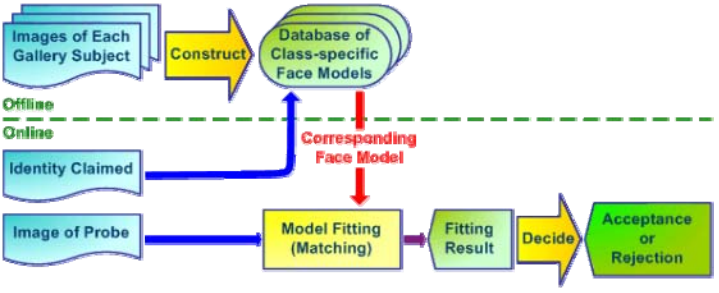


Fig. 3. Scheme based on class-specific models for face verification

3.2.2 Face Verification Using AAM

We have used the technique of AAM in our system to realize the proposed face verification scheme. We don't want to introduce the technique of AAM in this paper, because it is a common and known tools of face modeling, please refer to [9] for details. The proposed scheme contains three main steps (refer to [12]):

- Face modeling: By applying the modelling algorithm of AAM to the face images which are prepared for the construction of models, the system constructs a class-specific face model for each class, or each person. By constraining the range of the model parameters, we assume that this class-specific face model can only synthesize legal face instances which belong to this class.
- Model fitting: After selecting the face model according to the claimed identify, the system will use this model to fit the input image by performing AMM fitting algorithm. This iterative procedure of fitting can be simply initialized by an automatic detection or localisation of two centres of eyes on the face. For the details of eye detection algorithm refer to [13]. The fitting result after convergence will be sequentially used to make the final decision.
- Final decision: We assume that the face model of one class can only synthesize the legal face instances of this class, so if the input image belongs to this class, a good fitting result will be obtained. On the other hand, if the image of probe does not belong to this class, a bad fitting result, or a great error, will be achieved. Thus, by simply setting a threshold, the system can make the final decision, acceptance or rejection, according to the value of fitting error obtained in previous step.

3.2.3 Experimental Results

We present here the experimental results of our face verification algorithms performed on the IMM face database [14]. IMM face database comprises 240 still images with 40 different human faces, all of them without glasses. The gender distribution is 7 females and 33 males. All images are in the same size: 640×320 pixels.

We choose randomly 4 faces image to construct a face model for each class. And the reminder 2 images of each class were used to do the test. Fig. 4 shows the ROC curve of these tests. For a real verification system, there is a trade-off between the false acceptance rate (FAR) and the false rejection rate (FRR). The choice of the factor, i.e., FAR or FRR should be low, depends on the nature of the application.

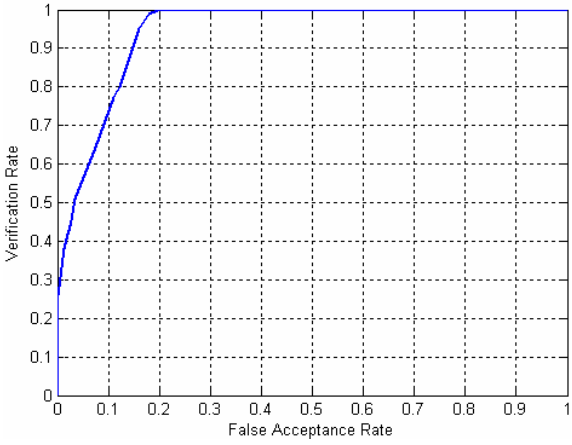


Fig. 4. ROC curve of experimental results

4 Application

We have developed a prototype of smart identification card system, named "ID-SecureWare", which intents on realizing the architecture proposed in this paper. This system is demanded to perform the identity verification by two ways: a fix way and a mobile way. That is to say, we can effectuate the identity control anywhere by the mobile way. So we have developed two versions of interfaces: one for PC and the other for Pocket PC. For Pocket PC, we use a special smart reader which can be connected to a handset device.

Actually, the function of "manual face verification online" has been completely realized, and combination of the function of "automatic face verification" into the system is in process. So, the current prototype system distinguishes two major steps. The first step is the fabrication process of smart identification card. During this process, we save into a smart card the holder's general identity information, including name, birthday, etc. A digital face image of the holder is also captured, and a compact signature is generated and stored into the smart card. Surely, the other part of data is stored into the database at the same time. The second step is the verification process of identification card. As we have said, this process can take place in two ways: a fix way with a PC, and a mobile way with a Pocket PC. The user introduces a holder's identification card into the smart card reader connected to the PC or to the Pocket PC, and then a connection is set up between the server and the PC or the Pocket PC in order to reconstruct the holder's face image. Finally, a manual authentification can then take place between the identification card holder and the reconstructed face image displayed on the screen of the PC or of the Pocket PC. Fig. 5 shows the interface of identity verification on Pocket PC.

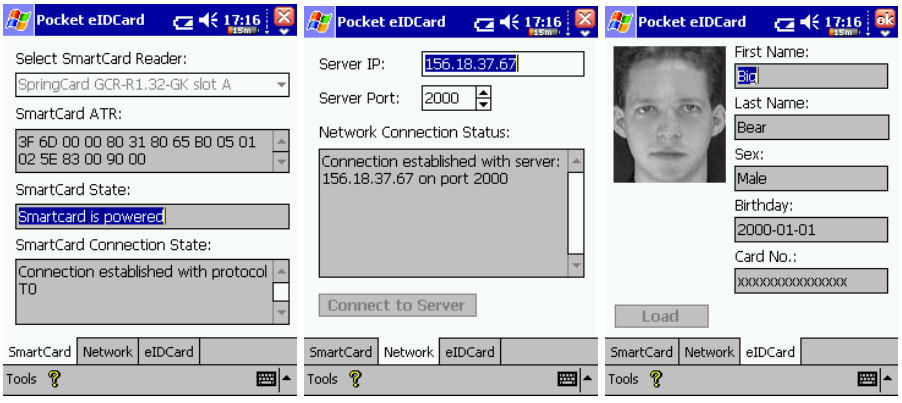


Fig. 5. Interface of identity verification on Pocket PC

5 Conclusion

This paper presents a smart identification card system using facial biometric information. For a trade-off between the security and the cost, this system utilizes an architecture containing three security levels of identity authentification, “manual face verification offline”, “manual face verification online” and “automatic (biometric) face verification”, which satisfy the different security requirements of various applications of identification cards. For the function of “manual face verification online”, we bring out an idea based on decomposing the face image into two parts which are stocked into the card and the database of system respectively. An algorithm based on Singular Value Decomposition of 2D matrix is proposed to realise this idea. For the function of “automatic face verification”, we proposed a novel face verification scheme based on class-specific face models. The technique Active Appearance Model is applied, as the way of face modelling, to realize the proposed scheme. A prototype application of such system, which contains a fix version for PC and a mobile version for Pocket PC, is also introduced in this paper.

As we have seen, the verification performance of the proposed face verification algorithm is not quite good and should be progressed. So the amelioration of this face verification algorithm will be our main future works. After anatomizing the algorithm and the experimental results, we found that the problem locates at the insufficiency of resources for face modelling. That is to say, a face model which is built from only four face images of a subject can not well interpret all cases of this subject, such as different light conditions, head poses, facial expressions etc. However, it is not realistic to gather so many face images which are various enough to construct an ideal class-specific face model in practical applications. Thus, we are developing an algorithm which can generate abundant virtual face images, varying in light conditions, head poses, facial expressions etc., of a subject based on a frontal face image with neutral facial expression of this subject. In this way, a more interpretative face model can be built for each subject by using the virtual face images, even if we have only one frontal face image of this subject.

References

1. X. Lu, *Image Analysis for Face Recognition*, personal notes, 2003.
2. M. A. Turk and A. P. Pentland, *Eigenfaces for Recognition*, Journal of Cognitive Neurosicence, Vol. 3, No. 1, pp. 71-86, 1991.
3. A. Hyvarinen, *Survey On Independent Component Analysis*, Neural Computing Surveys, Vol. 2, pp. 94-128, 1999.
4. K. Etemad and R. Chellappa, *Discriminant Analysis for Recognition of Human Face Images*, Journal of the Optical Society of America A, Vol. 14, No. 8, pp. 1724-1733, 1997.
5. B. Scholkopf, A. Smola and K. Muller, *Nonlinaer Component Analysis as a Kernel Eigenvalue Problem*, Neural Computation, Vol. 10, No. 5, pp. 1299–1319, 1998.
6. T. Kanade, *Picture Processing System by Computer and Recognition of Human Faces*, PhD. Thesis, Kyoto University, 1973.
7. L. Wiskott, J. M. Fellous, N. Kruger and C. V. D. Malsburg, *Face Recogniton by Elastic Bunch Graph Matching*, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 19, No. 7, pp. 775-779, 1997.
8. M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. V. D. Malsburg, R. P. Wurtz and W. Konen, *Distortion Invariant Object Recognition In the Dynamic Link Architecture*, IEEE Transations on Computers, Vol. 42, No. 3, pp. 300-311, 1993.
9. T. F. Cootes, G. J. Edwards and C. J. Taylor, *Active Appearance Models*, Proc. of European Conference on Computer Vision, Vol. 2, pp. 484-498, 1998.
10. V. Blanz and T. Vetter, *A Morphable Model for the Synthesis of 3d Faces*, Proc. of SIGGRAPH, pp. 187–194, 1999.
11. L. Chen, K. Peng and X. Liu, *Smart Identification Card*, PCT/FR200400685, France, 2004.
12. K. Peng, L. Chen and S. Ruan, *A Novel Scheme of Face Verification Using Active Appearance Models*, Proc. of IEEE Conference on Advanced Video and Signal based Surveillance (AVSS 2005), pp. 247-252, 2005.
13. K. Peng, L. Chen, S. Ruan and G. Kukharev, *A Robust and Efficient Algorithm for Eye Detection On Gray Intensity Face*, Proc. of 3rd International Conference on Advances in Pattern Recognition (ICAPR 2005), LNCS, Vol. 3687, pp. 302-308, 2005.
14. M. B. Stegmann, B. K. Ersboll and A. R. Larsen, *Fame - a Flexible Appearance Modeling Environment*, IEEE Transaction on Medical Imaging, Vol. 22, No. 10, pp. 1319-1331, 2003.

Architectures and Functions of the TMO Kernels for Ubiquitous and Embedded Real-Time Distributed Computing

JungGuk Kim[1], MoonHae Kim[2], and Shin Heu[3]

[1] Hankuk University of Foreign Studies, Korea
jgkim@hufs.ac.kr
[2] Konkuk University, Korea
mhkim@konkuk.ac.kr
[3] Hanyang University, Korea
shinheu@cse.hanyang.ac.kr

Abstract. The TMO (Time-triggered Message-triggered Object) model is a real-time object model for real-time distributed computing. A TMO instance is a kind of autonomous object having two special types of member threads. The first type is a group of time-triggered threads activated by pre-defined timing constraints such as periods and deadlines. And the second type is a group of message-triggered threads that are activated by distributed IPC messages from remote TMO's and finish their computing within pre-given deadlines. With these properties of the TMO, ubiquitous and real-time distributed applications can be easily organized as a logical network of TMO's. Since the TMO model has been proposed, there have been many successful progresses in developing TMO engines based on open-source kernel platforms. The kernels are TMO-Linux for general embedded systems and TMO-eCos for small embedded/ubiquitous systems. In this paper, after introducing the general architectures and functions of the developed TMO kernels for embedded systems, some comparisons of their characteristics are also given to specify their suitable usage domains.

1 Introduction

The TMO (Time-triggered Message-triggered Object) model formalized earlier by Kim and Kopetz [1] is a robust model in developing hard and soft real-time applications. With the TMO model, both functional and timing behaviors of system can be explicitly specified at design time. To support real-time executions of TMO applications in distributed environments, several types of kernels and middlewares[3,4] have been developed on multiple OS platforms. The kernel engines are the TMO-Linux[6] and TMO-eCos[8]. While middleware engines have been mainly used in soft real-time applications such as multimedia services[7] and real-time simulations[2,7], the embedded TMO kernels have been mainly used for real-time control systems such as robots[8].

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 71–82, 2006.

Based on the recognition that supporting of time/message-triggered tasks and distributed IPC between collaborating nodes makes developments of real-time embedded/ubiquitous applications very easier and robust, some research groups of the KETI (Korea Electronics Technology Institute) also has started porting a TMO engine to an RTOS for sensor network. In this paper, the general architecture and key functions of the developed TMO kernels are introduced briefly and some comparisons of their characteristics are also given to specify their suitable usage domains. In section 2, the concept of TMO and TMO programming are introduced briefly and in section 3 the general architecture of TMO kernels is described with their implementation techniques. After some comparisons of the existing TMO engines are given in section 4, we will conclude in section 5.

2 Related Works

2.1 TMO Model and Programming

The TMO model was formalized as a timeliness-guaranteed computing model for real-time distributed systems. As a new real-time programming paradigm, the TMO model aims at combining object-oriented programming with concurrent programming, network transparent object collaborations and imposing timing constraints to object methods. Figure 1 shows the structure of a TMO instance. A TMO instance consists of three major parts: an ODS (Object Data Store) and two types of active methods (Spontaneous Methods and Service Methods). The characteristics of the TMO model can be summarized as follows.

- A time-triggered member-thread, named SpM (Spontaneous Method), is activated automatically by a real-time clock when a given timing constraint; called AAC (Autonomous Activation Condition); is satisfied. An AAC consists of a period, a finish-deadline and a start-stop duration.
- A message-triggered member-thread, named SvM (Service Method), is activated by a message from a source outside a TMO. Once an SvM is invocated, it must be scheduled to finish its service within a given deadline. However, SvMs cannot disturb the executions of SpMs. This rule is called the *Basic Concurrency Constraint* (BCC) [1]. The number of inputs, outputs and logic gates in it.

Following is the typical execution scheme of an SpM.

```
SpM_register (AAC constraint, SCHED policy)
while (SpM_wait_invocation ()) { // wait invocation
      do a periodic job within the given deadline;
}
```

In the above, *SpM_register*() is used to transform a normal thread into an SpM with a timing constraint. The timing constraint (AAC class) includes an invocation-period, a finish-deadline of each turn of execution and a start-stop duration. The *policy* argument of SCHED type will be one of SCHED_EDF

(Earliest Deadline First), SCHED_LLF (Least Laxity First) and SCHED_FIFO. A call to *SpM_wait_invocation*() function lets the calling thread to be blocked until the next periodical invocation by the deadline scheduler.

Management of a message-triggered SvM is always done in conjunction with the logical multicast IPC of a TMO kernel, because an SvM is always activated by a message received from a multicast IPC channel. Following is the programming pattern for an SvM.

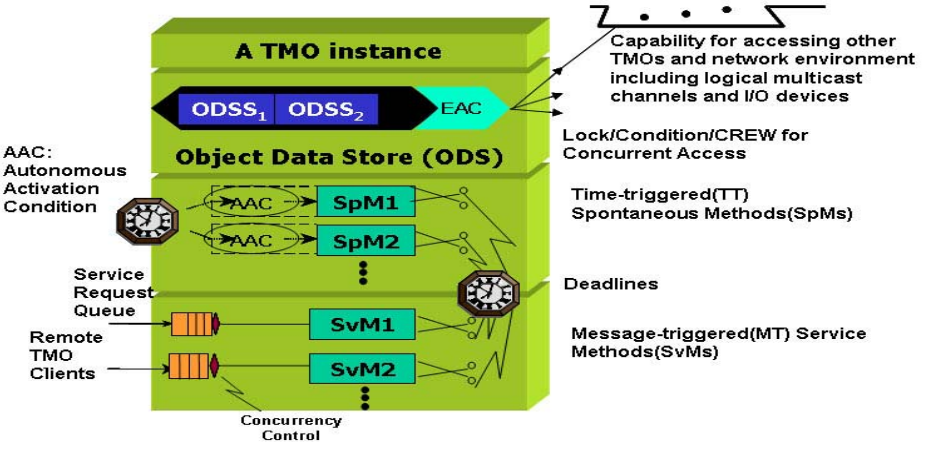


Fig. 1. Structure of the TMO model [1]

SvM_register (deadline);
alloc_channel (channel_id, channel-attribute);
while (*SvM_wait_invocation*(channel_id,
message-pointer) == TRUE) {
do a service job within its given deadline;
}

The deadline given at the time of registration will be the time limit form an invocation to the next call to *SvM_wait_invocation*(). A call to *alloc_channel*() is a request to allocate a new channel or to get a right to access to an existing channel. A channel maybe a local or a distributed one as designated by the attribute. When a remote or local TMO method wants to request a service to an SvM, it will use the following call.

bool_t *send_message*(int channel_id, Message *pMessage, int mode);

The *mode* argument designates the way of wake-up. FIFO will invocate the first thread waiting on the channel and BROADCAST will wake up all threads in the waiting queue. Following is an example of a TMO class program in C++ with the help of TMOSL (TMO Support Library). TMOSL is a macro/object library to support TMO programming in C++.

Table 1. A TMO class program example

```
class TMO_name : public TMO
{

private :
   void method1(void);
   void method2(void);
   SpM (method1);
   SvM (method2);
private:
    rt_cond Cond_name;
// Condition for sync.
    rt_lock Lock_name;
// Lock for sync.
    time-critical-data;

public :
   void InitInstance(void);
       :
};
void
TMO_name::InitInstance(void)
{
// SvM, SpM initialization
    SvM_Init (method1);
    SvM_Init (method2, 3);
       :
       :
}
```

```
SpM_Body (TMO_name, method1)
void TMO_name::method1(void)
{
   SpM_register (AAC, SCHED_LLF);
   while(SpM_wait_invocation()){
      do something periodic;
      Lock_name.ex_lock();
         update ODS;
      Lock_name.ex_unlock();
    }
    SpM_deregister();
    //returns to a normal thread
}
SvM_Body(TMO_name, method2)
void TMO_name::method2(void)
{
    SvM_register (deadline);
   alloc_channel (channel_id,
      channel-attribute);
    while(SvM_wait_invocation
      (channel_id, messagep)) {
         do something;
         Lock_name.ex_lock();
         update ODS;
         Lock_name.ex_unlock();
    }
    SvM_deregister();
}
```

3 Key Components of a TMO Kernel and Implementation Techniques

A TMO kernel must have the following components to support real-time executions of TMO's.

- A deadline driven scheduler that performs on-time activations of SpM's and scheduling SpM's and SvM's by their deadlines;
- An inter/intra-node logical multicast IPC subsystem that performs local/distributed communications and activates SvM's by arrived messages;
- A clock synchronization module among distributed nodes;

Among the above key components, introductions to the scheduler and IPC system will be given in this paper.

3.1 Deadline Scheduler

TMO kernels support three policies for deadline-driven scheduling. They are the EDF (Earliest Deadline First), LLF(Least Laxity First) and FIFO-based serialized (context-switching-free) scheduling for hard real-time computing. All three types of schedulers currently work well, however, to aid the selection of a suitable scheduling policy, several kinds of pre-analysis tools have been developed. Before introducing the tools and process, the implementation of the scheduler is introduced first.

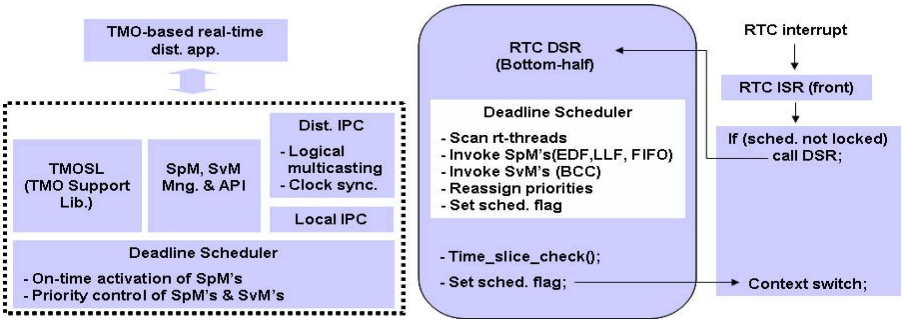


Fig. 2. Architecture of a TMO kernel and the deadline scheduler

For all scheduling policies, time-tick hooking is necessary to let the scheduler perform periodical monitoring and controlling of real-time tasks. Actually, the deadline-scheduler has been inserted into the timer interrupt bottom-half handler of a kernel (Linux, eCos) as shown in figure 2. The scheduler mainly does activation of waiting SpM's on time and controlling priorities of active SpM's and SvM's by their deadlines, laxities or invocation order. In case of EDF and LLF, a preemption can be occurred either at the end of an interrupt handling or at the end of a system-call if the scheduler-lock is not set. In case of FIFO-based serialized scheduling, the priorities of real-time tasks are set according to the order of invocation times and preemptions are not allowed. Of course, all SvM's are scheduled by the BCC policy.

3.2 The Process for Selecting a Scheduling Policy with a Serializability Analyzer

To aid the selection of a suitable scheduler, several kinds of pre-analysis tools have been developed. Following is an introduction to a tool called a schedulablilty analyzer. In the selection process, the schedulability-level for a group of SpM's on a single-CPU system is found to be one of three levels, called "Static-seriablizable", "Dynamic-serializable" and "Preemption-required" by the analysis tool. And the level found is used to determine a suitable scheduling policy. If

the schedulability-level is found to be static or dynamic serializable, the FIFO-based serialized scheduling policy can be chosen. If the schedulability-level is found to be preemption-required, the EDF or LLF policy must be taken. The followings are some definitions relevant to the selection process.

The schedulability-level of a system that includes a set of SpM's; $SpMT = \{SpM_0\ (P_0, E_0, D_0),\ SpM_1\ (P_1, E_1, D_1),..,\ SpM_{n-1}\ (P_{n-1}, E_{n-1}, D_{n-1})\}$; where i is the serial number of an SpM, P_i is the period, E_i is the WCET and D_i is the deadline of SpM_i; is defined as follows.

Definition 1. *A group of SpM's is called static-serializable if there exists at least an execution scenario in which all SpM's can start their periodic executions exactly at their designated times and can finish their executions within the pre-given deadlines without any preemption on a single CPU system. In this definition, it is assumed that the initial-start-time of each SpM_i can be adjusted in the range of [0, P_i) at design time. The times of subsequent periodic invocations are fixed by the initial-start-time.*

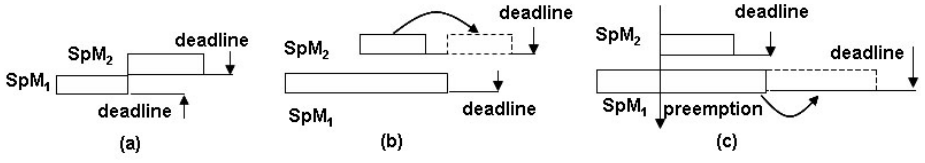


Fig. 3. (a) static-serializable (b) dynamic-serializable (c) preemption-required

Definition 2. *A group of SpM's is called dynamic-serializable if there exists al least one execution scenario in which all SpM's can finish their executions within their pre-given deadlines without any preemption on a single CPU system. In this schedulability level, some instances of periodic invocations of each SpM may be delayed by case-by-case amount of times, however, all SpM's can finish their periodic executions in their deadlines without any preemption. The assumption on the adjustments of initial-start-times of SpM's is same as in the static-serializable case.*

Definition 3. *A group of SpM's is called preemption-required if the group is neither static-serializable nor dynamic-serializable, but all SpM's can finish their executions within the pre-given deadlines under the condition that preemptions and invocation-delays are possible.*

Example 1 (a static-serializable case 1). Let's consider the case that there are four SpM's: $SpM_0\ (T, E_0, D_0)$, $SpM_1(2T, E_1, D_1)$, $SpM_2(2T, E_2, D_2)$ and $SpM_3\ (4T, E_3, D_3)$. To cover all possible execution scenarios, locating executions of SpM's must be considered in the time-range of $[0, 6T)$ where $6T$ is the L.C.M of periods of SpM's. For this range of interval, all alternatives of execution scenarios that are created by adjusting initial start-times of SpM's must be considered. For easier analysis, let's fix the initial-start-time of SpM_0 (with the shortest period)

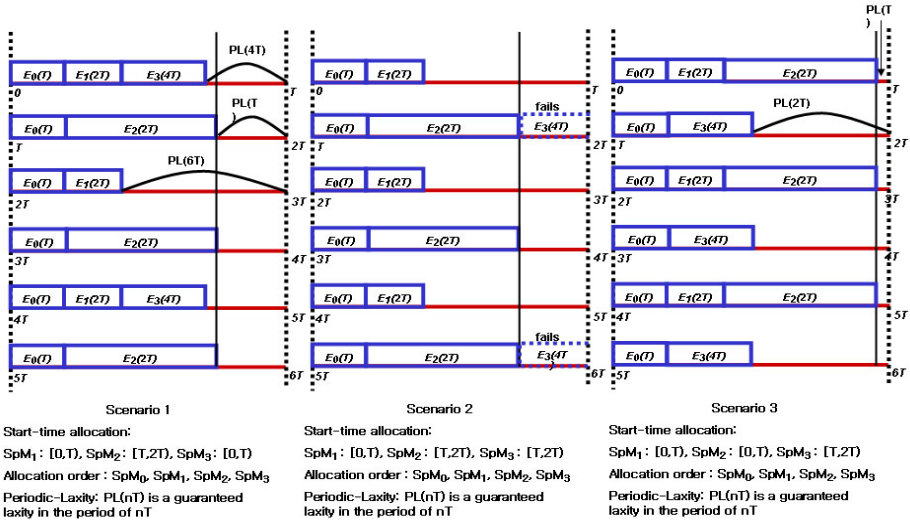


Fig. 4. Three representative scenarios generated by the analyzer

at time 0 at first, so that SpM_0 can be invocated at $0, T, 2T$,,, subsequently. And then, by the assumption of adjusting initial-start-times in definition 1, the start time of SpM_1 may be located at somewhere possible ether in $[0, T)$ or in $[T, 2T)$ to avoid an overlap. This means that we have two choices in selecting intervals for locating the initial start time of SpM_1, two for SpM_2, and three for SpM_3 respectively. So we come to have 12 (= 1*2*2*3) choices in total. For each choice of selecting intervals for initial-start-times, the order of allocating executions of SpM's must be considered too. That is, for each choice of 12 alternatives for initial-start-times, all permutations (6 ways) of orders of locating three SpM's except SpM_0 must be tried. This is because the distribution-patterns of idle time-slots may differ from each other according to the allocation order in some cases.

Figure 4 shows three representative scenarios from all possible 12 * 6 allocation alternatives. In figure 4, the scenario 2 needs some overlaps, however, other two scenarios does not. So, by definition 1, the example SpM-group is classified into "static-serializable". The analyzer also finds out the pattern of continuous idle time-slots denoted by PL_{nT} (Periodic Laxity) for each serializable scenario. The PL_{nT} means the size of an idle time slot that appears once per every interval of nT in a certain scenario. They can be used to specify the guaranteed service ratios of SvM's. For example, the scenario 1 has three kinds of idle time-slots: $PL_T = T - (E0 + E2)$, $PL_{4T} = T - (E0 + E1 + E3)$ and $PL_{6T} = T - (E0 + E1)$. If we select to use PL_T for an SvM with lesser execution time than PL_T, the scheduling system comes to be capable of executing the SvM for every event-message occurred at most once for every interval of T. Of course, the finish deadline of an SvM must be adjusted by considering the worst case because the arrival time of an event is not known.

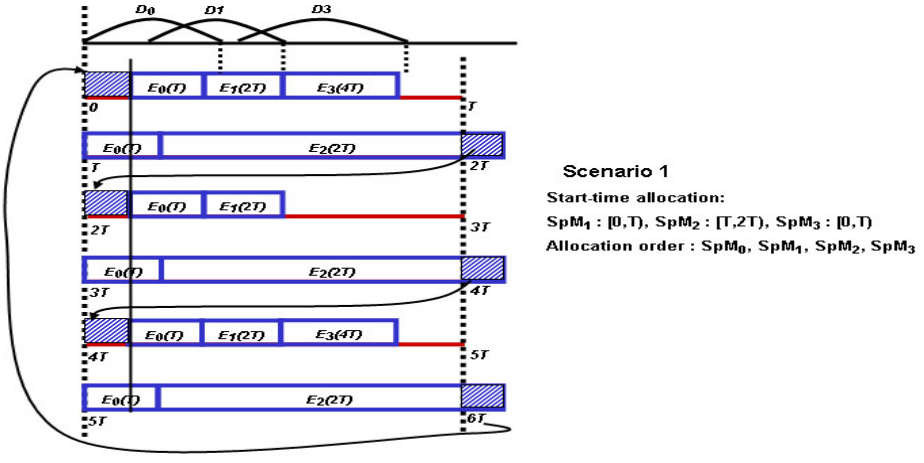


Fig. 5. A dynamic-serializable case

Example 2 (a static-serializable case 2). Let's consider the case that there are four SpM's: SpM_0 (T, E_0, D_0), $SpM_1(1.5T, E_1, D_1)$, $SpM_2(3T, E_2, D_2)$ and SpM_3 $(4T,\ E_3,\ D_3)$. In this case, not all the periods of SpM's are integer-multiples of the smallest period of the group. To simplify the process, we assume that there is a dummy SpM: $SpM_d(0.5T, 0, 0.5T)$: whose WCET is zero. By substituting 0.5T by T', periods of SpM's will be represented as T', $2T'$, $3T'$, $6T'$ and $8T'$ respectively. After this substitution, the same analysis process can be applied as in integer-multiple cases except the fact that some allocations of executions can be overlapped with the SpM_d.

Example 3 (a dynamic-serializable case). If the analyzer cannot find any static-seriablizable scenario for an SpM-group, it then starts a process to find dynamic-serializable scenarios by shifting (delaying) overlapped executions as much as required. In the scenario of figure 5, the execution of SPM_2 cannot avoid an overlap with the execution of SpM_0. Then the analyzer tries to shift the executions of SpM's that appear at the next interval by the required time-amount repetitively. Of course, a shift of an execution is only allowed under the condition that they can finish their executions within the given deadlines. Since, a deadline is a time-limit that is applied from the original invocation time, a shit will be allowed only when the time to the deadline covers the amount of shift-delay and WCET. If the analyzer finds out at least one scenario that meets the condition of no-deadline-violation in spite of case-by-case delays, the SpM group is said to be "dynamic-serializable". As in the static-serializable case, periodic laxities can also be found for the declaration of SvM services under the BCC policy. If a group of SpM's is found to be neither static-serializable nor dynamic-serializable, then the EDF or LLF scheduling policies with preemptions must be applied.

3.3 Real-Time Distributed IPC Subsystem

The distributed IPC subsystem of the TMO kernels is a network-transparent communication tool for collaborating nodes. It supports same local and remote API's based on logical multicast channels. It consists of two-layers (figure 6); the local chan-nel IPC and the distributed IPC layer. This layered architecture is to support flexible target configurations. The local IPC layer supports intra-node IPC between real-time tasks in kernel mode and this is done by managing message/task queues for channels and waking-up/queuing of SvM's. The interfaces for SvM's such as *send_message*() and *SvM_wait_invocation*() perform sending and receiving (block/wake_up) of messages using these queues.

The distributed IPC layer is a library that performs receiving/multicasting messages from/to remote TMO nodes. Collected messages from remote nodes are delivered to the bottom layer for message-queuing or for waking-up waiting SvM's. This layer is composed of a protocol adaptation module, protocol-independent IPC functions and a real-time task, named IMMT (Incoming Message Management Task) waiting for event/control messages from remote nodes.

Currently, the distributed IPC layer has been built only on UDP-broadcast/multicast, IEEE1394 and serial-Bluetooth, however, the protocol platform can be easily exchanged to other wired/wireless protocols such as Zigbee, etc., because the library has a separated protocol adaptation module. This layer also has a module for dealing with handshaking, joining and leaving of distributed nodes. This module also can be easily extended to have an ad-hoc routing for sensor networks.

Among several protocol platforms used for TMO-kernels, the 1394-based distributed IPC system has been developed for industrial applications in which a

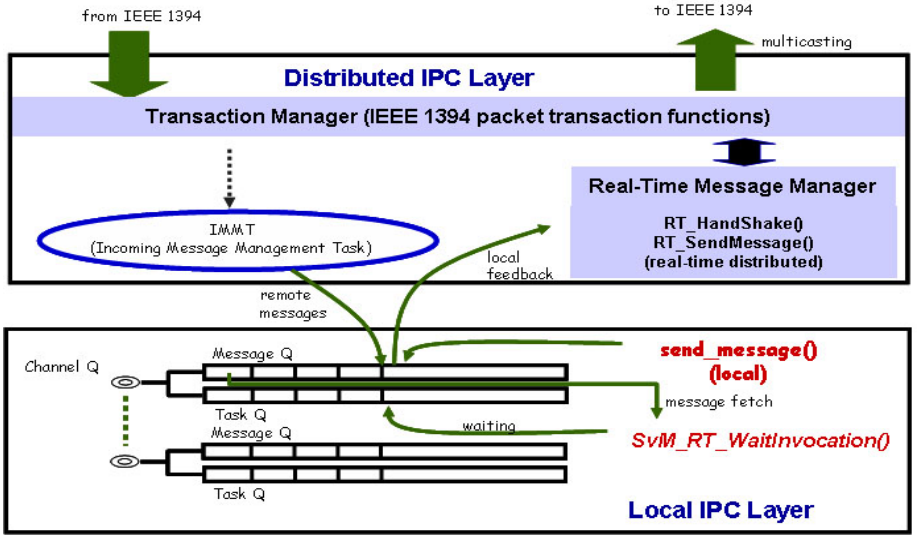


Fig. 6. A dynamic-serializable case

system is controlled by time-critical data from several sensing devices. This work has been done by considering the fact that a common critical issue in many industrial distributed control systems is to establish a real-time communication infrastructure (fieldbus). In this paper, the structure of the 1394-based IPC system is briefly introduced for further extension to a wireless protocol-based IPC system for sensor networks.

With this IPC system, TMO nodes are physically connected by the IEEE 1394's tree-topology and a group of dedicated isochronous slots are allocated to each node for its logical channels. This means that a logical channel is composed of multiple physical isochronous channels each of which is one for a node. That is, each receiving node using a logical channel collects data form multiple corresponding iso-channels. Since each node sends sensor data using different iso-channels in time-triggered way, time-critical processing of sensor data is possible in a predictable amount of time. And for handshaking of kernels, each kernel uses its own iso-channel respectively.

4 Potentiality of TMO and Its Kernels as Ubiquitous Computing Platforms

In this section, some characteristics of the TMO model and its kernels are discussed to figure out their possibilities of being used in ubiquitous computing [table 2].

Table 2. Characteristics of the TMO model and its kernels

	units/functions		TMO-eCos
TMO model	TMO object instances		Componentization of member-threads in an object-oriented way
	SpM		Real-time data acquisition, Event polling, time-critical processing, Deadline-driven power management
	SvM		Sporadic event processing, Real-time data acquisition
	Distributed IPC		Protocol independent IPC bet, neighbor nodes, Make TMO migration easy, Collaboration of heterogeneous kernels
TMO Kernels	TMO-Linux	Characteristics	From 600KB,
		Usage domain	Sensor data acquisition server(clustering), Safety-critical processing of data, Control host for digital devices
	TMO-eCos,	Characteristics	From 30KB, highly configurable, Higher scheduling accuracy
		Usage domain	Small embedded devices, Mote devices, Robots, Higher scheduling accuracy
	TMO modules in a kernel		Highly modular, Easy porting to a smaller RTOS for mote devices (TMO-NanoQ+ is under development)
Tools	Schedulability analyzer		Safety-critical modeling with TMO

Time-triggered and message-triggered methods of TMO instances offer a very natural and robust way to build time critical applications that handle sensors and actuators. The distributed IPC system makes collaborative processing of control and data-acquisition nodes very easy and moreover it offers an easy TMO-migration mechanism due to location independency of the IPC. Migration of TMOs can be used for saving memory and power resources in a sensor network. As concerns with kernel usages, the TMO-Linux is suitable for a (embedded) server that handles real-time data collection, processing and high-level control while the TMO-eCos can be used as a platform for smaller embedded devices such as sensor nodes, actuator nodes and robots because of its compactness and higher scheduling accuracy. The TMO subsystems such as the deadline scheduler, the distributed IPC system and the TMO support library can be easily ported to smaller kernels for mote devices because they have highly modular structures. Actually, a new smaller TMO kernel for mote devices is under development based on the NanoQ+ which is a sensor node kernel developed by ETRI, KOREA.

5 Conclusions

The importance of providing a design-time framework on which one can easily specify the temporal behaviors of a real-time embedded/ubiquitous system, has been recognized broadly as well as the importance of faster service-latencies of an RTOS. The goal of the TMO model is focused on to provide such a framework that supports design-time guarantee and easy real-time programming tools. The development of TMO kernels such as the TMO-Linux and TMO-eCos has been done to accommodate such a demand. And because the key components such as the deadline scheduler and the distributed IPC system have been designed to be highly modular and interoperable, new TMO engines on other smaller RTOSes can be easily developed.

Base on the fact that the developed TMO-kernels are currently being used together successfully in developing real-time embedded applications, we hope that the TMO scheme becomes to be expanded into various embedded operating systems including RTOSes for sensor nodes.

Currently, a number of new TMO kernels are currently being developed based on some smaller RTOSes such as microCOS-II^{TM} and NanoQ+^{TM} (an RTOS for sensor networks developed by ETRI KOREA) by using the architecture and modules presented in this paper. All these efforts are to expand the usage domains so that the TMO scheme covers the areas of real-time server, embedded devices and sensor networks.

Acknowledgements. This research was supported by University IT Research Center Project funded by Ministry of Information and Communications of Korea and by Hankuk University of Foreign Studies.

References

1. Kim, K.H., Kopetz, H.: A Real-Time Object Model RTO.k and an Experimental Investigation of Its Potentials. In Proc. of the 18th IEEE Computer S/W & App. Conference (1994) 392–402
2. Kim, K.H.: Real-Time Simulation Techniques Based on the RTO.k Object Modeling. In Proc. of the 20th IEEE Computer S/W & App. Conference (1996)
3. Kim, K.H.: Object-Oriented Real-Time Distributed Programming and Support Middleware. In Proc. of ICPADS'00 (2000) 10–20
4. Kim, J.G., Cho, S.Y.: LTMOS: An Execution engine for TMO-Based Real-Time Distributed Objects. In Proc. of PDPTA'00 Vol. V. (2000) 2713–2718
5. Kim, M.H.: Time-triggered Message-triggered Object Modeling of a Distributed Real-time Control Application for its Real-time Simulation. In Proc. of the 20th IEEE Computer S/W & App. Conference (2000) 549–556
6. Kim, J. G., Kim, M.H.: TMO-Linux: A Linux-based Real-time Operating System Supporting Execution of TMO's. In Proc. of ISORC'02 (2002) 288–296
7. Jo, E. H., Kim, M. H., Kim, J. G.: Framework for Development of Multimedia Applications Based on the TMO Structuring Scheme. In Proc. of WSTFES'03 (2003) 35–38
8. Kim. J. G., Kim, M. H.: TMO-eCos: An eCos-based Real-time Operating System Supporting Execution of TMO's. In Proc. of ISORC'05 (2005) 182–189

An Embedded System Design for Ubiquitous Speech Interactive Applications Based on a Cost Effective SPCE061A Micro Controller

Po-Chuan Lin, Jhing-Fa Wang, Shun-Chieh Lin, and Ming-Hua Mo

Department of Electrical Engineering, National Cheng Kung University
701 No.1, Ta-Hsueh Road, Tainan City Taiwan R.O.C.
Tel.: 886-6-2757575 Ext. 62341, Fax: 886-6-2761693
{tony, wangjf, Jason, handson_mo}@icwang.ee.ncku.edu.tw

Abstract. Previous researches show that speech interactive systems provide users not only a user-friendly interface but a feedback scheme in speech. However, these systems are built on PCs but not available anytime or anywhere. Therefore, an embedded system design is proposed for ubiquitous speech interactive applications. In addition to provide multiform interfaces, this paper also proposes multiple speech interactions by four scenarios: remote control, dictionary query, programmable dialogue, and lexicon-in sentence-out. For cost down purpose, two issues are presented to allow real-time process on a low cost and resource-limited SPCE061A (49.152 MHz) micro controller unit (MCU). One is the short-time start/end point detection of speech streams and the other is the timing arrangement of feature extraction/saving. By utilizing 18 speech streams (average 2.56 seconds) in our experiments, the average processing time for one recognition operation is 0.92 second and the recognition rate is 89.39 %.

1 Introduction

With recent advances in multi-media technology and database processing, computer has emerged concerning the practicality of supplementing the human-machine interface. At the present days, the speech recognition technology makes computers become concrete for user to talk with it. Computers starts up speech recognition technique to understand what the speaker says, after appropriate flow path arrangements, users can interactively operate or talk with a machine. This interaction is finished by comparing input speech with speech templates saved in database. There are many speech interactive products which use speech recognition technique on the market such as EZ Talk [3], Talk to Me [4], MyET [5], LiveABC [6], etc. However, these products could be only worked on a desktop computer. It is not portable and not available anytime or anywhere. Nowadays, MCU (Micro Controller Unit) becomes more powerful and cheaper, it is possible to execute speech recognition calculation by a stand along embedded system. For portable devices, low-cost design is a key component in providing an infrastructure for developing regions [8]. We implement the entire ubiquitous speech interactive applications based on a low cost MCU-SPCE061A [2]

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 83 – 92, 2006.

MCU. The SPCE061A requires only a few peripheral components for workable speech applications. We implement the dynamic time warping (DTW) algorithm for speech recognition and use it as kernel to build four speech interactive scenarios: remote control, dictionary query, programmable dialogue, and lexicon-in sentence-out. The framework of this paper is as follows. The proposed interactive speech system is presented in Section 2. In Section 3, the embedded system design of the proposed system is presented. Section 4 shows the experimental results. Finally, generalized conclusions are presented in Section 5.

2 The Proposed Interactive Speech System

Operation of the proposed speech interactive system is divided into two phases – a scenario setting phase and an interactive phase. Figure 1 shows the framework of the proposed speech interactive system. During the scenario setting phase, users first select one scenario for interaction. The four scenarios are remote control, dictionary query, programmable dialogue, and lexicon-in sentence-out. Each scenario is used to provide different interactive response. After selecting one of the proposed scenarios, the proposed system loads the data of selected scenario including speech templates and response data. The proposed system uses a set of prompts for recording speech interactive data. In designing the prompts for read speech collection, we tried to pick speech templates and their response data, so that the proposed system could generate a response for users. The speech templates of selected scenario are used to recognize users' speech input for response generation. The response data related to the speech templates is used to generate system response for users.

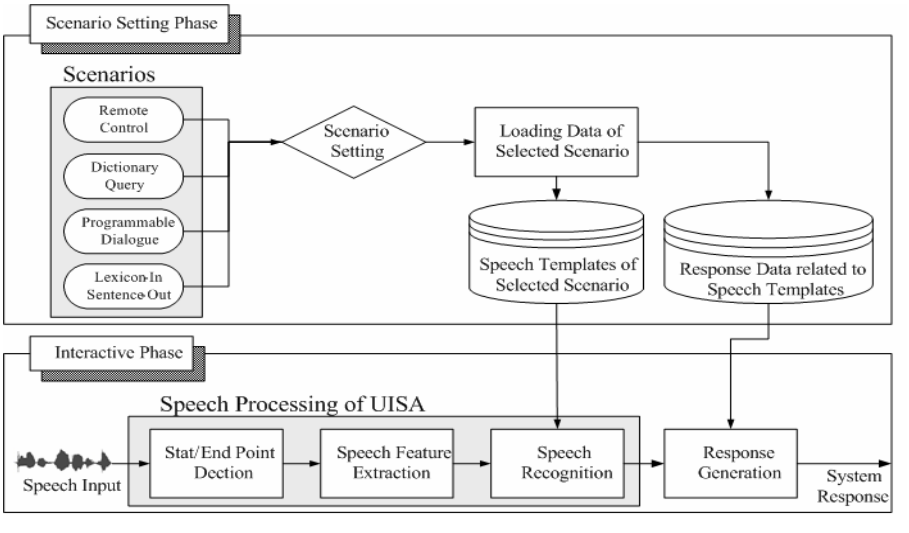


Fig. 1. The framework of the proposed speech interactive system

During the interactive phase, after detecting start/end points of one speech stream input, 10-dimension linear prediction coefficient cepstrums (LPCCs) are extracted due to the advantages of faster operation and simpler hardware design [7]. The speech recognition process is used to measure dissimilarity between speech input and each speech template of the selected scenario. After recognizing the input speech, the hypothesized response is determined using the template with the lowest dissimilarity. The hypothesized response is used to generate system response via the response data related to speech templates during the final response generation process.

2.1 Remote Control (RC)

Table 1 is an interactive example for a remote control car. When users record the speech templates of commands for remote control, the system establishes the link of response for commands. If users input a command, the system will feedback the corresponding control to the car by recognizing the input commands with speech templates.

Table 1. An interactive example for a remote control car

Speech Templates recording and response establishment	System	:The leading Type training. Please follow me. (prompts)
	System	:Go Forward
	User	:Go Forward
	System	:Back
	User	:Back
	System	:Turn Left
	User	:Turn Left
	System	:Turn Right
	User	:Turn Right
	System	:Begin to recognize
Interactive Phase	User	:Back
	System Response	(The remote control car is moving back)
	User	:Turn Left
	System Response	(The remote control car is turning left.)

2.2 Dictionary Query (DQ)

Table 2 is an interactive example for dictionary query by the language pairs of English and Chinese. When users record the speech templates of lexicons for dictionary query, the system establishes the link of translations from dictionary. If users query a lexicon, the system will show the corresponding translations by the recognized lexicon.

Table 2. An interactive example for dictionary query

Speech Templates recording and response establishment	System	:Please follow me …Apple
	User	:Apple
	System	:Please follow me …Book
	User	:Book
	System	:Please follow me …Basketball
	User	:Basketball
Recognition	User	:Apple
	System Response	:Apple A-p-p-l-e Apple 蘋果 This is an apple.

2.3 Programmable Dialogue (PD)

Figure 2 is an interactive example of using programmable dialogue. Users can be a questioner or answerer. For the programmable dialogue, the question-answer (Q&A) index table is established. The Q&A index table indexes that which question corresponds to which answer. When users record the question and answer into database, the system establishes the link of Q&A and record in Q&A index table. If users input a question, the proposed system feedbacks the corresponding answer to the users by checking index table. If users add new dialogue into database, the proposed system will modify the Q&A index table automatically.

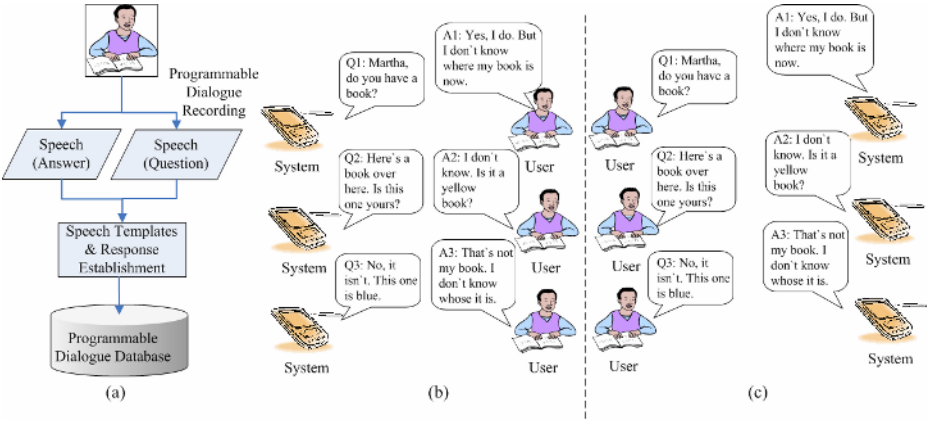


Fig. 2. An interactive example of using programmable dialogue: (a) the framework of programmable dialogue recording; for the same question-answer dialogue, (b) system query and user answer, and (c) user query and system answer

2.4 Lexicon-In Sentence-Out (LISO)

Table 3 shows an interactive example for lexicon-in sentence-out. When users record the speech data of lexicons by the system prompts, the system establishes the link of positions of lexicons in the selected sentence. After recording all lexicons, the system outputs a sentence which is generated from the recorded lexicons.

Table 3. An interactive example for lexicon-in sentence-out

System	:What is your name?
User	:TONY
System	:How old are you?
User	:Thirty
System	:Where do you live?
User	:Taipei
System	:What is your interest?
User	:Play basketball and watch TV
System	:My name is Tony. I am thirty and live in Taipei. I like to play basketball and watch TV.

3 The Embedded System Design of the Proposed System

3.1 Hardware Architecture

For cost-effective, low-power, and high-performance microcontroller solution in small die size. The Sunplus 4.5V SPCE061A (49.152 MHz) MCU [2] is adopted for our speech signal processing. Comparing to the common ARM series chips, this low cost 16-bit RISC controller features dedicated voice recognition products, intelligent interactive devices and advanced educational toys, etc. Besides, it also provides a flexible, powerful integrated development environment (IDE). The built-in components we adopted included 32K Words (64K Bytes) flash memory for instructions, and 2K Words (4K Bytes) serial SRAM for APIs, speech features and speech input buffers, 16-bit timer/counter, 10-bit ADC/DAC converter for concurrent real-time speech input/output, amplifier for microphone input, AGC function and PLL for clock generation. Figure 3 depicts the block diagram of our testbed architecture. In addition to the internal memory, we also use an external 256K Words (512K Bytes) flash memory for saving compressed speech streams. The photo of the prototype system is given in Fig. 4.

3.2 A Simple Manner for Start/End Points Detection

For DTW process with a resource-limit system, we simply detect the start/end points by speech energy (the square value of the acoustic amplitude). Fig. 5 indicates our simple manner to speed up the start/end point detection for recording speech streams with different length. For every speech streams, we arrange equal-size memory spaces for them and initialize their contents as silence data. When a recording process starts, our system start to wait until an energy sum of 80 speech samples exceeds an experimental threshold, the speech amplitudes from ADC buffer are starting to write into the memory. By the same method, the memory writing process is stopped by comparing incoming samples with an energy threshold.

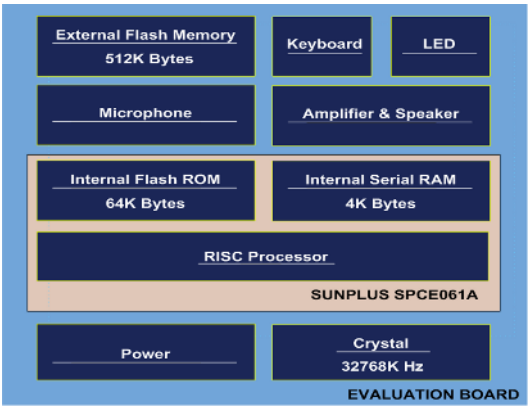


Fig. 3. Hardware block diagram

Fig. 4. Prototype of proposed system

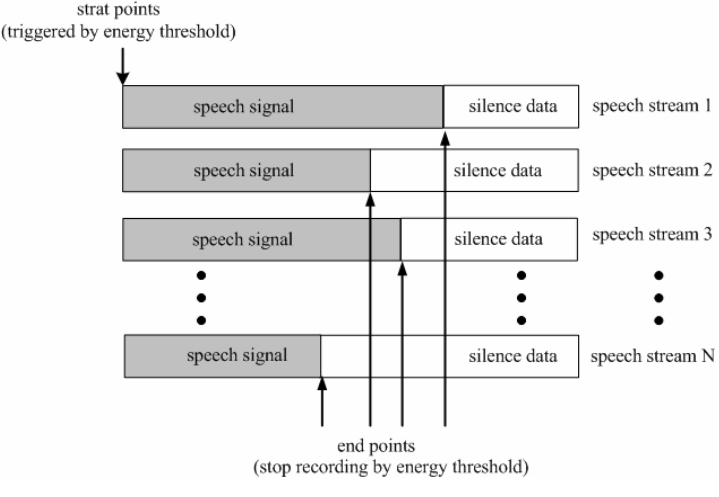


Fig. 5. A Simple manner for start/end point detection

When the DTW process starts to compute distance between speech stream pairs, we only take the non-silence data from memory for DTW processing. By using this memory access technique, the start points of different speech streams can be kept pace for DTW distance computing. This manner can easily record speech streams with different length. Considering to a resource limited device, it speeds up the start/end point detections for a real-time system.

3.3 Real-Time Feature Extraction and Memory Saving

We propose a timing arrangement to achieve the real-time speech feature extraction and saving. This work is finished by the timing arrangement among interrupt counting, input double buffers switching, memory access and ADC controlling. After a speech frame (32 ms) is received from the ADC, both of the LPCCs speech feature extraction and memory saving are finished before the incoming of next frame. Thanks to the high performance of the SPCE061A processor, we have enough time to deal with the continuous speech inputs and obtain the real-time LPCCs memory saving. The flowchart of our program scheduling is shown in Fig. 6. The total computation time of LPCC computation and memory saving only takes 0.115 ms which is faster than the speech sampling period (0.125 ms); thus making our system to be workable for real-time demand.

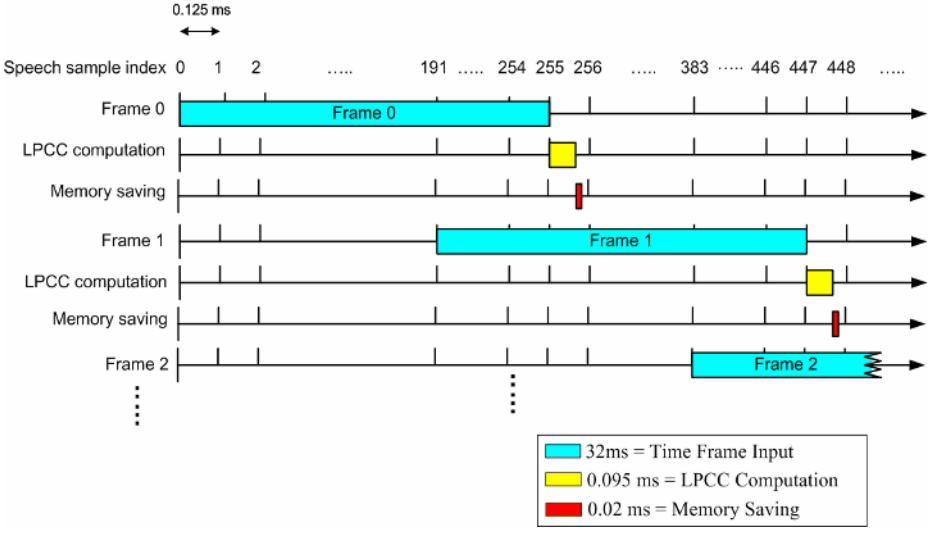


Fig. 6. Program scheduling for real-time speech feature extraction and recording

3.4 Memory Allocation and Reduction

Figure 7 depicts the memory allocation. Moving to an embedded platform meant that we had to conserve memory. We wanted to incorporate as little memory in the system as possible to save on both system area and power consumption. An optimizing C/C++ compiler was also adopted. We adopt the SCAM_S240 (2.4K bps) coding scheme [2] as our speech compression for saving speech streams. The final database size and memory consumptions was shown in Table 4.

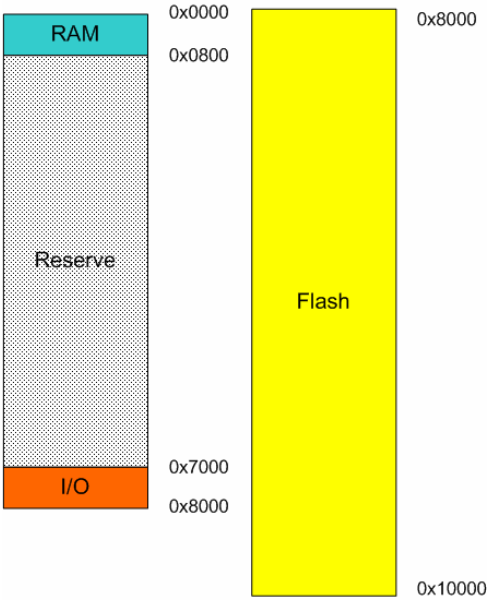


Fig. 7. Memory allocation

Table 4. Database size and memory consumptions

Scenarios	Remote Control	Dictionary Query	Programmable Dialogue	Lexicon-In Sentence-Out
Database Size	4 comments	10 queries/31 sentences	4 keywords/4 sentences	10 sentences
Average Length (seconds)	1.5	2/3	5/5	5
Serial RAM	2.242K Bytes	3.6K Bytes	3.4K Bytes	2.5K Bytes
Flash ROM	54K Bytes	278K Bytes	68K Bytes	312K Bytes

3.5 Software Architecture

Figure 8 shows the software architectures that focus on our embedded system for multiple speech interactions. The hardware APIs of external flash memory access, LEDs, keyboard and the ADC/DAC have been developed for the SPCE061A MCU. All of the interactive scenarios were implemented base on following speech processing units: DTW, feature extraction, start/end points detections and SACM_S240 speech coding. For speed up the DTW process, the fixed-point arithmetic of the SPCE061A processor; accuracy studies of the finite length effect was conducted.

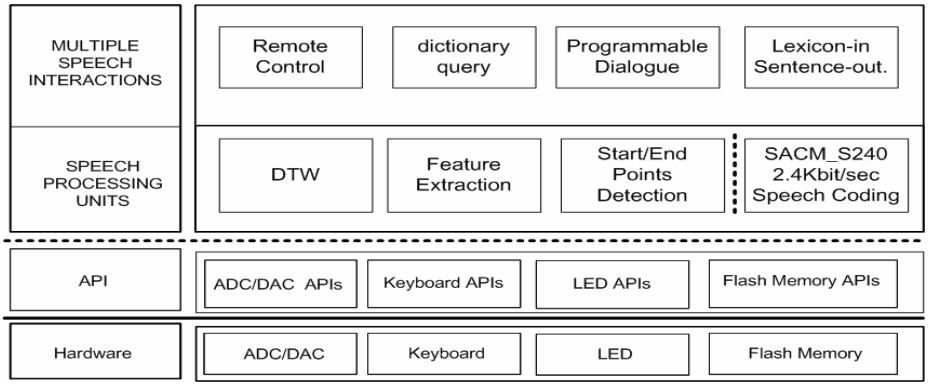


Fig. 8. The overall software architecture

4 Experimental Results

The sampling frequency was 8 kHz, and the frame size was 32 ms with 8 ms overlap. For noise reduction, a 16-order FIR digital filter was implemented as the speech pre-processing. The 10-dimension LPCCs were extracted to represent a speech frame. For average 2.56 seconds speech streams, the average processing time for one DTW operations is 0.92 second. Table 5 shows the performance comparisons for different interactive scenarios. Three 1.5V 2000 mAh batteries could be used for 66 hours and 53 minutes.

Table 5. Performance comparisons for different interactive scenarios

Scenarios	Remote Control	Dictionary Query,	Programmable Dialogue
Database Size	4 comments	10 queries	4 keywords
Recognition Rate	89.64%	86.42%	92.12%

5 Conclusion

A cost effective embedded system design for multiple speech interactions is presented in this paper. The start/end point detection of speech streams and program scheduling of feature extraction/saving are specially designed to suit with the resource-limit identity of an embedded system. For speech interactive applications, we propose four scenarios: remote control, dictionary query, programmable dialogue and lexicon-in sentence-out. Besides, some speech processing issues for embedded computing such as hardware/software design and memory reduction are also discussed. The average recognition rate is 89.39 %. This result shows that our design can be utilized for various speech interactive applications such as intelligent toys, hand-free controlling, machine learning, and speech retrieval, etc.

References

1. Ney, H., Nießen, S., Och, F.J., Sawaf, H., Tillmann, C., and Vogel, S.: Algorithms for Statistical Translation of Spoken Language. IEEE Trans. Speech and Audio Processing 8 (1) (2000) 24–36
2. Data sheet, SPCE061A, *Sunplus university program,* Sep. 2003.
3. EZ talk, http://www.eztalk.to/
4. Talk to Me, http://www.auralog.com/
5. MyET, http://www.myet.com/en/Index.htm
6. LiveABC, http://www.liveabc.com/english/
7. Wang, J.F., Lin, S.H., and Yang, H.W.: Multiple-Translation Spotting for Mandarin-Taiwanese Speech-to-Speech Translation. Int. J. Computational Linguistics and Chinese Language Processing 9 (2) (2004) 13–28
8. K. Sukun, N. Sergiu and P. Rabin K., "Hardware Speech Recognition in Low Cost, Low Power Devices", computer science division (university of California, Berkeley) cs252 class project, Spring 2003.

Prototyping Object-Based Ubiquitous Multimedia Contents Storage for Mobile Devices

Young Jin Nam

School of Computer and Information Technology
Daegu University, Kyungbuk, Republic of Korea
yjnam@daegu.ac.kr

Abstract. This paper proposes an object-based ubiquitous multimedia contents storage architecture for mobile devices that employs iSCSI protocol for ubiquitous storage access over IP network and the object-based storage device model for low power. It also provides a multimedia content player that operates directly with the proposed storage architecture. We prototype the proposed storage architecture and the multimedia content player upon Linux-based desktop environments. Performance evaluation by playing MP3 multimedia contents reveals that the proposed storage architecture reduces the total power consumption by 9%, compared with existing network storage. This enhancement is mainly contributed to the fact that a large portion of the file system is moved from the mobile device into the object-based multimedia contents storage.

1 Introduction

Technology advance in processors, codec, I/O peripherals, enables high-quality multimedia contents to play on mobile devices, such as PDAs, portable multimedia players, etc. The mobile devices are mostly equipped with a hard disk or flash memory to contain the multimedia contents. Of these, the hard disk are currently more commonplace than the flash memory, because of its cost-effectiveness [1]. However, it is expected that the motor-based hard disks will eventually be replaced by the flash memory-based solid state disks. The hard disk consume 5–10 times more power than the flash memory, as it spins a spindle motor at a high speed and moves read/write heads mechanically. For example, 1-inch microdrive consumes 190–310mA(3.3V) for read and write, whereas 2Gb-NAND flash memory consumes only 15–30mA(3.3V).

Meanwhile, the increase in Internet access speed from the mobile devices enables other alternatives to store multimedia contents: the network-attached storage(NAS) [2] and the IP-based storage [2,3,4]. Even though they can obviate any limits in storage capacity, the mobile storage devices require to have additional software components, such as NFS, CIFS, and iSCSI [2,3] protocol that conveys the SCSI protocol over IP network. The NAS servers are similar to dedicated NFS or CIFS servers. They provide storage applications with file-level storage services, such as file open/close, file read/write, etc. However, the NAS server suffers from scalability, because it is involved in every file service between the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 93–102, 2006.

storage applications and the underlying storage device; that is, the overhead of the NAS server increases proportionally as the storage capacity grows. By contrast, the IP storage provides storage applications with block-level storage services over the IP-based storage area network(SAN) [5]. Typically, it is known that the IP storage guarantees better scalability in terms of storage performance and capacity. However, it does not provide enough interfaces to upper-level storage applications. In result, the storage applications should take care of a part of file-level services, such as file locking, or they should have special file systems called SAN file systems, such as global file system [6]. Figure 1(a) shows various protocol stacks for the IP storage. As mentioned, the iSCSI protocol delivers the SCSI commands over the IP network. Similar to the SCSI protocol, the iSCSI protocol consists of the iSCSI initiator and the iSCSI target. The initiator operates on the host system and issues SCSI(I/O) commands to the storage side. The target receives the commands and processes them through internal disk device drivers.

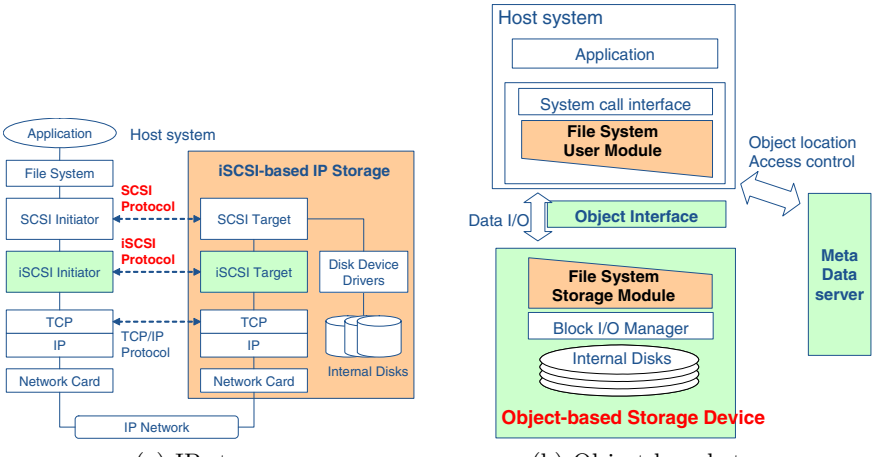


Fig. 1. Existing storage architectures: (a) IP storage architecture and (b) object-based storage architecture

Recently, the object-based storage(OSD) that exploits both advantages of the NAS server and the IP storage has been proposed as an emerging storage technology [7,8]. Figure 1(b) shows the object-based storage(OSD) architecture. The OSD treats user data or files as objects. For example, a multimedia content like a MP3 file, a database table, or a data block can be mapped onto a single object in the OSD architecture. Different from the previous block-level storage architecture, parts of a file system, such as free space management including block allocation and de-allocation are moved into the storage side. In addition, the storage not only contains the actual data blocks associated to an object, but also includes its associate object attributes for more intelligent storage management, such as data migrations, QoS management, etc. The object-based storage

includes a metadata server that is basically responsible for maintaining a location of each object contained in physical storage devices. Much research effort on OSD has been underway to design more intelligent OSD architectures [7], OSD-supporting file systems [9,10], QoS-guaranteed OSD [11], etc. Meanwhile, T10/SCSI ratified OSD commands as a new SCSI command set (version 1.0, revision 10) in Sep. 2004 [12]. They have currently been working on version 2 that includes new features such as multi-objects, snapshot, etc.

This paper proposes an object-based ubiquitous multimedia contents storage architecture for mobile devices based on the iSCSI protocol and the object-based storage device(OSD) model. The feature of the ubiquitous storage access is enabled by using iSCSI protocol that delivers SCSI commands over the typical IP network. The object-based storage model makes mobile devices reduce power consumption by moving a large portion of file system source codes into the storage itself. While the previous research works mainly emphasize various issues to apply the object-based storage model to the large-scale storage/server environments, this paper originally applies the object-based storage model to mobile embedded systems environments. The remainder of this paper is organized as follows. Section 2 describes the design and implementation of the proposed storage architecture. Section 3 provides the results of performance evaluation in terms of consumed power by running MP3 multimedia contents on the proposed storage architecture and the existing network storage(NAS). Finally, this paper concludes with Section 4.

2 The Proposed Storage Architecture

Figure 2 shows the I/O environment of the proposed storage architecture. The major software components at the mobile device encompass an extended initiator-mode iSCSI driver and a multimedia content player. The extended initiator-mode iSCSI driver (briefly, the initiator-mode iSCSI driver) sends SCSI commands to its corresponding target-mode iSCSI driver over the IP network. It additionally supports the OSD SCSI commands defined in the standard [12]. The multimedia content player is in charge of reading a multimedia content file from the object-based IP storage and then playing the file at the mobile device. It communicates the initiator-mode iSCSI driver through well-defined API. Our current design maps a single multimedia file into a single object within the physical storage. Next, the main software components at the target-side storage (also called object-based IP storage) include an extended target-mode iSCSI driver that contains an embedded file system, and disk device drivers to handle the internal disks. The extended target-mode iSCSI driver (briefly, the target-mode iSCSI driver) basically works upon the TCP/IP stack and processes OSD SCSI commands. It also contains an embedded file system to effectively serve a set of OSD SCSI commands associated to file-level services, such as object-creation and object-open. In what follows, we will describe the design of the key software components in detail.

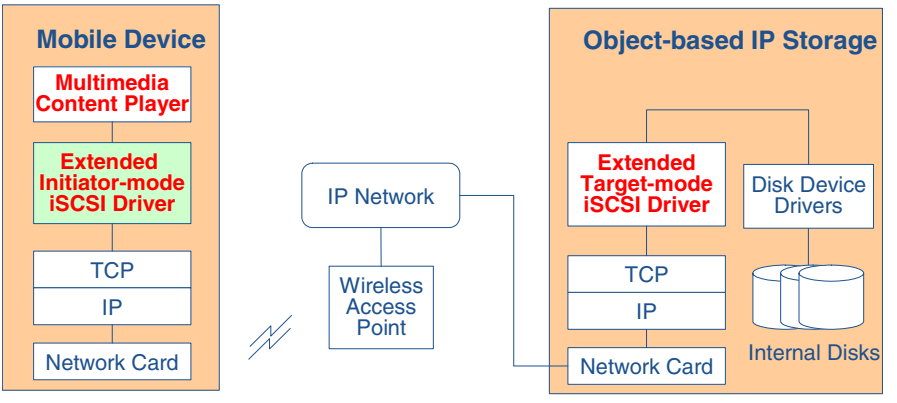


Fig. 2. The proposed storage architecture

2.1 Extended Initiator-Mode iSCSI Driver

The initiator-mode iSCSI driver exists as a form of a kernel device driver within the mobile device. This driver can process a set of OSD SCSI commands, as well as the existing SCSI command set. Table 1 shows a format of a OSD command descriptor block (CDB) that is used in our implementation. Note that this format is slightly different than that of the standard version 1.0 (revision 10). The OPERATION CODE(7Fh) field distinguishes the OSD CDB's from the existing SCSI CDB's, and the SERVICE ACTION field represents a specific OSD SCSI command. The GROUP ID, the USER ID, and the SESSION ID respectively identify an associate group object, user object, and session to fulfill the given SERVICE ACTION. The LENGTH and the OFFSET represent the length and the offset of data within an object to read and write. The remaining fields are related to the return values for the associate command.

Table 1. OSD SCSI command descriptor block(CDB) [12] (N represents the CDB size in bytes)

Byte(s)	Description	Byte(s)	Description
0	OPERATION CODE(7Fh)	1	CONTROL
2–5	RESERVED	6	SECURITY
7	ADDITIONAL CDB LENGTH(N-7)	10	OPTION BYTE 1
11	OPTION BYTE 2	12–15	GROUP ID
16–23	USER ID	24–27	SESSION ID
28–35	LENGTH	36–43	OFFSET
44–47	GET ATTRIBUTES PAGE	48–51	GET LIST LENGTH
52–55	GET ALLOCATION LENGTH	56–71	UNUSED
72–75	SET LIST LENGTH	76–N	UNUSED

The initiator-mode iSCSI driver provides the multimedia content player with a well-defined API to request the following service actions: CREATE GROUP, CREATE, WRITE, READ, REMOVE, and REMOVE GROUP. The structures of the associated service actions can be found in the OSD standard [12]. In addition, the specific API includes `osd_create_group()`, `osd_create()`, `osd_open()`, `osd_read()`, `osd_write()`, `osd_remove()`, and `osd_remove_group()`.

2.2 Extended Target-Mode iSCSI Driver

The target-mode iSCSI driver operates in the object-based IP storage. Similar to the initiator-mode iSCSI driver, it can handle a set of OSD SCSI commands, as well as the existing SCSI command set. This driver first receives an OSD command sent from its associated initiator-mode iSCSI driver. Next, it decodes the received OSD SCSI command and processes it according to its definition in the standard [12]. On receiving the CREATE GROUP command, the target-mode iSCSI driver generates a unique GROUP ID, creates a group object of the GROUP ID, and returns the GROUP ID to the initiator-mode iSCSI driver. For the CREATE command, it generates a unique USER ID within a given GROUP ID, creates a user object of the USER ID, and finally returns the USER ID. For the WRITE command, it searches the user object associated to the given GROUP ID and the USER ID, and writes data received from the initiator-mode iSCSI driver into the position pointed by the given offset. For the READ command, it finds the user object associated to the given GROUP ID and the USER ID, reads into a buffer data at the position pointed by the given offset, and transfers the buffered data to the initiator-mode iSCSI driver. For the REMOVE command, it finds the user object associated to the given GROUP ID and the USER ID, and removes it. Finally, for the REMOVE GROUP command, it first searches the group object associated to the given GROUP ID, and removes it. In addition, managing objects requires an embedded file system within the object-based IP storage. Our implementation employs a simple embedded file system that works at the user level, where each object is represented as a regular file in the Ext3 file system. Note that the design of an efficient file system for object-based IP storage is not of our concern in this paper.

2.3 OSD-Enabled Multimedia Contents Player

The multimedia content player is mainly responsible for reading a multimedia content file directly from the object-based IP storage and playing it at the mobile device. The current version of the multimedia content player can support only the MP3 audio format, while we have been extending its feature to support various video formats as well.

Storing/Removing Multimedia Contents: When storing a multimedia content to the object-based IP storage, it creates a 24-byte file called a list file that consists of a 4-byte magic number, 4-byte GROUP ID, 8-byte USER ID, and 8-byte file size information. The magic number informs the multimedia content

player that the MP3 file is stored at the object-based IP storage, not at the local file system. Note that the list file corresponds to the metadata server in the object-based storage architecture in Figure 1(b). It implies that our design collocates the metadata server with the initiator-mode iSCSI driver(host system). The multimedia content player knows an object location by reading the GROUP ID and USER ID information from the list file. Our design assumes that there exists a single object-based storage. Multiple object-based storage devices require extra information in the list file to identify specific storage location. Figure 3 shows how to store a multimedia content to the object-based IP storage in detail. In order to store a multimedia content to the object-based IP storage, the multimedia content player first invokes the `osd_create_group()` function to issue the CREATE_GROUP command that newly receives a group ID from the object-based IP storage. It next stores the obtained group ID into the `gid.dat` file. The current design assigns a different group ID to each multimedia content. However, the group ID can be used to categorize various multimedia contents stored in the object-based IP storage according to a genre, authors, etc. Next, the multimedia content player calls the `osd_create()` function to issue the CREATE command that receives a user ID and allocates storage space to store the multimedia content. The obtained user ID and group ID are stored in an associate list file, named as `osd_<filename>.mp3`. Note that the `gid.dat` and list files are stored in the local file system of the mobile device, not the object-based storage. Finally, the player invokes the `osd_write()` function to issue the WRITE command that actually writes the multimedia content into the allocated storage space. Recall that a single multimedia content is mapped into a single object in our design.

Figure 4 depicts how to remove a multimedia content from the object-based IP storage. Removing the multimedia content issues REMOVE and REMOVE_GROUP commands by invoking the `osd_remove()` and `osd_remove_group()` functions. As mentioned, the location of the multimedia content can be found by opening its associate list file.

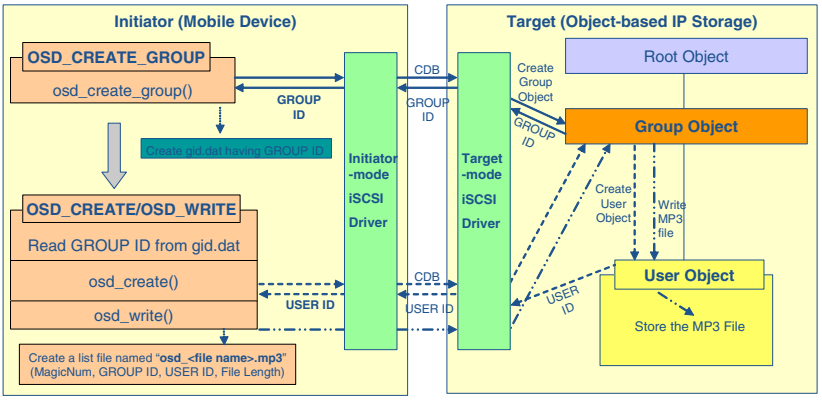


Fig. 3. Storing a multimedia content to the object-based IP storage

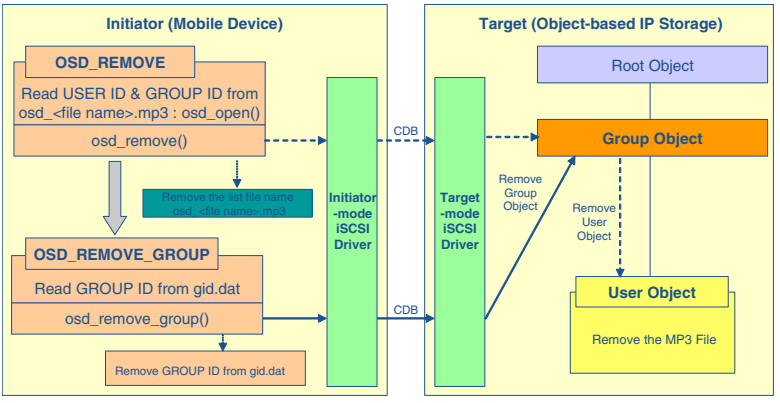


Fig. 4. Removing a multimedia content from the object-based IP storage

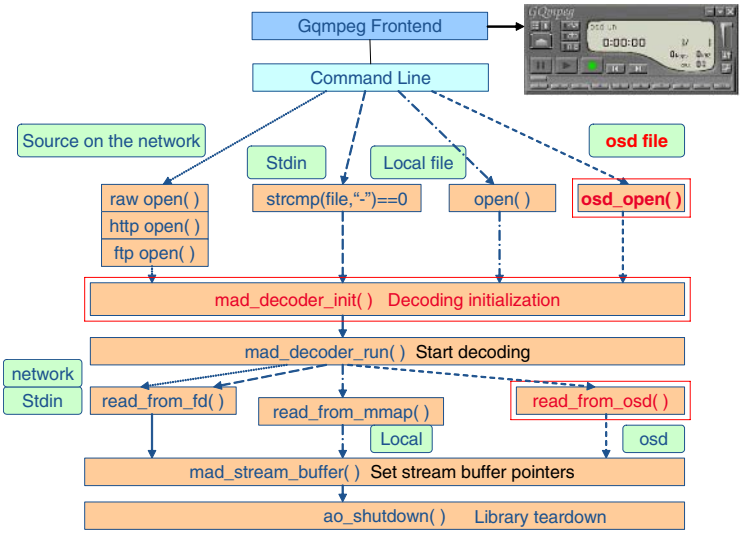


Fig. 5. The software architecture of the OSD-enabled multimedia content player

Reading/Playing Multimedia Contents: The OSD-enabled multimedia content player is implemented based on a well-known command-line MP3 player in Linux, mpg321 [13]. The mpg321 can play mp3-format audio files both in the local file system via the mmap function and on the network via ftp or http protocols. Figure 5 presents the revised mpg321 software architecture to work with the underlying object-based IP storage through the extended initiator-mode iSCSI driver. The `osd_open()` function is used to open a requested multimedia content file and read the group and user IDs. The `read_from_osd()` function calls the `read_osd()` function repeatedly to fill up its stream buffer with the

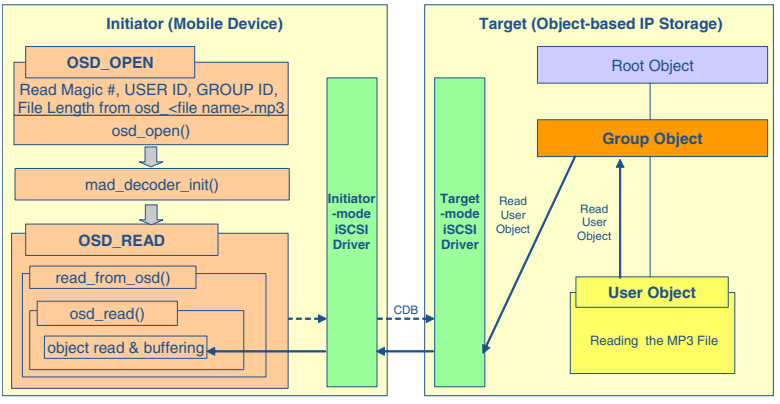


Fig. 6. Reading/playing a multimedia content from the object-based IP storage

multimedia content data. We added GQmpeg [14] in front of the mpg321 to provide a graphic user interface.

Figure 6 explains how to read a multimedia content from the object-based IP storage and play it. Once the multimedia content player opens the associate object, it initializes an audio decoder(mad library) and subsequently calls the `mad_decoder_run()` function that invokes the `read_from_osd()` function and decodes/plays the read data.

3 Performance Evaluation

We compare the performance of the proposed storage architecture with existing network storage (NAS server) that exploits the NFS network file system in terms of consumed power, while running a MP3 audio file of 4.93MB size, 192Kbps bit-rate, 2 channels, 48KHz sampling-rate, and CBR. The consumed power is measured in an indirect manner by observing the CPU utilization. For the fast prototyping, we rapidly prototyped the proposed architecture in Linux-based desktop environments (Pentium 4 1.5GHz, 384MB main memory, Linux kernel 2.4.20-8), where the initiator and the target are networked via 10Mbps Ethernet. For performance evaluation, the MP3 file is initially copied into the NFS file server and the object-based IP storage. Next, we run the MP3 file at each system, while measuring the CPU utilization, the memory utilization, and the kernel time. We repeated this experiment for each system more than ten times and averaged out the obtained results. The kernel time represents the time spent in running in the kernel mode to play the MP3 file. Table 2 shows that the proposed storage saves the consumed power by 9 percent, compared with the existing network storage(NAS server). This gain is mainly attributed to the fact that a large portion of the file system source code is moved into the object-based IP storage itself. Observe that the kernel time of the proposed storage architecture is reduced by 36 percent compared with the existing network storage.

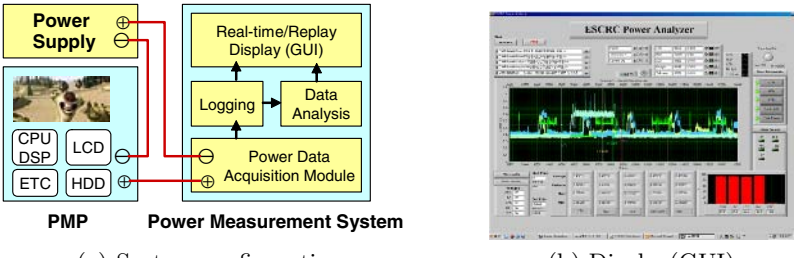
Table 2. Result summary of the consumed power by the proposed storage(object-based multimedia content storage) and the existing network storage(NAS server)

	Existing network storage	Proposed storage
CPU utilization	4.7%	4.3%
Memory utilization	2.0%	2.1%
Kernel time	0.210sec	0.135sec

4 Concluding Remarks

This paper designed object-based ubiquitous multimedia contents storage and its associated multimedia content player for mobile devices. By originally applying the object-based IP storage model to the mobile embedded systems, we intended to provide the features of ubiquitous storage access and low power. We verified the effectiveness of the proposed storage architecture by rapidly prototyping it under Linux-based desktop environments. The obtained results convinced us that the proposed storage architecture is very relevant to the mobile devices equipped with limited battery.

Currently, we have been porting our current implementation into wireless mobile environments. Hardware and software specifications of our mobile device include a PXA255 CPU, 2.5-inch TFT/LCD, touch-screen, 54Mbps-WLAN, 20GB-HDD, Linux 2.6.14, and a mplayer-based multimedia content player. The mobile device provides a set of external ports to measure consumed power at hardware components including CPU, LCD, WLAN, IDE-HDD, battery, etc. To exploit this, we built up a power measurement system that employs NI PXI-based DAQ cards and LabView-based consumed power-gathering/analyzing software, as shown in Figure 7. While the mobile device consumes 0.43watts on average when idle, it consumes 0.62watts when playing a video multimedia content from its local HDD (44% increase). We also observed that simply adding a WLAN-card to the mobile device increased the consumed power of the idle system by



(a) System configuration (b) Display(GUI)

Fig. 7. Power measurement system: (a) system configuration and (b) real-time & replay display (GUI) interface

37 percents (0.59watts). We expect that the power consumption with WLAN becomes much higher when the WLAN is actively used to access objects in the object-based IP storage. Thus, it is crucial to devise an efficient low power control scheme for the WLAN device, when applying the proposed storage architecture into wireless mobile environments.

Acknowledgments

This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment) (IITA-2005-C1090-0501-0018).

References

1. Handy, J.: Flash memory vs. hard disk drives - which will win? http://www.storagesearch.com/semico-art1.html (2005)
2. Farley, M.: Building Storage Networks. McGraw Hill (2000)
3. Hufferd, J.: iSCSI: The Universal Storage Connection. Addison-Wesley (2003)
4. Meth, K., Satran, J.: Design of the iscsi protocol. In: Proceedings of the Mass Storage Systems and Technologies/20th IEEE/11th NASA Goddard Conference. (2003)
5. Orenstein, G.: IP Storage Networking: Straight to the Core. Addison-Wesley (2003)
6. RedHat: Red hat global file system. White Paper (2004)
7. Factor, M., Meth, K., Naor, D., Rodeh, O., Satran, J.: Object storage: The future building block for storage systems. In: Proceedings of International IEEE Symposium on Mass Storage Systems and Technologies. (2005)
8. Research), E.R.: Object-based storage device(osd) basics. http://www.snia.org/education/tutorials/spr2005/storage (2005)
9. Lustre: Lustre file system. http://www.lustre.org (2005)
10. Wang, F., Brandt, S., Miller, E., Long, D.: Obfs: A file system for object-based storage devices. In: Proceedings of the 21st IEEE - 12th NASA Goddard(MSST2004) Conference on Mass Storage Systems and Technologies. (2004)
11. Lu, Y., Du, D., Ruward, T.: Qos provisioning framework for osd-based storage system. In: Proceedings of the 21st IEEE - 13th NASA Goddard(MSST2005) Conference on Mass Storage Systems and Technologies. (2005)
12. T10, T.C.: Osd standard version 1.0 (rev.10). http://www.t10.org/ftp/t10/drafts/osd (2004)
13. Drew, J., Depooter, S.: Project mpg321. http://sourceforge.net/projects/mpg321 (2001)
14. Ellis, J.: Project gqmpeg. http://sourceforge.net/projects/gqmpeg (2000)

CATA: A Garbage Collection Scheme for Flash Memory File Systems*

Longzhe Han[1], Yeonseung Ryu[1], and Keunsoo Yim[2]

[1] Department of Computer Software, Myongji University
Nam-dong, Cheoin-gu, Yongin, Gyeonggi-do, Korea
ysryu@mju.ac.kr, longzhehan@gmail.com
[2] Computing Lab., Samsung Advanced Institute of Technology
Kiheung-gu, Yongin, Gyeonggi-do, Korea
keunsoo.yim@samsung.com

Abstract. The problem of flash memory is that it cannot be overwritten unless erased in advance. In order to avoid having to erase during every update, non-in-place-update schemes have been widely used. In case of non-in-place update mechanism, garbage collection is needed to reclaim the obsolete space. In this paper, we study a new garbage collection scheme to reduce its cost such as the number of erase operations and the number of data copies. The proposed scheme determines the victim blocks by exploiting usage information of data blocks such as age, utilization and erase count. In addition, the proposed scheme predicts the future I/O workload and controls the number of victims to avoid disturbing the normal I/O operations. Experimental results show that the proposed scheme can perform well especially when the degree of locality is high.

1 Introduction

Flash memory is becoming important as nonvolatile storages because of its superiority in fast access speeds, low power consumption, shock resistance, high reliability, small size, and light weight [7,11,8,6,13]. Because of these attractive features, and the decreasing of price and the increasing of capacity, flash memory will be widely used in consumer electronics, embedded systems, and mobile computers.

Though flash memory has many advantages, its special hardware characteristics impose design challenges on storage systems. First, flash memory is organized in terms of *blocks*, where each block is of a fixed number of *pages* [8]. A block is the smallest unit of erase operation, while reads and writes are handled by pages. The size of page is fixed from 512B to 2KB and the size of block is somewhere between 4KB and 128KB depending on the product. Second, flash memory cannot be written over existing data unless erased in advance. Besides

* This work was supported by the Korea Research Foundation Grant funded by the Korean Government(MOEHRD)(R08-2004-000-10391-0).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 103–112, 2006.

Table 1. Characteristics of different storage media. (NOR Flash:Intel 28F128J3A-150, NAND Flash: Samsung K9F5608U0M)

Media	Access time		
	Read (512B)	Write (512B)	Erase
DRAM	2.56μs	2.56μs	
NOR Flash	14.4μs	3.53ms	1.2s (128KB)
NAND Flash	135.9μs	226μs	2-3ms (16KB)
Disk	12.4ms	12.4ms	

erase operation can be performed in a larger unit than the write operation and it takes an order of magnitude longer than a write operation (See Table 1). Third, the number of times an erasure unit can be erased is limited (e.g., 10,000 to 1,000,000 times). Therefore, data must be written evenly to all blocks to avoid wearing out specific blocks to affect the usefulness of the entire flash memory device, that is usually named as *wear leveling*.

Since blocks should be erased in advance before updating, updates in place is not efficient. All data in the block to be updated must first be copied to the system buffer and then updated. After the block is erased, all data must be written back from the system buffer to the block. This results in poor update performance. Moreover, the blocks of hot spots would soon be worn out. To solve these problems, data are updated to empty spaces and obsolete data are left at the same place as garbage, which a garbage collector later reclaims.

Recently, some garbage collection algorithms for log-structured flash memory file systems are proposed. These garbage collection algorithms should deal with issues such as how many blocks to erase, which blocks to erase, and where to migrate valid data from erased blocks. The primary concern of garbage collection algorithms has been to reduce the cleaning cost such as the number of erase operations and the number of data copies. In addition, because the erase operations due to garbage collection could disturb normal I/O operations, the number of victim blocks erased during a garbage collection should be carefully determined.

In this paper, we study a novel garbage collection algorithm, called CATA (Cost-Age-Time with Age-sort) to reduce its cost such as the number of erase operations and the number of data copies. The proposed scheme determines the victim blocks by exploiting usage information of data blocks such as utilization, age, and erase count. In addition, the proposed scheme predicts I/O workload of the near future and controls the number of victims to avoid disturbing the normal I/O operations.

If we predict the I/O workload such as the number of I/O request arrivals during the next garbage collection execution, we can determine the number of victim blocks so that the interference of garbage collection is minimized. For example, when the number of I/O request arrivals during the next garbage collection is estimated as high, garbage collector may select at most one victim block. Otherwise, garbage collector may select several victim blocks. In the latter case,

garbage collector can gather more valid data from victim blocks and efficiently perform the data migration by grouping the data according to their characteristics (i.e., cold data and non-cold data). Experimental results show that the proposed scheme can perform well especially when the degree of locality is high.

The rest of this paper is organized as follows. In Section 2, we review previous works that are relevant for this paper. In Section 3, we present the architecture of proposed garbage collection and deal with the problem of predicting the I/O workload. We also propose a new garbage collection algorithm. Section 4 presents the experimental results to show the performance of proposed scheme. The conclusions of this paper are given in Section 5.

2 Background

2.1 The Cost of Garbage Collection

Flash memory cannot be written over existing data unless erased in advance. Besides erase operation can be performed in a larger unit than the write operation and it takes an order of magnitude longer than a write operation. The erase operation can only be performed on a full block and is slow that usually decreases system performance and consumes power. Therefore if every update is performed in place, then performance is poor since updating even one byte requires one erase and several write operations. In order to avoid having to erase during every update, a *logging approach* has been recommended since it is quite effective in several ways (see Fig. 1). First, logging solves the inability to update *in situ* since an update results in a new write at the end of the log and invalidation of the old. The natural separation of asynchronous erases from writes allows write operations to fully utilize the fixed I/O bandwidth, and thus prevents performance degradation that may occur when writes and erases are performed simultaneously.

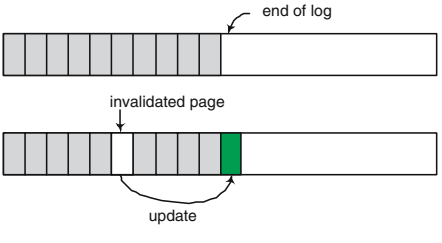


Fig. 1. Log-structured management

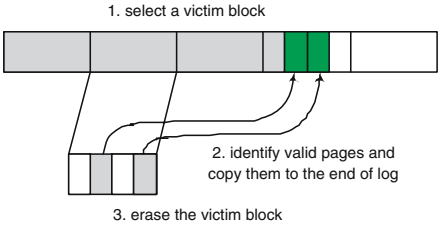


Fig. 2. Three steps of garbage collection

When data are updated to empty spaces at the end of the log, obsolete data are left at the same place as *garbage*, which a *garbage collector* process later reclaims. Since garbage collection can be performed in the background, update operation can be performed efficiently. The operation of garbage collection usually involves three stages as Fig. 2. It first selects victim blocks and then identifies valid data

that are not obsolete in the victim blocks. And it copies valid data from the victim blocks to the end of log. After that, the victim blocks are erased and available for rewriting.

The cleaning cost and the degree of wear-leveling are two primary concerns of garbage collector. The garbage collector tries to minimize cleaning cost and wear down all blocks as evenly as possible. Sometimes the objective of minimizing cleaning cost conflicts with that of wear-leveling. For example, excessive wear-leveling generates a large number of invalidated blocks, which degrades cleaning performance.

First, we define the cleaning cost. Let MC_i be migration cost and EC_i be erasure cost for i-th garbage collection, respectively. The migration cost denotes the number of copies of valid data from the victim blocks to free space in other blocks. The erasure cost denotes the number of erasure. Then the cleaning cost of garbage collection can be described as follows:

$$\sum MC_i + EC_i \tag{1}$$

The cost to erase a block is much higher than to write a whole block. The erasure cost dominates the migration cost in terms of operation time and power consumption. Therefore, the number of erase operations determines the garbage collection costs. For better performance and power conservation, the primary goal is to minimize the number of erase operations.

Next, we define the degree of wear-leveling as follows:

$$\epsilon = E_{max} - E_{min} \tag{2}$$

where E_{max} is the maximum erase count and E_{min} is the minimum erase count, respectively. The smaller ϵ is, the longer the lifetime of system is. Since excessive wear-leveling does cleaning performance more bad than good, it is sufficient that ϵ should be below a predefined threshold.

2.2 Related Works

There are some issues of garbage collection algorithms:

When When is garbage collection started and stopped? It usually executes periodically or is triggered when the number of free blocks gets below some threshold.

How many How many blocks are cleaned at once? The more blocks are cleaned at once, the more valid data can be reorganized. However, cleaning several blocks needs much time, which can disturb normal I/O execution. Thus, most garbage collection algorithms select only one block.

Which Which block is selected for erasing? One may select a block with the largest amount of garbage or select blocks using information such as age, update time, etc. This is referred to as *victim selection algorithm*.

Where Where is the valid data written out? This is referred to as *data migration algorithm*. There are various ways to reorganize valid data, such as enhancing the cleaning performance by grouping pages of similar age together or grouping related files together into the same block, etc.

A number of victim selection algorithms based on the *block utilization* have been studied [9,15,10,4,12]. The *greedy* algorithm selects blocks with the largest amount of garbage for erasure, hoping to reclaim as much as possible with the least cleaning work [12]. The greedy policy tends to select a block in a FIFO order irrespective of data access patterns. It is known that the greedy policy works well for uniform access, but does not perform well for high localities of access [15]. The *cost-benefit* algorithm chooses blocks that maximize the formula [12,9]: $\frac{age \cdot (1-u)}{u+1}$, where u is the utilization of a block (the fraction of space occupied by valid data) and *age* is the time since the most recent modification, respectively. The cost is derived from migration and erasure, which are reflected by the denominator $(u+1)$. The benefit is given as the space-time product form. The term $(1-u)$ reflects how much free space it acquires. Because of term *age*, cold blocks can be cleaned at a much higher utilization than hot blocks.

There are several methods to migrate valid data to the cleaned blocks. Most schemes try to gather hot data together to form the largest amount of garbage to reduce garbage collection cost. The *age-sort* algorithm used in Log-Structured File system(LSF) sorts valid data by age before writing them out to enforce the gathering of hot data [12]. For better effect, several blocks are cleaned at once. The *separate block cleaning* algorithm uses separate blocks in cleaning: one for cleaning not-cold blocks and writing new data, the other for cleaning cold segments [9]. The separate segment cleaning was show to perform better than when only one segment is used in cleaning, since hot data are less likely to mix with cold data. The *dynamic data clustering* algorithm clusters data according to their update frequencies by active data migration [5].

As for wear-leveling, simple swapping approaches have also been proposed. However, swapping data between two blocks requires buffer memory, erasing two blocks and rewriting swapped data. Thus the swapping methods consume a lot of available system resources and time, which could disturb normal I/O execution.

3 New Garbage Collection Scheme

3.1 Motivation

The performance of garbage collection depends on the combination of victim selection policy and data migration policy. The cost-benefit policy, a representative victim selection algorithm, generally performs better than the greedy policy. But it does not perform well for high localities of access without combining efficient data migration policy. We will show it in the next section. Under high localities of access, after a large number of logging and cleaning operations, cold data becomes mixed with non-cold data within each block. After that time, cold data moves around uselessly together with non-cold data. For this reason, the utilization of cleaned blocks remains stable at a high value and the amount of free space collected becomes small. In other words, migration cost and erasure cost could be increased. In order to overcome this problem, the cost-benefit

policy has to combine with data migration policy that separates cold data and hot data when migrating valid data.

Many flash memory file systems are based on Log-Structured File system [14,1]. But, the separate block cleaning policy and the dynamic data clustering policy cannot be used for log-structured file systems. The age-sort policy was used in Log-Structured File system. It sorts the valid pages of victim blocks by the time they were last modified and migrates them at the end of log. For example, it migrates the oldest pages first at the end of log.

The problem of this policy is that garbage collector should select several victim blocks for better separation of cold and non-cold data. When it collects valid data from many victim blocks, there may be high probability of isolating cold data and non-cold data and of migrating them to separated blocks. But, previous garbage collection algorithms choose one victim block since erase operation takes much time and thus erasure of several victims at once can disturb normal I/O operation.

If we can predict the I/O workload such as the number of I/O request arrivals during the next garbage collection execution, we can control the number of victim blocks to be erased according to the estimated I/O workload. When it is predicted that the number I/O request arrivals for the next garbage collection is high, garbage collector selects one or no victim block. Otherwise, garbage collector can select several victim blocks and thus improve its performance.

3.2 Architecture

The proposed garbage collection module consists of three components: (i) a *monitor* that measures the request arrival rate, (ii) a *predictor* that uses the measurements from the monitor module to estimate the workload characteristics in the near future, and (iii) a *garbage collector* that performs garbage collection task.

The monitor is responsible for measuring the request arrival rate. The monitor tracks the number of request arrivals a_i in each measurement interval (I) and records this value. The monitor maintains a finite history consisting of the most

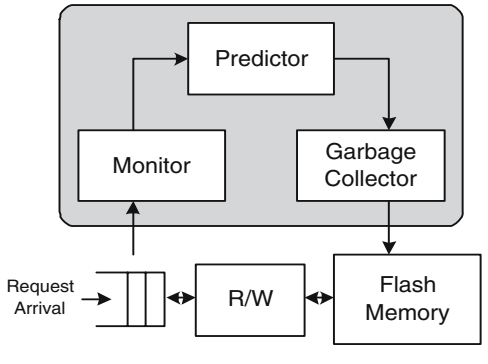


Fig. 3. Proposed garbage collector architecture

recent H values of the number of arrivals. Let A_i be the sequence $a_i^1, ..., a_i^H$ of values from the measurement history. Let W be the time units to execute garbage collector.

The predictor uses the past measurements to predict the number of arrivals $\hat{n}_i$ and the arrival rate λ_i for the W time units. In order to predict the number of arrivals $\hat{n}_i$, we use the model of AR(1) process [3,2](autogressive of order 1). Using the AR(1) model, a sample value of A_i is estimated as

$$\hat{a}_i^{j+1} = \bar{a}_i + \rho_i(1) \cdot (a_i^j - \bar{a}_i) + e_i^j, \tag{3}$$

where, ρ_i and $\bar{a}_i$ are the autocorrelation function and mean of A_i respectively, and e_i^j is a white noise component. We assume e_i^j to be 0, and a_i^j to be estimated values $\hat{a}_i^j$ for $j \geq H + 1$. The autocorrelation function ρ_i is defined as

$$\rho_i(l) = \frac{E[(a_i^j - \bar{(a)}_i) \cdot (a_i^{j+l} - \bar{a}_i)]}{\sigma_{a_i}^2}, 0 \leq l \leq H - 1, \tag{4}$$

where, σ_{a_i} is the standard deviation of A_i and l is the lag between sample values for which the autocorrelation is computed.

Let $M = W/I$. Then we estimate $\hat{a}_i^{H+1}, ..., \hat{a}_i^{H+M}$ using Equ. 3. Then, the estimated number of arrivals in W time units is given be $\hat{n}_i = \sum_{j=H+1}^{H+M} \hat{a}_i^j$ and finally, the estimated arrival rate, $\hat{\lambda}_i = \frac{\hat{n}_i}{W}$.

3.3 CATA : Cost-Age-Time with Age-sort

We propose a new algorithm called *Cost-Age-Time with Age-sort*(CATA). The CATA algorithm chooses blocks that maximize the formula:

$$\frac{1-u}{1+u} \cdot age \cdot \frac{1}{erase\_count} \tag{5}$$

where u is the utilization of a block (the fraction of space occupied by valid data), *age* is the time since the most recent modification, and *erase_count* is the number of times a block has been erased, respectively. Here, age is normalized by a transformation function, aiming to avoid age being too large to overemphasize age. The age transformation function is similar to that of [4].

The CATA uses the predicted workload to determine the number of victim blocks. If CATA predicts that the number I/O request arrivals for the next garbage collection is high, it selects one or no victim block. Otherwise, it selects two or three victim blocks according to the predicted number of I/O request arrivals. After selecting the victim blocks, the CATA sorts the valid pages of victim blocks by the time they were last modified and migrates them at the end of log. It migrates the oldest pages first at the end of log.

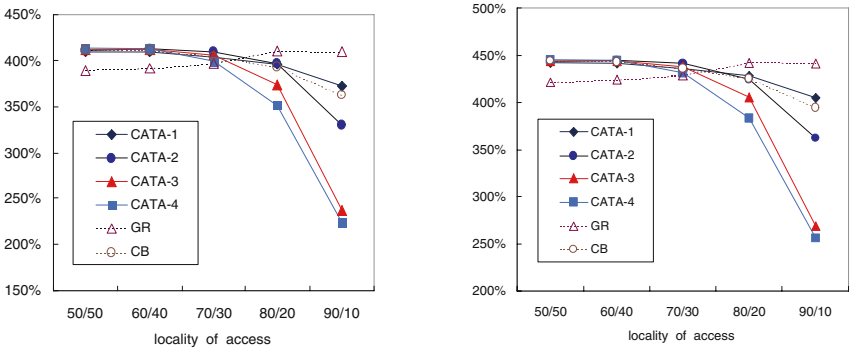
4 Experiment

We have performed simulations in order to investigate the CATA policy by varying the number of victim blocks. We have also implemented three algorithms

for comparison: GR represents the greedy policy with no separation of hot and cold data; CB represents the cost-benefit policy with no separation of hot and cold data; and CATA-x represents the CATA policy, where x is the number of victim blocks.

Since at low utilization garbage collection overhead does not significantly affect performance, in order to evaluate the effectiveness, we initialized the flash memory by writing data sequentially to fill it to 90% of flash memory spaces. The created workloads then updated the initial data according to the required access patterns. We used the notation for locality of reference as 'x/y' that x% of all accesses go to y% of the data while $(100 - x)$% go to the remaining $(100 - y)$% of data.

We define the number of extra erase operations as the number of erase operations minus the number of erase operations from an ideal scheme. The ideal scheme is defined as a scheme that performs one erase operation for every n-page write requests, where n is the number of pages per block. Similarly, the number of extra write operations is defined as the number of write operations minus the number of writes requested. Performance metrics are the ratio of the number of extra erase operations to the number of erase operations from ideal scheme, the ratio of the number of extra write operations to the number of write requests and the degree of wear-leveling.



(a) Erase performance (b) Write performance

Fig. 4. Erase and write performance

Fig. 4 (a) shows the ratio of the number of extra erase operations to the number of erase operations from ideal scheme and (b) shows the ratio of the number of extra write operations to the number of write requests. In both figures, as the locality is increased, the performance of CATA-2, CATA-3, and CATA-4 increased rapidly. This is because CATA with several victims can separate data, such that cold data are less likely to mix with hot data as compared with the other policies. This effect is more prominent under higher locality of access. Fig. 4 also shows that the performance of CATA-3 and CATA-4 are nearly the same. This implies that it is sufficient to select at most three victim blocks.

Table 2. Comparison of wear-leveling degree

	CATA-1	CATA-2	CATA-3	CATA-4	GR	CB
50/50	3	3	4	3	6	5
60/40	3	4	4	3	6	6
70/30	3	3	4	5	13	7
80/20	4	5	7	9	17	10
90/10	4	11	13	12	34	17

Table 2 shows the simulation results about the degree of wear-leveling. It illustrates that CATA provides stable wear-leveling effects. This is because CATA considers wear-leveling by exploiting the erase count of data block when it selects the victim blocks. In case of GR and CB, which do not consider the erase count of data block, the degree of wear-leveling is increasing when the access pattern becomes skewed.

5 Concluding Remarks

There have been researches on garbage collection algorithms for flash memory file systems. Garbage collection algorithms should deal with some issues such as how many blocks to erase, which blocks to erase, and where to migrate valid data from erased blocks. The primary concern of garbage collection algorithms has been to reduce the cleaning cost. But, the number of victim blocks erased during a garbage collection should be carefully determined because the erase operations due to garbage collection could disturb normal I/O operations.

In this paper, we propose a novel garbage collection architecture, which predicts I/O workload of the near future and determines the number of victim blocks according to the predicted I/O workload. We also study a garbage collection algorithm, called CATA (Cost-Age-Time with Age-sort), which efficiently performs the data migration by gathering valid data from several victim blocks as much as possible. It controls the number of victim blocks by using the estimated I/O workload during garbage collection execution so that the interference of garbage collection is minimized. Experimental results show that the proposed scheme can achieve its goal such as reducing its cost and minimizing the wear-leveling degree especially when the access pattern is highly skewed.

References

1. Yaffs (yet another flash filing system). http://www.aleph1.co.uk/yaffs/.
2. G. Box and G. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, 1976.
3. A. Chandra, W. Gong, and P. Shenoy. Dynamic resource allocation for shared data centers using online measurements, 2002.
4. M. Chiang and R. Chang. Cleaning policies in mobile computers using flash memory. *Journal of Systems and Software*, 48(3):213–231, 1999.

5. M. Chiang, P. Lee, and R. Chang. Using data clustering to improve cleaning performance for flash memory. *Software Practice and Experience*, 29(3):267–290, 1999.
6. T. Chung, D. Park, Y. Ryu, and S. Hong. Lstaff: System software for large block flash memory. *Lecture Notes in Computer Science*, 3398:704–710, 2005.
7. F. Douglis, R. Caceres, F. Kaashoek, K. Li, B. Marsh, and J. Tauber. Storage alternatives for mobile computers. In *Proceedings of the 1st Symposium on Operating Systems Design and Implementation*, 1994.
8. Samsung Electronics. 256m x 8bit / 128m x 16bit nand flash memory. http://www.samsungelectronics.com.
9. A. Kawaguchi, S. Nishioka, and H. Motoda. Flash memory based file system. In *Proceedings of USENIX95*, pages 155–164, 1995.
10. H. Kim and S. Lee. An effective flash memory manager for reliable flash memory space management. *IEICE Trans. Information and Systems*, E85-D(6):950–964, 2002.
11. B. Marsh, F. Douglis, and P. Krishnan. Flash memory file caching for mobile computers. In *Proceedings of the 27th Hawaii International Conference on Systems Sciences*, 1994.
12. M. Rosenblum and J. K. Ousterhout. The design and implementation of a log-structured file system. *ACM Trans. Computer Systems*, 10(1):26–52, 1992.
13. Y. Ryu and K. Lee. Improvement of space utilization in nand flash memory storages. *Lecture Notes in Computer Science*, 3820:766–775, 2005.
14. David Woodhouse. Jffs: The journalling flash file system. In *Proceedings of the Ottawa Linux Symposium*, 2001.
15. M. Wu and W. Zwanepoel. envy: A non-volatile, main memory storage system. In *Proceedings of the 6th Internation Conference on Architectural Support for Programming Languages and Operating Systems*, 1994.

A Robust Location Tracking Using Ubiquitous RFID Wireless Network*

Keunho Yun, Seokwon Choi, and Daijin Kim

Department of Computer Engineering, POSTECH, Korea
{rootyun, capriso, dkim}@postech.ac.kr

Abstract. A dangerous workplace like the iron production company needs a durable monitoring of workers to protect them from an critical accident. This paper concerns about a robust and accurate location tracking method using ubiquitous RFID wireless network. The sensed RSSI signals obtained from the RFID readers are very unstable in the complicated and propagation-hazard workplace like the iron production company. So, the existing particle filter can not provide a satisfactory location tracking performance. To overcome this limitation, we propose a double layered particle filter, where the lower layer classifies the block in which the tag is contained by the SVM classifier and the upper layer estimates the accurate location of tag owner by the particle filter within the classified block. This layered structure improves the location estimation and tracking performance because the evidence about the location from the lower layer makes a effective restrict on the range of possible locations of the upper layer. We implement the proposed location estimation and tracking system using the ubiquitous RFID wireless network in a noisy and complicated workplace (100m $\times$ 50m) where which 49 RFID readers and 9 gateways are located in the fixed locations and the maximally 100 workers owning active RFID tags are moving around the workplace. Many extensive experiments show that the proposed location estimation and tracking system is working well in a real-time and the position error is about 2m at maximum.

1 Introduction

Location based technology [1,2] is an emerging application field that uses physical, geometric and logical location information of people or objects and the location estimation and tracking of people or objects are the most important tasks among many constituent technologies in the location based technology. Several sensors using infrared light, ultrasound, RF (Radio Frequency), and UWB have been proposed for implementing the location based technology.

* This work is financially supported by the Ministry of Education and Human Resources Development(MOE), the Ministry of Commerce, Industry and Energy(MOCIE) and the Ministry of Labor(MOLAB) through the fostering project of the Lab of Excellency, and also partially supported by the Regional Innovation System(RIS) of the Ministry of Commerce, Industry and Energy(MOCIE).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 113–124, 2006.

In this work, we choose the RF system for estimating the location due to its many advantages such as the relative low cost, the wide operating range and easy operating compatibility with the existing network environment. Often, the RF system is using Zigbee protocol that is widely used in the wireless network, which enables an ad-hoc network, a low resource consumption, and a reliable data transfer additionally. Also, we take the particle filter among many existing location tracking methods such as Bayesian filters [3], Kalman filter [4], and particle filter [5], due to its simplicity and diversity of nonlinear/non Gaussian modeling.

In RFID system, we use the RSSI to measure the strength of RF signal. Generally, the RSSI signal is attenuating as the distance of the emission source is far from the receiver. Unfortunately, the sensed RSSI signal in the active RFID system has some bad characteristics for location estimation and tracking because of large sensitivity, various operating environment, and its nonlinearity. In order to overcome this shortcoming, we need to modify the existing location tracking method to complement a defect of RSSI signal as much as possible. To meet the goal, we propose a double layered particle filter that combines the classification into the existing particle filter as follow. In the lower layer, we use the SVM classifier[6] to determine where a tag is included in the block level. In the upper layer, we use the existing particle filter technique to estimate the tag's location, velocity and direction accurately based on the classification result obtained from the first layer. This layered structure improves the location estimation and tracking performance because the classification result of the first layer restricts effectively the possible locations for the upper layer.

2 Backgrounds

2.1 Particle Filter

Bayesian filters try to estimate the motion state x_t of a dynamic system from a noisy collection of observations $Y_{1:t} = (y_1, y_2, \cdots, y_t)$. Fig. 1 illustrates a graphical model [7] of the Bayesian filters [8], where x_t and y_t are the hidden variable and the observed variable, respectively. Bayesian filters take three important

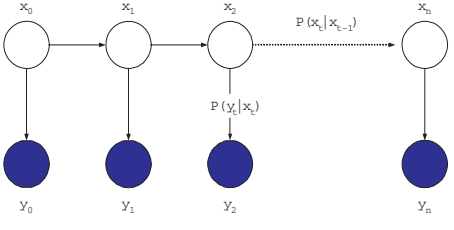


Fig. 1. A graphical model of Bayesian filters

assumptions such as (i) conditional independence, (ii) probabilistic relationship like Gaussian model, and (iii) the first order Markov process. Also, they include the motion model $p(x_t|x_{t-1})$ that is used to update the temporal change

of the motion state x_t and the sensor likelihood model $p(y_t|x_t)$ that is used to update the measurement weights. When we denote an observation sequence as $Y_{1:t} = (y_1, y_2, \cdots, y_t)$, the Bayesian filters can be described by the following two equations as

$$p(x_t|Y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|Y_{1:t-1})dx_{t-1}, \tag{1}$$

$$p(x_t|Y_{1:t}) = \frac{p(y_t|x_t)p(x_t|Y_{1:t-1})}{p(y_t|Y_{1:t-1})},$$

$$p(y_t|Y_{1:t-1}) = \int p(y_t|x_t)p(x_t|Y_{1:t-1})dx_t. \tag{2}$$

Here, Eq. (1) and (2) represent the time update that predicts the next state by the motion model $p(x_t|x_{t-1})$, and the measurement update that predicts the next observation by the sensor likelihood model $p(y_t|x_t)$, respectively. According to Bayes rule, the posterior distribution $p(x_t|Y_{1:t})$ can be computed by the product of the prior distribution $p(x_t|Y_{1:t-1})$ and the sensor likelihood model $p(y_t|x_t)$, where the evidence $p(y_t|Y_{1:t-1})$ is used as a normalization factor.

Particle filter is often called as SMC (Sequential Monte Carlo) [9] or SIR (Sequential Importance sampling with Resampling) [10]. Fig. 2 illustrates the working principle of particle filter, which can be summarized by the following. First, we create N initial samples. Second, we compute the weights of samples in the importance sampling process using the likelihood of the observed distribution. Third, we perform the re-sampling process to solve the degeneracy

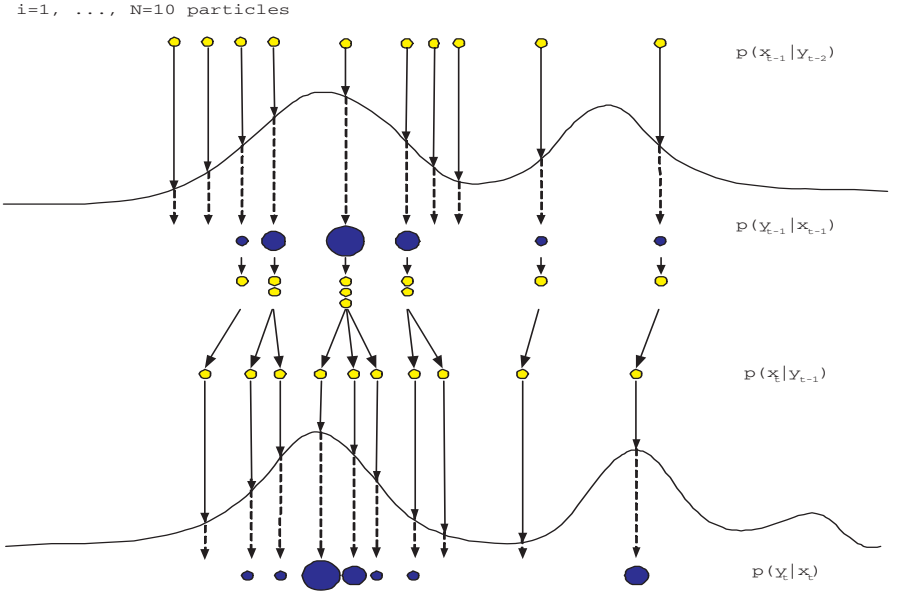


Fig. 2. An operating process of particle filter [10]

problem which samples are centralized in a specific region. Fourth, we perform the time update process. This process will be repeated until no more observation is founded.

In the practical implementation of particle filter, we take the sampling method to reduce the computational cost. According to the central limit theorem, the summation can approximate the integration when the number of samples N approaches to infinity as

$$I_N(f) = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)}) \overset{N\to\infty}{\Longrightarrow} I(f) = \int_x f(x)p(x)dx. \tag{3}$$

So, if we substitute the integration for an approximation sampling, the time update process will be replaced by

$$p(x_t|Y_{1:t-1}) \approx \sum_{m=1}^{M} w_{t-1}^{(m)} p(x_t|x_{t-1}^{(m)}),$$
$$w_{t-1}^{(m)} = \frac{p(y_{t-1}|x_{t-1}^{(m)})}{\sum_{m=1}^{M} p(y_{t-1}|x_{t-1}^{(m)})}, \tag{4}$$

where $w_{t-1}^{(m)}$ is an importance weight of the mth sample of the observed distribution in the time $t-1$.

3 Location Tracking Using a Double Layered Particle Filter

Fig. 3 shows an approximated graphical model of the proposed double layered particle filter, where z_t is the observation variable that is the RSSI signal from all RFID receivers, y_t is the hidden variable that is a coarsely estimated tag's location (u', v') in the block level using the SVM classification, and x_t is the state variable that is an accurately estimated tag's location (u, v) in a fine level. The role of the first layer is to estimate the tag's coarse location y_t given the RSSI

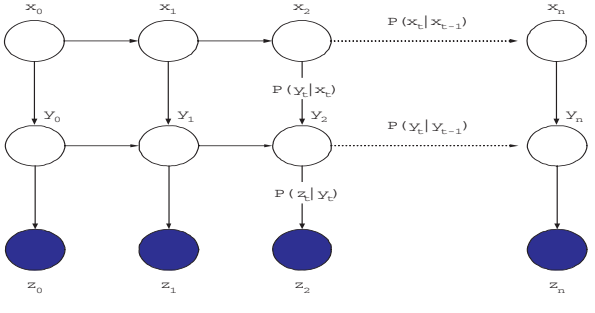


Fig. 3. A graphical model of the proposed double layered particle filter

observations z_t and the role of the second layer is to estimate the tag's precise location x_t given the coarsely estimate tag's location y_t. So, we can treat the function of the first layer as a preprocessing for the conventional particle filter given in the second layer. The preprocessing provides a more reliable evidence about the tag's location and this will improve the tracking performance of the conventional particle filter.

3.1 The First Layer: Coarse Location Estimation

The role of the first layer is to provide a reliable estimate of the tag's coarse location y_t given the RSSI observations z_t. For doing this, we consider three consecutive strategies by the following.

First, we consider the measurement update of the first layer. For the measurement update, the SVM classifier determines in which block a tag is included. The estimated tag's location would be represented in a probabilistic way because there is a uncertainty about the estimated tag's location due to the instability of the the observed RSSI signals. In our work, the estimated tag's location y'_t obeys the Gaussian distribution model $G(\mu_t, \sigma_t^2)$ and the center of block in which the tag is included is treated as the mean μ_t of the Gaussian distribution. When we perform the measurement update in the first layer, the estimated location y'_t is given by

$$y'_t \sim G(\mu_t, \sigma_t^2), \tag{5}$$

where $\mu_t =$ mean of $\text{block}_{\text{SVM}}(z_t)$.

Second, we consider the *restricted* time update of the first layer. In a practical situation, we obtain several number of the observed data in a given time interval and we assume that the tag's owner can not walk so fast. Then, the estimated locations within a given time interval would not change abruptly within a short interval. We adopt this constraint by restricting the possible estimated locations during a given time interval as follows.

In the practical application, the difference of the estimated locations between two consecutive time intervals $\delta = |y'_t - y'_{t-1}|$ should be restricted because the estimated locations between two consecutive time intervals can not be changed abruptly. To reflect this constraint, we determine the current tag's location at the time t as follows. If the difference D is smaller than a given threshold value D, we take the estimated location at the time t as the current tag's location. Otherwise, we throw away the estimated location at the time t and take the previous estimated location at the time $t-1$ as the current tag's location. When we consider this restricted time update, the estimated tag's location y''_t is given by

$$y''_t = \begin{cases} y'_t, & \delta < D, \\ y'_{t-1}, & \text{otherwise.} \end{cases} \tag{6}$$

When we consider only the restricted time update, the instability of the sensed RSSI signals can make the estimated locations stuck to the same position for

some time intervals. This situation will produce a substantial error in the estimated location because the restricted time update keeps the consecutive estimated locations to be same for some time intervals even though the real tag's position is changed very much. One way to avoid this situation is to accept the estimated location at the time t as the true tag's position when the previous k consecutive estimated locations containing the estimated location at the time t keep the same position. When we consider this persistence of the estimated locations, the tag's location y''_t is given by

$$y''_t = y'_t, \ when \ equal(y'_{t-k+1:t}) \ is \ true, \tag{7}$$

where the function $equal(y'_{t-k+1:t})$ is true only when the previous k consecutive estimated tag's blocks are identical.

Third, we take the *majority voting* scheme to make the current estimated location more reliable. In this scheme, we take the majority of the three consecutive estimated locations $\{y''_{t-2}, y''_{t-1}, y''_t\}$ as the current estimated location, i.e., when more than two estimated locations among three consecutive estimated locations are coincident each other, it is treated as the majority. When we consider the majority voting scheme, the coarsely estimated tag's location y_t is given by

$$y_t = \text{Majority of } \{y''_{t-2}, y''_{t-1}, y''_t\}. \tag{8}$$

3.2 The Second Layer: Accurate Location Estimation

The role of the second layer is to provide an accurate tag's location x_t based on the coarsely estimated tag's location y_t that is obtained from the first layer. For doing this, we perform two kinds of updates as follows.

First, we perform the measurement update of the second layer that modifies the weights of all samples. The center of the estimated block obtained from the first layer is treated as the mean of the Gaussian distributed samples. The weight of each sample is updated in accordance with the distance between the mean and the sample's location using the policy that the more close to the mean a sample is, the larger value of weight it has. Further, in the case of using the SVM classifier, the samples within the estimated block are more weighted because the SVM classifier provides more strong evidence about the existence of samples within the estimated block. When we consider the above two effects, the weight of the mth sample at the time t can be represented as

$$\begin{aligned} w_t^{(m)} &= \frac{p(y_t^{(w)} y_t^{(l)} | x_t^{(m)})}{\sum_{s=1}^{M} p(y_t^{(w)} y_t^{(l)} | x_t^{(s)})}, \\ &= \frac{p(y_t^{(w)} | x_t^{(m)} y_t^{(l)}) p(y_t^{(l)} | x_t^{(m)})}{\sum_{s=1}^{M} p(y_t^{(w)} | x_t^{(m)} y_t^{(l)}) p(y_t^{(l)} | x_t^{(s)})}, \\ &= \frac{\alpha(m) p(y_t^{(l)} | x_t^{(m)})}{\sum_{s=1}^{M} \alpha(s) p(y_t^{(l)} | x_t^{(s)})}. \end{aligned} \tag{9}$$

Second, we perform the time update of the second layer that predicts the next location by the predefined motion model $p(x_{t+1}|x_t)$ as

$$p(x_{t+1}|Y_{1:t}) \approx \sum_{m=1}^{M} w_t^{(m)} p(x_{t+1}|x_t^{(m)}). \tag{10}$$

The obtained posterior distribution $p(x_{t+1}|Y_{1:t})$ provides an accurate location estimation because the weight $w_t^{(m)}$ used in this work is taken precisely by considering the regression or classification result in the first layer.

Fig. 4 illustrates a schematic flowchart that shows how the proposed location estimation method is working, where the process is repeated until no more observation is founded.

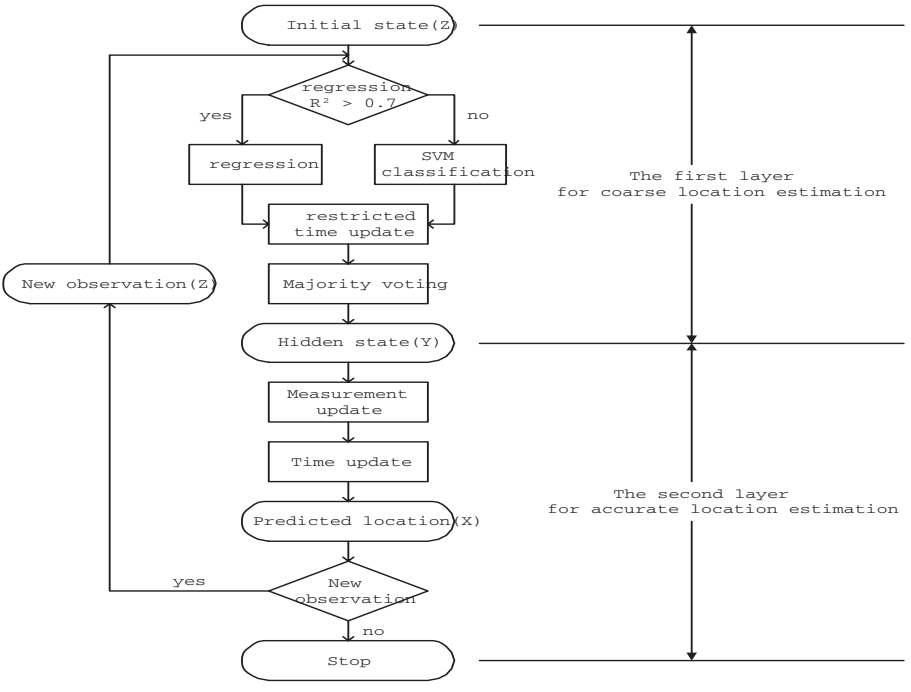


Fig. 4. A schematic flowchart of the proposed location estimation method

4 System Architecture

4.1 Transmission and Reception Modules

We use the smart tag which uses Zigbee protocol of the MAXFOR company, where it works on the Berkeley TinyOS and it periodically sends a fixed packet

twice for a second using the 2.4GHz frequency band and the RFID reader[1] receives the packet signal after receiving all tag's header data. The ULG includes a data receiver and a wireless LAN for sending to AP (Access Point). Then, the AP sends the tag data to the server through the gateway. For a reliable data transmission, the ULG includes Zigbee nodes which have a function of ad-hoc network.

4.2 Network Architecture

In this work, we take a centralized star network architecture to interconnect several ULGs and an ad-hoc mesh structure to interconnect the Zigbee nodes. This mixed type of star and mesh network structure guarantees a reliable communication because the ad-hoc function plays a role of reliable data transfer among the moving tags. Fig. 5 illustrates the star-mesh structure that is used in our work. If people with a tag moves around the office, the ULG readers that are in the fixed locations are receiving the tag data through the star-mesh network and are generating the RSSI signal depending the distance between the ULG reader and the tag. Then, the server receives the tag data and the sensed RSSI signal from the ULGs using the wireless LAN through the gateway. Then, the server starts to estimate the tag's location by the proposed double layered location estimation method.

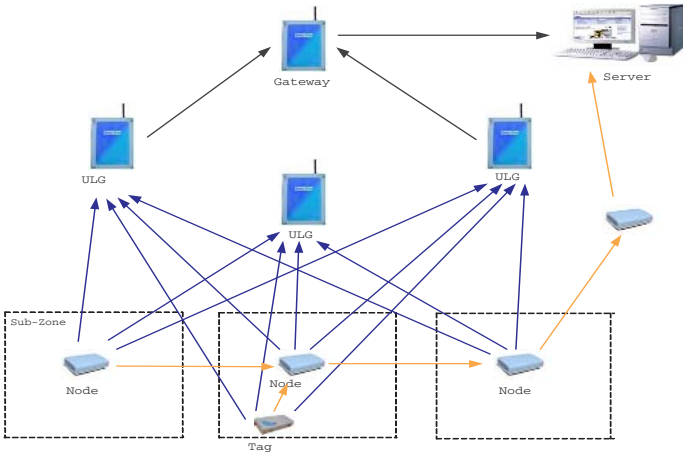


Fig. 5. Star-mesh network architecture

5 Experiment Result and Discussion

We performed the location estimation experiments in our office, where nine working desks are located regularly with the square shaped partitions in the size of

[1] In this paper, we call the RFID reader as the ULG (Ubiquitous Location Gator).

10 m × 9 m and there are many obstacles like partitions plates, bookshelves, and computers, etc. We adjust the height of RF reader as similar as people's stature because RF signal is sensitive to the direction.

5.1 Experiment for Estimating the Location of a Person in an Arbitrary Movement

In this experiment, we evaluate the estimation performance of the proposed location estimation algorithm, where a person is moving arbitrarily around the office. We choose 25 designated positions in the office, and we collect a total of 8,400 training samples, which consists of 21 positions × 4 different orientation × 100 training samples. The reason why we collect the sampling data from four different orientations (E, W, S, N) is that the sensed RSSI signals are too sensitive to the direction of RFID reader antenna, the orientation of RFID tag, and human interference and we have no idea about the orientation of the tag under test.

Since the RSSI value at each position is very sensitive to the orientation, we classifier the RSSI values at each position into four different orientations. So, we need to build 21 × 4 SVM classifiers, where each classifier uses 100 training samples and is trying to discriminate the RSSI values into one combination of the location and the orientation among 100 possible combinations of 21 locations and 4 different orientations.

Fig. 6 illustrates the results of location estimation when a people is moving arbitrary in the office. Here, the • marks are the training positions, and the triangular mark near a specific • mark denotes the position that is estimated by the proposed location estimation method, where $\triangleright$, $\triangleleft$, $\bigtriangledown$, and $\bigtriangleup$ represent that the tag is orientated toward the east, west, south, and north orientation at the specific location, respectively. Here, we only show some partial test points denoted by the • mark because it is difficult to discern the total estimated locations corresponding to all test pest points. As you can see, some points are misleadingly estimated due to the incompleteness of the proposed location estimation method. Table 1 compares the MSE between the true location, the execution time, and the SVM's location and orientation classification performance between two different location tracking methods: the conventional particle filter and the proposed double layered particle filter. From this table, we know that (i) The conventional particle filter method shows a great value of MSE due to a large error due to the large variation of the RSSI data itself, and the (ii) the proposed particle filter method shows the improvement of the MSE performance by 69%, and the location and orientation classification performances a little, at the cost of a little more execution time. This improvement is due to the preprocessing of SVM classification for the coarse location and orientation and the use of filtering technique such as the restricted time updates. This result is also very important because the use of SVM classification enables the accurate estimation of orientation and this makes us to wear the RFID tag everywhere of our body, like helmet, breast or waist.

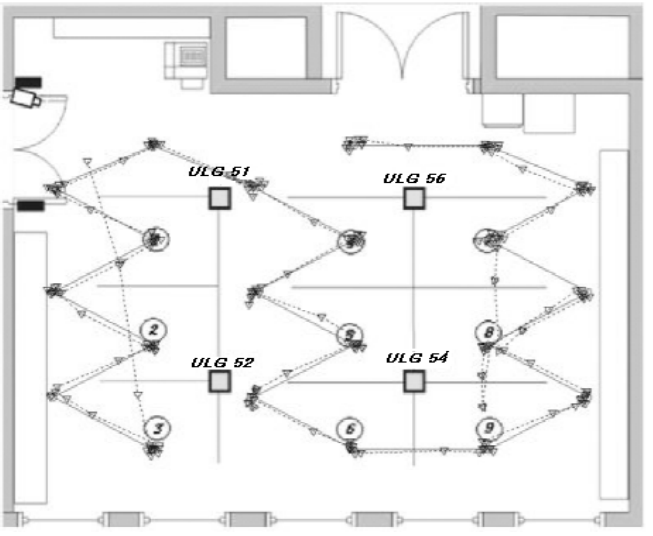


Fig. 6. Result of people tracking in an arbitrary movement

Table 1. Comparison of two location estimations in the case of people's arbitrary movement

Method	MSE (m)	Location accuracy (%)	Orientation accuracy (%)
conventional	3.462	·	·
proposed	1.064	87.14	91.19

5.2 Application to the Real Environment

We applied for the proposed tracking method in a very noisy and complicated working environment such as the PCM (Pickling and Tandem Cold Rolling Mills) workplace in the POSCO (Pohang Steel Company). Since the workplace is so noisy and complicated that the workers can miss the fire alarm in the emergency, it is very important to monitor the worker's positions for their safety and easy rescue. Fig. 7 shows a layout of the PCM workplace, where it consists of two stories and has a physical size of 100m × 50m.

We determined the number and locations of ULGs and gateways such that there existed no non-propagation region of RF signals over the PCM. Throughout many trials and errors, we decided to use 49 ULGs and 9 gateways whose positions were marked as the rectangles. The server collecting all tag data is located in the PCM's operating room. The server collected all RSSI signals from the ULGs at 49 distinct locations, estimated the worker's locations using the proposed tracking algorithm, and displayed the estimated worker's locations in the real time manner. Currently, the implemented system can track 100 workers at the same time and its maximum error between the true location and the

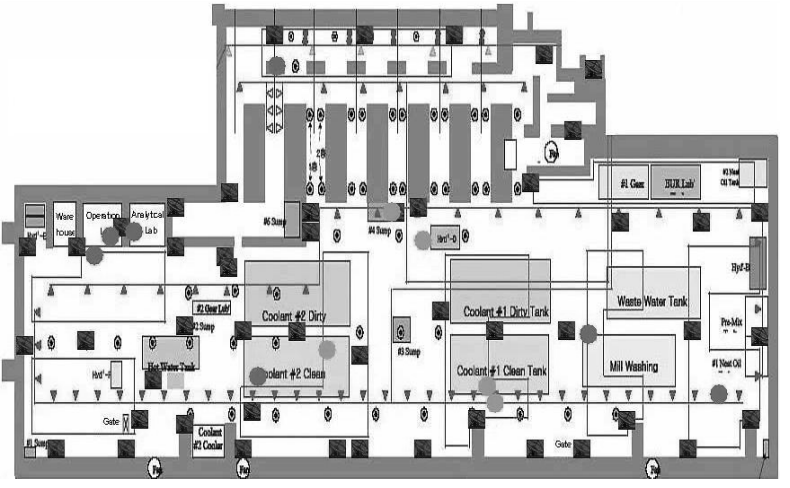


Fig. 7. A partial snapshot of people monitoring system

estimated location would be less than 3m. Fig. 7 illustrates a snapshot of the monitor screen of the implemented real-time people monitoring system, where 12 workers are successfully tracking in the real time, whose current locations are marked as the circles.

6 Conclusion

We proposed an epoch-making tracking algorithm combined with a preprocessing like a regression or SVM classifier. The proposed method consisted of a double layered structure, where the first layer performed the coarse location estimation using the regression or SVM classification, and the second layer performed the accurate location estimation using the particle filter whose initial location cues came from the coarsely estimated location from the first layer. This complementary working principle provided a better tracking performance than the existing tracking method such as the particle filter.

The proposed location tracking algorithm is so general that it can be applied to any kinds of location tracking modules like infrared light, ultrasound, and UWB sensors. And, it can provide a people's existence and/or location in the indoor corridor and the office within a limited size resolution ($2m \times 2m$). Also, the current location tracking system can track many peoples independently with a real-time even in the iron production company with a noisy, complicated, and wave-propagation hazard environment.

References

1. Hightower, J., Borriello, G.: Location systems for ubiquitous computing. Computer. **34** (2001) 56-66
2. Hightower, J.: The location stack. Ph.D. thesis (2004)

3. Fox, D.: Bayesian techniques for location estimation. (2003) 16-18
4. Guvenc, I., Abdallah, C., Jordan, R., Dedeoglu, O.: Enhancements to rss based indoor tracking systems using kalman filters. International Signal Processing Conference and Global Signal Processing Expo. (2003) 91-102
5. Hightower, J., Borriello, G.: Particle filters for location estimation in ubiquitous computing : A case study. The Sixth International Conference on Ubiquitous Computing(Ubicomp). (2004)
6. Burges, C.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery. **2** (1998) 121-167
7. Jordan, M.: An introduction to probabilistic graphical models. (2003)
8. Fox, D., Hightower, J., Liao, L., Schulz, D., Borriello, G.: Bayesian filters for location estimation. IEEE Pervasive Computing. **2** (2003) 24-33
9. Doucet, A., Godsill, S., Andrieu, C.: On sequential monte carlo sampling methods for bayesian filtering. Statistics and Computing **10** (2000) 197-208
10. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. International Journal of Computer Vision. **29** 5-28

Hybrid Predictors for Next Location Prediction

Jan Petzold, Faruk Bagci, Wolfgang Trumler, and Theo Ungerer

Institute of Computer Science,
University of Augsburg, Germany
{petzold, bagci, trumler, ungerer}@informatik.uni-augsburg.de

Abstract. Neural networks, Bayesian networks, Markov models, and state predictors are different methods to predict the next location. For all methods a lot of parameters must be set up which differ for each user. Therefore a complex configuration must be made before such a method can be used. A hybrid predictor can reduce the configuration overhead utilizing different prediction methods or configurations in parallel to yield different prediction results. A selector chooses the most appropriate prediction result from the result set of the base predictors. We propose and evaluate three principal hybrid predictor approaches – the warm-up predictor, the majority predictor, and the confidence predictor – with several variants. The hybrid predictors reached a higher prediction accuracy than the average of the prediction accuracies of the separately used predictors.

1 Introduction

A context aware application should be customized to the user's preferences. Furthermore the application could take over decisions from humans which are dictated by human habits. Therefore next contexts should be predicted such that the system can act proactively. Location is commonly viewed as the most important context and many techniques to predict the next location of a user exist: neural networks, Bayesian networks, Markov models, and state predictors are representatives. Typically numerous parameters must be determined for every user to set up one of the methods.

We investigated in our previous work different methods for next location prediction, e.g. different state predictors [1], the multi-layer perceptron [2], and a dynamic Bayesian networks [3]. We investigated for all techniques different parameter settings. The state predictors were evaluated with local and global patterns based on orders of 1 to 5. We also evaluated in this investigation the Markov predictor with the same parameters, pattern types and orders. Neural networks require several parameter settings for the structure like e.g. the number of input neurons and for the learning algorithm the learning rate. Parameters for the Bayesian network could be the history length and different time parameters like weekday and time of day. We observed for all techniques that the parameters that deliver the best prediction accuracy differ for each person.

In this paper we investigate hybrid predictors. A hybrid predictor consists of a set of base predictors which will be used to predict the next context. All

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 125–134, 2006.

base predictors deliver a prediction result and a selector determines the result of the hybrid predictor from the set of delivered results. The advantages could be a better prediction accuracy or with the same prediction accuracy a higher quantity. A disadvantage of the hybrid predictor is the additional expenses in storage and computing costs opposite to the single predictors.

Two critical choices must be made for a hybrid predictor: the choice of the set of base predictors and the selector. Our evaluations are done with a large number of base predictors from the state and Markov prediction methods (see table 1). We introduce for the selector three principal possibilities – a warm-up, a majority, and a confidence selector – and discuss several variants.

The paper is structured as follows. The next section describes the base predictors. The three proposed hybrid predictors are introduced in section 3. Section 4 shows the evaluation results. Related work is described in section 5. The paper ends with a conclusion.

2 Base Predictors

All known prediction techniques like neural networks or Bayesian networks can be used as base predictors in the predictor set. But in our evaluation we investigated only Markov and state predictors as base predictors of hybrid predictors.

2.1 Markov Predictor

Markov models seem a good approach for the next location prediction based on location histories. A Markov model regards a pattern of the last visited locations of a user to predict the next location. The length of the regarded pattern is called the order. Thus a Markov model with order 3 uses the last three visited locations. For all patterns the model stores the probabilities of the next location which is calculated from the whole sequence of the visited locations by the user. A simple Markov model is the Markov predictor [4,5]. A Markov predictor stores for every pattern the frequencies of the next locations.

2.2 State Predictor

A disadvantage of the Markov predictor is its bad relearning capability because of the frequency counter. After a habit change the new habit must be followed as often as the previous habit before the prediction is changed. The state predictors [1] prevent this problem. They use a finite automaton which is called two-state predictor for every pattern thus replacing the frequency counter of the Markov predictor. Because of the selection of the pattern and the use of the automaton the predictor is called two-level two-state predictor.

2.3 Local Predictors

The Markov predictor and the state predictor above considered a global location history. A local Markov predictor or a local two-level two-state predictor use only history of neighbor locations and disregard the movement sequences through all

locations. The current pattern does not contain the global history, i.e. which locations were entered before, but the local history, meaning the locations, which the user visited after the current location in the past.

2.4 Confidence Estimation

The hybrid predictor with the confidence selector uses the confidence of the base predictor to select the prediction result. In [6] three confidence estimation techniques were proposed – the strong state, the threshold, and the confidence counter method. By using the confidence estimation the prediction accuracy could be significantly improved. The confidence will be considered separately for every pattern of the state predictor and the Markov predictor respectively.

The strong state method can only be used with the two-level two-state predictors and provides two confidence levels. The current state of the two-state predictor is strong if the predictor needs two misprediction to change the prediction. In this case the predictor is assumed as confident. Otherwise the state is weak and the predictor is viewed as unconfident.

The threshold method considers the prediction accuracy of the last predictions. If this accuracy exceeds a specified threshold the predictor's confidence is classified as high. Otherwise the confidence of the predictor is classified as low.

The confidence counter method estimates the prediction accuracy with a saturation counter. Figure 1 shows a counter that consists of 4 states. The counter is adapted appropriate to the predictor's correctness. If the counter is in the state 3 or 2 the predictor is assumed as confident, otherwise the predictor is unconfident.

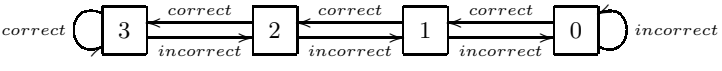


Fig. 1. Confidence counter

3 Hybrid Predictors

3.1 Warm-Up Predictor

On one hand there are predictors with a short learning time and low prediction accuracy, e.g. state or Markov predictors with a small order. On the other hand predictors exist which need a long time for learning but provide a better prediction accuracy e.g. predictors with a high order. The idea of the warm-up predictor is now to combine a fast learning predictor and a slow learning predictor. In the warm-up phase, where the slow learning predictor cannot deliver a result since a context pattern occurs the first time, the fast learning predictor is used to predict the next context. Successively, the better slow learning predictor is used. This principle will be implemented by Prediction by Partial Matching (PPM) and Simple Prediction by Partial Matching (SPPM) respectively.

The Prediction by Partial Matching (PPM) [4] from the area of data compression works as follows in the case of state predictors: A maximum order m is applied instead of the fixed order. Then, starting with this maximum order m, a pattern is searched according to the last m locations. If the pattern matches, the two-state predictor of this pattern is used to predict the next location. If no pattern of the length m is found, the pattern of the length $m-1$ is looked for, i.e. the last $m-1$ locations. This process can be accomplished until the order 1 is reached. If even a predictor of order 1 doesn't generate any prediction there will be no prediction. We propose to stop the PPM with order 1 because location prediction doesn't make sense when the current location isn't known.

A simplification of the PPM is the Simple Prediction by Partial Matching (SPPM). Also here a maximum order is applied. If no pattern with the length of the maximum order is found, the pattern of the length 1 is looked for. If a predictor of order 1 doesn't generate a prediction there will be no prediction. The SPPM method reflects the idea of the warm-up predictor perfectly.

3.2 Majority Predictor

The selector of a majority predictor considers only the predictors from the predictor set which deliver a prediction result. The prediction result of the hybrid predictor is the result which obtains the majority in the set of results delivered by the base predictors. There are three majority selection principles - the relative majority selector, the simple majority selector, and the absolute majority selector. A *relative* majority selector gives the result which appears most frequently in the set of results. A result is selected by the *simple* majority selector when more than half of the predictors deliver this result. If no result reaches the simple majority the hybrid predictor doesn't deliver a prediction. The *absolute* majority selector works like the simple majority selector but it considers all base predictors including the predictors which can't deliver a prediction result.

Figure 2 shows a majority predictor with simple majority. The predictor with relative majority predicts the same location like the predictor with simple

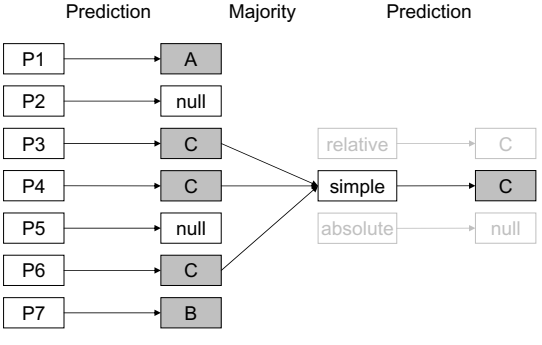


Fig. 2. Majority predictor with simple majority

majority. But the predictor with absolute majority cannot make a prediction in this example. Due to its small quantity of results reached, we omit the absolute majority selection principle in our evaluations.

3.3 Confidence Predictor

The confidence predictor operates as follows. First every predictor from the base predictor set is making a prediction. The results of the predictors will be classified by one of the confidence estimation method (see section 2.4) into confidence levels. The predictors on the highest level will be used to predict the next location. One of the proposed majority methods is used to select the result from the set of results of the predictors on the highest confidence level.

In the use of a confidence predictor there are three decision steps between the predictions of the base predictors in the predictor set and the final prediction of the hybrid predictor. Primary, which confidence estimation method will be used. There are three methods – the strong state, the threshold, and the confidence counter method. Thereby the confidence of every base predictor of the predictor set must be calculated by the same confidence method, since a comparison of the confidence level of different confidence methods isn't possible.

Secondary, a majority method must be chosen. After the classification of the prediction results in the confidence levels the predictor on the highest confidence level should be selected. In most cases this won't be a single predictor but many predictors will reach the highest level. Therefore a decision must be made by a majority method (see section 3.2) which prediction result is used for the hybrid predictor.

If the highest reached confidence level is very low, there is the problem that an unconfident predictor or a set of unconfident predictors are used for the prediction of the hybrid predictor. To eliminate this problem a tertiary criterion can be used which decides by a barrier whether the prediction result will be selected. By using a barrier we consider only predictors whose confidence is equal or higher than the barrier.

4 Evaluation

We show the evaluation results measured with the Augsburg Indoor Location Tracking Benchmarks [7] – a collection of movement data of four persons over several months. We compare the prediction accuracy and the quantity of the different predictors. The accuracies were calculated only with known patterns. That means prediction caused by patterns which occur the first time and deliver no useful prediction excluded from the accuracy. The quantity expresses the rate of made predictions to desired prediction which corresponds to the number of location changes.

In the evaluation we used only state and Markov predictors with different orders in the predictor sets. The predictor set of the warm-up predictor is predetermined by the PPM and SPPM principle respectively. To evaluate the majority and the confidence predictors the four predictor sets in table 1 are used.

Table 1. Predictor sets for the evaluation of the majority and the confidence predictors

	P_1	P_2	P_3	P_4
local one-level one-state predictor	•			
local one-level two-state predictor	•			
local two-level two-state predictor order 1 - 5	•	•	•	
local two-level two-state predictor PPM(5)	•			
local two-level two-state predictor SPPM(5)	•			
local two-level Markov predictor order 1 - 5	•	•		•
local two-level Markov predictor PPM(5)	•			
local two-level Markov predictor SPPM(5)	•			
global two-level two-state predictor order 1 - 5	•	•	•	
global two-level two-state predictor PPM(5)	•			
global two-level two-state predictor SPPM(5)	•			
global two-level Markov predictor order 1 - 5	•	•		•
global two-level Markov predictor PPM(5)	•			
global two-level Markov predictor SPPM(5)	•			

4.1 Warm-Up Predictor

We evaluated PPM and SPPM in four variants: local two-level two-state predictors, global two-level two-state predictors, local Markov predictors, and global Markov predictors all with maximum order 5. These we compared to the state and Markov predictors with order 1, 2, 3, 4, and 5. Figure 3 shows the average prediction accuracies reached for the four persons.

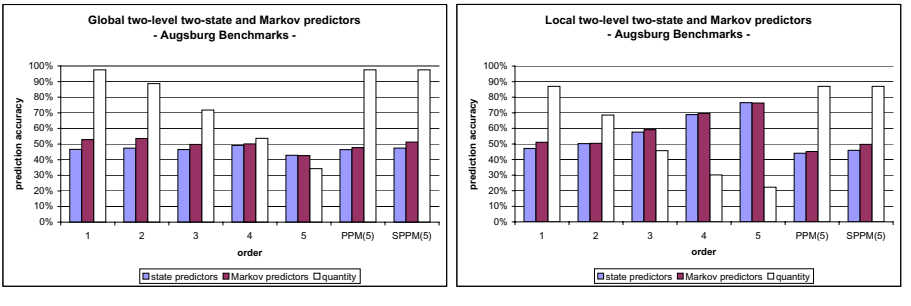


Fig. 3. Warm-up predictors: PPM and SPPM

The local base predictors work better than the corresponding local PPM and SPPM predictors. The global warm-up predictors work better than the global predictors with order 5. In the comparison of PPM and SPPM we see a better performance of the SPPM predictor than the PPM predictor in all cases. The charts show that the warm-up predictors yield no improvement in the used scenario.

4.2 Majority Predictor

Figure 4 shows the average prediction accuracies of the majority predictors. The measurements base on the four predictor sets P_1 to P_4.

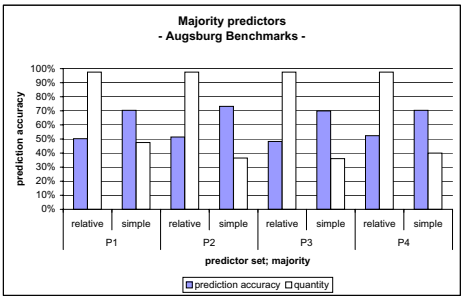


Fig. 4. Majority predictors

The predictors with the simple majority reach a better prediction accuracy than the predictors with relative majority. The best prediction accuracy was reached with the predictor set P_2 – state and Markov predictors. When we compare the result with the predictor from the predictor set P_2 (see figure 3), we can see that only the local two-level two-state and the local Markov predictors with order 5 work better. But the quantities of these predictors are lower than the quantity of the hybrid predictor. The predictors with relative majority reach a higher quantity than the predictors with simple majority, since when one predictor from the predictor set could deliver a result, the hybrid predictor is delivering a result.

These results show that the majority predictor with relative majority has no advantage over the single predictors. The predictor with simple majority reaches an accuracy which is clearly higher than the average of the accuracies of the single predictors. A good predictor set seems to be the set P_2. This set includes the state and Markov predictors as well as the local and global variants.

4.3 Confidence Predictor

The strong state confidence method can only be used with the predictor set P_3 which contains only state predictors. In the evaluation we investigated all variants of the confidence predictor described in section 3.3. The threshold method and the confidence counter method were evaluated with the four sets. The comparison of the four predictor sets showed that the confidence predictors reach similar prediction accuracies. The results showed only discrepancies of one percent. For this reason we consider in the following only the predictor set P_2 which

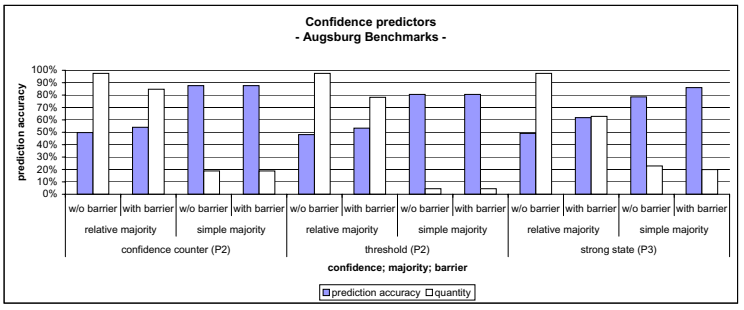


Fig. 5. Confidence predictors

provides a combination of the advantages of the local and global predictors as well as the state and the Markov predictors. Figure 5 shows the reached average prediction accuracies.

In the measurements the threshold method uses a barrier of 60% and the confidence counter uses four states. The confidence predictor using the simple majority with and without barrier perform identically for the threshold method and the confidence counter method. This fact shows that if more than half of the predictors on the highest confidence level deliver the same prediction result, then this confidence level is equal or higher than the used barrier.

For the strong state method a significant increase can be noticed from simple majority without barrier to simple majority with barrier. The reason is that the strong state method provides only two confidence levels. Thus predictors are often classified in the lower level. The use of the barrier in the predictor with the relative majority improves the prediction accuracy, but opposite the predictor with simple majority the results are very bad.

As expected from the investigation of the majority predictors the confidence predictors with simple majority perform essentially better than the predictors with relative majority. Predictors with relative majority selector work worse than the single predictors of the predictor set using the same confidence method. But the confidence predictors with simple majority are always better than the single predictors using the same confidence method.

The predictor using the threshold method delivers the lowest prediction accuracy. The use of the confidence counter method reaches the highest accuracy. If we consider additionally the quantity the threshold method shows a disadvantage opposite the other methods. All methods show that a higher prediction accuracy can only be reached with a lower quantity. A big difference exits between the quantity with relative majority and simple majority. Here the threshold method performs worst. The reason can be the continuous confidence level, so only few predictors from the predictor set reach highest confidence level.

The predictors with the simple majority and the barrier reach the highest prediction accuracy in all measurements. The threshold method shows always the lowest quantity which don't seems acceptable.

5 Related Work

Most context prediction approaches apply only a single prediction method [8,9, 10,11,12]. To our knowledge no hybrids are used in this application domain up to now.

The basic state predictor methods, the confidence estimation methods, and the hybrid predictor approach are motivated by branch prediction methods from the area of processor architecture. The idea of a warm-up predictor was proposed by Young and Smith [13] for hybrid branch predictors. During the warm-up phase of a hybrid branch predictor a static or a simple dynamic predictor should be used. This idea was improved by our warm-up predictor since our warm-up predictor can switch between the complex and the simple predictor already during the warm-up phase. In the processor architecture domain a static predictor is used for the first n million branches. After this n million branches the adaptive predictor is used only. The idea of the confidence predictor is based on the confidence estimation for branch prediction by Grunwald et al. [14]. For a hybrid branch prediction the predictor is used which reached the highest confidence level. The confidence in this case was the up to now reached prediction accuracy. This corresponds with our confidence predictor with threshold method.

6 Conclusion

For prediction methods a lot of parameters must be set up which differ for each user. Therefore a complex configuration must be made before a prediction method can be used. The paper presented three hybrid approaches which reduce the configuration overhead by automatically selecting the probably best result from differently configured base predictors and base predictors with different prediction methods respectively.

The warm-up predictor didn't achieve the desirable improvements. Opposite this the majority and confidence predictors reached a increase of the prediction accuracy or a prediction accuracy which was higher than the average of the prediction accuracies of the single predictors of the base predictor set. The decision of the majority in both cases – the majority predictor and the confidence predictor – fall on the simple majority. For the confidence predictor there wasn't a favorable confidence method. The use of the barrier achieves a small improvement for the strong state method. For the other methods there wasn't an improvement. The confidence predictors reach always slightly better prediction accuracies than the majority predictors. But the majority predictors show a significant better quantity than the confidence predictors in all tests. P_2 seems a good candidate for the predictor set. P_2 combines the advantages of the local and globale as well as of the state and the Markov predictors.

Future investigations of hybrid approaches could select also other prediction techniques like neural networks or Bayesian networks in the base predictor set.

References

1. Petzold, J., Bagci, F., Trumler, W., Ungerer, T.: Global and Local Context Prediction. In: Artificial Intelligence in Mobile Systems 2003 (AIMS 2003), Seattle, WA, USA (2003)
2. Vintan, L., Gellert, A., Petzold, J., Ungerer, T.: Person Movement Prediction Using Neural Networks. In: First Workshop on Modeling and Retrieval of Context, Ulm, Germany (2004)
3. Petzold, J., Pietzowski, A., Bagci, F., Trumler, W., Ungerer, T.: Prediction of Indoor Movements Using Bayesian Networks. In: Location- and Context-Awareness (LoCA 2005), Oberpfaffenhofen, Germany (2005)
4. Chen, I.C.K., Coffey, J.T., Mudge, T.N.: Analysis of Branch Prediction via Data Compression. In: ASPLOS VII, Cambridge, Massachusetts, USA (1996) 128–137
5. Ross, S.M.: Introduction to Probability Models. Academic Press (1985)
6. Petzold, J., Bagci, F., Trumler, W., Ungerer, T.: Confidence Estimation of the State Predictor Method. In: 2nd European Symposium on Ambient Intelligence, Eindhoven, The Netherlands (2004) 375–386
7. Petzold, J.: Augsburg Indoor Location Tracking Benchmarks. Context Database, Institute of Pervasive Computing, University of Linz, Austria. http://www.soft.uni-linz.ac.at/Research/Context_Database/index.php (2005)
8. Ashbrook, D., Starner, T.: Using GPS to learn significant locations and predict movement across multiple users. Personal and Ubiquitous Computing **7** (2003) 275–286
9. Bhattacharya, A., Das, S.K.: LeZi-Update: An Information-Theoretic Framework for Personal Mobility Tracking in PCS Networks. Wireless Networks **8** (2002) 121–135
10. Kaowthumrong, K., Lebsack, J., Han, R.: Automated Selection of the Active Device in Interactive Multi-Device Smart Spaces. In: Workshop at UbiComp'02: Supporting Spontaneous Interaction in Ubiquitous Computing Settings, Gothenburg, Sweden (2002)
11. Mozer, M.C.: The Neural Network House: An Environment that Adapts to its Inhabitants. In: AAAI Spring Symposium on Intelligent Environments, Menlo Park, CA, USA (1998) 110–114
12. Patterson, D.J., Liao, L., Fox, D., Kautz, H.: Inferring High-Level Behavior from Low-Level Sensors. In: 5th International Conference on Ubiquitous Computing, Seattle, WA, USA (2003) 73–89
13. Young, C., Smith, M.D.: Improving the Accuracy of Static Branch Prediction Using Branch Correlation. In: Proceedings of the Sixth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-VI), San Jose, USA (1994) 232–241
14. Grunwald, D., Klauser, A., Manne, S., Pleszkun, A.: Confidence Estimation for Speculation Control. In: Proceedings of the 29th Annual International Symposium on Computer Architecture, Barcelona, Spain (1998) 122–131

Psychology-Aware Video-Enabled Workplace

Marco Anisetti[1], Valerio Bellandi[1], Ernesto Damiani[1], Fabrizio Beverina[2], Maria Rita Ciceri[3], and Stefania Balzarotti[3]

[1] Department of Information Technology, University of Milan
via Bramante, 65 - 26013, Crema (CR), Italy
{anisetti, bellandi, damiani}@dti.unimi.it
[2] STMicroelectronics Advanced System
Research group Italy
fabrizio.beverina@st.com
[3] Communication Psychology LAB, Catholic University
of Milan Italy
{maria.ciceri, stefania.balzarotti}@unicatt.it

Abstract. The ability to recognize, interpret and express emotions plays a key role in human communication. Current computer interfaces have become able to "see", thanks to advanced video sensors and video processing algorithms; however, until recently they could not plausibly "guess" user intentions, because available feature extraction techniques could not provide the adequate level of service needed to support sophisticated interpretation capabilities. Our approach relies on a set of novel face and posture recognition techniques efficient and robust enough to be at the basis of a fully video-enabled intelligent pervasive workplace, capable of providing value added services based on the real time of facial and postural data. We propose to build on our current work in this area to create an infrastructure for lightweight facial and posture analysis allowing a variety of extended interactions between users and their work, market and entertainment environments.

1 Introduction

Video acquisition devices show a threefold trend of development: miniaturization to permit their bundling within mobile devices, large scale diffusion for allowing access to 3G services like video-calls, automatic surveillance and acquisition quality improvement for better performance. Considering both the current diffusion trend and the growing presence of wireless networks, some interesting scenarios are becoming realistic. A major one involves a new generation of wireless video sensors sharing information on the presence or identity of a subject in order to certify the subject's posture, movements and even emotional state. Much research has been carried out in the last ten years regarding face detection, identification and tracking. Early works had to deal with low computational power and low quality of videos; nevertheless some interesting results were achieved especially regarding emotional interpretation via optical flow [1], by 2D morphing [2] or feature tracking [3][4]; facial identification using eigenfaces [5] or fisherfaces [6]; facial tracking or more recent body tracking using even contours than 2D template tracking [7] [8]. These early works had a significant influence on more recent

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 135–146, 2006.

approaches, which can rely on a growing quality of sensors and a much greater computational power, together with decreasing costs. In human interaction non-verbal signals (in particular, face and posture) are a very important sources of information (e.g. about identification of the subject, decoding of his level of alertness, interest and emotional engagement). In the interaction between human and artificial agents a particular interest has concerned design and implementation of interfaces able to recognize the user's emotions from real-time capturing and processing of sensory modalities input via various media [20,21]. In fact non verbal emotional expression allows to disambiguate the verbal message content and to support the multi-modal richness of face-to-face communication [22]. Emotional models have been proposed as a critical component of more effective human computer interaction: explicit attention to the emotional aspects aims at increasing the system performance in terms of usability and acceptance. In this sense, encoding facial expression of emotions play an important role in the design of interfaces: they should not be considered a simple optional providing pleasantness but they represent crucial cues as they are involved in the selection, regulation and motivation to action [23]. Besides the well-known use of face recognition in the framework of hybrid biometric authentication system, face data interpretation can lead to enforcing declarative policies including conditions on human attitude or behavior (For instance, it can be thought a system that control the attention level of the operators that, in an airport, have to check each piece of luggage passing under the X-ray machine). Among many interesting works on video analysis [9], only a few turned out to be suited to general purpose applications. A major research problem is making the video analysis system robust to occlusion, morphing, and changes in illumination without previous information about the application environment. Also, much work remains to be done to make video analysis efficient enough to be executed in real-time with while retaining high precision and without dependency on the training set. About video analysis the most promising techniques use 3D models of faces [10] or 3D information on the facial shape extracted by 3D scan or inference [11] by multiple cameras (or stereo cameras). Recently, partial but promising solutions have been proposed to the problem of matching a face model with a video stream (often called the identification problem). Such solutions use 3D shape matching [12], coupled with revised versions of the original eigenface ideas. Another fundamental problem is using the video stream to infer a subject's emotional state or intentions. Recent approaches use complex morphable models built with a high number of correlated 3D scans, created from an optical flow correlated with a 3D motion vector [13]. All these techniques rely on the presence of an initialization step for selecting a good initial position for video tracking of facial features [14]. These results notwithstanding, a major problem still remains to be solved: the applicability of 3D techniques in real workplace environments. For that reasons our work focuses on general applicability, using a generic 3D rough model capable of morphing and moving in a realistic manner without any training. This model permits to satisfy the real-time constraint both for facial and gesture data. Using our 3D model together with new correction techniques we are also able to deal with illumination variations without any a priori knowledge about environment illumination conditions, and to maintain the tracking or identification of a subject during long periods of time.

2 Design Details

As anticipated above, the goal of our design is a layered generic architecture, where the lowest layer (Layer 1) provides functionalities to manage and collect information about gesture and facial expressions from inexpensive video sensors. Layer 2 processes this information in order to extract high level knowledge regarding people identity, attitude and emotional status and shares this knowledge with applications (layer 3)[1]. In order to develop such an architecture, we aim to customize the core technologies for facial expression recognition and identification; in particular, we are working on security oriented technologies for face expression inference and identification based on recognition and tracking with standard 3D morphable models. Another goal is designing, developing and evaluating prototypes of smart user-adaptive interfaces for internet-based communication, supporting non-intrusive user profiling models employing emotional states representation to complement other information like navigation behaviour. Our generic architecture will be tailored to specific pervasive computing scenarios, including the following ones:

- *Implicit and explicit remote interfaces*. Implicit interfaces provide ambient safety, automatically reacting in case of aggressive behaviour or when some video detected action occurs. For explicit interfaces, our goal is to create a gesture based remote interface providing a human-centred ambient interaction. Also, information on users emotional state will be used to tune the systems reactions. A sample application of explicit interfaces is an automatic meeting monitor, based on streaming video analysis for speakers identification and automatic speech recognition for report creation. Such a monitor can flag violations to corporate policies, e.g. on need-to-know knowledge sharing.
- *Intelligent shop-floor and virtual marketplace adaptation*. A goal of our project is to provide an intelligent interface to a real commercial environment (e.g., a shopping mall) using the standard surveillance cameras. The aims of this intelligent environment are verifying the reaction of a buyer by emotional and gesture recognition, store and modify buyer profiles by facial identification and automatic fidelity card management. Another sample application is the online automatic adaptation of virtual commercial environment (e.g., a web shop, or a remote interview site) using inferences on the emotional state of the user and an automatic profiling process.

2.1 Architecture

Our proposed architecture can be functionally decomposed into three layers: a high performance layer for data collection and conditioning, an SLA (Service Level Agreement) intermediate layer extracting knowledge and certifying its quality to a pervasive application layer evaluating conditions on knowledge items. We shall now describe in some detail the functionality of each layer.

Layer 1: Video analysis layer. This layer is aimed at designing a software layer collecting the data from single/multiple video streams, process them using novel low-level

[1] A major challenge to be faced is deploying this architecture in a real environment, reusing where possible existing video sensors such as surveillance cameras.

video analysis algorithms and produce data conditioned for the upper layers. Some examples of conditioned data are features for identification or emotional inference, as well as for any kind of video analysis for knowledge extraction. All these techniques are very adaptable to environmental conditions because of the diffusion of video acquisition sensors that can be placed outdoors, indoors, or even on mobile devices. All features extracted by this layer are functional to the application; for instance, only AU (Action Unit of Facial Action Coding System FACS) are extracted for facial expression detection and local features for identification. In this layer sensor technologies have a great importance as well as algorithms and processing techniques.

Layer 2: SLA and interpretation. This layer is aimed at designing a software layer supporting a standard interface to video-enabled applications. Today, digital business transactions are increasingly performed in diverse situations, using a variety of mobile devices and across multiple communication channels. In the new paradigm of distributed access to the communication and computing infrastructure, a much richer context representation must be provided to application in a standard format. Considering for example identification predicates, the main characteristic of identification during streaming video is that level of confidence can evolve over time. Our goal is to create a dynamically managed infrastructure providing applications with clearly defined, semantically uniform dynamic and static predicates, as well as their associated SLA metrics. This characterization, when applied to expression-based facial features, suggests the creation of an avatar for expression evaluation testing and SLA validation. Besides implementing these ideas on a standard server-based architecture, our aim is also to investigate its feasibility in a peer-to-peer (P2P) setting where independent peers equipped with sensors will share video information and take decisions based on it.

Layer 3: Application layer. We plan to implement and deploy some application pilots. We plan four types of application pilots: *i)* A Shelf space environment reacting to subject action (attention level, motion etc). *ii)* A Human Resources Selector where the environment is able to analyze emotional state of the candidate during work parley. *iii)* A Surveillance environment capable of monitoring an environment and taking some crucial security action. Using the knowledge of the position of other cameras to request frames from them can improve the quality of location or produce a more reliable identification using face information. *iv)* A Pointer device application where cooperation between pointer device traditional transducers and video sensors that analyze a video stream showing the pointer device could produce a more robust pointer precision. This new pointer could be applied, for instance, to video-games.

3 Core Technologies

Our overall work is based on core technologies for video streaming analysis (face recognition, identification and expression recognition) and for subject behaviour clustering. Following we present a brief description of each core technologies starting from the general purpose to focused ones.

Motion and face detection. For this topic there are many different approach in literature, a dissertation on this techniques not is the scope of this paper. For that reason we

focus this section on our techniques that relies on skinmap and motion detection for skin validation as a starting point of area clustering followed by two principal techniques:

- Manual feature selection for mask placing.
- Automatical mask placing with one step morphing algorithm.

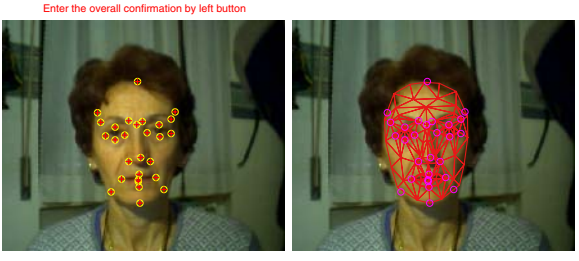


Fig. 1. One example of feature point selection and mask fitting

Figure 1 shows an manual feature selection process that relies on local selection of some points of interest (27 points). The mask placing process use POSIT [29] approach for postural inference and shape unit estimation to obtain the best fitting mask also described in figure 1. The second approach, the Automatic Mask placing is currently on advanced testing pase and produce a very encouraging results. The principle is to use a general purpose face texture as a starting point for a morphable process that try to fit the mask with the real face using our tracking algorithm for posture evaluation and shape morphing. This kind of approach is similar to [12] but include also posture evaluation do not need any training phase and work wit our general purpose 3D face model. The initial results on our database, on Cohn-Kanade database [28] and on UMST database produce an encouraging percentage of correct fitting around 70%. Many works must be done in this issue to reinforce this automatic fitting process.

Face tracking with expression recognition. The primary challenge in this complex tracking task is to extract simultaneously and with high computation efficiency many parameters, including posture-related ones. To overcome with numerous problem of facial tracking issue we use several techniques starting from occlusion management system for self and external occlusions (mask and IRLS [27]) and dissimilarity analysis for quality of tracking testing and automatic correction and template update. Figure 2 shows an example of self occlusion and dissimilarity techniques applies to tracking based application. Expression recognition and tracking must work also for unknown persons (person that do not have an entry in identification database) i.e. tracking must work with a general 3D facial model when a subject specific one is not available. In [18] we have described an innovative robust expression face tracker including a posture confidence evaluation. Our algorithm is robust w.r.t. morphing and illumination changes in spite of the difference between the 3D face model and the real subject face.Regarding expression recognition our team has built an automatic expression features extraction system, described in [19]. Our system is able to extract features according to the Facial Action Coding System (FACS)[15] proposed by Ekman and Friesen. FACS is a

Fig. 2. Examples of face tracking techniques. Starting from the left: self occlusion mask in template (Eyes regions are masked as a very noisy area due blincking and frequent motion), IRLS weight and local feature selection on retrieved face for dissimilarity analysis.

comprehensive, anatomically based system for measuring all visually discernible facial movement. FACS describes all visually distinguishable facial activity on the basis of 44 unique action units (AUs), as well as several categories of head and eye positions and movements. Each AU has a numeric code, the designation of which is fairly arbitrary. FACS coding procedures allow for coding of the intensity of each facial action on a 5-point intensity scale, for the timing of facial actions, and for the coding of facial expressions in terms of events. An event is the AU-based description of each facial expression, which may consist of a single AU or many AUs contracted as a single expression. Tables 1 and 2 show our good results of our system for AU inference.

Face Normalization with local analysis. Face normalization is directly linked to face expression and posture tracking. In fact using our tracking approach it becomes possible to have a normalized neutral face for each frame of the videos stream as occurs for dissimilarity evaluation. On this normalized face some interesting local analysis can be easily carried out. In this issue we developed a robust and precise techniques for eyes and lids analysis [19]. Our eyes state algorithms is based on the analysis of lids positions, and iris position, on normalized face. The lids are fundamental for determination the state of the eyes. In application like gesture recognition, state-based model had been used [4]. In our algorithm, the transaction between opened eyes and closed eyes states is defined by the tracking of the lids in conjunction with possibility of finding the iris. In fact iris can provided an important information about the eyes state. For eyelid tracking we used a 1D (vertical) Lucas-Kanade feature tracker, in inverse compositional implementation and with linear-appearance variation (LAV) technique for illumination

Table 1. AU classification by fuzzy rules. Because of nature of our features some coupled AU have a better recognition results like 1+2.

AU	# of Sample	Correct Recog	Misses	False positive	% Correct Recog
AU 1	19	15	4	1	79%
AU 2	12	10	2	0	83%
AU 4	33	30	3	0	90%
AU 5	10	9	2	0	90%
AU 6	16	15	1	1	93%
AU 7	25	20	5	7	80%
AU 1+2	14	13	2	1	92%
AU 1+2+5	8	7	1	0	87%
AU 1+2+7	4	4	0	1	100%
AU 1+4	5	2	3	1	40%

Table 2. AU classification by fuzzy rules on eyes feature for first half of table and by head tracking algorithm for the second half. Because of precision of our head tracking algorithm and nature of the features (from 51 to 56) we have a perfect classification.

AU	# of Sample	Correct Recog	Misses	False positive	% Correct Recog
AU 41	15	15	0	1	100%
AU 42	10	7	3	2	70%
AU 43	30	30	0	0	100%
AU 44	2	0	2	5	0%
AU 45	40	33	5	1	82%
AU 46	2	2	0	0	100%
AU 61	20	18	2	0	90%
AU 62	32	31	1	0	96%
AU 63	34	30	4	1	88%
AU 64	32	28	4	7	87%
AU 51	33	33	0	0	100%
AU 52	10	10	0	0	100%
AU 53	16	16	0	0	100%
AU 54	25	25	0	0	100%
AU 55	16	16	0	0	100%
AU 56	25	25	0	0	100%

changes. With this implementation we are able to track exactly the lids with height confidence and to determinate the eyes state. Because of the fast movement of lids and the extremely changeability of the area around the eyes, the tracking may fails. In this case we used an alternative technique based on skin-map and Canny edge-detector operator. The main idea of this lids tracking strategies be founded on assumption that the lids position could be well defined by no-skin region on eye area. From our experience this idea work well for lower lids because of better general illumination condition. For upper

lids because skin-map techniques may produce a wrong tracking results we combine the skin results with Canny operator on the region of iris. Therefor to perform this technique we need an estimation of the iris center to evaluate the reliability of the extracted values. Considering the iris location in cases of lids tracking fail we proceed directly on the estimation of the iris center on the default area of the face related to the eyes positions and not on the specific one delimited between the hight confidence position of the eyelids produced by tracking. (Figure 3). Other interesting furrows analysis can be carried

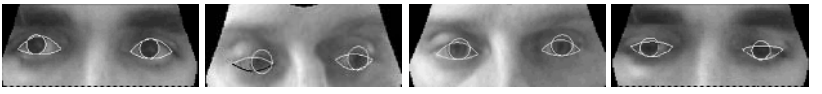


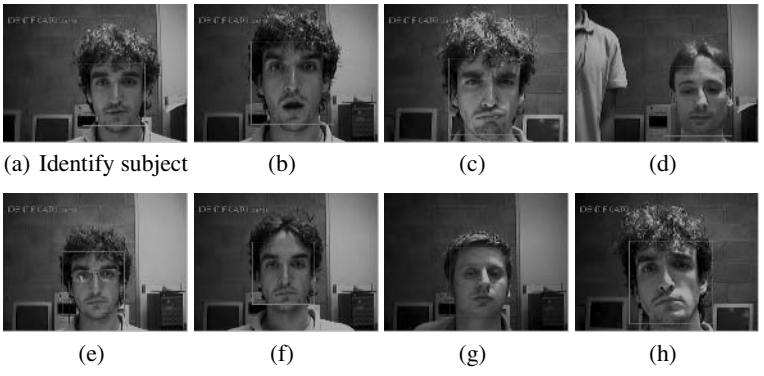
Fig. 3. Eye Tracking: in black eyelids low confidence estimation that is defined by Skin-map-Canny tracking technique

out on normalized face, for instance for finding certain local muscular movement or for validate the expression estimation obtained by tracking approach explain before.

Face Identification. Our identification process is based on head and expression tracking for facial normalization. We have developed a promising face identification technique using an innovative one step tracking algorithm for facial posture and expression estimation [17]. Our work will take advantage of using lightweight automatic feature extracting techniques and movement prediction techniques like the one developed by Microsoft Research [16]. This identity certification process, in case of correct identification, provides a measure of identification quality for each frame of the captured video. With this technique we are able to prevent at last 21% of FAR (False Acceptance Rate) with maximum 2% of FRR (False Rejection Rate). To improve the FAR of our identification technique we have developed a technique which we have called "dissimilarity analysis". This technique works on a normalized frontal face and estimates the imprecision of the tracking process as the probability of correct identification. In the framework of this work, further investigation will be done on normalized faces to eliminate ambiguous cases and improve FAR, in this framework we are currently testing eigenface approach for identification on normalized retrieved face obtaining great results. Figure (4) shows some examples of identification robustness w.r.t. appearance changing. Some frames captured by a video-camera are shown to underline the robustness of our identification process.

Another experiment was focused on identifying expression invariants. We used the Cohn-Kanade data base and many different subjects including some neutral frames from which we could construct our 3D template database. In the following tables (Tab.(3), Tab.(4)) we summarize our results. Based on intensity of expression we classify expressions from neutral to average, up to maximum intensity.

Thus, we are able to extract simultaneously information about expression and identity.



(a) Identify subject (b) (c) (d)

(e) (f) (g) (h)

Fig. 4. Example of identification tests across expression and illuminations changing. Subfigure (a) represent the subject that wants to be identified. In Subfigure (d)(g) false identity declarations without system accept response.

Table 3. Percentage of identifications in cases of expression variation using Cohn-Kanade, our own authentication database and database of A-lea project

database	neutral	low	average	high
Cohn-Kanade	99%	98%	92%	89%
Authentication DB	98%	96%	89%	88%
A-lea	97%	96%	91%	85%

Table 4. Percentage of identification in cases of pose variations using ELITE2002 database (misured in degree)

database	0-5	5-25	25-45	>45
ELITE2002	98%	96%	89%	79%

4 One Case Study: Alea Project

An enabling system was set up at the Psychology Communication Lab of the Catholic University of Milan for implementing experimental sessions. Different kinds of devices were used to record the subject's behaviors and all instruments were synchronized: *i)* Two high resolution web cameras, *ii)* Physiological recordings were taken using the BIOPAC System (BIOPAC System, Inc.; Goleta, CA) and *iii)* A high quality microphone to record vocal reports. Subjects were asked to use the computer where an avatar guided them across the three different kinds of computer games. All sessions started with 2 minutes of free exploration of a web site for the baseline measure of physiological signals. Total duration was of about 20 minutes. They were divided into two different groups, according to the kind of information received by the avatar. In particular, this experimental research will make use a sort of the Wizard of Oz approach

[26], that is it employs an initially simulated prototype of emotional intelligent interface: the subjects are told they are interacting with an emotional intelligent computer, though in fact they are not. In the experimental condition, the subjects were exposed to a simulated emotional-intelligent computer, where the avatar provided a simulated intelligent feedback to the user to decode his emotional state and to adapt the tasks accordingly. For example, the avatar used sentences like: "You seem to be bored, so here it is a new game", "You are in difficulties: I repeat the instructions for you". The simulated emotional-intelligent computer appeared to be automatic to the subjects, but it was actually controlled by an out of sight experimenter. In the control condition the avatar guided the subjects in the same activities but did not simulate to decode emotion. All subjects were alone in the room with the computer. A rigid protocol was set in order to create some emotion eliciting stimuli in an **ecological way** (i.e. the subject is unaware of the reasons of the experiment and that he is recorded by a web-camera). Three different kinds of computer games were projected to modify the subject's attention level and to elicit specific emotional reactions. Specifically, game events were supposed to support four emotional evaluation checks: 1. novelty (a change in the situation that captures the subject's attention); 2. hedonic value (intrinsic pleasantness or unpleasantness of the stimulus); 3. goal conduciveness (events that help or damage the subject to reach a goal); 4. coping, (increasing levels of difficulty of tasks that change the estimated ability to deal with them). All games were previously tested on 10 subjects to assess their efficacy. Data were collected in different response domains: non verbal expressive behavior, physiological signals and self-report. Non verbal Behavior: A first level of analysis consisted of a behavioral micro-analysis. All video tapes were codified frame by frame (25 fps) with our software. Four macro-categories were then considered. **Facial movements**: the fundamental muscle movements that comprise Facial Action Coding System ([15]) were selected. We considered action units relating to upper face and lower face. **Gaze** direction: we considered if the subject looks at the screen, at the keyboard, around, etc. **Posture**: behavioral units of moving near to/far from the screen were considered.

5 Conclusion and Discussion

Humans have an inherent tendency to interact with computer systems by adopting ways that are common to human-human interaction. In "The Media Equation" Reeves and Nass [24] argue that human-machine interaction is inherently natural and social, so that the rules of human-human interaction apply to human-machine interaction: in many ways people seem to respond psychologically to interactive computers as they were human actors and not tools [25]. Agents that show to understand emotion and behave like humans in the environment where they interact with users (such as computer games or tutoring environments) are more enjoyable, engaging and efficiency. Current computer interfaces may be able to "see" thanks to video acquisition devices; however, they cannot plausibly guess user intentions, because they lack interpretation capabilities. Our project is aimed at producing "interfaces that can guess" with a sufficient level of quality to be of real help to human users and application. As anticipated above, the goal of our project is to provide an intelligent system able to manage and collect feature infor-

mation about gesture and facial expressions from inexpensive video sensors (layer 1), extract high level knowledge regarding people identity and emotional status (layer 2) and use this information for novel applications in the scenarios above mentioned (layer 3). The main challenge of our project is to deal with real environment setting problems, reusing where possible the existing video acquisition infrastructure such as surveillance cameras. Our aim is to improve the core technologies for facial expression recognition and identification; in particular we intend to develop technologies for face expression inference and identification based on recognition and tracking with standard 3D morphable models. Another major aim is to design, develop and evaluate prototypes of smart user-adaptive interfaces for internet-based communication, supporting non-intrusive user profiling models employing emotional states representation to complement other information like navigation behaviour. We plan to demonstrate an infrastructure for lightweight, online facial and posture analysis suitable for complementing ambient interfaces, allowing a variety of extended interactions between users and their work, market and entertainment environment. We are currently developing the line of research described in this paper. Some encouraging results have been achieved, especially regarding the core technologies of video analysis [17][18][19]. These encouraging results reinforce our vision of interfaces that can guess as a basic component of a human-centered pervasive environment, where security policies including a new type of conditions on human attitude and behavior can be enforced in a distributed, efficient way.

References

1. Essa,I. A., Pentland, A.: A Vision System for Observing and Extracting Facial Action Parameters. IEEE CVPR 1994 Conference (1994) 76-83.
2. Baker, S., Matthews,I.: Active Appearance Models Revisited. Carnegie Mellon University School of Computer Science, Pittsburg (2002).
3. Black, M.J., Yacoob,Y.: Recognizing facial expressionin image sequences using local parameterized. Int'l j. Computer Vision, Vol. 25 (1997) 23-48.
4. Tian, Y., Kanade, T., Cohn, J.: Recognizing action units for facial expression analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, February (2001) 97-115.
5. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience, Vol. 3 (1991) 71-86.
6. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, July (1997) 711-720.
7. Bregler, C., Malik, J.: Tracking People with Twists and exponential maps. In CVPR98 (1998) 8-15.
8. Hager, G.D., Belhumeur, P.N.: Efficent Region Tracking With Parametric Models of Geometry and Illumination. IEEE Transactions on Pattern Analysis and Machine Intelligence (1998) 1125-1139.
9. Baker, S., Matthews, I., Xiao, J., Gross, R., Kanade, T., Ishikawa, T.: Real-Time Non-Rigid Driver Head Tracking for Driver Mental State Estimation. 11th World Congress on Intelligent Transportation Systems, October (2004).
10. Xiao, J., Kanade, T., Cohn, J.F.: Robust Full-Motion Recovery of Head by Dynamic Templates and Re-registration Techniques. Proceeding of the fifth IEEE international Conference on automatic face and gesture recognition (2002).

11. Lu, X., Jain, A.K., Colbry, D.O.: Matching 2.5D Face Scans to 3D Models. IEEE Transaction on pattern analysis and machine intelligence, Vol. 28 (2006).
12. Blanz, V., Vetter, T.: Face Recognition Based on Fitting a 3D Morphable Model. IEEE Transaction on pattern analysis and machine intelligence, Vol. 25 (2003).
13. Vlasic, D., Brand, M., Pfister, H., Popovic, J.: Face Transfer with Multilinear Models. SIGGRAPH (2005).
14. Feris, R.S., Gemmell, J., Toyama, K., Kruger, V.: Facial features detection using a Hierarchical Wavelet Face Database. 5th Int conf of an Automatic Face and Gesture Recognition.
15. Ekman, P., Friesen, W.: Facial action coding system: A technique for the measurement of facial movement. Consulting Psychologists Press (1978).
16. Perez, P., Vermaak, J., Blake, A.: Data fusion for Visual Tracking with particles. Proc of the IEEE (2004) 495-513.
17. Damiani, E., Anisetti, M., Bellandi, V., Beverina, F.: Facial identification problem: A tracking based approach. IEEE International Symposium on Signal-Image Technology and Internet-Based Systems (IEEE SITIS'05). Yaoundé Cameroon (2005).
18. Anisetti, M., Bellandi, V., Damiani, E., Beverina, F.: 3D espressive face model-based tracking algorithm. Signal Processing, Pattern Recognition, And Applications (SPPRA 2006). Innsbruck (2006).
19. Bellandi, V., Anisetti, M., Beverina, F.: Upper-Face Expression Features Extraction System for Video Sequences. Visualization Imaging, and Image Processing (VIIP05). Benidorm Spain, Sept. (2005) 83-88.
20. Picard, R.W.: Affective Computing. MIT Press. Cambridge (1997).
21. Picard, R.W.: What does it mean for a computer to "have" emotions?. In: Trappl, R., Petta, P., Payr, S.: Emotions in Humans and Artifacts. MIT Press (2003).
22. Scherer, K.R.: Appraisal considered as a process of multilevel sequential checking. In: Scherer, K., Schorr, A., Johnstone, T. (eds): Appraisal processes in emotion: Theory, methods, research. Series in affective science, London, Oxford University Press, xiv, 478 (2001).
23. Ciceri, R., Balzarotti, S.: Analysis of the human physiological responses and emotional multimodal signals to an interactive computer. In: Agents that Want and Like: Motivational and Emotional Roots of Cognition and Action. UH The University of Hertfordshire, Hattfiled, (2005) 25-34, ISBN 902956 41 7
24. Reeves, B., Nass, C.: The Media Equation. How People Treat Computers, Television and New Media Like Real People and Places. CSLI Publications, Centre for the Study of Language and Information. Cambridge University Press (1998).
25. Gratch, J., Marcella, S.: Evaluating the modelling and use of emotion in virtual humans. Proceedings of the 3rd International joint Conference on Autonomous Agents and Multi agent Systems (2004).
26. Picard, R.W., Klein, J.: Computers that Recognise and Respond to User Emotion: Theoretical and Practical Implications. Interacting with Computers, (2002) 141-169.
27. Baker, S., Matthews, I.: Lucas-kanade 20 years on: A unifying framework. International Journal of Computer Vision, Vol. 56, February (2004) 221-255.
28. Kanade, T., Cohn, J., Tian, Y.: Comprehensive database for facial expression analysis. Proc. 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00). Grenoble France (2000) 46-53.
29. DeMenthon, D., Davis, L.: Model-based object pose in 25 lines of code. International journal of computer vision, IJCV, Vol. 15, June (1995) 123-141.

Distributed Embedded Intelligence Room with Multi-agent Cooperative Learning

Kevin I-Kai Wang, Waleed H. Abdulla, and Zoran Salcic

Department of Electrical and Computer Engineering, University of Auckland. Private Bag 92019, Auckland, New Zealand
{kevin.wang, w.abdulla, z.salcic}@auckland.ac.nz

Abstract. In this paper, a novel Multi-agent control system with fuzzy inference learning and its physical testbed are presented. In the Multi-agent system, distributed controlling, monitoring and cooperative learning are achieved through ubiquitous computing paradigm. The physical testbed named Distributed Embedded Intelligence Room (DEIR) is equipped with a fair amount of embedded devices interconnected in three types of physical networks, namely LonWorks network, RS-485 network and IP network. The changes of environment states and user actions are recorded by software agents and are processed by fuzzy inference learning algorithm to form fuzzy rules that capture user behaviour. With these rules, fuzzy logic controllers can perform user preferred control actions. Comparative analysis shows our control system has achieved noticeable improvement in control accuracy compared to the other offline control system.

1 Introduction

Ubiquitous computing is a new paradigm of information processing technology where the computation is carried out by a group of invisible embedded devices rather than by visible computers. By assuming ubiquitous computing in the background, ambient intelligence aims to achieve a system that is aware of the environment context, able to model and adapt to user's behaviour and should respond on user's behalf [1], [2].

In this paper, a novel Multi-agent (MA) control system with cooperative fuzzy inference learning is proposed. In the proposed system, massive amount of embedded devices are connected via different physical networks. A middleware layer above the physical network provides a unique control interface for the high level MA control system to all the physical networks. In the MA control system, software agents are distributed in nature, each monitoring and controlling its own embedded device. The Multi-agent Fuzzy Inference System (MAFIS) is developed to capture and model user's daily activities and to perform sensible control actions on user's behalf. In addition to the software based MA control system, physical testbed named Distributed Embedded Intelligence Room (DEIR) built in the University of Auckland (UoA) is also introduced.

The rest of the paper is organised as follows. Section 2 gives a brief description of our target environment and related researches. Section 3 details our physical testbed, DEIR, including the implementation of physical network infrastructure, middleware

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 147 – 156, 2006.

and the MA control system. Section 4 explains the process of fuzzy inference learning of MAFIS. Section 5 discusses the comparative analysis carried out between MAFIS and the other offline control system. Section 6 indicates some future works within our project and section 7 concludes the paper.

2 Target Intelligent Environment

2.1 SOHO Environment

The target Intelligent Environment (IE) in our research is the so-called "small office, home office" (SOHO) environment. In such kind of environment, working, living and entertaining functionalities are fully integrated. Typical devices that could be in such environments include TV, Hi-Fi audio system, mobile devices such as cell phone or PDA, air conditioning system, light control system and a variety of sensors and actuators.

In order to model human activities in such environments and to test the developed control system in a realistic way, a physical testbed is essential. In the UoA, a physical testbed called DEIR has been constructed by the Embedded Systems Research Group (ESRG). This physical testbed is equipped with a number of sensors including light intensity, temperature, pressure, smoke, and occupancy sensors to monitor the environment states. It also contains a number of actuators for automating windows, blinds, dimmable lights and house appliances. These devices can be controlled via traditional switch interface, remote control, and PC interface. Users need not to be aware of the existence of vast number of embedded devices. DEIR forms a comprehensive testbed which allows various types of experiments such as human activity analysis, control system verification, and remote monitoring and controlling. More information on physical network, middleware and control system will be given in section 3.

2.2 Related Works

The MIT AI Lab started IE researches around mid 90s [3]. At that time, their research focus was to introduce intelligence via smart sensors and camera networks and can be considered as Human-Computer Interaction (HCI) and sensors network research. In 1999, a MAS called Metaglue which had no built-in intelligence, was developed in that lab to control the IE [4]. However, in the past few years, intelligent knowledge-based resource management [5] and reactive behaviour systems [6] had been developed and integrated into the MAS to introduce intelligence.

The University of Essex is also one of the pioneers in this research field. Their research focuses on online learning of personalised behaviour which is inline with our research [7]. The core learning process of the proposed MAFIS is developed based on their Adaptive Online Fuzzy Inference System (AOFIS) [8].

The Adaptive Building Intelligence (ABI) project collaborated by several Swiss universities uses MAS approach as well. This project is different to ours in the way

that it is aimed at providing intelligent building services rather than intelligent living spaces [9].

There are also many other research efforts such as the Microsoft Smart House [10], IBM BlueSpace [11] and MASSIHN project [12]. However, most of them focus on integrating working behaviour and device automation into the control system. These control systems neither capture or model the human behaviour nor adapt to human needs, and do not reveal the true meaning of ambient intelligence.

3 Distributed Embedded Intelligence Room (DEIR) Architecture

In this section, DEIR architecture will be discussed in three components. The first component consists of a collection of physical device networks. The second component includes the corresponding control software for each device network and the middleware layer. The third component is the MA control system with cooperative fuzzy inference learning. The conceptual system architecture can be depicted in Fig. 1.

3.1 Physical Network Infrastructure

In the current version of DEIR, three types of device networks are implemented, namely the IP network, LonWorks network [13] and RS-485 network. In general, LonWorks network is mainly used for connecting embedded sensors whereas RS-485 network is for controlling automatic devices such as windows, blinds and lights. In a higher level, IP network is responsible for controlling IP-based network devices such as IP-cam.

In LonWorks network, an iLon 100 router is used as a hardware interface between the control software and embedded sensors. RS-485 network uses a combination of smart switches and a hardware gateway server to connect a group of automatic devices with the control software. Each smart switch has 2 microcontrollers, Motorola HC11 and PIC, and a infra-red receiver integrated in it which allows up to 3 devices to be controlled via traditional switch interface or infra-red remote control interface. All the switches are connected with the gateway server which in turn communicates with the control software. By using the control software, PC control interface is also possible.

Similar to the other two device networks, multiple IP-based network devices can be connected to the system using one or more network routers or switches depending on the system topology. However, as shown Fig. 1, the IP network is at a higher level in the system architecture. Therefore, IP network is expected to provide other functionalities. By adding another layer of middleware, different low level device networks can be mapped onto the IP network, and hence the high level control system can treat the whole physical network in IP network as one entity. Further, IP network also has the flexibility of integrating other advanced wireless technologies such as Wi-Fi, Zigbee and Bluetooth.

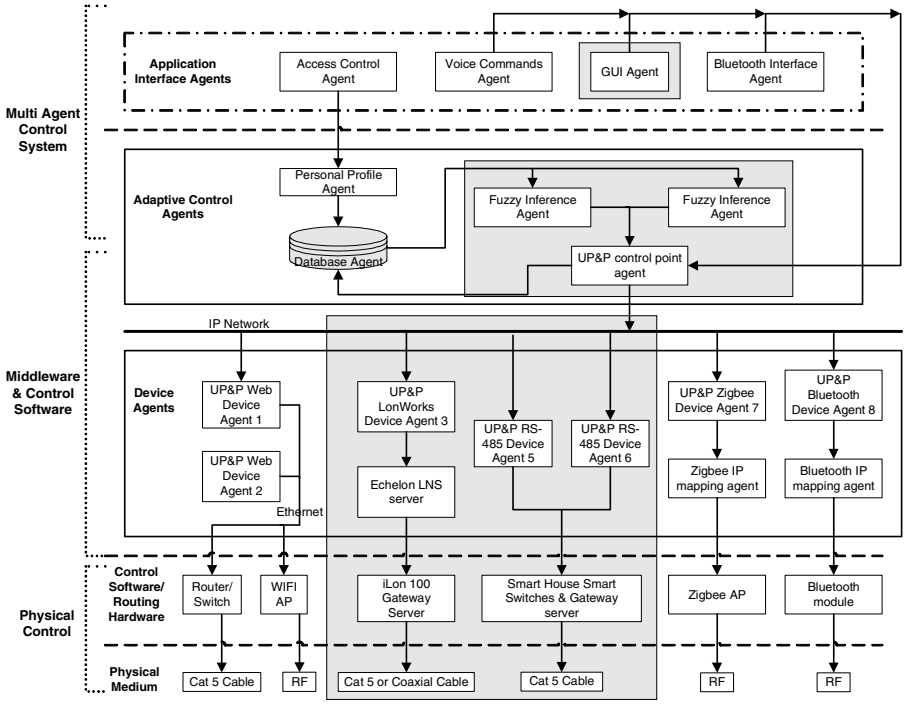


Fig. 1. DEIR control system architecture

3.2 Middleware and Device Control Software

In DEIR, Universal Plug and Play (UPnP) is used as the middleware. In general, UPnP is an IP-based device discovery protocol which enables automatic device discovery, configuration and management. The implementation of UPnP protocol can be considered in two components: the control point and software devices. UPnP control point is the server component which keeps all the information of registered software devices and acts as a software interface between high level MA control system and software devices. UPnP software device is the client component which links the control point with the corresponding device control software. There is a one to one relationship between UPnP software devices and physical devices. The UPnP protocol is incorporated in DIER using CyberGarage UPnP Java API.

LonWorks Network Operating System (LNS) is the control software which controls all the embedded sensors via the iLon 100 router. The corresponding UPnP software devices are implemented using the Java API provided by the LNS developer's kit. On the other hand, the UPnP software devices for RS-485 network devices are implemented in a slightly different way. As RS-485 control software is simply a software interface which converts the high level control commands into RS-485 network commands, this conversion can be integrated directly into the UPnP software devices and the control software is neglected in the actual implementation. The difference in LonWorks and RS-485 networks can be seen in Fig. 1.

3.3 Multi-Agent Control System

According to the functional requirements of an IE defined by Information Society Technologies Advisory Group (ISTAG), a true ambient intelligence should be able to model the environment and human behaviour, to interact with the user, and to provide secure and quality services [1], [2]. While meeting these functional requirements, one must not forget the real ambient intelligence is achieved through ubiquitous computing paradigm. This means, the environment consists of a large number of embedded devices which are smaller in size and have limited computational power. Therefore, the control system must be a distributed system and MAS offers a good solution to this application.

There had been a number of reliable, widely used and tested agent platforms in the past few years. Therefore, rather than developing one of our own MAS for DEIR, an existing MAS, JADE (Java Agent DEvelopment Framework), is used [14]. Agents of JADE are developed using the provided Java API. This allows UPnP components to be incorporated in JADE platform as low level device agents (refer to Fig. 1), which makes the system architecture much more coherent. Different to the device discovery ability that comes with the UPnP protocol, JADE has its own device discovery routine which enables direct communication between any pair of agents in the runtime environment. This extends the capability of UPnP which only allows communications between software devices and control point. With the ability of direct communication, agents can exchange information and achieve cooperative learning and controlling.

4 Multi-Agent Fuzzy Inference System

Multi-Agent Fuzzy Inference System (MAFIS) is implemented in DEIR. The learning technique used in MAFIS is adopted from the Adaptive Online Fuzzy Inference System (AOFIS) developed in the University of Essex [8]. AOFIS uses an unsupervised, data-driven, one pass fuzzy inference learning. The fuzzy rule bases are learned based on captured user's activities and corresponding environmental states. Despite the learning process of MAFIS and AOFIS is the same, the life-long online adaptation ability is yet implemented in MAFIS. The following subsections provide more detailed descriptions for each step of the learning process in sequence.

4.1 Double Clustering Algorithm

Double clustering algorithm [15] is a combination of Fuzzy C-Mean (FCM) algorithm and agglomerative hierarchical clustering algorithm. It takes recorded numerical data (i.e. the environment states and user actions) and converts the data into fuzzy granules which can be represented by fuzzy sets. FCM algorithm [16] is a multi-dimensional clustering process which generates a predefined number of P clusters according to the geographical closeness between the data instances through an iterative process. With r input/output features, each of the P cluster centre is a r-dimensional vector $\bar{c}_i = (c_{i1}, c_{i2}, \cdots c_{ir})$, consists of the one dimensional centres of each feature.

Different to FCM algorithm, agglomerative hierarchical clustering [17] is a one dimensional clustering process. Based on the results of FCM, there are P one dimensional centres, usually called prototypes, for each input/output feature. The

prototypes of each feature are grouped into a sorted sequence from minimum to maximum. The number of prototypes in each sequence should equal to the number of fuzzy sets to be derived in the next step. In order to reduce the number of prototypes, two consecutive prototypes which are closest in value are replaced by their mean value until the required number of prototypes is reached.

4.2 Generation of Membership Function

In order to generate interpretable fuzzy sets, the prototypes of each feature need to be quantified into membership functions. Gaussian membership function is used to quantify the prototypes and to interpret the fuzzy sets of each input and output features. The Gaussian membership function is generated based on the centre and spread, which can be worked out using the prototypes formed in the previous step [8]. Notice that, however, the membership values of the boundary fuzzy sets are fixed to 1 so the membership function can be extended indefinitely beyond their centres.

4.3 Fuzzy Rule Extraction

In order to extract the fuzzy rule base from the previously defined membership functions and captured data instances, the enhanced version of Mendel Wang (MW) method [18], [19] is used. This approach generates fuzzy rules that map multiple input features to multiple output features. Assuming there are *N* data instances for *n* input features and *k* output features and *K* derived fuzzy sets for each feature. The first step is to run through all the data instances, *t*, in the dataset and compute the Gaussian membership values $\mu_{A_s^q}\left(x_s^{(t)}\right)$ for each membership function q = 1, 2, ….., *K*, and for each input feature x_s, s=1, 2, …, n. Then find a particular membership function $q^* \in \{1, \cdots, K\}$ which outputs the maximum membership values among all the other membership functions for each data instance as in Eq. (1):

$$\mu_{A_s^{q^*}}\left(x_s^{(t)}\right) \geq \mu_{A_s^q}\left(x_s^{(t)}\right). \tag{1}$$

The fuzzy set $A_s^{q^*}$ that outputs maximum membership value is assigned to the particular input x_s. In the end, there should be *N* rules generated, one for each data instance. However, a lot of rules generated will have the same antecedent parts but possibly different consequent parts. This forms a group of conflict rules. To resolve these conflicts, the weight $w(t)$ of a rule, which measures the degree of membership of the input instance to the fuzzy region covered by the rule, is calculated as:

$$w^{(t)} = \prod_{s=1}^{n} \mu_{A_s^q}(x_s(t)) \ . \tag{2}$$

Assuming there is a group of *M* conflict rules, which the weight average of all the rules in the conflict group can be calculated using the weight value calculated in Eq. (2), the mathematical expression of the weight average is shown as follows:

$$av = \frac{\sum_{u=1}^{M} y^{(t_u)} w^{(t_u)}}{\sum_{u=1}^{M} w^{(t_u)}} \quad . \tag{3}$$

where u is the index of the conflict rules, M is the number of the conflict rules and t_u is the index of data instance corresponding to the conflict rule u.

Based on the weight average, the consequent parts of the rule can be evaluated to resolve the conflicts and generate a final set of fuzzy rules. Among the possible K output fuzzy sets $B^1, \cdots, B^K$, find a B^* such that,

$$\mu_{B^*}(av) \geq \mu_{B^\kappa}(av), \quad * \in K \quad . \tag{4}$$

The process of calculating weight average and finding the output fuzzy set that generates the maximum membership value of the weight average is carried out for each output feature to cope with rules of multiple output features. The final fuzzy rule derived would be in form of Eq. (5).

$$\begin{aligned} &\text{If } x_1 \text{ is } A_1^{(l)} \text{ and } \ldots\ldots \text{ and } x_n \text{ is } A_n^{(l)} \\ &\text{Then } y_1 \text{ is } B_1^{(l)} \text{ and } \ldots. \text{ and } y_k \text{ is } B_k^{(l)} \quad . \end{aligned} \tag{5}$$

where n is the index of input features, k is the index of output features and l is the index of the fuzzy rule.

Once the fuzzy inference agents have generated membership functions and fuzzy rule bases, Fuzzy Logic Controller (FLC) agents are ready to control the environment on user's behalf using the learned fuzzy rules. In MAFIS, singleton fuzzification, max-product composition, product implication and height defuzzification are deployed in the FLCs [8].

5 Results and Discussions

To examine the control accuracy of MAFIS, a comparative analysis with its centralised version, AOFIS, has been performed. Refer to Fig. 2, the Scaled Root Mean Square Error (SRMSE) is used to measure the control accuracy. The traditional RMSE is scaled to take into consideration the different output ranges. For example, dimmable lights can output values range from 0 to 100 according to its intensity whereas non-dimmable lights output 0 to 1 according to switch status. In order to conduct a fair comparison, the same dataset is used to evaluate the performance of MAFIS and AOFIS. The dataset used for the analysis contains 7 input features, namely internal light sensor, external light sensor, internal temperature sensor, external temperature sensor, chair pressure sensor, bed pressure sensor and time; and 10 output features including 4 dimmable lights, blinds, desk light, bed light, heater and two PC applications: MS Word and MS Media Player. This particular dataset

contains 408 data instances collected over 3 consecutive days monitoring real user activities. The data instances are split into 272 data instances in the training set and 136 data instances in the testing set. Different to AOFIS which models the relationship between all the inputs and outputs, MAFIS associates relevant inputs and outputs into separate groups and models each group separately. Refer to Fig. 3, 4 device groups are defined. Each device agent contributes its own data to the others in the group to achieve cooperative learning. These four groups are modelled by four independent fuzzy inference agents at the same time and each device group can be modelled with optimised number of fuzzy sets. As shown in Fig. 2, MAFIS achieves 5% reduction in overall control errors comparing to AOFIS.

Scaled Root Mean Squared Error (SRMSE)					
No. of Fuzzy Sets	MAFIS				AOFIS
	Group 1 SRMSE	Group 2 SRMSE	Group 3 SRMSE	Group 4 SRMSE	SRMSE
2	0.6726	0.7963	0.6608	0.5415	0.2148
3	0.2102	0.2406	0.3157	0.1390	0.1476
4	0.1798	0.2057	0.3127	0.1094	0.1461
5	0.1389	0.1775	0.2687	0.0819	0.1364
6	0.1204	0.1705	0.2199	0.0853	0.1352
7	0.0979	0.1739	0.2220	0.1047	0.1261
8	0.0931	0.1911	0.1536	0.0735	0.1326
9	0.0893	0.1496	0.1249	0.0665	0.1472
10	0.0974	0.1354	0.1004	0.0652	0.1537
11	0.0835	0.1290	0.0972	0.0697	0.1696
12	0.0865	0.1335	0.1330	0.0735	0.1999
13	0.0716	0.1126	0.1566	0.0912	0.2246
14	0.0731	0.1140	0.1404	0.0866	0.2337
15	0.0742	0.0976	0.0735	0.0735	0.246
16	0.0807	0.1080	0.0891	0.0547	0.2459
17	0.0792	0.1154	0.0976	0.0773	0.2732
18	0.0780	0.1258	0.0857	0.0676	0.2747
19	0.0900	0.1080	0.0819	0.0971	0.2771
20	0.0783	0.1349	0.0949	0.0488	0.2839
Optimised SRMSE for MAFIS			0.0729		

Fig. 2. Comparison of 2 offline control systems

Group 1 (9 features)		Group 2 (9 features)	
Input set	Output set	Input set	Output set
Int. Light Sensor Ext. Light Sensor Chair Pressure Bed Pressure Time	Dimmable Light 1 Dimmable Light 2 Dimmable Light 3 Dimmable Light 4	Int. Light Sensor Ext. Light Sensor Int. Temp. Sensor Ext. Temp. Sensor Chair Pressure Bed Pressure Time	MS Word MS Media
Group 3 (6 features)		**Group 4 (7 features)**	
Input set	Output set	Input set	Output set
Int. Light Sensor Ext. Light Sensor Int. Temp. Sensor Ext. Temp. Sensor	Blind Heater	Int. Light Sensor Ext. Light Sensor Chair Pressure Bed Pressure Time	Bed Light Desk Light

Fig. 3. Relevant input/output devices grouping

6 Future Works

With the proposed control system architecture, future work can be carried out in both control and object level. In control level, online adaptation characteristic can be integrated into MAFIS to achieve a life long learning control system which provides more satisfactory services by adapting itself according to the changes of user behaviour. Also, different machine learning techniques can be mixed in the MA control system to handle different groups of devices to achieve a better performance.

In terms of the object level, rather than having predefined groups of relevant input and output devices, automatic grouping of relevant devices should be implemented. As the system complexity grows with increasing number of embedded devices, it is not always possible for human to predefine optimised device groups. In addition, UPnP software devices can be integrated into the smart switches to achieve real plug and play of those known devices.

7 Conclusions

In this paper, a novel Multi-agent Fuzzy Inference System (MAFIS) and its physical testbed, DEIR, are presented. DEIR consists of a fair amount of embedded devices. Embedded devices are interconnected and have limited computational powers which make DEIR a true ubiquitous computing testbed. MAFIS takes the advantage of ubiquitous computing to achieve cooperative learning and ubiquitous intelligence. Multiple groups of relevant input and output devices are monitored by separate fuzzy inference agents in parallel to model the user activities and to perform control actions on user's behalf. MAFIS and DEIR are well integrated to meet the requirements of ambient intelligence such as the ability to understand environmental context, to model human behaviour and to make sensible controls on human's behalf. The comparative analysis shows that our system has achieved a notable improvement in control accuracy compare to the centralised control system, AOFIS. In our future work, life-long cooperative learning with mixed machine learning techniques and automatic device configuration abilities are targeted.

Acknowledgement. This research is supported by UARC research grant 3604552 and top achiever doctoral scholarship. The authors would like to thank Faiyaz Doctor and Prof. Victor Callaghan for their kind contribution of providing the dataset for comparative analysis and various helps regarding to the use of their AOFIS learning technique.

References

1. Ducatel, K., Bogdanowicz, M., Scapolo, F., Burgelman, J.-C.: Scenarios for Ambient Intelligence in 2010. Information Soc. Technol., Advisory Group (ISTAG), Inst. Prospective Technol. Studies (IPTS), Seville (2001)
2. Riva, G., Loreti, P., Lunghi, M., Davide, F.: Presence 2010: The Emergence of Ambient Intelligence. Amsterdam, The Netherlands:Ios Press (2003)
3. Brooks, R. A.: The Intelligent Room Project. In: Second International Conference on Cognitive Technology, (1997) 271-278
4. Philips, B. A.: Metaglue: A Programming Language for Multi-Agent Systems. M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA (1999)
5. Gajos, K.: A Knowledge-Based Resource Management System For The Intelligent Room. M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA (2000)
6. Kulkarni, A. A.: A Reactive Behavioral System for the Intelligent Room. M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA (2002)
7. Hagras, H., Colley, M., Callaghan, V., Clarke, G., Duman, H., Holmes, A.: A Fuzzy Incremental Synchronous Learning Technique for Embedded-Agents Learning and Control in Intelligent Inhabited Environments. In: Proc. of the 2002 IEEE Int. Conf. on Fuzzy Syst., (2002) 139-144
8. Doctor, F., Hagras, H., Callaghan, V.: A Fuzzy Embedded Agent-Based Approach for Realizing Ambient Intelligence in Intelligent Inhabited Environment. In: IEEE Trans. Sys. Man Cybern., vol. 35, no. 1, (2005) 55 – 65

9. Rutishauser, U., Schaefer, A.: Adaptive Building Intelligence: A multi-Agent approach. Diploma thesis, University of Applied Science Rapperswil, Switzerland and Institute of Neuroinformatics, Swiss Federal Institute of Technology and University of Zurich , Switzerland (2002)
10. Brumitt, B., Cadiz, J. J.: Let There Be Light! Comparing Interfaces for Homes of the Future. Microsoft Research, Redmond, WA 98052, MSR-TR-2000-92 (2000)
11. Yoshihama, S., Chou, P., Wong, D.: Managing Behavior of Intelligent Environments. In: Proc. of the First IEEE Int. Conf. on Pervasive Comp. and Communications, (2003) 330-337
12. Tsai, C. F., Wu, H. C.: MASSIHN: A Multi-Agent Architecture for Intelligent Home Network Service. In: IEEE Trans. on Consumer Electronics, vol. 46, (2002) 505-514
13. Echelon Corporation, "LonWorks Overview, " February 2006, http://www.echelon.com/solutions/overview/default.htm
14. Wang, K. I., Abdulla, W. H., Salcic, Z.: A Multi-Agent System for Intelligent Environments using JADE. In: IEE Int. Workshop on Intell. Environ., (2005) 86-91.
15. Castellano, G., Fanelli, A. M., Mencar, C.: Generation of interpretable fuzzy granules by a double clustering technique. In: Arch. Contr. Sci., vol. 12, no. 4, (2002) 397-410
16. Bezdek, J.: Pattern Recognition with Fuzzy Objective Function Algorithms. New York: Plenum (1981)
17. Jalin, A. K., Dubes, R. C.: Algorithms for Clustering Data, Englewood Cliffs, NJ: Prentice Hall (1998)
18. Wang, L. X.: The MW method completed: A flexible system approach to data minig. In: IEEE Trans. Fuzzy Syst., vol. 11, no. 6, (2003) 678-782
19. Wang, L. X., Mendel, J. M.: Generating fuzzy rule by learning from examples. In: IEEE Trans. Syst. Man Cybern., vol. 22, no. 6, (1992) 1414-1427

Intelligent Pervasive Middleware Based on Biometrics

Jonghwa Choi, Dongkyoo Shin, and Dongil Shin*

Department of Computer Science and Engineering, Sejong University,
98 Kunja-Dong Kwangin-Gu, Seoul, Korea
jhchoi@gce.sejong.ac.kr, shindk@sejong.ac.kr,
dshin@sejong.ac.kr

Abstract. This paper presents IPD (intelligent pervasive middleware) that provides automatic home services (consumer electronics: TV, DVD, audio, light, and air-conditioner) for human through analysis of the biometrics and environment contexts. The IPD receives the biometrics context (pulse, facial expression and body temperature, human location in smart home and human motion) from sensor devices. We handled the context's pattern analysis in two steps. The first step selects consumer electronics (TV, DVD, audio, air-conditioner, light, project) from IPD's rules. In the second step, IPD predicts detailed home service (for example, a detailed home service of the TV includes news, sports, and drama), using the supervised algorithm-based pattern analyzer. We used the SVM (support vector machine) for detailed service pattern analysis. We experimented on the intelligent pervasive middleware in two directions, and it was shown to have an effective performance in practical application. We are currently studying the association technique of home service (by using data mining) that can happen when IPD predicts home service by the home service predictor.

Keywords: Biometrics, Smart Home, Intelligent Home, Pattern Recognition.

1 Introduction

Mark Weiser described a vision for 21st century computing that countered the ubiquity of personal computers. "The most profound technologies are those that disappear," he wrote. "They weave themselves into the fabric of everyday life until they are indistinguishable from it" [1]. This is pervasive computing. Pervasive computing refers to visionary new ways of applying information and communication technologies (ICT) to our daily lives [2]. In this paper, we present the system architecture for pervasive computing in smart home. A smart home for pervasive computing is a space in which all consumer electronics and electronic devices are connected with the network. That is a ubiquitous network. Ubiquitous computing aims to "enhance computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user [3]. The smart home that provides an automatic home service for humans is a space that applies intelligence middleware in a ubiquitous computing environment [4]. In this paper, we

* Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 157 – 165, 2006.

present the intelligent pervasive middleware that provides an automatic home service for humans. Studies of smart homes have proceeded from a lot of directions. The MS Easy Living offered a home service according to the user's position [5]. The MavHome presented an architecture-of-pattern recognition method for the smart home, and applied a data mining technique for algorithm implementation [6]. We present intelligent pervasive middleware that predicts a home service for human by analysis of the biometrics contexts. Section 2 explains previous researches about intelligence home (or smart home). In section 3, we present the architecture of intelligent pervasive middleware for the smart home. Section 4 describes an algorithm that predicts the user's consumer electronics service pattern. In section 5, we present implementation and experimental results. We conclude with section 6.

2 Related Studies

The smart home is our future home. It provides various home services according to the human's situation. Representative studies of smart homes are Easy Living of Microsoft [7] and Aware Home of AHRI (Aware Home Research Initiative) [8]. The Easy Living system is consisting of four parts (*person tracking*: user authentication and user's location tracking, *world model*: agent lookup, *room control*: room control UI, rules engine, *authentication*: PC logon, fingerprint logon). However, the Easy Living has a set focus for user position and user certification, and it does not have the function of intelligence middleware with user and environment context analyses. The Aware Home brings man who lives alone into the focus. For example, if an old man doesn't move, the aware home provides an alarm service. In the neural network house project, studies that provide an air-conditioning service and light control through neural networks, were presented [9]. The MavHome project presented a synthetic study for the intelligent home [10]. It includes system architecture and a prediction algorithm of human behavior, and a resources optimization method in user position recognition [11, 12]. The health smart home was presented as new trend of the smart home [13]. This shows that the smart home's role is diversified. In this paper, we discuss the use of a support vector machine for learning and prediction of a supervised algorithm-based pattern analyzer in the intelligent pervasive middleware [14].

3 Architecture and Flowchart of Intelligent Pervasive Middleware

Figure 1 shows the architecture of intelligent pervasive middleware and the structure of our smart home. We present research that predicts home services automatically through pervasive computing. We applied OSGi(Open Gateway Initiative) as the home network framework. The OSGi takes charge of a role that integrates network protocols of the multiplex used at the home. Also, all processes that operate in the intelligent pervasive middleware act as OSGi bundles, and the OSGi controls the life-cycle of all bundles [15]. The intelligent pervasive middleware includes five main bundles (the sensor recognition utility, the context manager, the network data manager, the home service predictor, and the home service controller), and the integration bundle controller helps with their interoperation.

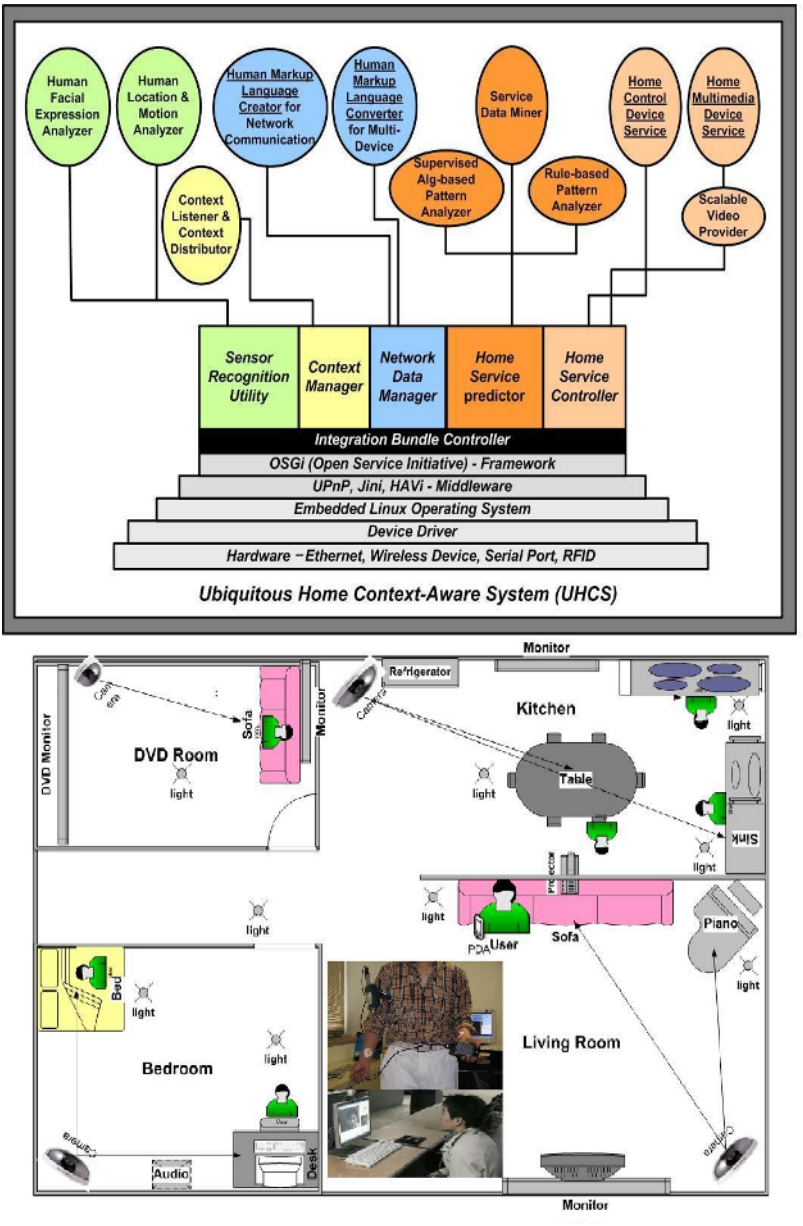


Fig. 1. Intelligent Pervasive Middleware Architecture and Smart home Structure

The sensor recognition utility manages additional operation for context abstraction (facial expression, user location and motion). If new sensor device, which is necessary for additional operation, is installed on the intelligent pervasive middleware, it will be included in the sensor recognition utility. The context manager receives human and environment contexts from the sensor devices, and normalizes all contexts. The home

service predictor predicts home service for the human through the SAPA (supervised algorithm-based pattern analyzer) and the RPA (rule-based pattern analyzer). The service data miner presented in figure 1 predicts possible home service when home service is offered by SAPA and RPA. The service data miner uses data mining technique, and we are studying it now. The home service controller is delivered home service command delivered from SAPA and RPA, and is classified into two groups (control-service and multimedia-service). Figure 1 shows the structure of the smart home in this study. Our smart home consisted of living room, bedroom, kitchen, and DVD room. In the living room and the DVD room, when there is a human in the sofa location, the intelligent pervasive middleware automatically provides home service (<TV: drama-sports, news>, <DVD: action, romantic, horror>, <light: off, on [brightness low, brightness high]>) through context analysis. In the bedroom, when there is a human in the bed location, the intelligent pervasive middleware provides the human with a home service (<audio: dance, rock, and classics>) through pattern analysis of the contexts. If the user is watching TV and then moves to the kitchen, intelligent pervasive middleware transmits the screen picture from the TV monitor (in living room) to a monitor in the kitchen.

4 Process Bundle in the Intelligent Pervasive Middleware

The intelligent pervasive middleware acquires seven contexts (pulse, body temperature, facial expression, eye point, human location & motion, and room temperature) from five sensor devices and creates automatically one context (time).The intelligent pervasive middleware must acquire the context from the human and the environment to predict a home service for the human. The context manager acquires seven contexts from sensor devices. All contexts are delivered to the home service predictor by the context manager. The context listener takes charge of network management to acquire all contexts from sensor devices. The context listener requests contexts from sensor devices every three seconds. All contexts that are acquired by the context listener confirm the scope of all contexts from the context property table that is defined in the database, and errors are then checked. When the intelligent pervasive middleware initializes, the context manager loads the context property table from the database. Figure 2 shows the context property table that is defined in the intelligent pervasive middleware. In figure 2, the scope expresses limitation value for all contexts. The facial expression is classified by seven expressions and the normalization value is set between 0.1 and 0.7. Our smart home was divided into 9 sectors for human position tracking (normalization value: 0.1-0.9). The context that is normalized is transmitted for each pattern recognition method according to the context property table (pattern recognition method in figure 2). The intelligent pervasive middleware includes three pattern recognition methods (supervised algorithm-based pattern analyzer, rule-based pattern analyzer, and service data miner). In this paper, we provide a home service for a human through analysis of SVM and a rule that is defined in intelligent pervasive middleware.

The home service predictor presented in this paper includes two processes (supervised algorithm-based pattern analyzer and rule-based pattern analyzer). Figure 3 shows the relation between the home service predictor's processes.

Context Property Table

Context	Scope	Pattern Recognition Method	Emergency	Not Null	Normalization
Facial Expression	1 ~ 7	RBM/SDM (011)	no	yes	yes
Room Temperature	-10 ~ 50	PRM/RBM/SDM(111)	no	yes	yes
Body Temperature	0 ~ 50	PRM/RBM/SDM(111)	yes	yes	yes
Pulse	0 ~ 240	PRM/RBM/SDM(111)	yes	yes	yes
Human Location	1 ~ 9	PRM/RBM/SDM(111)	no	yes	yes
Human Motion	1 ~ 9	RBM/SDM (011)	no	yes	yes
Time(month/day/hour)	1 ~ 9	RBM/SDM (011)	no	yes	yes
Eye Focus	Non fix	RBM(010)	no	no	no

1. Pattern Recognition Manager : PRM
2. Rule-based Manager : RBM
3. Service Data Miner : SDM

Fig. 2. Context Property Table

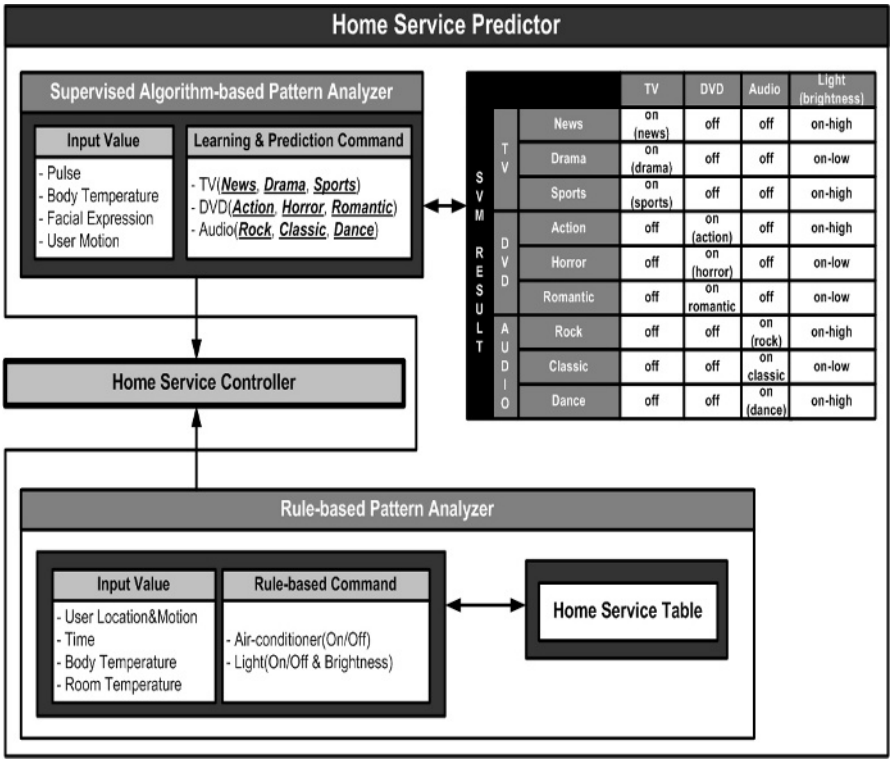


Fig. 3. Home Service Predictor

The rule-based pattern analyzer decides all actions from a rule that is defined by the home service table. Figure 3 shows rule that is defined to intelligent pervasive middleware by a tree structure. The rule-based pattern analyzer decides only a kind of home service that is selected by rule. For example, if the user's position is sofa in the DVD room, the rule-based pattern analyzer selects DVD as home service to execute. But the rule-based pattern analyzer doesn't decide about the detail of the DVD service (action DVD, horror DVD, romantic DVD). The supervised algorithm-based pattern analyzer predicts detailed home services of consumer electronics (TV, DVD, audio). In figure 3, the supervised algorithm-based pattern analyzer accepts the human's biometrics context (pulse, body temperature, facial expression, and motion) as input value for the SVM, which can predict one of nine detailed home services (TV<news, sports, drama>, DVD<action, horror, romantic>, audio<dance, rock, classic>).

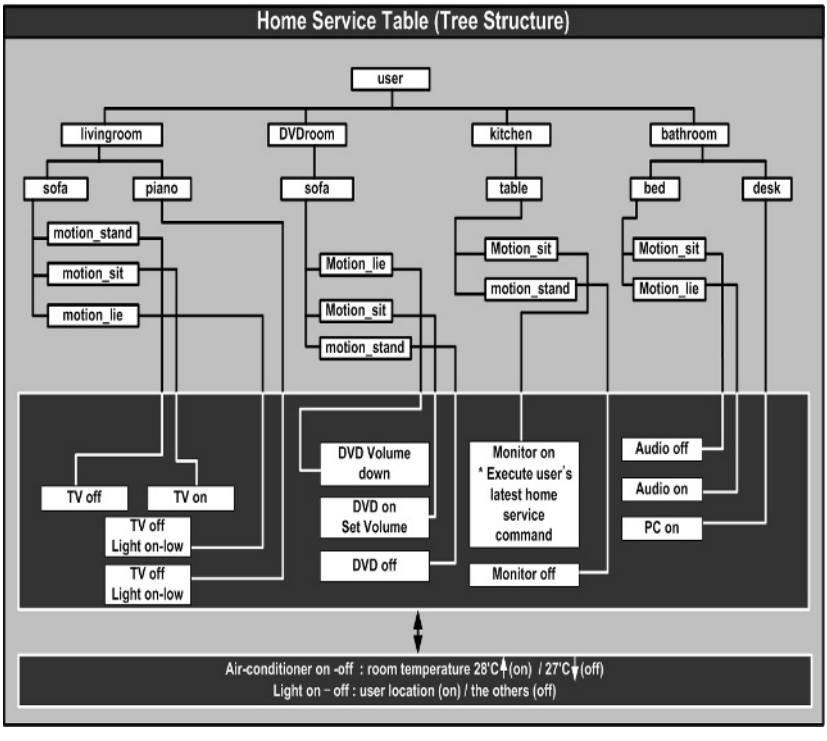


Fig. 4. Home Service Table

We used LSVM (linear support vector machine) to predict one of nine home services. All LSVMs act by hierarchical structure. The supervised algorithm-based pattern analyzer selects one LSVM among nine LSVMs according to the human's location, and provides results of pattern recognition to the home service command creator.

5 Experimental Evaluations

In this paper, we defined context to provide automatic home service in the smart home, presented a middleware model that provides home service to human through the context's pattern recognition method. Our intelligent pervasive middleware used human context and environment context to recognize the home service pattern of a human. We applied human location as key context to prevent incorrect home service by the intelligent pervasive middleware. First, the intelligent pervasive middleware decides what kind of services (TV, audio, light, air-conditioner, DVD) to offer, according to the human's location. Then, it decides a detailed home service (for example, TV includes news, sports, and drama). The intelligent pervasive middleware can decrease wrong predictions of home services through the application of two steps. It can decrease the problem of complex context's recognition by the definition of a key context (human location). Also, the intelligent pervasive middleware provides various home services for human by applying SVM for analysis of the biometrics context. Table 1 shows estimation of performance in the intelligent pervasive middleware. Runtime cost is operating time of each component. Because our pulse sensor device requires a lot of time to extract the exact value, the sensor device takes an average of 1.24 seconds to transmit all detail. The error rate shows the inaccuracy rate for execution of all components.

Table 1. Component Performance in Intelligent pervasive Middleware

	Runtime Cost (second)	Specific Gravity by time	Error rate
Sensor Device	1.24	15.5%	12.5%
Context Manager	3.24	40.6%	42.5%
Home Service Predictor	2.67	33.5%	10.2%
Home Service Execution	0.82	10.2%	4.5%

The sensor recognition utility decides facial expression, motion, and location by a raw image analysis that is acquired from a camera sensor. Because the context manager includes the sensor recognition utility, it requires the most time in the intelligent pervasive middleware. The home service for a human is predicted by two processes. First, the intelligent pervasive middleware decides what kinds of services (TV, audio, light, air-conditioner, DVD) to offer according to the human's location. Then, it decides the detailed home service (for example, TV includes news, sports, and drama). Table 2 shows the experimental result that is predicted by SVM.

We executed home service commands 1000 times for the experiment. In table 2, the average accuracy rate for detailed home service is shown as 89.8%. The precision shown in table 2 is the accuracy rate for classification in SVM. In future studies we will apply ECG (electrocardiogram) as biometrics context to increase the algorithm accuracy.

Table 2. Experiment result of Supervised Algorithm-based Pattern Analyzer

		Number of Support Vector	Number of kernel evaluations	Norm of longest vector	Precision on test set
TV	News	58	13435	2.23552	89 %
	Drama	42	14232	2.15432	92 %
	Sports	69	12435	2.29594	88 %
DVD	Action	72	13867	2.28184	94 %
	Romantic	46	12532	2.11132	91 %
	Horror	52	12814	2.31532	88 %
Audio	Dance	69	13265	2.63213	90 %
	Rock	52	12545	2.19334	89 %
	Classic	47	11942	2.10543	87%

6 Conclusions

This paper presented IPD (intelligent pervasive middleware) that provides automatic home services (consumer electronics: TV, DVD, audio, light, and air-conditioner) for human through analysis of the biometrics and environment contexts. The IPD receives the biometrics context (pulse, facial expression and body temperature, human location in smart home and human motion) from sensor devices. We handled the context's pattern analysis in two steps. The first step selects consumer electronics (TV, DVD, audio, air-conditioner, light, project) from IPD's rules. In the second step, IPD predicts detailed home service (for example, a detailed home service of the TV includes news, sports, and drama), using the supervised algorithm-based pattern analyzer. We used the SVM (support vector machine) for detailed service pattern analysis. We experimented on the intelligent pervasive middleware in two directions, and it was shown to have an effective performance in practical application. We are currently studying the association technique of home service (by using data mining) that can happen when IPD predicts home service by the home service predictor.

References

1. M. Weiser.: The Computer for the 21st Century. Scientific Am. (1991) 94-104; reprinted in IEEE Pervasive Computing. (2002) 19-25.
2. Koehler, A.. Som, C.: Effects of pervasive computing on sustainable. Technology and Society Magazine, IEEE. vol 24. Issue 1. (2005) 15 - 23
3. M. Weiser.: Some Computer Science Issues in Ubiquitous Computing. Commun. ACM. vol 36. no 7. (1993) 75-84
4. Das, S.K. Cook, D.J.: Guest Editorial - Smart Homes. Wireless Communications, IEEE. vol 9. Issue 6. (2002) 62 – 62
5. EasyLiving.: http://research.microsoft.com/easyliving/
6. MavHome.: http://www.darmstadt.gmd.de/ambiente/i-land.html
7. B, Brumitt. J, Krumm and S, Shafer.: Ubiquitous computing & the role of geometry. IEEE Personal Communications. (2000) 41-43

8. I,A, Essa.: Ubiquitous sensing for smart and aware environments: technologies towards the building of an aware home. In Position Paper for the DARPA/NSF/NIST workshop on Smart Environment. (1999)
9. M,C, Mozer.: The neural network house: An environment that adapts to its inhabitants. In Proceedings of International Symposium on Handheld and Ubiquitous Computing. (2000)
10. The MavHome Smart Home Project.: http://mavhome.uta.edu/information.html
11. D,J, Cook. M, Youngblood. E, Heierman. K, Gopalratnam. S, Rao. A, Litvin. and F, Khawaja.: MavHome: An Agent Based Smart Home. Proceedings of the IEEE International Conference on Pervasive Computing and Communications. (2003) 521-524
12. A. Roy. S,K, Das Bhaumik. A. Bhattacharya. K, Basu. D,J, Cook. S,K, Das.: Location aware resource management in smart homes. Proceedings of the IEEE International Conference on Pervasive Computing and Communications. (2003) 481-488
13. N, Noury. G, Virone. P, Barralon. J, Ye. V, Rialle. J, Demongeot.: New trends in health smart homes. Proceedings of the Enterprise Networking and Computing in Healthcare Industry. (2003) 118-227
14. Vapnik V. N.: The nature of statistical learning theory. New York, Springer-Verlag, (1995)
15. Choonhwa Lee. Nordstedt, D. Helal, S.: Enabling smart spaces with OSGi. Pervasive Computing. IEEE. vol 2. Issue 3. (2003) 89 - 94

An Arrival Time Anticipation Approach for Real-Time Tracking of Moving Object in Mobile Networks

JungHee Jo, JuWan Kim, KyungWook Min, KwangSoo Kim, and YongJoon Lee

Telematics-USN Research Division,
Electronics and Telecommunications Research Institute,
161 Gajeong-dong, Yuseong-gu, Daejeon, Korea
{dreamer, juwan, kwmin92, enoch, yjl}@etri.re.kr

Abstract. In real-time moving object tracking, each time an object moves, a new position should be sent from the moving object to the server. In this case, numerous moving objects cause a very large volume of location updating. It could bring communication, computation and update overhead between moving object and server. This paper proposes a new location tracking method that is applicable to continuous updating of current position of moving object. The techniques are based on movement anticipation with pre-evaluated arrival time information; the moving object compares actual position with anticipated position at predefined time interval and checks deviation between two positions. Only if predefined threshold exceed, the location updating, from moving objects to the server, is occurred. Using predefined route information, this approach is effective with update reduction for real-time tracking of moving object.

1 Introduction

A location-based service (LBS) is any product, service, or application that uses knowledge of a subscriber's location to offer value to the subscriber or to a third party. It use the connectivity provided by a wireless network to generate position information, deliver it to the consumer and mobile-resident application.

One of applications of LBS is moving object tracking to judge a user's current location. Ideally, each time an object moves, a new position should be sent from the moving objects to the server. A large population of moving objects causes a very large volume of updates. Such increasing updates entail following cost [1]: (1) Communication costs: Subjecting a wireless network to a high volume of updates may overload the network and degrade performance; (2) Server-side update costs: Database management systems typically do not support very large volumes of updates well. This is particularly true if spatial indexing of the moving objects is employed; (3) Client-side costs: Communication carries a significant computational overhead on a mobile client. This leads to a shorter operational time for battery powered, handheld devices. Therefore, it is important to reduce update rate with guaranteeing the minimum accuracy of moving object's location.

The contributions of this paper are as follows: (1) we suggest how to reduce the update rate between moving object and a server (2) we propose the method how

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 166 – 175, 2006.

moving object tracking component can be operated as one of software components of LBS platform (3) we introduce the testbed system to simulate our tracking approach.

The paper proceeds as follows: Section 2 explains the well-known location tracking works and limitations of such approaches. Section 3 introduces our tracking approach based on anticipated arrival time information. Section 4 covers the testbed organization for examining our approach. The final section provides concluding remarks and offers suggestions for future research.

2 Background and Previous Works

This section is divided into two subsections. In subsection 2.1 this paper introduces well-known tracking approaches including segment-based tracking. In subsection 2.2 this paper presents the limitation of segment-based tracking.

2.1 Location Tracking Approaches

There are well-known researches of tracking approaches of moving object [1]. One of approaches is location polling [2]. It is classified into server-based location polling and terminal-based location polling according to the position of location-acquisition agent. Server-based polling approach use the MS-assisted positioning technology such as Cell-ID and EOTD, and terminal-based polling approach use the MS-based positioning technology using GPS.

Sever-based location polling approach is a method that polls the location information from location server in mobile network. In that case, the location-acquisition agent is placed in location server and periodically request location of moving object to MPC (Mobile Positioning Center) in mobile network. On the other hand, terminal-based location polling approach is a method that terminal calculate its position using GPS. In this approach, the location-acquisition agent is placed in terminal. Synchronizing with terminal side, the server also could continuously track the terminal's location. This approach has an advantage such as not having to consider the network overload between MPC and server.

In the terminal-based location polling approach, each time an object moves, a new position should be sent from the moving object to the server. In that situation, a large volume of moving objects causes a very large volume of updates. It causes communication and update costs between moving object and server as explained in introduction. To reduce such overheads, several tracking approaches were introduced such as point-based tracking, vector-based tracking, and segment-based tracking [1].

First, point-based tracking assumes that the object to be tracked is located at the position given by the most recent updates. Using this policy, the movement of an object is represented on the server side as a jumping point. An update is issued by a moving object when its distance to the previously reported position deviates from its current GPS position by the specified threshold. Second, vector-based tracking uses the object's position as well as the object's speed and direction of movement. It assumes that the object moves linearly and with the constant speed received from GPS device in the most recent update. Using this policy, the movement of object is represented as a jumping vector on the server side. Lastly, segment-based tracking is similar to the

vector policy. The difference is that it assumes that an object moves along the shape of a known road segment and moves constant speed.

These approaches are constrained by a road network and they are capable of obtaining their positions from the terminal, using GPS receiver. The terminal initially obtains its position data from the GPS receiver and after establishing the connection with server, the server receives terminal's actual location information from the terminal. After obtaining this update, the server determines threshold and sends it to terminal. Then, the terminal calculates its anticipated position and compares it with real GPS position. If the difference between theses two exceeds the given threshold, the client issues an update to the server. Using such methods, it is possible to track a moving object's current positions with as few updates as possible.

According to the Civilis's experiment [1][7], segment-based tracking has the greatest potential compared with other two approaches. The reason is that this tracking technique knows the road on which the object moving, it enables various services that rely on information.

2.2 The Limitation of Segment-Based Tracking

The segment-based tracking, object moves along the shape of a known road, is closely correlated with the numbers of road segments. The reason is that the update is occurred whenever road segment is changed as well as deviation reaches some threshold.

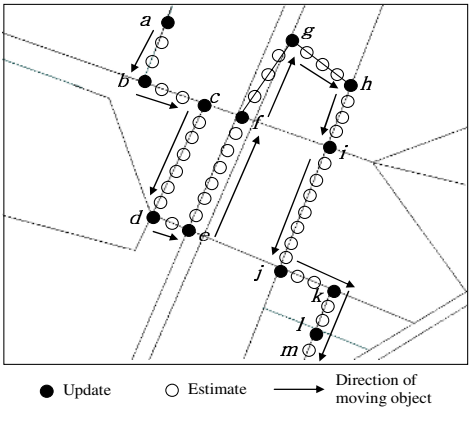


Fig. 1. The description of segment-based location tracking

In Fig. 1, the black circle represents the issuing of update and empty circle represents movement of object and the arrow indicates the direction of moving object. Each segment corresponds to the road segment between two crossroads. In other words, all crossroads are located at the ending points of segments. In the start position *a*, the moving object sends its actual position, which is obtained from GPS receiver, to server. On the server, the moving object's current position is determined using the location data received from the last update from the moving object and a prediction algorithm. A moving object also predicts its position using the same data and algorithm, as the server.

The moving object continuously compares the predicted location to its actual position, and it sends an update to the server on condition that the distance of two locations exceed the threshold. The empty circle implies the predicted position of moving object in server side and the black circle indicate position updating. If the road is composed of lots of road segments, definitely the number of update is also increased.

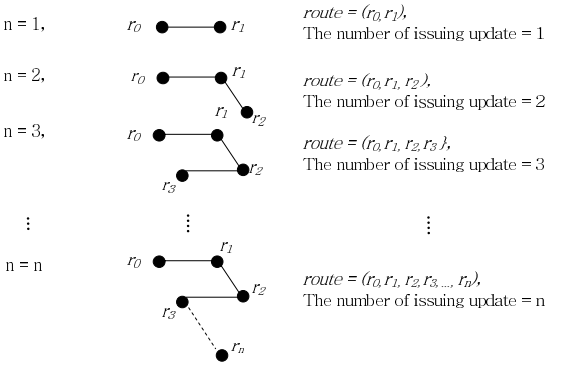


Fig. 2. The number of updates in segment-based location tracking

The route of moving object is defined as a set of polylines, $route = (r_0, r_1, \ldots, r_n)$ where $n \geq 1$. In this set, randomly r_k is a point defined as a pair of coordinates $r_k = (x_k, y_k)$ where $r_0 = (x_0, y_0)$ is origin and $r_n = (x_n, y_n)$ is destination of the route. The Fig. 2 shows that as the number of polylines increased, the number of issuing updates is also linearly increased. Therefore, reducing update rate is critical issue to enhance overall performance of location tracking. This paper suggests the way to eliminate such pitfall of segment-based tracking approach by proposing further efficient tracking approach based on moving object arrival time anticipation.

3 Proposed Tracking Approach of Moving Object

As shown in Algorithm 1 and Algorithm 2, the proposed tracking approach utilizes the intermediate arrival time information to reduce the update rate of moving object.

Algorithm 1. Terminal-Side Operation

```
realCurrentPos = GPSlog()
  if (FirstUpdate) then
    UpdateRequest(realCurrentPos,destination)
  else
    while (realCurrentPos != destination)
      if(currentTime ∈ anticipatedArrivalTime[]) then
        index = ExIndex(currentTime,anticipatedTime[])
        anticipatedCurrentPos = route[index]
        if((|realCurrentPos - anticipatedCurrentPos|)≤
threshold) then
```

```
          ContinueAnticipation()
        else
          UpdateRequest(realCurrentPos,destination)
        endif
      endif
    endwhile
  endif
```

Algorithm 2. Server-Side Operation

```
UpdatePositionDB(realCurrentPos)
route[]=RouteFromRoadDB(realCurrentPos,destination)
anticipatedArrivalTime[] = CalculateArrival-
Time(route[])
  if (FirstUpdate) then
    SendToTerminal(route[], anticipatedArrivalTime[],
threshold)
  else
    SendToTerminal(route[], anticipatedArrivalTime[])
  endif
```

In the terminal-side, terminal initially obtains its location information from GPS receiver. In the beginning of tracking, the terminal sends update-request to server with sending start position and destination of moving object. Having received this information, the server stores it to the database in server. Then, the server divides the entire route into several intermediate positions and calculates the arrival time for each intermediate position. Then the server sends it with predefined threshold to terminal.

While estimating arrival time of moving object at intermediate position, traffic congestion is one of the major problems. It is possible to make use of real-time and historical data in evaluating the time-of-arrival information of moving objects. With such information on traffic flow, travel time estimation can be achieved with precision [3]. In our approach, we assumed that real-time traffic server took such a role.

Having received route, anticipated arrival time per intermediate positions, and threshold, the terminal periodically checks whether current time corresponds to one of the intermediate arrival times. If a set of intermediate arrival times contains current time, the terminal compares its GPS position to the anticipated position. If the difference between these two exceeds the given threshold, the terminal issues an update to the server. Otherwise, the moving objects continuously moves without communication with server until it meet next intermediate arrival time. This procedure continues before the operation is unexpectedly terminated or the terminal arrives at destination.

For example, as shown in Fig.3, the entire route of moving object can be represented as $Route = (r_0,...,r_a,...,r_b,...,r_n)$. The term, r_0, represent origin and r_a and r_b are defined as the intermediate positions in entire routes and r_n is the destination of the route. In between two positions r_0 and r_a, there are other intermediate route, $Route = (r_1,...,r_{a-1})$. Same manner, there might be intermediate routes, $Route = (r_{a+1},...,r_{b-1})$, and $Route = (r_{b+1},...,r_{n-1})$.

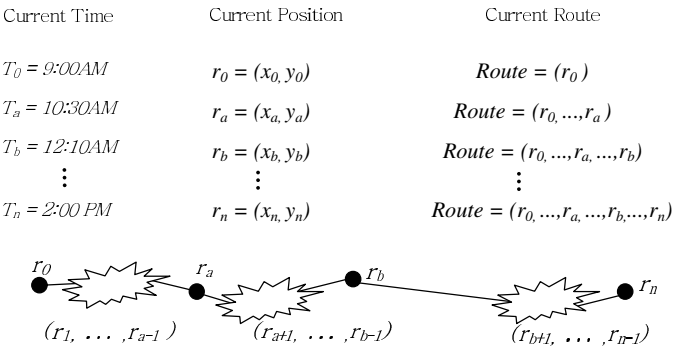


Fig. 3. The representation of entire route of moving object

According to the above scenario, we assume that the real-time traffic server estimated the intermediate arrival time at r_a as t_a, at r_b as t_b considering current and historic traffic condition. The object moves from r_0 until it meets first predefined intermediate time, t_a. If current time is same with t_a, the terminal compares current actual position (x_a,y_a) with anticipated position (x_a',y_a') and check the gap between two positions. Only if the gap exceeds a given threshold, the terminal send current actual position (x_a,y_a) to server and request updated route from (x_a,y_a) to (x_n,y_n).

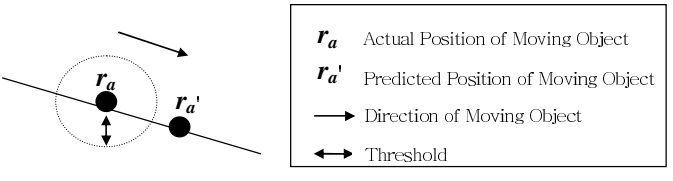


Fig. 4. The comparison of distance between anticipated and actual position of moving object

Using our tracking method, the range of total number of updates, *n(update)*, can be represented as $0 \le n(update) \le n(intermediate\ position)$ excluding initial update where *n (intermediate position)* is the total number of intermediates positions. If we assume th at the meaning of *n(total position)* is the total number of update including occurring c hanging of road segment and deviation from threshold, in case of traditional segment-based approach, the range of total number of updates, $n(update)'$, can be represented a s $0 \le n(update)' \le n(total\ position)$ where $n(intermediate\ position) < n(total\ position)$. Th erefore, we can conclude as $n(update) < n(update)'$.

This approach is more efficient if a large number of short strings are consisted of one straight road. In Fig.5, if an object moves from P_1 to P_5, the total number of update is at least 5, occurred at P_1, P_2, P_3, P_4, and P_5 in segment-based approach. However, using our approach, it could reduce less than 5. For example, if we set intermediate position as P_2 and P_4, the maximum total number of update are 2. The term 'maximum' means that at the P_2 and P_4, the terminal compares actual GPS position

with anticipated position and check the gap between two positions. If the gap doesn't exceeds a given threshold, no update is happened. In that case, the number of total update between P_1 to P_5 is zero.

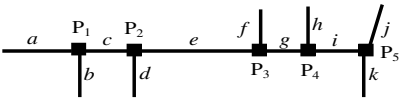


Fig. 5. The road network composed of large number of short strings

This simple example explains that knowing the entire routes and intermediate arri val time of moving object can significantly reduce the number of updates caused by c hanging of road segment in segment-based tracking approach. In between intermediat e times, the accuracy of the object's position on the server can be guaranteed if we ass ume the object moves at constant speed. Thus, the performance of this approach could be improved comparing with segment-based tracking by removing continuous 1) calc ulation of anticipated position, 2) obtaining actual position from GPS receiver, 3) com parison between anticipated position and actual position.

4 Testbed Organization

This section is divided into two subsections. In subsection 4.1 this paper describes the data set and simulation environment. In subsection 4.2 this paper compares our approach with traditional segment-based approach by simulation.

4.1 Data Set and Simulation Environment

We made a synthetic data set on Seoul city in Korea to simulate complicated and similar road network conditions of Seoul. While the network data generator, developed by Prof. Brinkoff [4] has been used to generate trajectories of mobile objects, we have used real road network data including network connectivity. The network data generator provides very realistic trajectories of mobile objects as well, by distinguishing the types of roads. With this data set, we can observe and examine the performance of our approach including segment-based tracking methods.

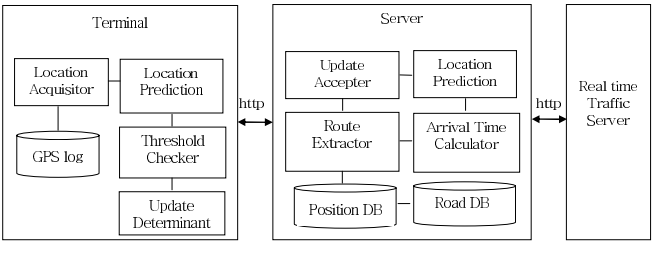


Fig. 6. The organization of testbed system

Instead of porting each tracking method into mobile device, we developed a simulation module for mobile device where we put tracking method. This module contains software components for feeding GPS data, tracking methods, and interface with server as shown in Fig. 6. In order to observe and simulate the performance of tracking methods, we developed a module to analyze and visualize the behavior and performance of tracking method. The module provides several static, visualization tool and accuracy measures for each tracking method.

In order to validate our tracking approach, we ported Tracking Manager component, which performs the real-time location tracking, on the LBS platform that is enables to handle subscriber's location, presence, privacy, and manage location related contents. The Tracking Manager receive update request from terminal through LBS Server Interface Manager in Fig.7. Specifically, LBS Server Interface Manger responsible for parsing, building, and transferring the message from Location Clients. Among the Location Clients, Tracking Assistant is the GPS equipped terminal that is tracked by Tracking Manager.

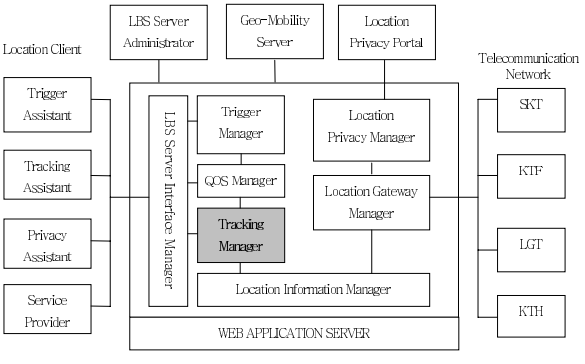
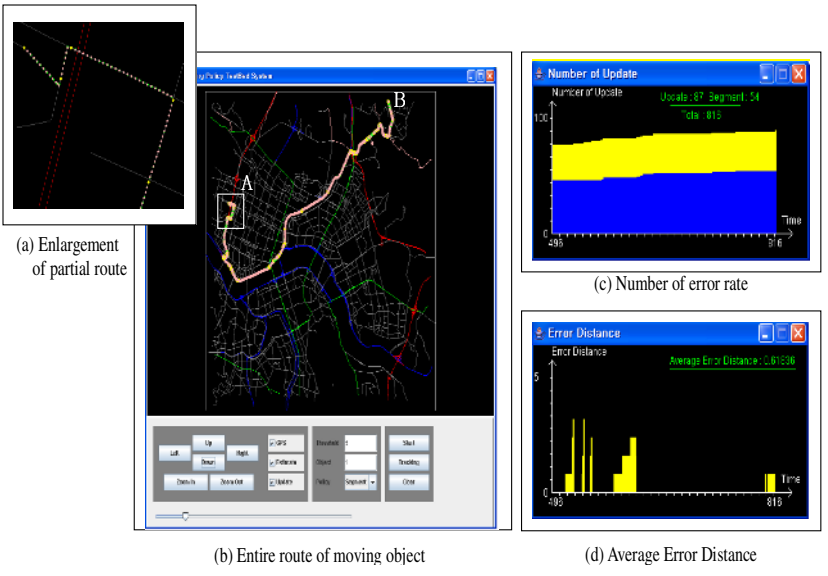


Fig. 7. The system architecture of LBS platform containing Tracking Manager

4.2 Simulations Results

First, we experiment the performance of segment-based tracking approach. As shown in Fig.8 (b), along the road network the object start to move from position A to position B. Fig.8 (a) is the enlargement of partial trajectory of object, that is indicated with square in Fig.8 (b). According to Fig.8 (c) while object moves whole route, the total number of acquisition of GPS position is 816. The total number of segments that is composed of entire route is 54 and total number of update request from moving object to server is 87. This result means the number of unexpected update, that is occurred because difference between actual and anticipated position is deviated from threshold, is 33. Fig.8 (d) shows the average error distance is 61.8m when we set the threshold as 50m for whole travel time, 816 seconds. We repeated this experiment 10 times by choosing another routes and collect statistical data. According to the analysis, the average total number of update in segment-based tracking is decreased 10



(a) Enlargement of partial route

(b) Entire route of moving object

(c) Number of error rate

(d) Average Error Distance

Fig. 8. The simulation results of segment-based tracking approach

times compared with the average total number of update that occurred whenever moving object acquires GPS position.

Based on those simulation results, we could estimate the performance of our approach. In previous experiment, if we assume that there are k intermediate positions such as $k<$*total number of segments in entire route*$=54$, the total number of updating in our approach is same or less than k, as we explained in section 3. It is definitely much smaller value than the number of total update, 87, in previous experiment. Therefore, the arrival time anticipation based tracking is more efficient than well-known segment-based tracking approach from a view of performance by reducing total number of update; it decrease 1) the overhead of continuous anticipation, 2) the obtaining rate of actual position from GPS receiver, and 3) the comparison rate between anticipated position and actual position.

5 Conclusion

In this paper, we proposed an effective real-time moving object tracking approach that utilized the anticipated arrival time information instead of depending on road segment. The principle idea is to reduce update rate comparing with segment-based tracking approach such a way that moving object would have to update as few times as possible while they travel in the road network. We described the organization of testbed system to simulate and validate the performance of tracking approach. In the future, we plan to evaluate our tracking approaches more concretely and examine their accuracy upon using adaptive threshold mechanism.

References

1. Civilis, A.; Jensen, C.S.; Nenortaite, J.; Pakalnis, S.; Efficient tracking of moving objects with precision guarantees, MOBIQUITOUS 2004, pp164-173, 2004
2. Byung-Ik Ahn; Sung-Bong Yang; Heui-Chae Jin; Jin-Yul Lee; Location Polling Algorithm for Alerting Service Based on Location, W2GIS 2005, pp104-114, 2005.
3. Karbassi, A.; Barth, M.; Vehicle route prediction and time of arrival estimation techniques for improved transportation system management, Intelligent Vehicles Symposium, 2003. Proceedings. IEEE, 9-11 June 2003 Page(s):511 – 516
4. http://fh-oow.de/institute/iapg/personen/brinkhoff/generator/, Network-based Generator of Moving Objects
5. Location Based Services 2005-2010: Market realities, recommendations and forecasts, Visiongain
6. Wolfson, O.; Chamberlain, S.; Son Dao; Liqin Jiang; Mendez, G., Cost and imprecision in modeling the position of moving objects, Data Engineering, 1998. Proceedings, 14th International Conference on 23-27 Feb. 1998 Page(s):588 – 596
7. Civilis, A.; Jensen, C.S.; Pakalnis, S.; Techniques for efficient road-network-based tracking of moving objects, Knowledge and Data Engineering, IEEE Transactions on Volume 17, Issue 5, pp698 - 712, May 2005.
8. Kam-Yiu Lam; Ulnsoy, O.; Lee, T.S.H.; Chan, E.; Guohui Li; An efficient method for generating location updates for processing of location-dependent continuous queries, Database Systems for Advanced Applications, 2001. Proceedings. Seventh International Conference on 18-21 April 2001 Page(s):218 – 225
9. Chartier, E.; Hashemi, Z.; Surface surveillance systems using point sensors and segment-based tracking, Digital Avionics Systems, 2001. DASC. The 20th Conference Volume 1, 14-18 Oct. 2001 Page(s):2E1/1 - 2E1/8 vol.1

Behavior Analysis with Combined RFID and Video Information

Hui-Huang Hsu[1], Zixue Cheng[2], Tongjun Huang[2], and Qiu Han[2]

[1] Department of Computer Science and Information Engineering
Tamkang University, Taipei, Taiwan
h_hsu@mail.tku.edu.tw
[2] School of Computer Science and Engineering
University of Aizu, Fukushima, Japan
z-cheng@u-aizu.ac.jp

Abstract. In order for the hidden computer to interact with human beings, a variety of sensors are used to collect data of human behavior. Using video cameras is quite straightforward on this matter. However, it is still a difficult task to do human motion analysis through only video data. In this paper, we discuss the necessity and the way to combine information from video sensors and other types of sensors, especially, the RFID reader. Data acquisition of RFID and video data is discussed first. Combinatorial information fusion techniques for multiple sources of information are then introduced. On the other hand, rule based methods provide an alternative solution for combining the two sources of information. An example in monitoring real-life learning behavior is also given.

1 Introduction

In a ubiquitous environment, the computers are distributed and hidden in the environment to provide needed service to human beings. To automatically provide context-aware service, interactions between the human and the computer is necessary. However, traditional human-machine interfaces like the keyboard and the mouse seem not suitable in this environment. It is better that the computer can understand what the human needs automatically simply through monitoring the behavior of the human. The less the human needs to give/input commands or data the better. To understand a person's behavior, it is straightforward to use a video camera to capture what the person does. It is easy for the human to judge who the person was and what happened in a video sequence. However, it is still a very complicated and difficult task for the computer. Although there is a lot of great effort in computer vision, limitation to a great extent still remains [1]. So other ways of sensing human's behavior are needed.

Biometrics methods have been developed to identify a person through speech, face, or fingerprint recognition. All of them require heavy computation. On the other hand, simpler sensors like RFID (Radio Frequency Identification) sensors, ultrasound sensors and pressure sensors can provide information of the existence of a person and his/her identity. Among them, the RFID is a very easy way to identify the existence of a certain

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 176 – 181, 2006.

person via the embedded object code in the RFID tag. The price of RFID tags has become cheaper that makes the tags more generally available. Furthermore, the RFID tags have long been used in goods and animals. Detection of certain objects existing with a person can also help the computer to understand the behavior the person.

In [2], an RFID-based tracking system was introduced. The system can detect the location of humans or other objects with the RFID tags. And location-based services can be provided automatically. In [3], implantation of RFID tags to humans was discussed. A few people have conducted the experiments. The authors described the usability in the contexts of control, convenience and care. RFID tags with applications to humans themselves are now generally discussed. They are not just been used in goods and animals any more. In [4], the algorithms for cooperation of multiple video sensors in surveillance are presented. To supervise the movement of a person in real time in an area, say a neighborhood, multiple cameras are needed. How to capture the transition of the object from one camera to another camera is the main problem. In this paper, we will focus on how to take advantage of information collected from both live-captured video and RFID sensors. The ways to properly combine or fuse the two sources of information is the major task here.

Data acquisition of human motions in video and RFID sensor data is described in Section 2. Two possible ways to combine information collected from different types of sensors are then discussed. The first one is an information fusion method through ranks and/or scores. The second one is a rule-based method. The methods are presented in Section 3 and Section 4, respectively. In Section 5, a scenario for ubiquitous learning is presented. In the example, the learner can receive advice from the computer about his/her learning behavior in front of his/her desk. This can be done automatically, and the learner can benefit from the advice to know his/her own learning behavior and to make adjustment. In Section 6, a brief conclusion is drawn.

2 Data Acquisition

To understand the behavior of a person in a video sequence, we have to do motion analysis. It includes three phases. First, the human object needs to be segmented from the background. A simple subtraction of the image with the background image and some further image processing are sufficient. A silhouette of the human object can be obtained from the segmentation in each frame of the video sequence. Then, a stick model representing the skeleton and the major joints of the person is applied to each silhouette. Genetic algorithms have been proved to be quite accurate in estimating the stick model of a human silhouette [5]. Finally, the behavior/movement of the person can be analyzed by examining the changes of the stick models in consecutive frames. The whole process is very complicated and time-consuming. It also faces problems regarding the estimation of the stick model, e.g. self-occlusion.

In this research, information that can be acquired by simpler methods is targeted. First, we can simply detect the existence of a person by subtracting the image of the current frame from the pre-captured background image. If a person does exist in the video sequence, his/her silhouette is segmented from the background. We then can

classify the silhouette into a few classes by a trained classifier. Each class represents a certain body posture. The system can detect the posture change that represents a certain behavior of the human. The posture change can be seen as an event and the time stamp of the change time is recorded in the system database.

To identify who the human is needs biometric methods in speech recognition, image processing, and computer vision. All of them are not an easy task, either. If an RFID tag can be attached or implanted to the human, the identification of the human can be easily achieved. The RFID tags can also be applied to other objects in real life. The tag can be used to detect and identify the existence of a certain object with proper deployment of the RFID readers. So for a certain reader, the data to be recorded would be the time stamp of the show-up time of an object and the time stamp of the disappearing time of the object. With the information about the objects appearing with the human, the behavior of the human can be analyzed.

3 Fusion of RFID and Video Information

A variety of sensors are used in the real environment to collect data/information. With the collected data, systems can be developed to do automatic/intelligent analysis of the data and provide appropriate response or service. However, how to combine/fuse the collected data from different sensors has been an essential research topic. In [6], a method called *combinatorial fusion analysis* is introduced. The method characterizes each decision making system by its scoring function, rank function, and/or rank/score function. With multiple sources of information used in different systems, a proper combination can make better decision and produce higher classification accuracy.

Here, we can represent the data collected from RFID readers and cameras in the three-dimensional space, respectively, $D_l(O_i, F_j, T_k)$, where O_i are objects, F_j are features or events, and T_k are time stamps. f_l: $D \rightarrow S$ is the score function that maps the data into scores for different classes. f_l can be estimated by machine learning or statistical methods. For example, neural networks or support vector machines can be applied here. With the scores of the classes, ranks can be decided. And a rank/score graph can be obtained for visualization [6].

To combine the scores resulted from the two information sources, linear combination can be utilized. It can be a weighted sum of the two scores or simply the average. However, it should be noted that the scores from the two sources need to be normalized first. These combined results should outperform those produced by each individual scoring system.

4 Rule-Based Information Combination

The rule-based system is a well-developed and widely-used technique since the 1980's. Production rules can be formulated in the IF-THEN format.
Rule i: if (A_1 *L.O.* A_2 *L.O.* … A_m), then C_i.

The antecedent part of the rule contains conditions of features. The conditions of features (A_1~A_m) can be combined with logical operations (*L.O.*) like AND, OR, and NOT. The consequent part of the rule (C_i), on the other hand, is the action or decision to be taken when the antecedent part is valid/true. The rule is called *fired* when this happens.

To analyze the behavior of a person in a specific problem domain, we can identify a few types/classes of behavior first. The features are the data collected from the RFID readers and the video cameras. And the system is used to find the class of behavior according to the features at a certain time. A rule base should be constructed for each specific problem domain. Then an inference engine can be used to infer the results (behavior classes) from the data.

5 An Example in Learning Behavior Monitoring

For students, how to learn class materials in a more efficient way is very important. Some students can stay in the study or even sit in front of the desk for hours, but learn very little. Furthermore, the student makes study plans. But sometimes it is not easy to know the real achievement unless effort is put into recording the learning time manually. Here, we present the idea of using both video cameras and RFID sensors to capture the learning behavior of a student in his/her room. With analysis methods mentioned in the previous two sections, the system can provide personalized learning advice to the student. This idea was called the *ubiquitous learning room* and was first proposed in [7]. Here we further investigate the realization of the system. The ubiquitous learning room can have a setting like the following:

A video camera is installed beside the desk to monitor the learner's behavior when he/she sits in front of the desk. All books, stationery, and related objects are attached with an RFID tag of a unique ID number. An RFID reader is also embedded under the surface of the desk. It is assumed that the reader can detect the RFID tag only when the object is within a certain range of the desk surface. The setting is shown in Fig. 1.

For data acquisition, the video camera can capture first the existence of a person in front of the desk. The show-up time is recorded and the existence feature is marked True. If the person leaves, the disappearing time is recorded and the existence feature is marked False. Furthermore, if a person is in front of the chair, a few behavior features can be identified and recorded, e.g., standing, sitting and head up (thinking or stretching), sitting and looking at the surface of the desk (studying), sitting but looking at other ways (distracted), and sitting but head down near the surface of the desk (sleeping). The behavior features and the time stamps are recorded only when the behavior is changed. On the other hand, an RFID reader is embedded under the surface of the desk. Objects like books, notes, pencils, …, etc. are attached with an RFID and can be detected when they appear on the desk. The data about the existence of the objects are recorded when changes occur. True or False of the existence and the time stamp of the change are recorded. The identification of a specific person appearing in front of the desk can also be done by the RFID. However, if the desk is used only by a specific person, then this is not necessary.

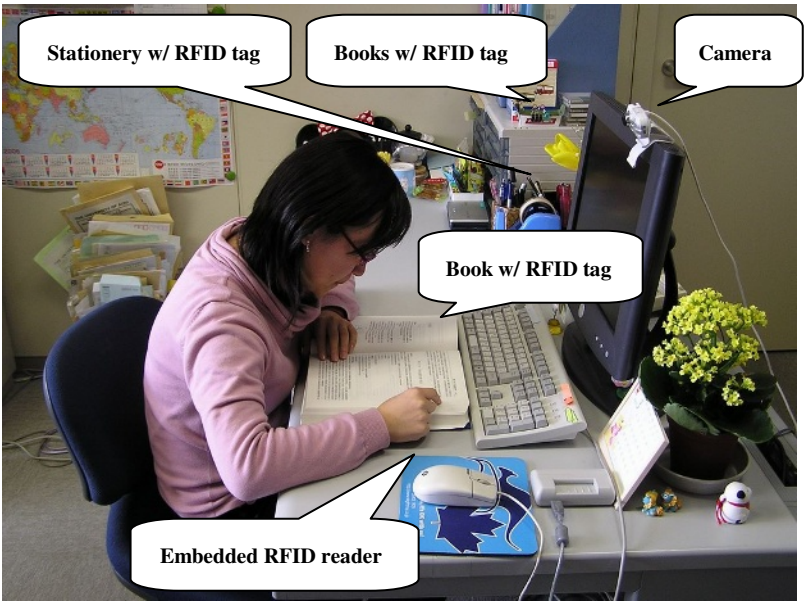


Fig. 1. Setting of a ubiquitous learning desk

Finally, the human behavior can be classified into several predefined categories with the combined RFID and video information. The categories can be, to name a few, away, sleeping, distracted, studying mathematics, studying English, writing mathematics homework, and writing English homework. Examples in the rule base are shown in Table 1.

Table 1. A sample rule base for behavior inference

R1: IF(Human existence = False), THEN Away.
R2: IF(Sitting but looking at other ways = True), THEN Distracted.
R3: IF(Sitting but head down near the surface of the desk = True), THEN Sleeping.
R4: IF(Sitting and looking at the surface = True AND Mathematics book existence = True), THEN Studying Mathematics.
R5: IF(Sitting and looking at the surface = True AND English book existence = True), THEN Studying English.

The recorded time stamps can be used to calculate the duration of each identified behavior. The results are then compared with the study schedule planned by the learner. A report can thus be generated to provide information and advice about his/her study to

the learner. The learner can adjust his/her learning habit to improve his/her study efficiency according to the report.

6 Conclusion

Understanding the learner's behavior is essential to provide better support to the learner, the parents, and/or teachers. The developed setting can be utilized in implementing a useful ubiquitous environment for analyzing the learning behavior. Information from both RFID sensors and video cameras is necessary for such a task. And proper combination/fusion is the key to the success of the analysis. A full implementation will be done to verify the usefulness of the idea. This work can also be extended to other problem domains when analysis of human behavior is needed.

Acknowledgement

This work was done during H. Hsu's visit to the University of Aizu in early 2006. H. Hsu thanks the financial support from the University of Aizu.

References

1. Hu, W., Tan, T., Wang, L., Maybank, S.: A Survey on Visual Surveillance of Object Motion and Behaviors. IEEE Trans. on Systems, Man, and Cybernetics—Part C: Applications and Review, Vol. 34 (3). (2004) 334-352
2. Satoh, I.: Location-Based Services in Ubiquitous Computing Environments. In Orlowska, M.E. et al. (eds): ICSOC 2003, Lecture Notes in Computer Science, Vol. 2910. Springer-Verlag, Berlin Heidelberg New York (2003) 527-542
3. Masters, A., Michael, K.: Humancentric Applications of RFID Implants: the Usability Contexts of Control, Convenience and Care. Proc. the 2nd Workshop on Mobile Commerce and Services (IEEE WMCS). (2005)
4. Collins, R.T., Lipton, A.J., Fujiyoshi, H., Kanade, T.: Algorithms for Cooperative Multisensor Surveillance. Proceedings of the IEEE, Vol. 89 (10). (2001) 1456-1477
5. Shoji, K., Mito, A., Toyama, F.: Pose Estimation of a 2D Articulated Object from its Silhouette Using a GA. Proc. the 15th Int'l Conf. on Pattern Recognition, Vol. 3. (2000) 713 – 717
6. Hsu, D.F., Chung, Y.-S., Kristal, B.S.: Combinatorial Fusion Analysis: Methods and Practices of Combining Multiple Scoring System. In: Hsu, H.-H. (ed): Advanced Data Mining Technologies in Bioinformatics. Idea Group Publishing, Hershey, PA, USA (2006) 32-62
7. Cheng, Z., Han, Q., Sun, S., Kansen, M., Hosokawa, T., Huang, T., He, A.: A Proposal on a Learner's Context-Aware Personalized Education Support Method based on Principles of Behavior Science. Proc. the 20th International Conference on Advanced Information Networking and Applications (AINA 2006). (2006)

Well-Being Store: A New Channel in U-Commerce for Insurance Industry

Jong Hwan Suh[1], Sung Min Bae[2], and Sang Chan Park[1,3]

[1] Department of Industrial Engineering and [3]Graduate School of Culture Technology
Korea Advanced Institute of Science and Technology (KAIST),
Guseong-dong, Yuseong-gu, Daejeon, Republic of Korea
{SuhJongHwan, sangchanpark}@major.kaist.ac.kr
[2] Department of Industrial & Management Engineering, HANBAT National University
DuckMyoung-dong, Yuseong-gu, Daejeon, Korea
loveiris@hanbat.ac.kr

Abstract. With new channels and devices based on ubiquitous computing, U-Commerce provides an environment where buyers and sellers are literally able to commerce anytime, anywhere, and anyway they like. In U-Commerce, a lot of opportunities are expectable in various industrial areas, and especially in insurance industry. In this paper, we propose new types of insurance services in U-Commerce. And as a new channel to practice it, we propose Well-Bing Store with three concepts: 'Corporeal Insurance Product', 'U-Cart', and 'Health Examination'. And supportive processes in Well-Being Store are depicted in detail as well. By introducing Well-Being Store, we transform a physical place into an intelligent space which cares insured customers previously before they get sick. This is actualized by managing a list of foodstuffs in a U-Cart accordingly, and by keeping a health examination regularly with help of an insurance company and a hospital.

1 Introduction

The words such as 'ubiquitous', 'pervasive' and 'ambient' have become renowned by the prospect of world where a computing network is available anytime and anywhere. With an influential paper published in 1991, Mark Weiser introduced the concept of 'ubiquitous computing' as the method of enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user. Since then, there have been a lot of researches on the realization of an environment saturated with computing and communication capabilities [1]. And prototype ubiquitous computing environment is being created within the laboratory focusing on networking, data management, security, and user interfaces [2]. Although there remains a long way to go until the ubiquitous computing comes true in our life, ubiquitous computing is getting closer to the realization.

By ubiquitous computing, objects become intelligent and serve as interfaces to information systems. This makes it possible to create something improved through new channels and devices, and provides an environment where buyers and sellers are literally able to commerce anytime, anywhere, and anyway they like. On this wise,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 182 – 191, 2006.

ubiquitous computing is quickening a new era for commerce, and we call this new era's commerce as 'U-Commerce'. It is a dynamic convergence of the physical and the digital, the interface of brick mortar commerce with web-based wireless and other next generation technologies in ways that create new levels of convenience and value for buyer and sellers [3].

In U-Commerce, numerous applications are possible by new channels where additional information on objects, processes, and individuals is gathered, exchanged, and proposed in a cost efficiency way. Above all, insurance industry in U-Commerce seems to be the most important place where the evolution by ubiquitous computing takes place more briskly than any other areas [4, 5]. In this paper, we propose new types of insurance services based on ubiquitous computing technologies: Well-Being Store as a new channel in U-Commerce for insurance industry. And to practice it, we propose three key concepts: Corporeal Insurance Product, U-Cart, and Health Examination. And interactive processes among those concepts in Well-Being Store are depicted in detail. The rest of the paper is structured as follows. Section 2 begins by introducing related works which have been motivations and references. Section 3 describes three concepts which we suggest to achieve Well-Being Store as a new channel in U-Commerce for insurance industry. And section 4 shows in detail how processes in Well-Being Store go on with the concepts on the basis of ubiquitous computing technologies. Section 5 discusses issues related to system architecture and implementation of our schemes, and finally we conclude in section 6.

2 Related Works

Applications like a U-Cart of this paper have been researched. For example, Pocket BargainFinder is a web-based comparison shopper developed for wireless mobile devices to find the cheapest one-line price for an object encountered in the physical world [6]. And Personal Shipping Assistant (PSA) is a handled-sized computer with touch screen, which is attached to the shopping cart [7]. However, U-Cart provides user's shopping list according to his/her health condition while Pocket BargainFinder only provides information about an on-line object for the cheapest price. Comparing to PSA, the U-Cart provides knowledge as well as information resulted from the central health advisory system in Well-Being Store while PSA only provides some proper information queried form database without any data-processing.

The impact of ubiquitous computing on the managerial and industrial areas has been investigated all around world reflecting its significance as a competitive power in the future [8]. In this paper, we choose insurance industry as our applicable area, and focus on proposing new types of insurance services on the basis of a new channel in U-Commerce. The application of ubiquitous computing to insurance company has been discussed a lot as well [4]. The study here also focuses on same issues, but is beyond those scenario based discussions. With practical frameworks, we introduce a corporeal insurance product, and also suggest its concrete purchase & payment processes in innovative ways.

3 Concepts: Corporeal Insurance, U-Cart, and Health Examination

3.1 Corporeal Insurance Product

The first concept is realized by the adoption of a corporeal insurance product. It is the result of efforts we have made to solve problems of insurance industry caused by agent system such as poor interaction and high commission cost. We considered the outline of channels in U-Commerce, and we found out a pathway as a new channel from a customer to an insurance company through a discount store [9]. To distribute an insurance product through a discount store, the insurance product needed to be transformed into the shape proper for the discount store. So, a corporeal insurance product was suggested [10]. Figure 1 describes a sample of corporeal insurance product and its prototype. On the surface of its inside, we attach RFID tag which contains an identifiable unique code for an insurance product. The way how we attach RFID tag may be various, but we prefer to stick it on the surface of the box so we can use it from the box later for additional services. And contract and description of insurance product are also contained in the box. According to the kind of an insurance product, we can diversify stuffs which we put into the box. There may be many alternatives for the stuffs. For example, we may put a free membership card of a fitness club if a customer joins a policy of a life insurance product. Or we may put some discount tickets or necessities of life if the customer is a sort of a housewife.

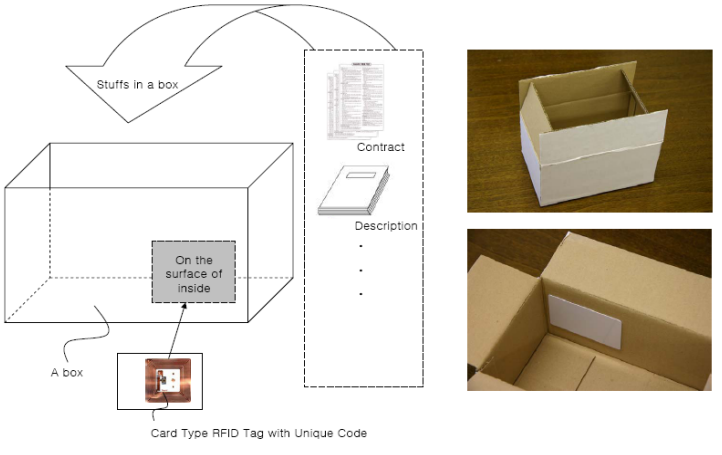


Fig. 1. A sample of a corporeal insurance product (left), and its prototype (right)

3.2 U-Cart: A Cart Based on Ubiquitous Computing

The second concept is a U-Cart which helps an insured customer to select foodstuffs according to the condition of his/her health. When a customer inserts membership

card with RFID tag into the U-Cart, it starts its operations. Soon the U-Cart identifies the customer by reading his/her membership id from the membership card, and it recognizes whether the customer is insured by a corporeal insurance product in Well-Being Store or not. For all customers, the U-Cart provides information about groceries inside of itself through its front display. And the U-Cart gives body to a lot of convenient ways such as quick and automatic figuring up, or guiding to the place where a certain product is located. Further, especially for insured customers, the U-Cart sends foodstuffs' ids to the central health advisory system of Well-Being Store and it shows analyses about their ingredients on its screen with their health conditions examined. By doing so, the U-Cart plays as a dietary advisor over just a carrier or a shopping assistant (see Figure 2).

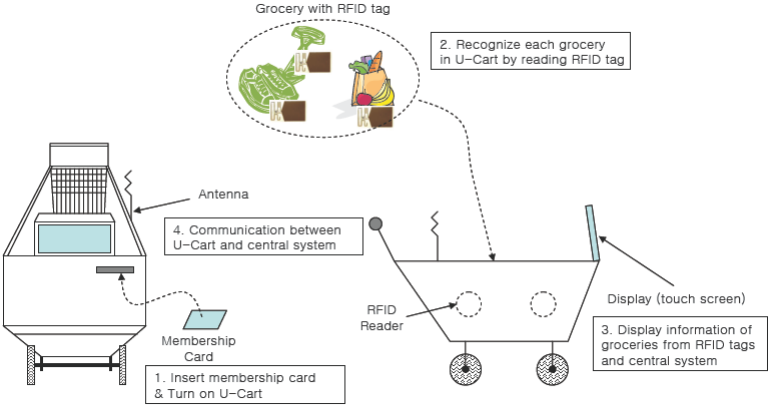


Fig. 2. A sample of a U-Cart - Not only a Shopping Assistant, Dietary Advisor about foodstuffs in a Shopping Cart - helps life-insured customers to select foodstuffs appropriate to their health conditions

3.3 Health Examination

As the third concept, we suggest a space for a health examination in Well-Being Store. This is used to check up on an insured customer's health regularly, and to inform the result whenever he/she wants to know it in Well-Being Store. The result of a health examination can be used in three ways: (1) **Health Care**, a health examination is itself useful for protecting insured customers from any kinds of diseases in advance. Therefore an insurance company can provide all real values which insured customers demand by paying their premiums; (2) **With Corporeal Insurance Product**, we can arrange a premium according to the result of a regular health examination; (3) **With U-Cart**, whether each of foodstuffs in a U-Cart is appropriate to an insured customer's health condition is analyzed, and if a food does not fit to his/her health condition, then it indicates the result on the screen of a U-Cart (see Figure 3).

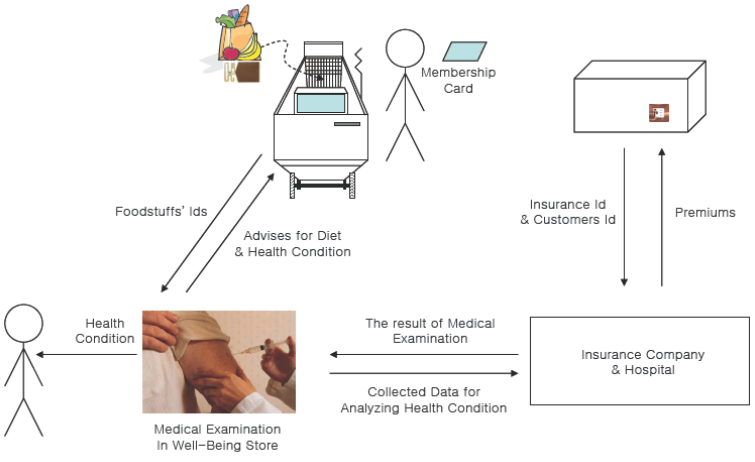


Fig. 3. A descriptive sample of using a Health Examination

4 Process Descriptions for Well-Being Store

4.1 Join/Keep Insurance in Well-Being by Corporeal Insurance Product

Figure 4 describes how a customer joins a policy of insurance services. When a customer visits Well-Bing Store, the customer who wants to be insured in Well-Bing Store just puts a box of a corporeal insurance product into a U-Cart. At once, the U-Cart identifies what the box is, and it displays insurance product's information on its screen. Here, the price of a corporeal insurance product is the first premium which the customer has to pay for being insured. After confirming insurance product's information on the screen, the customer completes the purchase of the insurance product by paying the price and signing on a receipt.

And to keep the policy of insurance services, the customer just needs to buy the same kind of a corporeal insurance product again when he/she visits Well-Being Store. Concrete steps for the payment process are described in figure 5. Surely a customer who is not insured does not belong to this process and just skip it. If a customer is turned out to be insured, then Well-Being Store decides if he/she needs to pay a premium in this time of visit. So, if the customer has to pay it currently, then the premium is paid through the same purchase steps as the other groceries. Here each premium at each time is calculated according to the customer's health condition stored in database system of Well-Being Store. Therefore, regular health examinations are necessary and somewhat mandatory.

Figure 6 shows how to keep the health examination regularly and how to use the result to calculate the customer's premium for each time when the customer visits Well-being Store. The concept of 'Health Examination' is applied to calculate a variable premium for each time, and to take care of customer's health condition regularly.

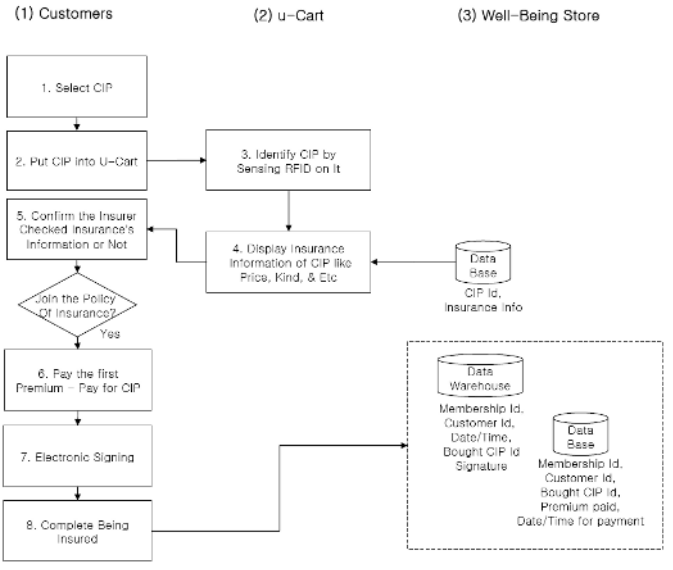


Fig. 4. Purchase process of a corporeal insurance product in Well-Being Store – 'CIP' is the abbreviation of 'Corporeal Insurance Product'

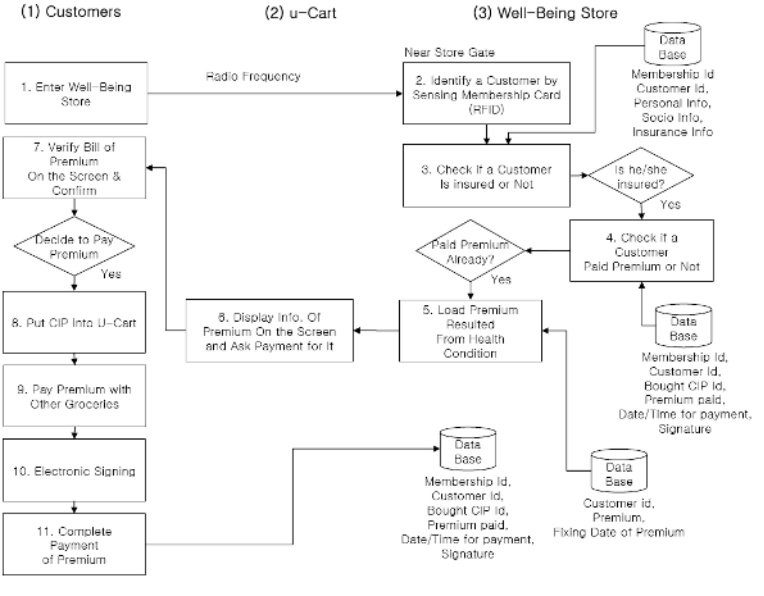


Fig. 5. Payment process of a corporeal insurance product in Well-Being Store – 'CIP' is the abbreviation of 'Corporeal Insurance Product'

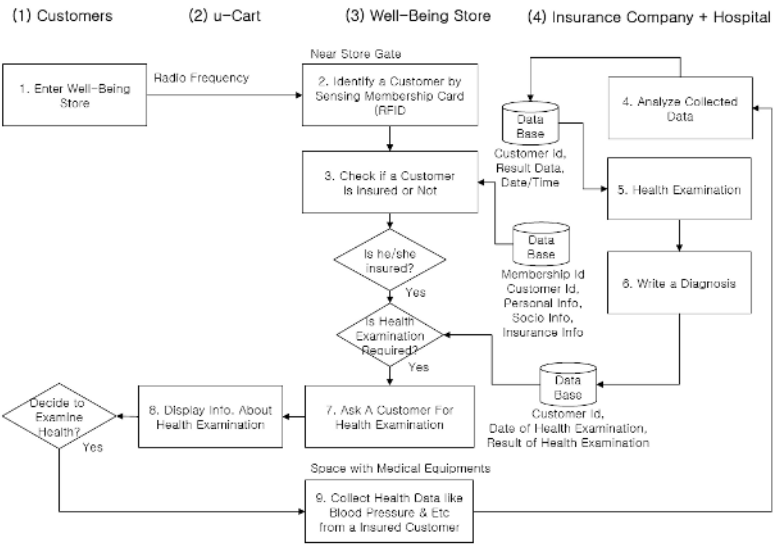


Fig. 6. Calculating a variable premium by customer's health conditions & Regular health tests

4.2 Just When a Customer Visits Well-Being Store with U-Cart

Figure 7 shows steps in order after a customer visits Well-Being Store. When a customer enters Well-Being Store, the customer is identified automatically by his/her membership card containing the customer's id. Well-Being Store analyzes a period of consumption for each grocery, and makes a list of groceries which the customer may need to buy in this time of visit. And then, it adds groceries which the customer bought just before but are not expected that the customer will not buy in this time of visit. For example, let's assume that there's a customer who bought an apple, a banana, and a potato lately. And one day, on Wednesday, a customer visited there again. As soon as he entered, Well-Being Store found that the customer buys an apple on every Thursday with 90% possibility. It also discovered that the customer buys an onion on every Wednesday with 95% possibility. The shopping list proposed on U-Cart's screen consists of three groceries except an apple: a banana, a potato, and an onion.

If the customer is insured by buying a corporeal insurance product in Well-Being Store, one more step should be progressed. The shopping list is checked and modified according to the customer's health conditions. For example, if the customer is found out that he/she has allergic to a potato, then Well-Being Store eliminates it from the previous shopping list. This kind of filtering out bad foodstuffs is performed by cooperation with an insurance company and a hospital. Figure 8 shows more specific steps for making the final shopping list where customer's health conditions are considered. Additionally in figure 9, we describe how a U-Cart provides customer's health conditions on its screen.

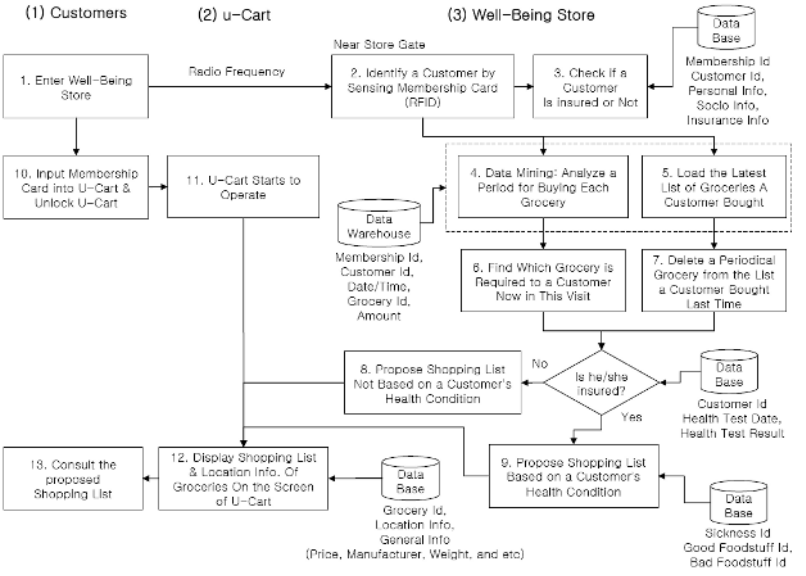


Fig. 7. Processing shopping list after a customer enters Well-Being Store

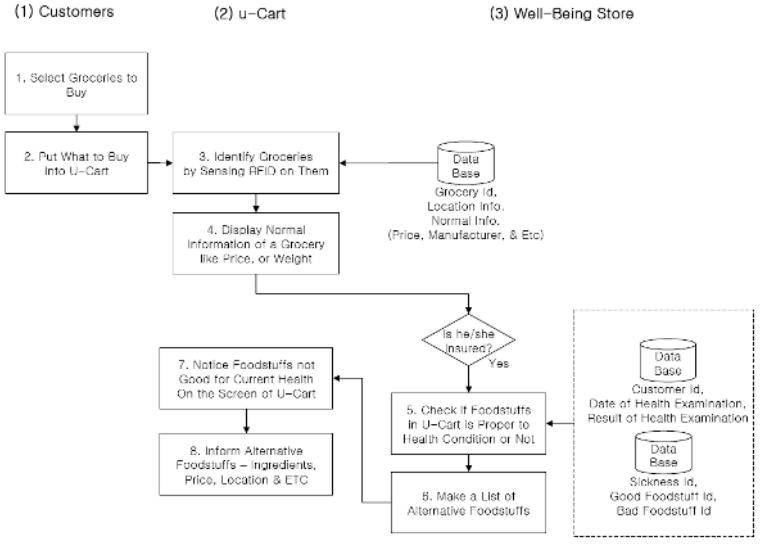


Fig. 8. Process for checking/modifying the foodstuffs' list in a U-Cart by health condition

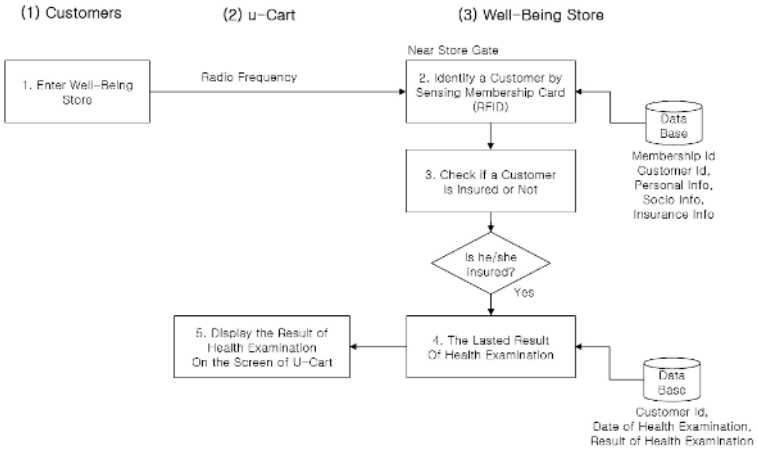


Fig. 9. Process for displaying health examination's result on the screen of a U-Cart

5 Discussions for System Architecture and Implementation

The method of this trial has several limitations. One of them is that we have to assume an insured customer who turns on a U-Cart selects his/her own foodstuffs only. If the other who has the insured customer's membership card starts U-Cart's operating, and receives advisory health information form Well-Being Store, those interactions become somewhat useless. Secondly, we need to consider how we turn this method to be a good investment. To make Well-Being Store a place which is equipped with U-Carts and for health examination, a lot of money has to been put into it. Therefore, we have to find solutions to overcome financial obstacles when we begin and manage Well-Being Store. Moreover, in this paper, we do not implement the actual system for the proposed methods. So, we want to discuss issues about its system architecture here. Here we suggest making use of 915 MHz RFID systems with EPC (Electronic Product Code) and 802.11g for the realization of three concepts (see Section 3) and their processes in Well-Being Store (see Section 4). And for building up information system in Well-Being Store, we employ the framework based on data warehouse and data base.

6 Conclusions and Future Work

In this paper, our goal is to provide insurance services through a new channel in U-Commerce by introducing concepts and processes of Well-Being Store based on ubiquitous computing technologies. By managing a list of foodstuffs accordingly with a U-Cart and keeping a health examination regularly, Well-Being Store become itself an intelligent space which cares insured customers previously before they get sick. Eventually, customers can keep being healthy through easier ways as long as they are insured in Well-Being Store, and an insurance company can create the real value of being healthy meeting customer's expectation for the premiums paid. Using regular

health examinations, it is possible for insured customers to pay reasonable premiums in accordance with their conditions of health. It also helps an insurance company to cut down insurance over a long period of time. In addition, a discount store also gets benefits with more customers increasing overall sales.

The study here focuses on describing concepts and processes of Well-Being Store. So, we do not make any prototypes for the actual systems. Therefore, its realization is required for further works. And prior to it, more specific system architectures need to be discussed and built up. Above all, how this paper will have effects on insurance industry needs to be investigated subsequently as well. We have to validate its economic efficiency in real world. We also need to improve our concepts and processes to provide more personalized insurance services based on ubiquitous computing technologies. We hope to perform further studies about how to expand the physical space beyond Well-Being Store. If we monitor customer's context-awareness related to insurance services in more places, we will be able to provide more services than proposed ones in the paper based on Well-Being Store.

Acknowledgement. This research is supported by the KAIST Graduate School of Culture Technology.

References

1. P. Boddupalli, F. Al-Bin-Ali, N. Davies, A. Friday, O. Storz, and M. Wu: Payment Support in Ubiquitous Computing Environments, Fifth IEEE Workshop on Mobile Computing Systems & Applications, pp. 110-121, 2003
2. M. Roman, C.K. Hess, R. Cerqueria, A. Ranganathan, R.H. Campbell, and K. Nahrstedt, Gaia: A Middleware Infrastructure to Enable Active Space, IEEE Pervasive Computing, pp. 74-83, Oct-Dec 2002
3. Stephen Schapp, and Richard D. Cornelius, U-Commerce: Leading the New World of Payments, White Paper, 2001
4. V. Corama, J. Bohn, and F. Mattern: Living in a Smart Environment: Implications for the Coming Ubiquitous Information Society, International Conference on Systems, Man and Cybernetics, pp. 5633-5638, IEEE Computer Society, 2004
5. Anatole Gershman: Ubiquitous Commerce: Always On, Always Aware, Always Proactive, Proceedings of the 2002 International Symposium on Applications and the Internet, IEEE Conmputer Society, 2002
6. Brody, A.B. & Gottsman, E.J.: Pocket BargainFinder: A Handheld Device for Augmented Commerce, Int. Symposium on Handheld and Ubiquitous Computing, pp. 44-51, Karlsruhe, Germany, 1999
7. METRO Group: Personal Shopping Assistant, http://www.future-store.org, 2004
8. Martin Strassner and Thomas Schoch: Today's Impact of Ubiquitous Computing on Business Process, Short Paper, Proceeding of the International Conference on Pervasive Computing, pp. 62-74, Pervasive2002, April 2002
9. Suh, J.H., and Park, S.C.: New Channel Management and Convergence in Service Industry: U-Commerce cases, the 4th Korea-China Quality Symposium, pp. 227-231, KSQM, August 2005
10. Suh, J.H., and Park, S.C.: A RFID System for Distribution of Corporeal Insurance through a New Channel in U-Commerce, Entrue Journal of Informational Technology, Vol.5, No.1, pp. 77-89, January 2006

Real-Time License Plate Detection Under Various Conditions

Huaifeng Zhang, Wenjing Jia, Xiangjian He, and Qiang Wu

Computer Vision Research Group,
University of Technology, Sydney, Australia
{hfzhang, wejia, sean, wuq}@it.uts.edu.au

Abstract. This paper proposes an algorithm for real-time license plate detection. In this algorithm, the relatively easy car plate features are adopted including the simple statistical feature and Harr-like feature. The simplicity of the object features used is very helpful to real-time processing. The classifiers based on statistical features decrease the complexity of the system. They are followed by the classifiers based on Haar-like features, which makes the final classifier invariant to the brightness, color, size and position of license plates. The experimental results obtained by the proposed algorithm exhibit the encouraging performance.

1 Introduction

A key technique in most of traffic related applications such as stolen car hunting, road traffic monitoring, and parking lots access control, is automatic car plate detection followed by the high accuracy car plate number recognition in un-controlled open environment. However, car plate detection is a difficult task because of ambient illumination conditions, visual angle, image perspective distortion, interference characters, etc., particularly under various complex conditions. Most of previous license plate detection algorithms are restricted in certain working conditions, such as fixed backgrounds [1], known color [2], or designated ranges of the distance between camera and vehicle [3, 4].

In these years, there were some researchers working on license plate detection under various conditions. Chang et al. [5] proposed a license plate detection algorithm using color edge and fuzzy disciplines. Their algorithm only can be used to detect the license plates with specific colors. In [6], Matas and Zimmermann proposed an algorithm to detect license plate and road sign in various conditions. They used character regions as basic units of license plate, which makes their algorithm robust to viewpoint and illumination. However, it can hardly distinguish interference characters from the true license plates. Kim et al. [3] proposed another license plate detection algorithm using both statistical features and templates. After the statistical features were used to select the Regions of Interest (ROI), license plate templates were employed to match the ROI. In most cases, however, general plate templates are very difficult to be constructed. Moreover, the sizes of the statistical features used in their algorithm were fixed. Hence the application of this algorithm is restricted extremely.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 192 – 199, 2006.

Recently, Haar-like features were widely used in object detection [7, 8]. Haar-like features were widely used in object detection [13]. It is a kind of simple object feature which is easy for computation. The major problem is that the number of Harr-like features involved by an object is huge, so it will easily increase the system complexity. Chen and Yuille [9] constructed a simple cascade classifier for text detection using statistical features. However, only statistical features were used in their algorithm, which always results in high false positive rate in practice.

In this paper, we use both statistical features and Haar-like features in the algorithm. The classifiers based on statistical features decrease the complexity of the system. They are followed by the classifiers based on Haar-features, which further improve the detection rate and low down the false positive rate. Moreover, in our algorithm, the statistical features are extracted from vertical gradient image, which makes algorithm extremely fast. The final cascade classifier is obtained by combining the above two kinds of classifiers.

The rest of the paper is organized as follows. The framework of our algorithm is introduced in Section 2. Vertical gradient image and two statistical features, Gradient Density and Density Variance, are defined in Section 3. Then Haar-like features and AdaBoost algorithm are described in Section 4. Experimental results are presented in Section 5. At last the paper is concluded in Section 6.

2 The Framework of the Algorithm

In our algorithm, we construct a cascade classifier [8] to increase the detection speed, in which the first two layers are based on statistical features and the following layers are based on Haar-like features. In this section, we introduce the algorithm in two aspects: testing and training.

2.1 Training

Positive samples and negative samples are needed in the training procedure. The positive samples are obtained through labeling the license plate regions from the vehicle images. The negative samples are extracted from different images which do not contain license plate. Some of the negative images are vehicle images without license plates. The others are the images taken randomly. All of the samples are scaled to 48*16 for the convenience of training.

Firstly, for all the samples, the values of one of the statistical features, Gradient Density, are calculated. A classifier is obtained by selecting the threshold which classifies all the positive samples as positive ones. Then, all the samples, including positive samples and negative samples, which are classified as positive ones (true positives and false positives) are used to train the classifier on the second layer. This classifier is based on the other statistical feature, Density Variance. The input samples are classified again and the positive ones are used to train the classifier on the third layer. Similarly, the samples classified as positive ones by the third layer are input to the fourth layer, and so on. The training finishes when the given false positive rate is reached. In our algorithm, we trained four layers of classifier based on Haar-like features and AdaBoost leaning procedures, which is the layer 3 through layer 6 in the final cascade classifier.

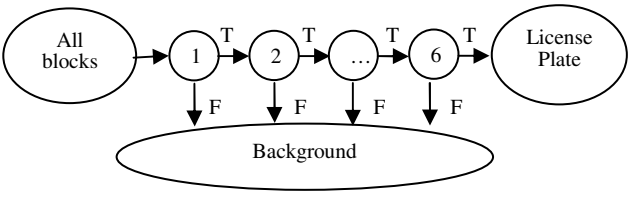


Fig. 1. The working flow of cascade classifier, where 1,2,…,6 represent the layers

2.2 Testing

When an image is input into the classifier, a mask of 48*16 is used to capture the same size of pixel block on the image. This mask will go through the whole image area. At each position, the cascade classifier is used to verify if the block covers a license plate. A cascade classifier can be taken as a degenerate decision tree as shown in Fig. 1. A positive result from the classifier on an upper layer triggers the classifier on the next layer. A negative outcome at any layer leads to an immediate rejection of the block. Then it slips to the next position and the same procedure is repeated.

The detection is implemented in multiple scales. In order to detect license plates of variant sizes, the block size is scaled up from 48*16 to 240*80, with a scaling factor of 1.2.

3 Statistical Gradient Features

Statistical analysis shows that the regions of license plates have some common characteristics. Firstly, a license plate region usually contains rich edge information. Secondly, most of the edges are vertical edges [1, 11]. Thirdly, the edges are distributed relatively uniformly in a license plate region.

Based on the observation, we define two statistical features of the block of a license plate. Both of the features are constructed from vertical edge image.

For algorithm simplicity purpose, the gradient information is investigated rather than edge information because an efficient general-purpose edge detector is usually difficult to obtain in practice.

3.1 Vertical Gradients Image

The vertical gradient image is generated by the convolution of the original image and the x-direction Sobel operator

$$S = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \tag{1}$$

The vertical gradient image emphasizes the differences on x-direction and diminishes the differences on y-direction. Because most of the edges in license plate

regions are vertical edges, the vertical edge image keeps most of the edge information in license plate region, at the same time, eliminates a large amount of edge information in background regions as shown in Fig.2.

(a) (b)

Fig. 2. The gradient images of a image (a) Normal gradient image; (b) Vertical gradient image

3.2 Gradient Density

The gradient density in a block is used to describe the edge density of the block using

$$D_G = \frac{1}{N}\sum_i \sum_j G(i,j) \tag{2}$$

where $G(i,j)$ represents the gradient magnitude at location (i,j) and N is the number of pixels in the block.

The x-direction Sobel gradient operator is employed to produce gradient map, where the resulted gradient magnitudes are normalized by the maximum gradient strength in the image.

During the training procedure, the size of the block is fixed to the size of the sample images, which is 48*16. During the testing procedure, the size of the block is changed depending on the scale of the searching block.

3.3 Density Variance

Besides the abundant edge information, note that the foreground characters in a license plate are usually distributed with relatively even interval. As its consequence, the gradient in the block of a license plate is distributed more evenly in space with similar strength, compared to most of the areas with simple structures. Fig.3 gives such an example.

Therefore, we modify and redefine the density variance feature [12] in order to discriminate license plates from background regions.

To obtain the feature, a block is divided into 12 equal-sized sub-blocks, as shown in Fig. 3. Let g_i denote the mean value of the gradient strength at sub-block i, and g

denote the mean value of the gradient strength of the whole block. Then, the density variance of the block, denoted as V_G, is defined as

$$V_G = \frac{\sum_{i=1}^{n} |g_i - g|}{n \cdot g} \tag{3}$$

where n is the number of the sub-blocks, e.g., $n = 12$ in above example.

(a) License Plate (b) Background

Fig. 3. Areas with different vertical gradient density variance distributions

The above defined density variance, which takes value from 0 to 1, is a ratio to the mean gradient strength of the block. In this way, no matter whether the gradient is strong or weak, the density variance keeps low as long as there are similarly strong or weak gradient distributed evenly through the block.

4 Haar-like Feature and AdaBoost

The Haar-like features originate from Haar basis functions [13]. They consist of a number of rectangles covering adjacent image regions (see Fig. 4). The value of a Haar-like feature is the difference between the average of the pixel values (in our algorithm, the gradient magnitude) in white rectangles and grey rectangles.

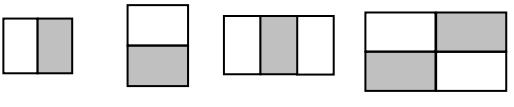


Fig. 4. Four types of Haar-like features

A Haar-like feature is determined by its type, the size and the position of the rectangles. The size and the position can be any as long as the feature is in the image block. Such Haar-like features dictionary can capture the interior structure of objects that are invariant to certain transformations. However the number of the features is too large in this features dictionary, e.g. there are hundreds of thousands features in a 48*16 image block. It is prohibitively time-consuming to compute all the features.

AdaBoost algorithm [10] is a good choice to select a small number of features from a very large number of potential features. The classifier trained through AdaBoost algorithm is the combination of a set of simple classifiers (called weak classifier), where each simple classifier uses one feature. The construction of weak classifier is

independent of AdaBoost algorithm. In our algorithm, perceptron [14] is selected as the weak classifier, in which the classifying threshold is determined by the given detection rate.

The basic idea of the AdaBoost algorithm is as follows. After constructing a weak classifier, the samples are re-weighted in order to emphasize those which are incorrectly classified. Then the next weak classifier is trained with the re-weighted samples. A number of weak classifiers are trained in this way till the given false positive rate is reached. The final classifier (called strong classifier) is constructed by combining these weak classifiers using a set of weights. These weights are determined by classification error of each weak classifier.

5 Experiments

We use 460 vehicle images in our experiments. 300 images are taken as training images, in which there are 305 visible license plates; the other 160 images are testing images, in which there are 169 visible license plates. The images used in our experiments were taken in various circumstances with various illuminations and view angles. The colors and the styles of the license plates are different. Some examples of the license plates are shown in Fig. 5.

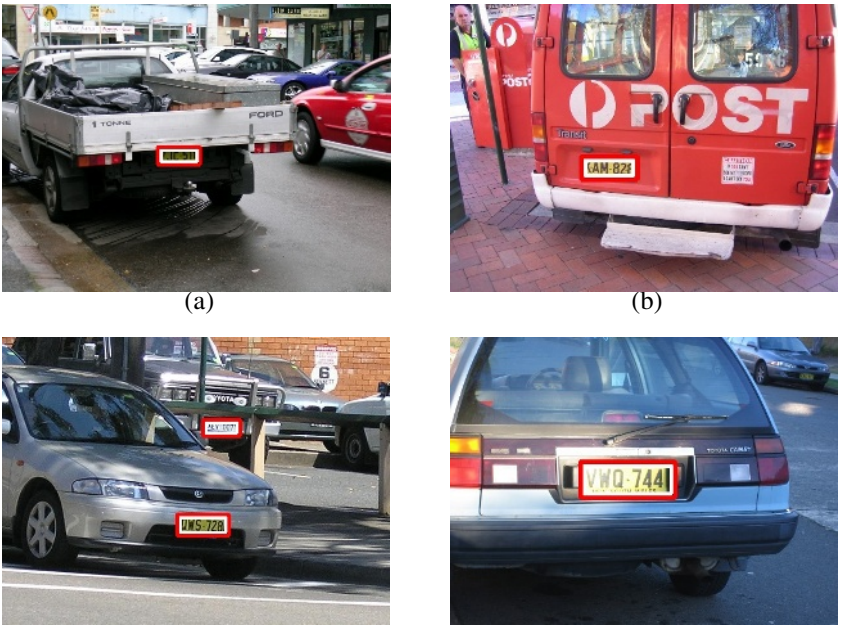
Fig. 5. Some examples of the license plates used in our experiments

The negative samples used to train the classifiers based on statistical features are collected by randomly selecting 28,000 sub-windows from 50 images which do not contain any license plate. The negative samples used in AdaBoost learning procedure are obtained from the incorrectly classified samples which are randomly extracted from 220 images that do not contain any license plate.

In the experiments, a six-layer cascade classifier is obtained. Each of the first two layers uses one of the statistical features defined in Section 3. On the last four layers, the numbers of the features in the strong classifiers are 18, 23, 55 and 67 respectively. So our final cascade classifier has 6 layers and uses 165 features. Compared to Viola's classifier having 38 layers and using 6060 features [8], our classifier is much simpler and efficient.

In our experiment, among the 169 visible license plates in 160 testing images, 156 license plates are detected, with detection rate 92.3%. At the same time, there are only 7 false positive regions. On a PC with Pentium 2.8GHz CPU, the detector can process a 648*486 image in about 50ms.

Fig. 6 shows some of the detection results, where the license plates are circled by white boxes. From the examples, we may see that our algorithm can work under various complex environments, various illuminations, and various view angles. The algorithm can detect the license plates with various sizes, positions and colors. Fig. 6(a) is an example with complex background; Fig. 6(b) shows the license plate detection against interference characters; Fig. 6(c) shows the result of detecting multiple license plates in one image; Fig. 6(d) shows that the algorithm is robust to the variance of illumination.



(a) (b)

(c) (d)

Fig. 6. Detection results of some vehicle images

6 Conclusions

In this paper, we construct a cascade classifier for license plate detection using both statistical and Haar-like features. The classifiers on the first two layers are based on statistical features. They can exclude more than 80% background regions from further training or testing. The classifiers on the next four layers, trained by AdaBoost learning procedure, are based on Haar-like features. In our algorithm, the training and testing are both extremely fast. With much less features, we obtain 92.3% detection rate and very low false positive rate when the license plate detection algorithm works in various complex environments.

References

1. Bai, H. and C. Liu. A Hybrid License Plate Extraction Method Based On Edge Statistics and Morphology. In Proceedings of the 17th International Conference on Pattern Recognition, pp. 831-834 vol.2, 2004.
2. Kim, S.K., D.W. Kim, and H.J. Kim. A Recognition of Vehicle License Plate Using a Genetic Algorithm Based Segmentation. In Proceedings of International Conference on Image Processing, pp. 661-664 vol.2, 1996.
3. Kim, S., et al. A Robust License-plate Extraction Method under Complex Image Conditions. In Proceedings of International Conference on Pattern Recognition, pp. 216-219 vol.3 2002.
4. Jia, W., et al. Mean Shift for Accurate License Plate Localization. In Proceedings of International Conference on Intelligent Transportation Systems, pp. 566-571, 2005.
5. Chang, S.-L., et al., Automatic License Plate Recognition. IEEE Transactions on Intelligent Transportation Systems, 2004. 5(1): pp. 42-53.
6. Matas, J. and K. Zimmermann. Unconstrained License Plate and Text Localization and Recognition. In Proceedings of IEEE International Conference on Intelligent Transportation Systems. pp. 572-577, 2005.
7. Mita, T., T. Kaneko, and O. Hori. Joint Haar-like Features for Face Detection. In Proceedings of the Tenth IEEE International Conference on Computer Vision. pp. 1619-1626, 2005.
8. Viola, P. and M.J. Jones, Robust Real-Time Face Detection. International Journal of Computer Vision, 2004. 57(2): pp. 137-154.
9. Chen, X. and A.L. Yuille. Detecting and Reading Text in Natural Scenes. in Proceedings of the International Conference on Computer Vision and Pattern Recognition, pp. 366-373 vol.2, 2004.
10. Freund, Y., An Adaptive Version of the Boost by Majority Algorithm. Machine Learning, 2001. 43(3): pp. 293-318.
11. Yu, M. and Y.D. Kim. An Approach to Korean License Plate Recognition Based on Vertical Edge Matching. In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, pp. 2975-2980 vol.4, 2000.
12. Chen, W.-Y. and S.-Y. Chen, Adaptive Page Segmentation for Color Technical Journals' Cover Images. Image and Vision Computing, 1998. 16(12-13): pp. 855-877.
13. Papageorgiou, C. and T. Poggio, A Trainable System for Object Detection. International Journal of Computer Vision, 2000. 38(1): pp. 15-33.
14. Duda, R.O., P.E. Hart, and D.G. Stork, Pattern Classification. 2nd ed. 2001, New York, NY, USA: Wiley.

RUIS: Development of Regional Ubiquitous Information System and Its Applications: Towards a Universal Ubiquitous Information Society

Susumu Konno[1], Kazuhide Koide[2], Shigeru Fujita[3], Tetsuo Kinoshita[1], Kenji Sugawara[3], and Norio Shiratori[2]

[1] Information Synergy Center, Tohoku University,
6-3 Aoba, Aramaki-aza, Aoba-ku, Sendai, 980-8579, Japan
skonno@isc.tohoku.ac.jp
[2] Research Institute of Electrical Communication, Tohoku University,
2-1-1 Katahira, Aoba-ku, Sendai, 980-8577, Japan
norio@shiratori.riec.tohoku.ac.jp
[3] Faculty of Computer & Information Network Science, Chiba Institute of Technology,
2-17-1 Tsudanuma, Narashino, Chiba, 275-0016, Japan

Abstract. Wide deployment of ubiquitous information environment makes it possible to provide various information to many users. "Regional information" should be one of the most valuable information if "context-aware" provision to community residents is made. But "contexts" of community residents are private information and should be treated very carefully. In this paper, we propose a novel information system called RUIS -Regional Ubiquitous Information System. RUIS can obtain and store regional information from network, realize robust security mechanism of personal/private information protection, and provide information that matches to residents' contexts. Using RUIS features we can realize"smart watch over" for children. We are working on the implementation of this application based on RUIS prototype.

1 Introduction

Rapid development of ICT, Information and Communication Technology, and wide deployment of ubiquitous information environment make it possible to provide various information to many users through ubiquitous network environment. One of the most valuable information from these ubiquitous environments is "regional information". The most remarkable feature of "regional information" is the dynamic changes in effectiveness depending on the "contexts" of community residents. A good example of change in effectiveness is bargain information. Its effectiveness increases as the peak shopping time comes closer and the distance between shops and people shorten. An information provider should provide the information to the residents where the information is most informative with the "context". Without this "context", if information providers notify information

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 200–208, 2006.

to "as many people as possible", a lot of people may get annoyed by the mostly useless information. There are two difficulties for the information provides to refer or utilize the "contexts" of community residents, as follows:

A) A risk of context information leaking: "Contexts" can be personal, and often contains private information. There is a risk of careless private information disclosure when information provider directly utilizes context information.
B) The dynamic changes of effectiveness of information: Another solution is a resident-side information filtering based on this context. But the dynamic characteristic of regional information makes filter configuration too complex. There is no universal filtering configuration for each resident.

Striking a balance between protecting private context information and providing context-aware information service is a big challenge. Very few works have been done yet to realize it in existing information security technologies and ubiquitous networking technologies. It needs regional sociality aware information control technologies. We named such technologies as "socialware", which is a part of "symbiotic computing" [1] [2] [3] [4] [5] [6] that we have been investigating for more than ten years.

In this paper, we propose a novel information system called RUIS – Regional Ubiquitous Information System. RUIS is an information infrastructure that supports various applications such as the bargain sell and the watch-over system. Design of these support applications is based on RUIS's "region-oriented socialware", where the concept of symbiotic computing can be applied effectively. RUIS can obtain and store regional information from network, realize robust security mechanism of personal/private information protection, and provide information to residents of the community that matches to their contexts. Using RUIS features we can realize "smart watch over" for children, senior, and handicapped. And we are working on the implementation of this application based on RUIS prototype.

In our proposed RUIS there are two very distinguishing features:

1) Socialware for security and safety:
   It provides seamless interface to community residents based on "social intelligence". It provides 'smart' and secure access to private information of community residents.
2) Regional information infrastructure based on the socialware:
   It has safety method of utilizing private information by applications. Information providers do not have to take the risk of exposing personal information.

2 The Structure of RUIS

2.1 The Concept of RUIS

RUIS is an information infrastructure that has region-oriented socialware built-in. It collects and stores regional information, that includes culture, history,

lifestyle, economy, employment, health, education, and personal information of community residents through ubiquitous information environments. RUIS plays a role of an infrastructure for sharing those types of information based on "sociality", that is, human and regional relationships and state of the community.

Example of information, RUIS stores are as follows:

- personal information (Name, Occupation, E-mail address, etc.)
- poasitional information of each community residents
- traffic information, event announcements
- situation of weather, disaster

RUIS manages those types of information based on "sociality", as described above, and "social intelligence" from region-oriented socialware. It enables to define the disclosure level for each information for different purposes. By these features, RUIS can provide "bare-bone" information, adequate to suit needs of application. By the term "bare-bone" we mean, "minimal set of information that keeps information structure, by completely removing the unnecessary contents", which makes it easier to disclose only a part of personal information that is needed for enjoying a service without careful consideration even for the people, those who are unaccustomed with Information and Communication Technology (ICT). Applications also do not have to implement mechanisms of secure storage and management about personal information.

2.2 The Framework of RUIS

Fig.1 shows the framework of RUIS. RUIS is build as a middleware between ubiquitous communication infrastructure and service applications. Regional information and personal information is collected from ubiquitous infrastructure, stored and managed by RUIS with only permitting internal reference.

When applications ask for the information with their requirement, RUIS will provide the "bare-bone" information and we call them as "RUIS-information". RUIS will not provide direct information access or raw information. This information is processed and filtered according to the proper disclosure level. All application services are realized by utilizing RUIS-information. Some kinds of application may want to communicate with each community resident directly. When these applications ask for relaying the service, RUIS also decides how to relay them actively.

We consider the example of bargain announcement, when information supplier needs to send the announcement of "supermarket XX bargain sell" to the "all housewives living in region YY" with e-mail. In this case, information supplier does not have to know about housewives' e-mail addresses. Instead they only need to send the announcement to RUIS. RUIS extracts RUIS-information that includes "e-mail address set of housewives who are interested in bargain of region YY", from its stored information, and send the announcement to them. Another example is multicast streaming, a direct type of communication application. For example, when promoter of a rock concert held in region ZZ want to multicast the promotion to teenagers, he does not have to know the names, addresses,

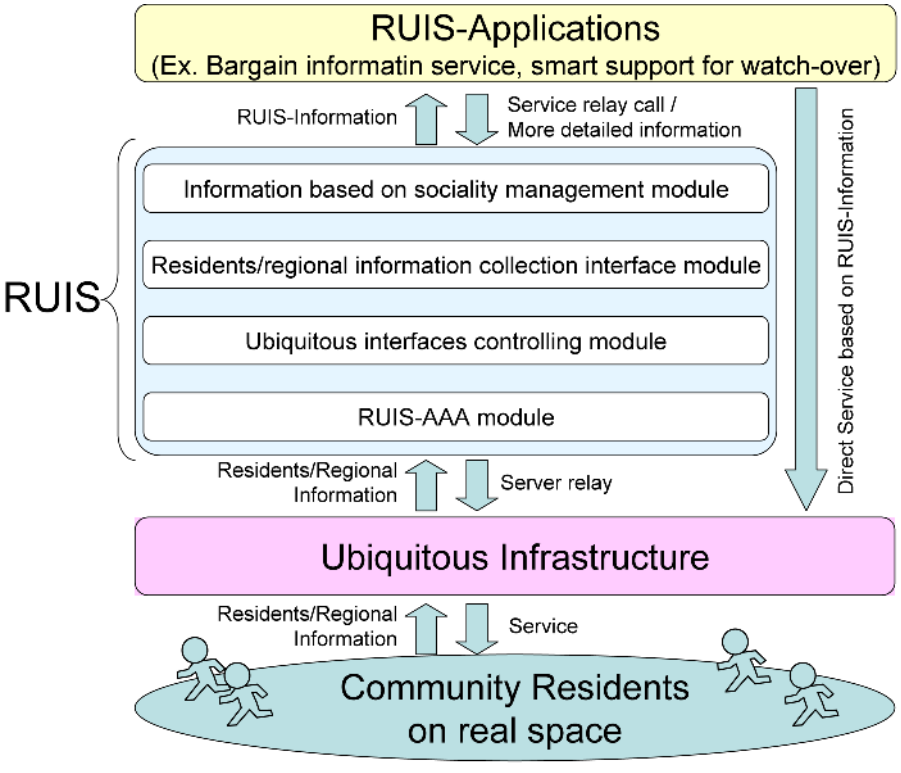


Fig. 1. The Framework of RUIS

birthdays, and other personal information about the recipients. RUIS provides only "IP addresses" of personal computers owned by the teenagers who want to go to rock concert in region ZZ. RUIS consists of four component systems. We will describe them in section 3.1.

2.3 RUIS Application

Applications that deal with regional information and personal information described as above, should provide not only filtering positional information but also context-aware information. We consider the bargain announcement example once again. Bargain information of supermarket XX delivered to residents living nearby to supermarket XX can be considered as imperfect from the view point of context-aware information delivery. It should be delivered to the "residents those who are now going to shopping, or able to go soon to the nearby supermarket XX". But "context" is often considered as private information, and there are many region-specific situations. Universal applications will be almost impossible to utilize these types of information safely. The main objective of RUIS is to deal with the critical information, such as personal/private information including contexts, and region specific information instead of applications.

RUIS can store these types of information independently from applications. Developers of applications have no need to consider about safe management of personal/private information or contexts. This will not only make the operation safe, without the risk of disclosing critical information, but also decrease costs of application development. We call those applications as "RUIS-application".

3 The Architecture of RUIS

3.1 RUIS Internal Components

RUIS is constructed of four components described below:

1) Resident/regional information based on sociality management module:
This component carries out classification of collected information based on sociality, generation of RUIS-information, management, and dynamic authentication of the information. It will be implemented as a region-oriented socialware, and can help various activities of community residents. "Sociality" contains relationships between residents, such as parent-child, friends, classmates, neighbors etc., and relationships between a resident and the regions he lives in, like, address, working place, etc. Dynamic authentication is realized based on 'sociality' as described above, and "social intelligence", which provides 'smart' access, as well as secure access, to private information of community residents. RUIS application will obtain RUIS-information that has proper information disclosure level managed by the dynamic authentication mechanism. For example, considering the case of Mr. A (Age.34/Male/Living in SENDAI city). Application that provides bargain announcement can only obtain information of Mr. A from RUIS as "A person, Male, in his thirties". But RUIS can provide detail information about Mr. A for his parents or friends. Sociality, in this case, relationship between residents, and social intelligence, like parents and friends those who are trustworthy, enable RUIS to control dynamic change of information disclosure level.
2) Residents/regional information collection interface module:
This is a universal interface to obtain regional and residential information from some ubiquitous infrastructures. Those are mainly collected from mobile hand-held equipments of residents (GPS unit, RF-ID, etc.).
3) Ubiquitous interfaces controlling module:
It realizes optimization of services provided by using RUIS to various ubiquitous interfaces (devices). It will collect information about status, reachability, and device property of ubiquitous interfaces. Although there are various ubiquitous devices, they seem to be integrated, based on IP, using Mobile-IP technology. We have already standardized MobileIPv6-MIB [7], which enables universal information collection of ubiquitous interfaces using SNMP.
4) RUIS-AAA (authentication, authorization, accounting) module:
This is a server system to realize AAA when external applications/people access to and utilize RUIS.

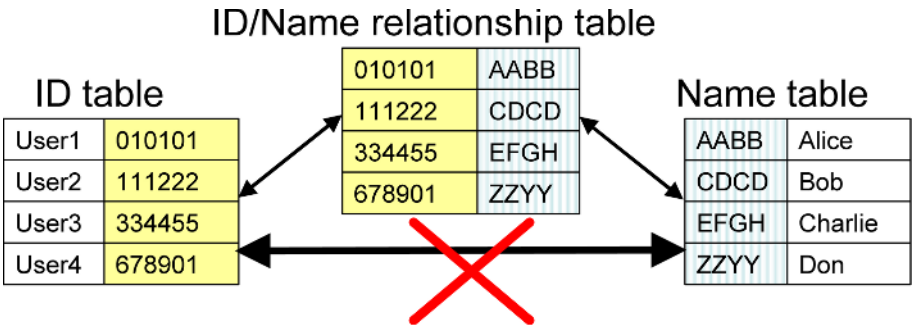


Fig. 2. Data structure for storing RUIS regional information/residents' activity model

3.2 Regional Information Storage

High security reservation is needed to store residents' personal/private information safely. Access from external network can be controlled by RUIS-AAA module as described in section 3.1. But in many cases leak of personal information occurs by the mistakes in internal operation. Therefore RUIS stores information in 'segmented' form. Data structure for storing information is shown in Fig.2. Each component of information is segmented, labeled with unique ID, and stored in each table. Relationships between unique IDs are separately stored in relationship table. Access permission to each relationship table is also separated with other information tables, and it will be controlled with its privacy importance. Separated authentications make it difficult to leak whole information at a time.

3.3 MobileIPv6-MIB

Controlling system of ubiquitous interfaces described in section 3.1 uses Mobile IPv6-MIB to collect information from various ubiquitous devices. MobileIPv6-MIB has a powerful feature of providing information about communication status of each ubiquitous device in the context of MobileIPv6. Communication status information is also useful as awareness information of residents. MobileIPv6-MIB also has mechanisms to control ubiquitous devices in the context of MobileIPv6. RUIS can control them to improve reachability to services using the MIB.

4 RUIS Application: Smart Support for Watch-over

4.1 Needs of Watch over Application

Recently many tragic incidents like kidnapping and fatally injuring the children is occurring in Japan. Japanese government adopts a serious stance against these crimes, and constructs "Regional Safe and Secure Information Network" [8]. Many telecommunication carriers in Japan start location notification service of children that utilizes feature of spatial localization equipped on mobile phone. However these services have the following drawbacks:

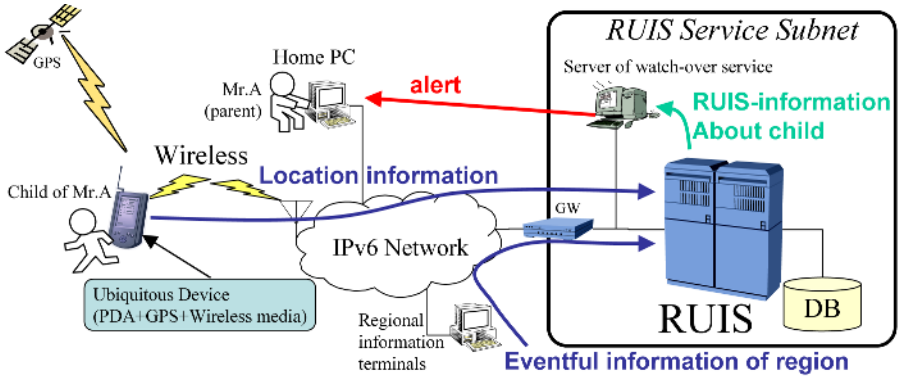


Fig. 3. Experimental system of smart support for childrenfs watch over

- Many useless notifications exist because the "contexts" of community residents are not considered
- Unilateral information provision from a system to users due to lack of interaction.
- Does not consider about the community

In order to overcome these drawbacks, we apply RUIS to a watch over application for children as a part of Symbiotic Computing Project [9] that we are promoting.

4.2 Smart Support for Watch over Based on RUIS

Smart support for watch over needs various personal information about the residents, just like, case history of past incidents, situation of residents, and so on. Generally these types of information should be treated very carefully considering protection of privacy. Therefore we try to apply RUIS for supporting watch over for children.

In general, watch over for children will alert when the behavior of child is different from the normal activity. However, by considering regional public event such as a festival, child's friendships, or context information, many alerts can be judged as "no problem". In this case, the child may go to the public event held in the region, or may visit a friend's home. Contexts related to the situation of the child will also help judgments. For example when Mr. B, a trusted friend of Mr. A, is accompanying Mr. A's child, even if the child visits the place he does not go normally, Mr. A does not need to worry about his child. 'Smart' judgment such as this example can be enabled by applying RUIS.

4.3 Experimental System

Fig.3 shows the overview of the experimental system that realizes smart support for watch over based on RUIS. The child who is watched over has a ubiquitous

Fig. 4. Children carries such ubiquitous device (PDA+GPS+Wireless media) in this experiment

terminal that is consisting of PDA device, GPS function, and wireless communication function. Location information of the child is obtained from GPS and sent through wireless device to RUIS that is constructed inside the RUIS Service Subnet. Regional information terminals cast over the region store eventful information that occurs inside the region. In this experimental system, we generate experimental eventful information through the terminal. Server of watch-over service is also placed inside RUIS Service Subnet, and monitors child's activities using location information stored in RUIS. If it judges as emergency (for example, some abnormal activity and no credible person around him), it sends alerts to the home PC of that child's parents.

5 Conclusion

We propose RUIS (Regional Ubiquitous Information System) information infrastructure to assist community residents considering their context by storing regional information with high security about personal information. We apply RUIS to the application of smart supporting for watch-over for children. At present, we are implementing and examining this prototype system.

References

1. Shiratori, N.: Post modern distributed system - flexible computing -. IPSJ Magazine (in Japanese) **36** (1995) 811–814
2. Fujita, S., Sugawara, K., Kinoshita, T., Shiratori, N.: An approach to developing human-agent symbiotic space. In: 2nd Joint Conference on Knowledge-Based Software. (1996) 11–18
3. Sugawara, K., Kinoshita, T., Shiratori, N.: Flexible network and human-agent symbiotic space. In: IPSJ DPS Technical Report. (1996) 109–114
4. Shiratori, N., Kinoshita, T., Sugawara, K.: Towards human-agent symbiotic space: Post-modern distributed system. IEICE Magazine (in Japanese) **80** (1997) 165–168
5. Shiratori, N.: Symbiotic computing for ubiquitous information society. In: Keynote speech at 2005 NEC TECHNOLOGY FORUM. (2005) 14–15
6. Suganuma, T., Uchiya, T., Konno, S., Kitagata, G., Hara, H., Fujita, S., Kinoshita, T., Sugawara, K., Shiratori, N.: Bridging the e-gaps: Towards post-ubiquitous computing. In: 20th International Conference on Advanced Information Networking and Applications. Volume 2. (2006) 780–784
7. Symbiotic Computing Project. (2005) [Online].Available: http://symbiotic.agent-town.com/.
8. M.Keeni, G., Koide, K., Nagami, K., Gundavelli, S.: Mobile ipv6 management information base (2006) [Online].Available: RFC4295, http://www.rfc-editor.org/rfc/rfc4295.txt.
9. Report on case studies about regional safety and secure information network (in Japanese). (2005) [Online].Available: http://www.lasdec.nippon-net.ne.jp/rdd/anshin/houkokukai/houkokukai_top.html.

Adaptive Service Delivery for Mobile Users in Ubiquitous Computing Environments*

Yong Zhang, Shensheng Zhang, and Hongxia Tong

Department of Computer Science and Engineering, Shanghai Jiaotong University
No. 800, Dong Chuan Road, Shanghai, 200240, China
{zycs926, sszhang, thx781212}@sjtu.edu.cn

Abstract. As an emerging computing model, Ubiquitous Computing (UbiComp) has become a source of challenging research. In this paper, we investigate the impact of mobility and resource-limitedness of UbiComp upon service provision, and design an adaptive service delivery model which can dynamically deliver satisfying services to mobile users in tune with the variation of context. In our design, we propose a user-oriented Quality of Service (QoS) model to measure whether services are suitable for being delivered to the mobile users in UbiComp environments. Moreover, we utilize first-order logic inference and fuzzy logic evaluation in the process of service delivery. We also perform several experiments to evaluate the service delivery model in terms of scalability and performance.

1 Introduction

The widespread deployment of wireless networks and portable devices is pushing computing toward the era of Ubiquitous Computing (UbiComp). In Ubicomp, a mobile user is the center of computing, and services need to be delivered to the user adaptively according to of the variation of context. Suppose a user with a smart phone enters an exhibition which provides various information services from simple text hint to complex multimedia introduction. After a brief electric registration through the smart phone, the user can receive a service list in which the services are available and ordered by the value of Quality of Service (QoS). The list is updated dynamically according to the status of current environment. Moreover, the user can click a service item, download the service proxy and invoke the service. This is a practical UbiComp scenario where the service infrastructure can provide right services to right users in right place at right time [1].

However, the implementation of the scenario is still a difficult task due to the lack of effective service middleware. Adaptive Service Provision Framework (ASPF) for UbiComp is one of the most important parts in the joint project between Shanghai Jiaotong University and Contec Innovation Inc., Canada [2]. The long-term objective of our research is to build an adaptive service middleware by which services, mobile users and diverse devices can be integrated seamlessly to facilitate the development of

* This work is funded by Shanghai Commission of Science and Technology and National Research Council of Canada / International Cooperation Project (05SN07114).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 209 – 218, 2006.

UbiComp applications. In this paper, we investigate the impact upon service delivery exerted by the mobility and resource-limitedness of Ubicomp, and develop an adaptive service delivery model which can dynamically deliver satisfying services to mobile users in tune with the variation of context. In our design, a service is implemented as the composition of service components, which is defined in a profile and published by service providers [3]. Especially, the same type of services with different compositions of service components is looked as different services (e.g. text navigation and graphics navigation.). The dynamic service configuration is beyond the scope of this paper.

The rest of this paper is organized as follows: Section 2 surveys related work. Section 3 describes the key components of service delivery model. In section 4, we propose a user-oriented QoS model with hierarchical structure and introduce the computation of QoS value. Section 5 discusses data provision and decision process in service delivery. Section 6 evaluates the service delivery model through the experiments in current implementation. Finally, we come to the conclusion of this paper.

2 Related Work

In this section, we discuss the related work in the area of QoS model and service provision. Recently, QoS model has been widely discussed in multimedia and Web-based applications by numerous researchers [4, 5]. Liu et al. proposed an extended QoS model which contained such domain-specific criteria as transaction, compensation and penalty rates for business model [6]. Maximlien et al. developed a dynamic service selection via an agent framework coupled with QoS ontology [7]. In the ontology, they defined several quality aspects derived from distributed system (e.g. interoperability, stability and integrity, etc.). These QoS models are not applicable for UbiComp. In our QoS model, the quality criteria are used to measure whether the services are suitable for being delivered to the mobile and resource-constrained environments, and whether the services are satisfying to the mobile users. The QoS value acts as metric to service delivery in UbiComp environments.

Several research efforts have addressed the general issue of service middleware to support adaptive service provision in mobile environments. To facilitate the development of context-aware applications, Aura aimed at the seamless integration of the blocks involving wireless communication, handheld computer, and smart phone [8]. Odyssey supported resource monitoring and application-aware adaptation [9], and Coda provided support for nomadic and bandwidth-adaptive file access [10]. Our work takes a similar view to the adaptability applied to services. We are interested in the adaptability in delivering appropriate services to mobiles users.

To support mobile users, HPL's CoolTown project developed a service model based on the convergence of Web technology, wireless networks, and portable devices [11]. CoolTown addressed location dependency and connectivity, and attempted to bridge physical and virtual worlds, whereas we address limitedness-resources and mobility, and emphasize QoS evaluation and service delivery. To develop context-aware mobile applications, Chan et al. designed environment monitors for MobiPADS to detect the change of service-specific context and implement dynamic service reconfiguration [12]. The system manipulated the configuration of service chains

based on Boolean logic decision, but ignored the fuzzy characteristic of context. Combining FOL inference and fuzzy logic evaluation in processing context, our approach indicates more flexibility in practical applications.

Dejan et al. presented the vision of a pervasive services infrastructure so as to offer desktop services to mobile environment [13]. In particular, they addressed the two technologies required to achieve the vision: adapting services to execute on resource-constrained devices and installing services on demand. They also presented preliminary results indicating the benefits of offloading and downloading services. In [14], Fouial et al. introduced a generic platform for value-added service provision for mobile environments. They identified service adaptation to user profile, preference and network capability, etc. These systems are similar to ours. However, they only focused on the generic functionalities and the architecture of service infrastructure. Inspired by their work, we firstly identify and classify context which influences the service delivery, then we design an adaptive service delivery model based on a user-oriented QoS model and a dynamic decision engine so as to implement practical UbiComp applications.

3 Components of Service Delivery

To make it clear in this paper, we define each pair of a mobile user and his/her portable device as a Mobile Peer (MP), denoted as (user, device), which is identified by MPID (userId and deviceId). Here, userId and deviceId are used to identify the mobile user and his/her device respectively.

Our work is based on the ASPF. In this section, we briefly introduce several components in ASPF which compose the service delivery model. The interaction of these components demonstrates the process of service delivery (see Fig.1).The components are described as follows:

1. Mobile components. Every MP is associated with a software component, named MP Client (MPC), running on the device, which acts as a launcher of service request, or a receiver of the result from decision engine, or even a trigger of services with user interface;
2. Context components. Context collectors obtain the context from physical sensors (e.g. RFID readers) and documents (e.g. service and device profiles). The context interpreters produce abstracted context through applying specific processing logics to the raw context, e.g. mapping the signal of RFID reader to a room number;
3. Service components. Besides the common components for registering, publishing, and discovering services, we design the decision engine based on a hierarchical QoS model (see Sect. 4). The decision engine can apply different evaluating methods to computing the QoS values of services depending on the service profiles and current context (e.g. device capability, network status and location of the user, etc.). We also design the service broker that serves as a surrogate between MPC and service providers. The service broker can rank the QoS values of services and select the satisfying services with high QoS values for each user.

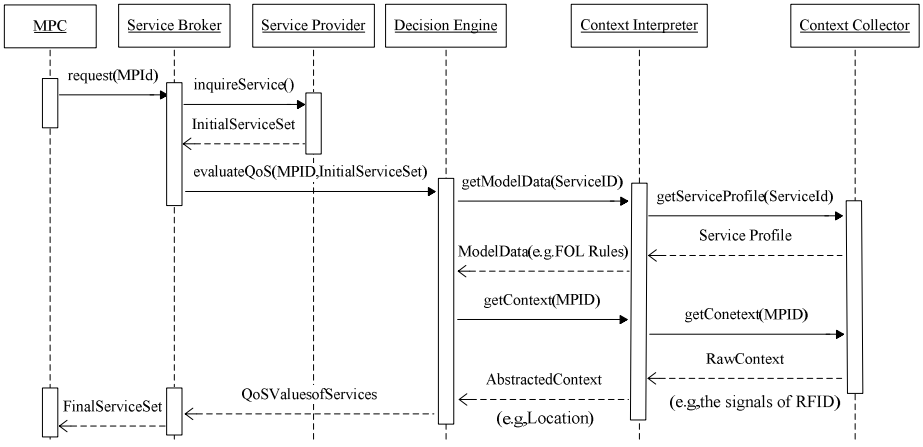


Fig. 1. The process of service delivery

4 QoS Evaluation

4.1 QoS Model

The quality criteria, derived from telecommunication and Web-based applications, mainly involve reliability, interoperability and robustness, etc. However, these quality criteria are not applicable for UbiComp scenario in which the user is the core of computation. QoS model should focus on quality criteria associated with the user's context in people's perception. We explore the definition of QoS and propose a user-oriented concept of measurement for service delivery in UbiComp, named as Degree of Service Satisfaction (DSS). DSS is an indicator which instructs the service to be delivered to satisfy the mobile user. Most importantly, the value of DSS is the metric to QoS evaluation in our approach. The DSS model is described as follows:

1. DSS is quantified and measured by a real number between 0 and 1. The higher value of DSS means the higher degree of the service suitable for being delivered.
2. To achieve scalability and flexibility, the DSS model is designed with a hierarchical structure. An example of the DSS model of our exhibition application is shown in Fig. 2. The model comprises such three sub-criteria as usability, device capability and network status. Furthermore, every sub-criterion contains leaf-criteria which involve collected context (e.g. memory, bandwidth, location and color depth, etc).

4.2 Computation of QoS Value

We identify two types of context which influence service delivery in different styles. We need to process them with different methods in computing DSS value:

1. *Boolean Type Context* (BTC). BTC refers to the type of context that does satisfy the service's requirement, or doesn't. For example, assuming that a print service

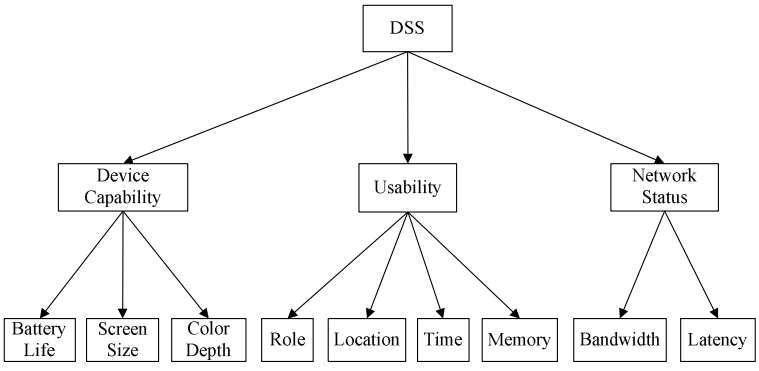


Fig. 2. Hierarchical DSS model

will be usable if a MP locates in an area, the location of the MP is a BTC. The case can be represented as:

folInfer(location of the MP, location specification of print service)→{0,1},

where, 1 indicates that the MP's location satisfies location specification in the service profile, 0 otherwise. DSS can be calculated by applying First-Order Logic (FOL) inference function *folInfer*() to determining the usability of the service.

2 *Fuzzy Type Context* (FTC). In people's perception, most kinds of context are fuzzy rather than crisp [15]. Given that a navigation service specifies bandwidth as 50kbps, considering the unstable characteristic, it does not mean that the service is unusable when current bandwidth is 45kbps, but we can say the bandwidth proposes less satisfactory service to the user. It is the same case in battery life and latency etc. In this regard, the case can be represented as:

fuzzyEvaluate(the fuzzy predicates of navigation service,
the set of FTCs in the service)→ ,

where, *fuzzyEvaluate*() is a function that executes fuzzy logic evaluation on all FTCs, ∈ [0,1]. The fuzzy predicates are the membership functions applied to the evaluation. In fact, the fuzzy predicates are organized as a fuzzy evaluation tree. Hence, computing the DSS value of a service requires hierarchical fuzzy evaluation on multiple grade levels (see Sect. 5). In addition, we adopt multiple methods to quantify FTCs to facilitate the calculation of DSS value. The detailed discussion of these quantitative methods can be seen in [16].

5 Service Delivery

5.1 Data Provision

Service providers utilize XML to maintain a service profile that describes the service's requirement of quality criteria associated with the context. Fig. 3(a) shows the partial listing of a service profile. Relational operators (e.g. "AND", "OR" and

"NOT") for requirement composition enable service providers to construct complex requirements associated with BTC. For example, as for a *btContext* of the service, "Usability" can be specified as: *Location*="*room* 321" .*and. Role*="*everyone*".

In initial phase, the service delivery model utilizes context interpreters to process all service profiles for QoS evaluation. Every *btContext* will be translated into a FOL rule as shown in Fig 3(b).The context interpreter can also produce a hierarchical fuzzy evaluation tree: every *ftContext* is mapped to a sub-node, while every FTC is mapped to a leaf-node (see Fig. 3(c)). Additionally, the context interpreter parses every FTC and produces a series of fuzzy predicates which are used to evaluate FTC on multiple grade levels. For example, suppose the FTC is specified as "Battery Life>50%" and there are three grade levels: "dissatisfactory", "satisfactory" and "very satisfactory", the corresponding fuzzy predicates may be *sigmf*(x, [0.2, 35]), *gaussmf*(x, [20, 50]) and *sigmf*(x, [0.2, 65]). The fuzzy predicates can be retrieved according to the name of node in the evaluation tree. In general, these fuzzy predicates derive from the experiences in practical applications.

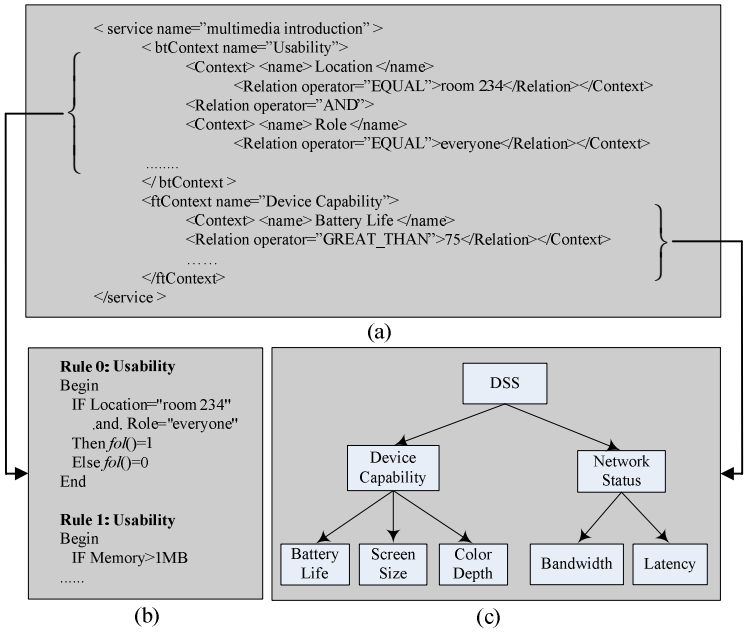


Fig. 3. Parsing service profile for service delivery: (a) service profile specified with XML, (b) FOL rules, (c) fuzzy evaluation tree

5.2 Process of Service Delivery

To put it more clearly, we define the initial service set (*ISS*) and the set of MP (*MPS*): $MPS = \{MP_1, MP_2, ..., MP_u\}$, $ISS = \{S_1, S_2, ..., S_v\}$. For a MP denoted as $MP_i \in MPS$, the process of service delivery includes two steps. We formalize the process as:

Step 1: Computing the DSS values of services by applying FOL inference to processing the BTCs of the MP_i, and obtaining its refined service set (*RSS*) as:

$$RSS_{MPi} = \{ S_j \in ISS \mid DSS_{BTC}(S_j) = 1 \} \text{ and } DSS_{BTC}(S_j) = \prod_{p \in P_i, r \in R_j} folInfer(p, r),$$

where P_i represents the set of FOL predicates translated from the delivered BTC of MP_i. $R_j = \Gamma(S_j)$ and $\Gamma : S \rightarrow Rule(S)$. Given a service name, the semantic function Γ can compute the set of FOL rules associated with the service. A service belongs to the RSS_{MPi} if and only if all the BTCs associated with the service completely satisfies the requirements specified by *btContext* in the profile.

Step 2: Calculating the final service set (*FSS*) by fuzzy evaluation on the FTCs of the MP_i:

$$FSS_{MPi} = \{ S_j \in RSS_{MPi} \mid DSS_{FTC}(S_j) \geq \lambda \} \text{ and}$$
$$DSS_{FTC}(S_j) = fuzzyEvaluate(FP_j, F_i),$$

where FP_j denotes the fuzzy evaluation tree derived from the profile of the *j*th service. F_i denotes the delivered FTC set of MP_i. $\lambda \in (0,1)$ denotes a threshold value of DSS for getting the fuzzy cut set of the RSS_{MPi}. In general, λ can be obtained from experiences in the practical applications.

In the initial phase, the decision engine needs to carry out two steps to compute FSS_{MPi} for MP_i. In the runtime phase, the decision engine only applies fuzzy evaluation to computing FSS_{MPi}, if all the delivered context is FTC. But if the context is mixed with BTC and FTC, the decision engine will reproduce RSS_{MPi} by using *folInfer*(), and then compute FSS_{MPi} through *fuzzyEvaluate*().

The function *fuzzyEvaluate*() applies hierarchical fuzzy logic evaluation to calculating DSS value based on current FTCs and fuzzy predicates. It firstly calculates each sub-criterion's DSS value based on leaf-criteria which the sub-criterion contains, then calculates the DSS value of the service depending on the values of all sub-criteria. The post-order traversal algorithm is applied in *fuzzyEvaluate*(). The pseudocode of *fuzzyEvaluate*() is described as follows:

```
float fuzzyEvaluation(TreeNode currentNode, Set ftcSet)
{
constant  int Max_Level = 3; // Assuming 3 grade levels
// Assuming the weights of grade levels: 0.2, 0.4, 0.4
constant float weightGL[]={0.2, 0.4, 0.4};
float  retVal = 0; temp= 0;
TreeNode  child = currentNode.firstChild();
if (child == NULL) {
 // Comprehensive fuzzy evaluation on all grade levels
  ftContext ftc = ftcSet.getFTC(currentNode.name());
  for (int gLevel =0; gLevel ++; gLevel < Max_Level){
 // Getting the fuzzy predicate of the grade level
    Function MF = currentNode.getPredicate(gLevel);
    temp = MF(ftc.value()) * weightGL[gLevel];
    retVal = retVal + temp ;}
```

```
    return  retVal;}
  else{
    while (child != NULL) {
      // Parent's DSS value is the sum of all children's
      temp= fuzzyEvaluate(child,ftcSet) * child.weight();
      retVal = retVal + temp;
      child = child.sibling();}
    return  retVal; }}
```

6 Experiments for Evaluation

The service delivery model has been implemented using jdk 1.4.1. We perform experiments on a COMPAQ ML350, equipped with one Intel Pentium III process, rated at 600MHz, 256MB RAM, and the operating system is Microsoft Windows 2000 Advance Server and the Java Virtual Machine version is 1.4.1. A PC machine installed Nokia Developer's Suit 3.0 for J2ME [17] enables us to emulate various Nokia's terminals as MPCs. A machine runs NIST net [18] as a router, which can emulate complex status of wireless network connection between MPC and decision engine. The overhead is the sum of the time taken by network delay, QoS evaluation and decision engine, etc.

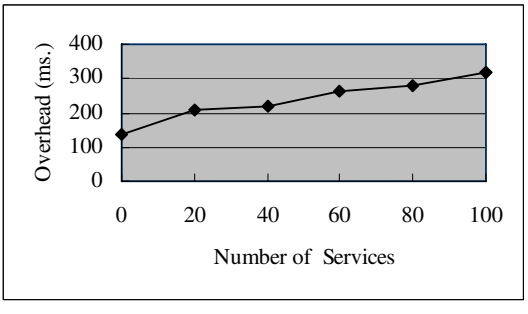


Fig. 4. Impact of services delivered for one MP

In order to estimate the scalability and performance of service delivery model in terms of the number of MPs and the number of delivery services, we develop a benchmark and conduct experiments by taking the following steps: we first estimate the basic overhead while only one MP is in the system, then investigate the variation of overhead under the circumstance of multiple MPs. In the experiments, we assume that necessary contexts have been prepared in advance. With NIST, we set Round Trip Time (RTT) to 100ms as the delay between MPC to service broker, and set bandwidth to 500kbps to test the model.

Fig. 4 illustrates the minimal overhead of the model where there is only one MP, three grade levels, and each service contains one BTC and four FTCs. As shown in Fig. 4, the overhead is basically linear in the number of services, and the increment of the overhead is minor.

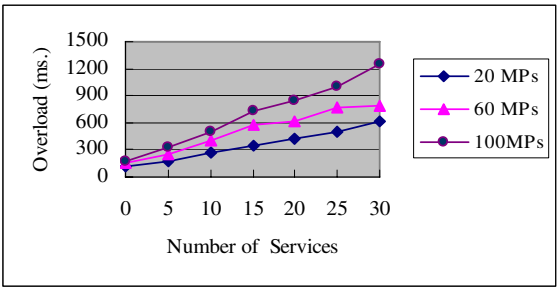


Fig. 5. Impact of services delivered for multiple MPs

We emulate the process of service delivery for multiple MPs by launching service requests from multiple MPCs. Fixing the number of BTC to 5 and the number of FTC to 10 in each service, we obtain the result of experiment shown in Fig. 5. The overhead does not increase greatly due to system adopts multiple threads to deal with multiple MPs in parallel, thus only the largest overhead of service delivery will be counted. The picture illustrates that the overhead is more or less linear in the number of services when the number of MPs is fixed. However, we find that the number of MPs has an impact on the slope of the curve. For example, the slope is 16.57 approximately when the number of MPs is 20, the slope rises to 39.03 while the number of MPs reaches 100. We believe that the rapid rise is mainly introduced by the increment of overhead of per thread. According to our experiment with the exhibition application, we fix the maximal number of MPs to 100 which has never been reached. In the future application, the scale limit will be removed by adding another processor to reduce the overhead of per thread.

7 Conclusion

In this paper, we design an adaptive service delivery model for UbiComp applications, which can dynamically deliver satisfying services to mobile users according to the variation of context. In our design, the user-oriented QoS model covers various quality criteria which are used to evaluate the impact of mobility and resource-limitedness of Ubicomp upon service delivery. Especially, the decision engine utilizes FOL inference and fuzzy logic evaluation to calculate the QoS values of services based on the classification of context. The service broker can deliver the services with high QoS values to mobile users so as to achieve high degree of service satisfaction. Experimental result has proved that our service delivery model is of better scalability and performance.

References

1. Roussos, G., Marsh, A. J., and Maglavera, S.: Enabling pervasive computing with smart phones. IEEE Pervasive Computing, Vol. 4, (2005) 20-27
2. Adaptive Service Provision Framework. http://hornet.sjtu.edu.cn

3. De Moor, A. and Van Den Heuvel, W.-J.: Web Service Selection in Virtual Communities. Proceedings of the Hawaii International Conference on System Sciences,Vol. 37. Big Island, HI., United States, (2004) 3105-3114
4. Menasce, D. A.: QoS Issues in Web Services. IEEE Internet Computing, Vol. 6, (2002)
5. Chalmers, D. and Sloman, M.: A Survey of Quality of Service in Mobile Computing Environments. IEEE Communications Surveys, (Second Quarter, 1999) 2-10
6. Liu, Y., Ngu, A. H. H., and Zeng, L.: QoS computation and policing in dynamic web service selection. 13th International World Wide Web Conference Proceedings, WWW2004. New York, NY, United States, (2004) 798-805
7. Maximilien, E. M. and Singh, M. P.: A framework and ontology for dynamic web services selection. IEEE Internet Computing, Vol. 8, (2004) 84-93
8. Garlan, D., Siewiorek, D. P., Smailagic, A., and Steenkiste, P.: Project Aura: Toward Distraction-Free Pervasive Computing. IEEE Pervasive Computing, Vol. 1, (2002) 22-31
9. Noble, B. D., Satyanarayanan, M., et al.: Agile Application-Aware Adaptation for Mobility. Proceedings of the 16th ACM SOSP, (1997) 276 - 287
10. Mummert, L. B., Ebling, M. R., et al.: Exploiting Weak Connectivity for Mobile File Access. Proceedings of the 15th ACM SOSP, (1995) 143-155
11. Kindberg, T., Barton, J., Morgan, J., et al.: People, Places, Things: Web Presence for the Real World. Proceedings of the third WMCSA,Vol. 7, (2002) 365 - 376
12. Chan, A. T. S. and Chuang, S.-N.: MobiPADS: A Reflective Middleware for Context-Aware Mobile Computing. IEEE Transactions on Software Engineering, Vol. 29, (2003) 1072-1085
13. Milojicic, D., Messer, A., Bernadat, P., Greenberg, I., Fu, G., Spinczyk, O., Beuche, D., and Schroder-Preikschat, W.: ψ - Pervasive services infrastructure. Technologies for E-Services, Second International Workshop, TES 2001, LNCS, (2001) 187-200
14. Fouial, O., Fadel, K. A., and Demeure, I.: Adaptive service provision in mobile computing environments. Proceedings of the 4th IEEE International Conference on Mobile Wireless Communication Networks (2002)
15. Mantyjarvi, J. and Seppanen, T.: Adapting applications in handheld devices using fuzzy context information. Interacting with Computers, Vol. 15, (2003) 521-538
16. Lum, W. Y. and Lau, F. C. M.: User-Centric Content Negotiation for Effective Adaptation Service in Mobile Computing. IEEE Transactions on Software Engineering, Vol. 29, (2003) 1100-1111
17. Nokia Developer's Suite for J2ME™. http://www.forum.nokia.com
18. National Institution of Standards and Technology, NIST Net Home Page. http://snad.ncsl.nist.gov/itg/nistnet

An Effective Message Flooding Method for Vehicle Safety Communication

Sukdea Yu[1] and Gihwan Cho[2]

[1] Div. of Electronic & Information Engineering, University of Chonbuk, 664-14 Duckjin-Dong, Duckjin-Gu, Jeonju, Chonbuk 561-756 South Korea
sdyu@chonbuk.ac.kr
[2] Div. of Electronic & Information Engineering, Research Center for Advanced LBS Technology, University of Chonbuk, 664-14 Duckjin-Dong, Duckjin-Gu, Jeonju, Chonbuk 561-756 South Korea
ghcho@dcs.chonbuk.ac.kr

Abstract. An intelligent vehicle safety system can be constructed by exchanging emergency-related information between any unrelated vehicles, such as urgency stop, traffic accident, and obstacles. In the most of vehicle safety communication applications, an emergency message is propagated in the form of broadcasting. However, it causes a lot of problems in terms of efficiency due to the multi-hop propagation and radio collision problems. This paper presents a selective message forwarding method by proposing the stem and branch structure. With benefit of the structure, only one vehicle performs forwarding a received emergency event among the vehicles that are included in the same wireless coverage. Moreover, the proposed scheme improves the efficiency of message transmission with the selective assignment of priority for forwarding message. To analyze its performance, it has been evaluated by network simulator.

1 Introduction

Running vehicles are threatening passenger's life by road status, trouble of vehicle, traffic accident and so on. Specially, simple hazardous factors become large traffic accidents in situation when drivers cannot notice those factors or cannot get enough time for reacting in time. Many vehicular manufacturers put emphasis in passive safety systems development. The communication among moving the vehicles and its usability present an affirmative opinion in aspect of the necessity.

In emergency situations, a driver typically relies on the tail brake light of the car immediately ahead to decide his or her own braking action. Under typical road situations, this is not always the best collision avoidance strategy for various reasons. This is particularly true in many situations where vehicles need to have an extended range of awareness beyond what drivers can immediately see or autonomous safety systems can detect. In many case, driver's reaction time typically ranges from 0.7 to 1.5 second [1]. According to OFCOM's investigation report, if drivers have time composure during 1 second that can correspond beforehand in traffic danger element, speaks that 90% of backside collision accident can be reduced [2,3]. Chain collisions can potentially decrease, or their severity can be lightened, by reducing delay between the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 219 – 228, 2006.

occurrence time of emergency event and the time at which the vehicles behind are informed about it [4].

Broadcast service is important to all kinds of networks. Whether new message needs to be sent to all participants across the network or the destination location is unknown, the message broadcastings are necessary [5]. Even if vehicles run on the same road, the broadcastings are necessary in the vehicle safety communication, because those vehicles do not have a pre-relationship each other. Simple broadcasting is a propagation method for one hop distance. So, vehicle safety communication uses a successive broadcasting (that is flooding) method to extend information transmission coverage. But simple flooding method drops dramatically bandwidth efficiency and outbreaks frequent message collisions, when the density of vehicles is high [5]. Therefore, many of inter-vehicle communication methods tried to overcome these problems, by using directional forwarding or selective re-forwarding method [1, 5-7].

The proposed scheme is a kind of the selective re-forwarding. It makes use of position of vehicle for propagating emergency messages. If a certain vehicle creates and broadcasts an emergency message, then only one vehicle rebroadcasts the message among vehicles that receive the message at the same time. An ideal position of the next forwarder is proposed for selecting next forwarder. The position is called DP (Designated Position). Vehicles that receive an emergent message defer re-forwarding for the proportional delay time that is obtained by the direct distance from current its position to the DP. A selected vehicle which has the shortest delay re-forwards the message and remains discard the procedure of re-forwarding. As a result, only one vehicle relays the emergency event among vehicles that are included in the same wireless coverage; so, broadcasting overhead for vehicle safety is extremely reduced.

The remainder of this paper is organized as follows. At first, it presents two related works that are related with selective rebroadcasting; the third section contains the detail procedure and protocol steps; the fourth section presents the simulation model, results and our analysis; the last section is the conclusion and further research.

2 Related Works

Broadcasting methods for vehicle service classifies two categories; one is the regional broadcasting, other is the directional broadcasting. But the directional broadcasting is more useful for vehicle safety communication, because vehicles that should receive the emergency message move to same direction. Representative directional broadcasting methods are NB (Naïve Broadcast) forwarding [9] and IB (Intelligent Broadcast) forwarding [10]. The NB forwarding considers only message arrival direction, but the IB forwarding considers additional factors such as implicit acknowledgment, message history and random duration for rebroadcasting.

The NB forwarding method serves as a baseline packet-routing mechanism for the target cooperative collision avoidance application. As soon as detecting an emergency event, the detecting vehicle sends CW (Collision Warning) message periodically at regular intervals [9]. Upon receiving a message, a vehicle decides whether to actuate related devices and start generating its own CW messages or ignore the message.

Vehicles ignore the message if it comes from back with respect to its direction of movement. Figure 1 describes NB forwarding procedure. If vehicle *B* detects an emergency event, it creates a CW message, and broadcasts it. Vehicle *A* and *C* receive the message. But vehicle *A* ignores the message, because it is arrived from back. Upon receiving the message, vehicle *C* actuates deceleration and rebroadcast the message.

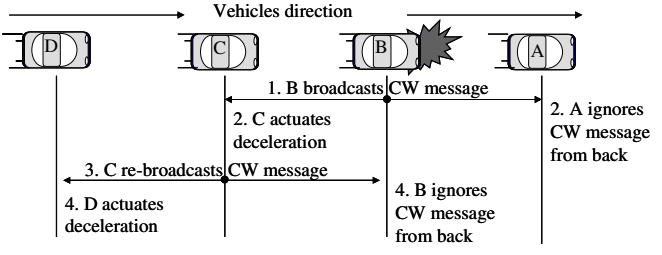


Fig. 1. The NB forwarding protocol sequence

The NB leads an excessive message forwarding which escalates message collisions for 802.11 MAC [10]. To avoid these, the IB forwarding uses implicit acknowledgment based message generation and transmission strategy. With this scheme, the protocol can improve the system performances by reducing the number of messages that are injected within a range for a given vehicle emergency event.

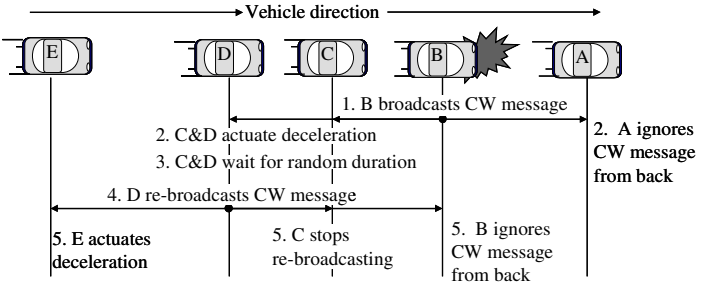


Fig. 2. The IB forwarding protocol sequence

After starting a broadcast, if a vehicle which detects the event receives the same message from back, it infers that at least one vehicle has received that message in the back and will be responsible for propagating it along the rest to the range. In Figure 2, as soon as vehicle *C* and *D* receive a CW message from *B*, they respectively wait random duration. If vehicle *D* has short duration, then vehicle *D* rebroadcasts the message. Because vehicle *C* receive same message from back, vehicle *C* stops rebroadcasting process.

3 Stem and Branch Flooding

3.1 Basic Structure

Most traditional protocols which make use of location information exchange a vehicles' current location periodically with neighbor nodes. The proposed protocol forwards an emergency message without location exchanging with neighbors. Each participant simply compares its own location with the stamped location in the received message, and starts the rebroadcasting process. But only one node can rebroadcasts the message among the receiving nodes. Remains give up the rebroadcasting process, because they receive duplicated message. Selected nodes become stem node, and unselected nodes become the branch node. A stem node temporally role-plays branch nodes' representative in the wireless coverage. If selected stem nodes are connected each other in a line, this becomes the SNB (Stem aNd Branch) structure.

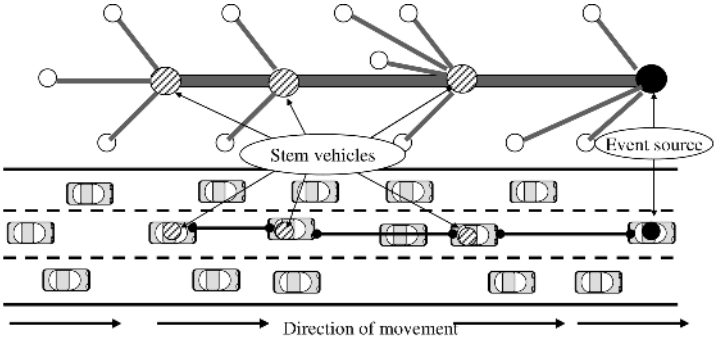


Fig. 3. The SNB structure and the mapping on road

Figure 3 shows the structure. Stem vehicles have a responsibility for forwarding to backward. Branch vehicles are potential candidates who become stem vehicles. But the other is selected as a current stem vehicle; the vehicles stop remained processes for forwarding the message. However, this structure is created whenever message is issued; it is not fixed structure as vehicles are moving around.

3.2 Stem Vehicle Selection

The DP is established with signal stability region, so that a stem vehicle may be unselected in the unstable boundary line or near the source vehicle. Normally, the strength of radio signal is decrescent being inverse proportion in distance square. The DP is a most suitable position for next forwarder to which a current vehicle considers. The DP is located on opposite direction that current vehicle moves and established between the current position and the radio boundary line on the stable signal region. The reason that the DP is established on more inside than radio boundary line is to reduce damages of messages maximally, and heighten message receptions. The stable signal region assigns about 150~200m in open field considering 802.11 Wireless LAN.

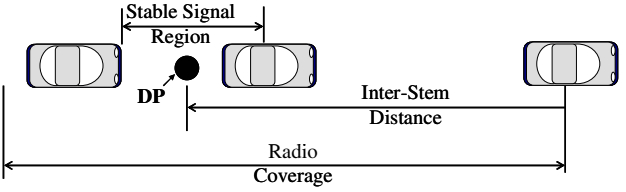


Fig. 4. DP and Inter-Stem Distance

Figure 4 shows the DP and the stable signal region. The inter-stem distance refers to the DP distance. The DP distance is not fixed; the DP's distance can be changed according to the situation of road. The DP distance is assigned longer to rural road and shorter to urban road respectively.

Vehicles which locate around the DP become the potential candidate of the stem vehicle. When consider particular situation of road - serialized node arrangement, same direction movements, similar speed - even if any vehicle around the DP becomes a stem vehicle, do not influence greatly in performance aspects. However, the nearest vehicle is more suitable as candidate of stem vehicle with the DP. Our work takes into account this feature. The competition problem which potentially can be happened is solved by the distance between the DP and the current vehicle position. A next stem vehicle has rebroadcasting delay time commensurately in distance with the DP. According to size of the delay time, each vehicle's priority is assigned naturally.

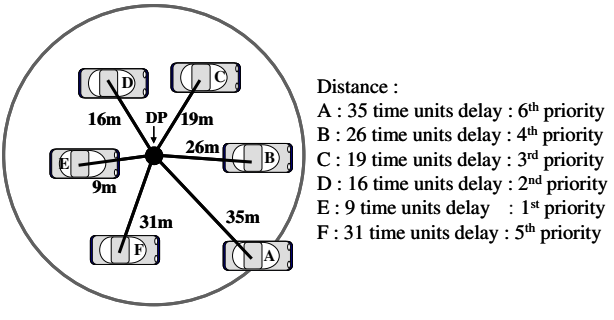


Fig. 5. Delay time assignment by the distance

Figure 5 describes the distance between the DP and the near vehicles. Each vehicle which receives the message calculates a DP with own linear distance. If it has done, each vehicle rebroadcasts the received message after the defer time as much as the distance. When another vehicle rebroadcasts the same message already, the others discontinue all processes for rebroadcasting the message and covert to reception stand-by status. If a vehicle which has the highest priority order does not attempt normally rebroadcasting the message because of its own problem, a vehicle which has next priority order rebroadcasts the message. Through this backup concept, the proposed scheme heightens transmission reliability. Repeat this rebroadcasting until the hop limitation.

3.3 SNB Flooding Protocol

The proposed flooding protocol is a kind of selective rebroadcasting method to propagate emergency-related information to the rear vehicles effectively. It is the most important function to select the vehicle which rebroadcasts the received message with a given conditions. Therefore, protocol description shows the procedural message reforwarding steps. It uses the inner product of two vehicle's directional vector for classifying vehicles on the same road [6].

Protocol steps:

1. Source vehicle does broadcasting including ID, event sequence number, event type, moving vector and DP.
2. If the message is come from other road or reverse lane, receiving vehicle ignores it.
3. When the message is duplicated, it is discarded. When the message is valid message, it is logged, and the rebroadcasting procedure is triggered.
4. The defer time and the DP are calculated for rebroadcasting.
5. The rebroadcasting is deferred for the defer time. If it does not receive any duplicated message during the defer time, the node broadcasts the message immediately.
6. Rebroadcasting is repeated to the hop limitation.

If a new message arrives at the current vehicle, the directional vector which is stamped in the message is compared with the current vehicle's directional vector. If the message was arrived from the vehicle on the same road, the protocol doesn't discard the message. Subsequently, the duplication of the message with message's source ID and message sequence number is checked. If the message is new, it is informed to the upper application. And then the TTL is checked. If the TTL is less than 1, the message doesn't need to rebroadcast. But the TTL is more than 1; the rebroadcasting procedure is triggered. At first, its defer time and the next DP are calculated. After the message was rebuilt, the node waits for the defer time. If the vehicle does not receive the same message for the defer time, the message is rebroadcasted. But the vehicle receives the same messages for the defer time, then it abandons the procedure of rebroadcasting that was scheduled by the fore message.

4 Simulation Results and Performance Analysis

4.1 Simulation Environment

Wireless network technologies (i.e. 5.9GHz DSRC [9]) for vehicle safety communication have the similar characteristic to 802.11 Wireless LAN transmissions by CSMA. Therefore, delayed transmission phenomenon can be emerged according as the number of packet increases. Also, bandwidth efficiency can be dropped because of repetition send-receive beyond necessity. The proposed protocol is an emergency message propagation protocol for vehicle safety communication, so the swiftness and stability of information transmission are very important.

The highway scenario was used for simulation. Every vehicle goes straight on the same road, and don't change direction during simulation time. But all vehicles depart at different position, and change speed as time goes periodically. The speed steps that each vehicle uses are 60Km/hr, 70Km/hr and 80Km/hr. So the topology of vehicles changes continuously. While simulation is gone, a certain vehicle belonging to fore-front group created and broadcasted emergency event occasionally.

The simulation has been implemented in NS-2 (version 2.29) [11]. The SNB protocol has been analyzed for its performance with the NB protocol [9] and the IB protocol [10]. The simulation measures the message transmission efficiency with message transmission ability in vehicular networks. The parameters used for simulation are listed in Table 1.

Table 1. Simulation parameters

Parameter	Value
Number of vehicles	10~100
Vehicle speed	60~80Km/hr
Simulation area	2500meter * 2500meter
Number of emergency event	10
Vehicle moving direction	All vehicles are same
MAC protocol	802.11 MAC
Emergency message size	200 bytes
Wireless coverage	250m
Emergency message's TTL	5
Simulation time	30 seconds

The simulation results have been analyzed in terms of following three performance criteria:

1. the number of sent packets : the total of sent packets
2. the number of received packets : the total of received packets
3. the ratio of effective packets : the percentage of effective packets in total received packets

4.2 Performance Evaluation

Figure 6 depicts the number of sent packets according to the number of vehicles respectively. The number of sent packet increases commensurately with number of vehicles in the NB. But, the number of sent packet is almost fixed regardless of the number of vehicles in the SNB and the IB. Because a random vehicle has been selected for rebroadcasting among the receiving message in the IB, it needs more packet transmissions to propagate to same distance as the SNB. The SNB can propagate the emergency message by half number of packets than the IB transmits.

Figure 7 and 8 show the number of received packets and the ratio of effective packet. Normally, if the number of vehicles increases, the number of received packets

increases exponentially. The effective packet means not duplicated packet. Once receiving new message, every vehicle discards the messages which are duplicated. Received packet is unnecessary except first message. Of course, once duplication is very important as implicit acknowledgment. So the ratio of effective packet may not pass over maximum 50% for reliability. As seen in Figure 7, all protocol increases the number of received packet according as the number of vehicles increases. However, if the number of received packet increases, the ratio of effective packets is apt to drop. The NB shows characteristic of typical flooding methods. But, the IB shows better efficiency than the NB, and the SNB is approximating to 50% that is an optimization ratio almost. The SNB is heightening the bandwidth efficiency by the packet that is less than the IB or the NB, and improves the availability of wireless resources maximum.

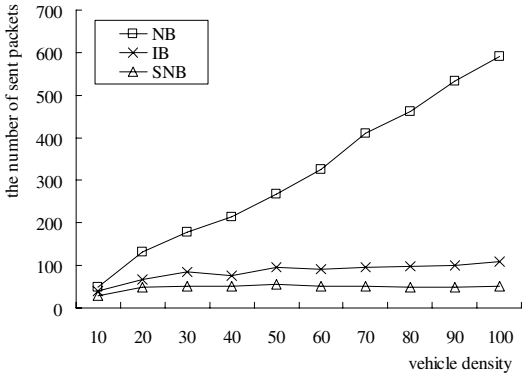


Fig. 6. The number of sent packets

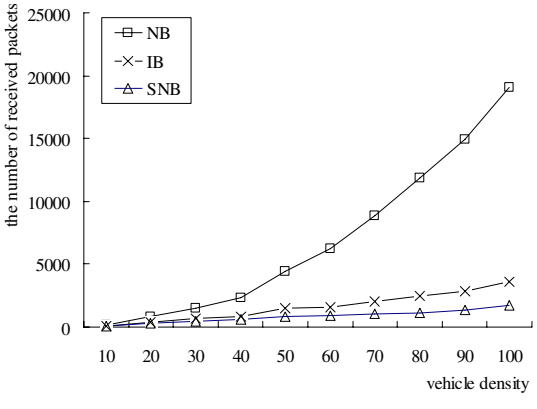


Fig. 7. The number of received packet

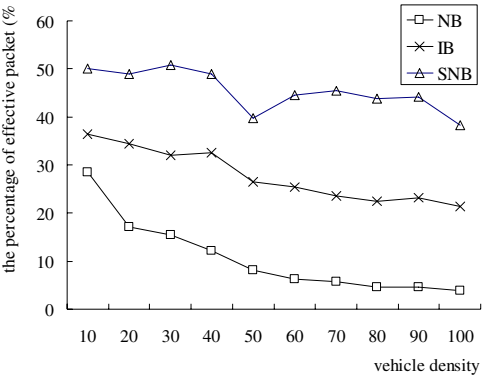


Fig. 8. The percentage of effective packets

4.3 Discussion

According to simulation results, the proposed SNB protocol is similar with the existent protocols in transmission performance aspects. But the number of packets that use for this performance is extremely less. In efficiency of aspect, the SNB protocol shows near optimal performance. Also, the protocol includes implicit acknowledgment function that is offered in the IB, the forwarding reliability highly is improved. When a vehicle which is assigned the highest priority does not rebroadcast in time because of some problems; the next priority vehicle naturally performs the rebroadcasting. This is a kind of backup function. In the contrast to other protocols, the SNB shows more good performance especially there is been many vehicles.

A partial result of measurements can be changed according to the DP distance variable. The DP distance has been assigned by 200m when implement this simulation code. The value is not optimized, but shows a comparative good performance.

5 Conclusion

The proposed method is a kind of selective re-forwarding with considering the vehicles' mobility. The method gives priority to each vehicle who receives packet, by the difference of its location. As soon as each vehicle receives a message, it can calculate a defer time that must wait for an opportunity for rebroadcasting. One vehicle which has the shortest deferred time rebroadcasts the message. Remains discard the procedure of rebroadcasting.

In the results of simulation, the proposed method is excellent than other methods for vehicle safety communication in the number of sent packet, the ratio of received packet. Specially, the number of transmission packets to propagate emergency information is kept almost changelessly to increase of the density of vehicles.

Future work includes an automatic the DP distance changing according to the traffic situation, analyses of the interrelation between the information reception time and the accident escaping rate. There is a scenario necessity to compose vehicles transfer pattern of the normal road with the highway movement.

References

1. Yang, X., Lui, J., Zhao, F., Vaidya, N.H.: A Vehicle-to-Vehicle Communication Protocol for Cooperative Collision Warning. Proc. Of 1st IEEE Int'l. Conf. on Mobile and Ubiquitous System Networking and Services (2004) 114-123
2. The Office of Communication in UK, http://www.ofcom.gov.uk/
3. Scherrer, D.,: Short Range Devices, Radio Frequency Identification Devices, Bluetooth, Ultra Wideband Systems, Automotive Short Range Radars, Overview and Latest Developments. OFCOM (2003)
4. Xu, Q., Sengupta, R., Jiang, D.: Design and Analysis of Highway Safety Communication Protocol in 5.9 GHz Dedicated Short-Range Communication Spectrum. Proc. of the 57th IEEE VTC (2003) 2451-2455
5. Sun. M.T., Feng, W., Lai, T.H., et al.: GPS-Based Message Broadcast for Adaptive Inter-Vehicle Communications. Proc. of the 52nd IEEE VTC (2000) 2685-2692
6. Fukuhara, T., Warabino, T.: Broadcast Methods for Inter-Vehicle Communications System. Proc. of IEEE Wireless Communications and Networking Conference (2005) 2252-2257
7. Torrent-Moreno, M., Jiang, D., Hartenstein, H.: Broadcast Reception Rates and Effects of Priority Access in 802.11-Based Vehicular Ad-Hoc Networks. Proc. of the 1st ACM Int'l Conf. on Vehicular Ad hoc Networks (2004) 10-18
8. Hasegawa, T., Mizui, K., Fujii, H., Seki, K.: A Concept Reference Model for Inter-Vehicle Communications (Report2). Proc. of IEEE ITS (2004) 810-815
9. ASTM Int'l: Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems - 5GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ASTM E2213-03 (2003)
10. Biswas, S., Tatchikou, R., Dion, F.: Vehicle-to-Vehicle Wireless Communication Protocols for Enhancing Highway Traffic Safety. IEEE Communications Magazine, Vol. 44, Issue 1, (2006) 74-82
11. Network Simulator Version 2, http://www.isi.edu/nsnam/ns/

RDF: Stores – A Lightweight Approach on Managing Shared Knowledge

Michael Schneider

German Research Center for Artificial Intelligence (DFKI),
66123 Saarbruecken, Germany
michael.schneider@dfki.de
http://www.dfki.de/~mschneid/

Abstract. One of the central ideas of ubiquitous computing research is to create smart real-world artifacts and environments that can spontaneously cooperate in order to enable novel applications that no single device could provide. A critical factor in such applications is the availability of relevant context knowledge, which may be provided by a variety of different sensors, domain models, and applications. Locating, accessing, and preserving such knowledge is not a trivial task, especially in dynamic environments where new knowledge sources and applications may unexpectedly appear while existing ones may vanish. In this paper we present a pragmatic yet powerful approach to manage shared knowledge by introducing a novel concept called RDF:Stores. We will present the general properties of RDF:Stores and describe how they can be implemented using off-the-shelf software components.

1 Introduction

One of the central ideas of ubiquitous computing research is to create smart real-world artifacts and environments that can spontaneously cooperate in order to enable novel applications that no single device could provide. A critical factor in such applications is the availability of relevant knowledge about the domain and the current state of the environment. This knowledge may be provided by a potentially huge number of heterogeneous sources: Sensor networks may monitor the state of the environment and the behavior of the user, domain models may provide general background knowledge, and applications may share user models or other relevant information. This knowledge may be accessed and consumed by a variety of different applications. Thereby, each application has its own focus and might only be interested in a small part of the globally available information.

Obviously, in an ad-hoc scenario as described above it is not possible to predict which concrete information is needed by whom and which source (if any) could provide the required information. Thus, some mechanism is needed to match knowledge sources and knowledge consumers on demand. Such a mechanism should provide a pull interface that allows applications to locate and access relevant knowledge as well as a push interface to notify interested applications whenever new information is available. Furthermore, such a mechanism should

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 229–239, 2006.

be able to perform caching of knowledge. Due to the dynamic nature of most ad-hoc scenarios, sources of important information may vanish from one moment to the next. In such a situation a cached copy of knowledge may be required to continue the operation of the remaining components. On the other hand, new knowledge consumers may appear unexpectedly. These consumers could have a justified interest in previously published knowledge that they where not able to collect at their own. Cached knowledge allows such applications to efficiently catch up on the current situation. Cached knowledge on the other hand may be the only source for historic and meanwhile outdated knowledge, which might no longer be shared by a knowledge source.

In this paper, we introduce the novel concept of RDF:Stores, which provides a simple yet flexible and powerful solution to the knowledge management problem described above. Special care was taken to keep the involved protocols and software components as simple and lean as possible, and to completely rely on established and well understood technology standards. Thus, the proposed concepts can be reliably and efficiently implemented using off-the-shelf software components. In the following we will give a general overview of the proposed framework. After discussing how knowledge is represented and shared, we introduce the concept of RDF:Stores as a broker between knowledge sources and consumers. Finally we describe our Java-based prototype implementation.

2 Knowledge Management Framework

In the proposed framework knowledge is represented and exchanged in form of minimalist RDF models. In its simplest form a model may describe just a single property of some part of the world at a given point in time, like the temperature of a certain room at a certain moment. Published on the web by the according knowledge source under a unique URL, these models may be accessed by interested applications via simple HTTP requests.

In order to allow knowledge consuming applications to locate relevant models, knowledge sources advertise the URLs of shared models to so-called RDF:Stores, which serve as a registry for semantic models. Applications may query these RDF:Stores and in response get a list of matching models, which they can download from the web at will. Furthermore, RDF:Stores provide a push service that notifies applications about newly available models and might implement caching strategies to keep knowledge from vanished sources.

Figure 1 (left) shows a simplified example of a typical use case of the framework described above. In the scenario a temperature sensor continuously monitors the temperature in a certain room. If a change in temperature is detected (1), an RDF model describing the new situation is created and made available via a web server (2). The URL of this new model is advertised to an RDF:Store (3). If an application asks the RDF:Store for a list of models containing statements about temperature (4), the URL of the previously published model is returned in a list together with other matching models. The application then fetches the full model from the web server (5).

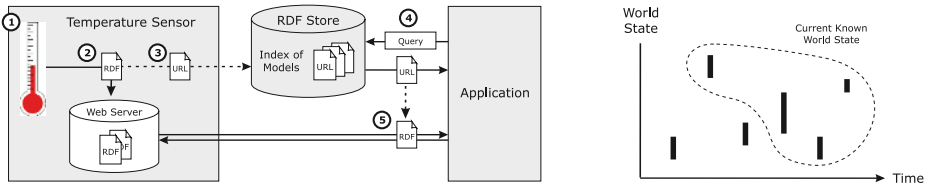


Fig. 1. Simplified use case of RDF:Stores (left part) and merging of partial knowledge models (black bars) to construct current world state (right part)

2.1 Knowledge Representation

The framework proposed in this paper is based on a sensor-centric knowledge model. This model assumes that each sensor's perception is limited to a very narrow part or aspect of the environment. Whenever a sensor discovers a change in state, it constructs a minimalist model that describes the new state of the changed property and publishes this model on the web under a unique URL. A model shared by a temperature sensor for instance could contain the only statement that the temperature in room C0.03 was 21 degree Celsius on March 16th, 2006 at 9 o'clock. Each model contains a timestamp that states when the knowledge was discovered[1]. In our model, also sources of conceptual or domain knowledge are seen as sensors[2].

Applications which require a more complete picture of the current situation may merge multiple individual models representing different aspects of the environment into a single, larger model. In the case of overlapping and contradicting partially models, applications have to resolve conflicts according to their trust into the reliability of the involved knowledge sources. To resolve conflicts on the application level has the big advantage, that it allows applications to apply their individual model of trust independently. Figure 1 (right) illustrates how partial models (black bars) provided by different sources may be merged into a more complex model of the current situation (dashed contour).

Although the proposed framework does not impose restrictions on the granularity of shared models, it is reasonable to separate independent statements into individual models. On the one hand, this minimizes the processing overhead (bandwidth, parsing, memory, and processing) for applications that are only interested in parts of the shared knowledge, and on the other hand reduces the danger of conflicts between models from different sources that describe similar properties of the environment. As description framework for the representation of knowledge we have chosen RDF [14], as it is the foundation of most of today's semantic web languages. A prominent example of such an RDF-based language is the Web Ontology Language (OWL) [15]. Nevertheless, most of the ideas described in this paper are general and not limited to RDF as description language.

[1] We assume that knowledge sources have access to a global clock.

[2] A user model for instance can be seen as a virtual sensor that provides knowledge about the user's current attitudes and preferences.

2.2 Publishing Knowledge

As mentioned earlier, each knowledge source has to publish its models on the web under a unique URL, either by using an embedded web server or by relying on extern web servers. We have discovered three reasonable approaches: Internal storage, external storage, and an approach called "semantic URLs".

Internal storage (Figure 2.i) means, that a knowledge source distributes all its models through an embedded web server. Publishing a new model this way is convenient for the knowledge source, but on the downside running a web server may consume potentially valuable resources. *External storage* (Figure 2.ii) means, that a knowledge source creates its models locally and then uploads them to an external web server. This frees the knowledge source from running a web server, but requires more effort to publish new models. *"Semantic URLs"* (Figure 2.iii) is a technique that can be applied whenever the generated models contain only a small amount of variable information. In our temperature example the only parts of the model that ever change are the timestamp and the temperature value. Thus, we could construct a URL of the form: `http://mysensor.net/<timestamp>_<temperature>.rdf` A script on the server `mysensor.net` then could extract timestamp and temperature from the URL upon request and insert both into a temperature model template. This way, every possible temperature model can be created on the fly. The according template and script can be located on any server in the world and no communication is necessary between the knowledge source and the server to upload the models. Thus, "semantic URLs" are especially well suited for applications in extremely resource constrained setups like sensor networks or mobile devices.

Knowledge sources must never change published models, as this could cause severe inconsistencies. Instead, if new or changed knowledge is discovered, a new model has to be created and published under a vacant URL. Knowledge sources may remove outdated models to free resources, but should try their best to keep and publish models as long as possible, as they may be still used or requested by some application.

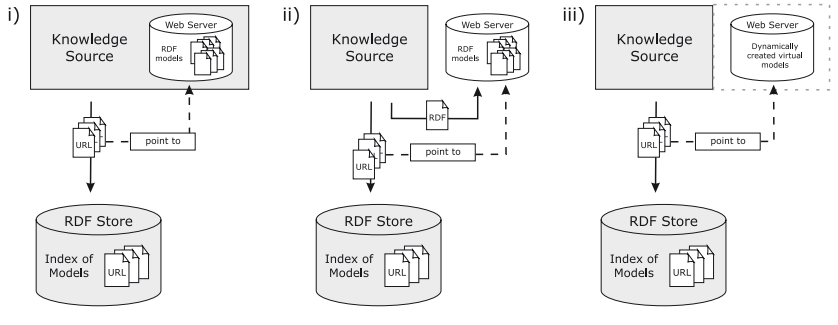


Fig. 2. Publishing knowledge using internal storage (i), external storage (ii), or semantic URLs (iii)

3 RDF: Stores

We will now introduce the concept of "RDF:Stores", which act as a registry for knowledge models. They hold an indexed list of published models and this way allow knowledge consumers to locate relevant knowledge models.

3.1 Interface

RDF:Stores provide two interfaces: The query interface allows to search the index for models with certain characteristics, while the advertisement interface allows knowledge sources to register new or revoke outdated knowledge models.

The *query interface*'s only method is "query" and accepts the parameters listed in Table 1. The *Query* parameter allows to search for models that match some filter like "all models that contain a statement about the temperature in room C0.03". The language that is used to specify these filters depends on the concrete RDF:Store implementation. In our prototype implementation we use WHERE clauses of RDQL [13] queries. The *Order* parameter allows to sort the returned list according to the models' timestamps. By default the models are returned in random order. The *Limit* parameter limits the number of returned models. By setting this value to 1 and applying descending temporal ordering, the most recent state of the environment can be acquired. The *Skip* parameter allows to omit a given number of results from the response. This is useful in conjunction with the *Limit* parameter to acquire a huge list of returned models in smaller chunks (this is also called "paging"). The *First* and *Last* parameters allow to specify a timeframe, that the returned models' timestamps have to match. This is not only useful to acquire historical data, but is also used to trigger the *push mechanism*: If the *Last* parameter is set to a point in the future, the requesting application will be notified on new models matching the current filter until the given point in time. In order to work the push service requires the *Callback* parameter to be set. If this parameter is given, the results are not returned in immediate response but are handed over to the given callback function in a separate call. A valid callback parameter is required in order for the push service to work, but may also be useful when issuing direct requests.

The *advertisement interface* allows knowledge sources to advertise new models and to revoke outdated models that they will no longer publish via the "advertise" respectively "revoke" method. The parameters of these methods are given in Table 2. Both methods expect a parameter *Model*, which has to contain the full URL of the knowledge model that should be advertised or revoked. The knowledge source must assure that this model is available for download under the given URL while the request is processed in order to give the RDF:Store the opportunity to update its index. In order to prevent malicious programs from revoking foreign models, knowledge sources can specify an authentication key on advertising that must be repeated when revoking a model. To protect and control the general access to an RDF:Store, additional existing techniques may be used. A HTTP interface for instance could be protected via HTTP authentication or session cookies.

Table 1. Methods and Parameters of the Query Interface

Parameter of "query" method	
Model: URL	URL of the model that is advertised (mandatory).
Filter: String	Only return models matching this filter. Language is implementation dependant, e.g. RDQL. Default is "all models".
Order: [ASC|DESC|NONE]	Sort models according to their timestamps. Default is "NONE".
Limit: Integer	Limit number of returned models. Default is "return all", but may be limited by implementation.
Skip: Integer	Skip the first n results. Default is 0.
First: Timestamp	Omit models with a timestamp earlier than specified. Default is "everything in the past".
Last: Timestamp	Omit models with a timestamp later than specified. If value is in the future, caller will be notified about new models (requires callback parameter to be set). Default is "now".
Callback: URL	Do not return results directly but post them to the given URL. Default is "no callback".

Table 2. Methods and Parameters of the Advertisement Interface

Parameter of "advertise" method	
Model: URL	URL of the model that is advertised (mandatory).
Key: String	Authorization key that will be required to eventually revoke model (optional).
Parameter of "revoke" method	
Model: URL	URL of the model that is revoked (mandatory).
Key: String	Authorization key that was given with "advertise" method (mandatory if given with advertisement).

3.2 Internals

RDF:Stores are a general concept that is independent of a concrete implementation. The presented concept neither prescribes how shared knowledge is represented, nor how an RDF:Store indexes the published models or if and how long published models should be cached. RDF:Stores may be applied in a multitude of different scenarios, and their actual implementation may depend on the concrete requirements.

To implement the *query and advertisement interface* describe above any remote invocation technique may be used. This includes (but is not limited to) universal web service protocols like XML-RPC [16], SOAP [11], and REST, as well as proprietary interfaces like Java's RMI [3], or Microsoft's .NET framework. In our prototype implementation, which will be discussed later, we use simple HTTP requests with key-value parameters encoded in the URL.

In order to build their index RDF:Stores need an idea of the content of the models that have been advertised to them. The amount of information that actually needs to be stored in the index depends on the expressiveness of the querying language. The required expressiveness as well as the amount of resources that are available to store the index are application-dependant. If maximum expressiveness of the search queries is required, one solution is to keep a local copy of any published model. This approach was also chosen in our prototype implementation, as it has the additional advantage of providing caching functionality at no additional costs (see below). If less expressiveness is required, it may be

sufficient to construct a compressed index, which for instance contains only the concepts mentioned in each knowledge model.

In dynamically changing environments or if access to historic information is required, RDF:Stores may need to cache knowledge models. Two major caching strategies can be distinguished: The *lazy caching strategy* tries to avoid caching as long as a model is published by its source. A model is cached only if the source revokes the model via the advertisement interface. The *greedy caching strategy* tries to keep a local copy of each shared knowledge model. This consumes more memory resources, but improves the robustness in the case of unexpected connectivity loss with a knowledge source. By deciding whether to return the original or cached models' URLs on query requests, the workload of serving the knowledge models can be dynamically distributed between the RDF:Store and the knowledge sources. The best strategy again is application dependant.

4 Prototype Implementation

We will now describe our prototype implementation of RDF:Stores that is currently applied in the projects SPECTER [9] and SharedLife [10]. The SPECTER project is about building a personal journal by keeping track of a user's real world actions and affective states. After learning a user model from the personal journal, an artificial agent could offer context-sensitive support adapted to the user's individual habits and needs. SPECTER applies RDF:Stores to realize its sensing framework as well as to manage its personal journal. The prototype is implemented in Java but can interact with any system that is able to perform HTTP requests.

Figure 3 shows the components involved in our RDF:Store implementation as well as the flow of information between them. The HTTP interface is provided via an embedded web server implementation called Jetty [5], which hosts two servlets that implement the query and the advertisement interface. The handling of RDF models is performed by the Jena semantic web framework [4]. This framework provides an API for parsing, storing, and searching RDF, RDFS and OWL models. Models advertised to the RDF:Store by a knowledge source are fetched via HTTP and are parsed into individual persistent Jena models. As Jena allows to store such models in a SQL database, a large number of models can be handled in parallel without performance issues. As database engine we use the freely available MySQL [12] database server. An additional servlet hosted by the embedded web server allows to access the knowledge models stored in this database and thus provides the RDF:Store's caching functionality.

The RDF:Store's interface is accessed by issuing HTTP requests of the form `http://<host>/<method>?<param_1>&...&<param_n>`, where `<host>` denotes the hostname of the machine running the RDF:Store, `<method>` is one of the methods "query", "advertise", and "revoke" (as described in the previous section), and `<param_i>` for $i = 1, \ldots, n$ are the request parameters (see Tables 1 and 2) given as key-value pairs `<name>=<value>`. To advertise the semantic

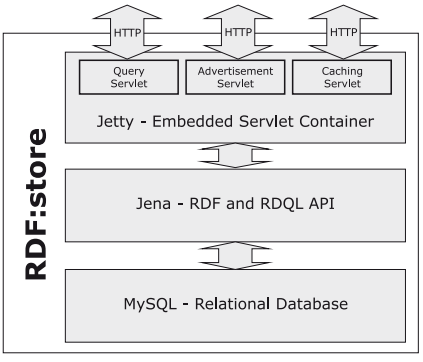


Fig. 3. Components of the RDF:Store prototype implementation

URL version of our temperature model, the temperature sensor would issue a request similar to the following:

```
http://myrdfstore.net/advertise?model=http://mysensor.net/20060316090000-21.rdf
```

Query filters can be specified using RDQL WHERE clauses [13]. A filter matching all models which contain a statement about a temperature exceeding 20 degree Celsius could look like this: `(?y, <ns:hasTemperature>, ?z) AND ?z>20` In order to get the latest known model that matches this filter, the following query request might be send to an RDF:Store:

```
http://myrdfstore.net/query?order=DESC&limit=1&filter=(?y,<np:hasTemperature>,?z)+AND+?z>20
```

Due to space restrictions, the filter as well as the URL given above are simplified: The "np" part of the filter has to be substituted by a complete namespace prefix, and special characters in the URL have to be escaped.

In response to a query requests a list of matching models is returned in plain text format. In the first line the number of returned models is given, followed by the URLs of matching models in separate lines each. If the callback parameter is set, the list is send to the specified callback URL via a HTTP POST request instead. If the *Last* parameter is given with a value situated in the future, the query is kept by the RDF:Store until that point in time and is repeatedly executed on every newly shared model. Internally, each query is resolved via Jena's RDQL processor. The WHERE clause given in the filter parameter is extended into a complete RDQL query and in turn is applied on all models stored in the Jena database. As Jena translates RDQL queries into SQL queries, the indexing capabilities of the SQL database system are automatically exploited to speed up such queries. This allows to efficiently search even large numbers of models.

5 Related Work

Existing information sharing and data management frameworks can be divided into two groups: Centralized infrastructure-based systems on the one hand and distributed peer-to-peer systems on the other hand.

Centralized (or blackboard-based) systems use a common repository provided by the infrastructure to store shared information. Often, this repository implements a tuple space [2]. Knowledge sources post the information they want to share to the tuple space. Information consumers may retrieve selected elements from a tuple space by specifying filters, which are applied in a binary comparison to all elements in the tuple space. Examples of tuple space implementations are JavaSpaces [7] and EventHeaps [6].

Peer-to-peer (or agent-based) systems store information by distributing it over multiple peers in the network. Thus, each peer stores only a fraction of the globally shared knowledge. In order to locate relevant knowledge, queries are propagated throughout the network according to some routing protocol until a node is found that could provide the requested information. The Kademlia algorithm [8] for instance establishes a distributed hashtable based on a "distance" measure between nodes. Other approaches like [1] use incentive-based routing protocols, which reward peers that actively contribute to others' information accesses. Furthermore, collaborative caching mechanism support the replication of popular information, while unused data might eventually fade away.

Centralized as well as distributed approaches have their particular strengths and weaknesses. The centralized approach is easy to implement but inflexible with respect to dynamic and changing environments. Imagine a user has his user model stored on a blackboard in some environment. If he changes location, his user model would have to be copied or moved from the original blackboard to the blackboard in the new environment. Obviously it would be more natural and efficient to store the user model near the user, e.g. on a mobile device like a PDA that the user is carrying with him. A distributed peer-to-peer system is by far more flexible, but puts a comparatively high demand on the involved peers: Sophisticated protocols for knowledge distribution and routing are often complex to implement and require considerable computational and communicational resources when applied. Furthermore, such approaches assume that most of the peers can actively contribute to the overall benefit. At least today this in general is not the case. The world is cluttered with small and limited devices like smart tags or sensor network nodes, that are easy and cheap to produce and accordingly possess only very limited resources. On the other hand, more and more resources are available in the infrastructure in form of powerful PCs or servers.

The RDF:Store approach presented in this paper tries to combine the best parts of both worlds in order to realize simple yet flexible knowledge sharing in dynamic intelligent environments. Knowledge is stored in a distributed manner as each piece of knowledge is conceptually attached to its sources. This allows knowledge sources to easily "take their knowledge around with them" if they are moved to another environment. In contrast to a purely distributed system, the complex and costly process of matching knowledge sources and consumers is performed by a powerful central component provided by the infrastructure, the so-called RDF:Store. RDF:Stores allow to divide the work between infrastructure and mobile components in a natural and efficient way. In comparison to a

purely centralized system, flexibility is preserved by introducing one level of indirection. Instead of the knowledge itself only pointers to distributed knowledge are centrally collected by the RDF:Store.

6 Conclusion

In this paper we presented a framework based on the concept of RDF:Stores, which allows for a lightweight management of shared knowledge. It keeps knowledge conceptually attached to its source but provides a central registry to query for relevant models. With the ability to implement caching mechanisms the framework can be used in dynamic environments where a continuous availability of actual and history knowledge is required. We described how RDF:Stores can be reliably and efficiently implemented using standard software components and mature semantic web technologies. Future work will have to perform an empirical comparison of the proposed approach with other existing frameworks.

Acknowledgements

The work presented in this paper was funded by the German Federal Ministry of Education and Research (BMBF) under the contract numbers 01 IW C03 (Specter) and 01 IW F03 (SharedLife).

References

1. Chen, W., Wang, C., Lau, F.: Collaborative and Semantic Data Managment Framework for Ubiquitous Computing Environments. Proc. of the International Conference on Embedded and Ubiquitous Computing (EUC-04) (2004)
2. Gelernter, D.: Generative communication in Linda. ACM Trans. Programm. Lang. Syst. 7(1) (1985)
3. Sun Microsystems: Java Remote Method Invocation (RMI). http://java.sun.com/products/jdk/rmi/ (March 2006)
4. Jena Semantic Web Framework. http://jena.sourceforge.net/ (March 2006)
5. Jetty Java HTTP Servlet Server. http://jetty.mortbay.org (March 2006)
6. Johanson, B., Fox, A., Hanrahan, P., Winograd, T.: The Event Heap: An Enabling Infrastructure for Interactive Workspaces. CS Tech Report CS-2000-02 (2000)
7. Mahapatra, S.: Introducing javaspaces. Java Developer's Journal (2000)
8. Maymounkov, P., Mazires, D.: Kademlia: A Peer-to-peer Information System Based on the XOR Metric. In Proc. of IPTPS02 (2002)
9. Schneider, M., Bauer, M., Kr"oner, A.: Building a Personal Memory for Situated User Support. Proc. of the Workshop on Exploiting Context Histories in Smart Environments (ECHISE) at Pervasive 2005 (2005)
10. SharedLife Project. http://www.dfki.de/sharedlife/ (June 2006)
11. W3C: Simple Object Access Protocol (SOAP). http://www.w3.org/TR/soap/ (March 2006)

12. MySQL AB: mySQL Relational Database Server. http://www.mysql.com/ (March 2006)
13. W3C: RDF Data Query Language (RDQL). http://www.w3.org/Submission/RDQL/ (March 2006)
14. W3C: Resource Description Framework (RDF). http://www.w3.org/RDF/ (March 2006)
15. W3C: Web Ontology Language (OWL). http://www.w3.org/2004/OWL/ (March 2006)
16. XML-RPC. http://www.xmlrpc.com/ (March 2006)

Vision Based Automatic Surveillance Towards a Smart Application

Dong-liang Lee[1] and Lawrence Y. Deng[2]

[1] Dept. of Information Management, St. John's University, 499, Sec. 4, Tam King Road Tamsui, Taipei, Taiwan
[2] Dept. of Computer Science and Information Engineering, St. John's University, 499, Sec. 4, Tam King Road Tamsui, Taipei, Taiwan
{lianglee, lawrence}@mail.sju.edu.tw

Abstract. Intelligent vision based surveillance has been an important research issue of distributed computing and communication in the future. Since the smart applications are implemented on cyberspace, how to control smart objects and to increase the degree of self awareness has been a challenging issue. This paper was addressed in a vision-based surveillance smart application. An Extended Petri Net was proposed for modeling the vision-based surveillance smart application. This proposed system not only provided the hyperspace facilities but also gave the certainly smart/intelligent fusion for receiver's needs. Our testbed were focused on the swing and slide those were very common facilities in the park field or playground. We constructed several surveillance cameras with the different specified position and view angle. To obtain the moving objects which were grabbed video and it could be analyzed the potential dangers in real time. We believe that the more smart applications can be developed the more smart real world will be come true in near future.

1 Introduction

Integrating both of cyberspaces and real spaces could be developed into smart applications. Smart applications will build a smart world. Many smart applications were suggested in [1] [2] that included the potential trends and related challenges toward the smart world and ubiquitous intelligence from smart things to smart spaces and then to smart hyperspaces. Likewise, they also showed their efforts in developing a smart hyperspace of ubiquitous care for kids, called Ubi-Kids. In [3], they focused on novel applications in a home environment utilizing communication protocol IEEE 802.11a/b/g as the underlying network. These networks constitute the foundations of an attractive framework, allowing the users the freedom of mobility coupled with feature-rich functionality. The system will include networked appliances such as refrigerator, oven, HVAC system, and personal video recorder (PVR), entertainment systems such as networked DVR, VCR players, home surveillance and security system. Algorithms for cooperative multi-sensor surveillance were proposed by [4]. They presented an overview of video understanding algorithms developed at CMU to perform cooperative, multi-sensor surveillance. A network of smart sensors were also deployed that were independently capable of performing real-time, autonomous

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 240 – 249, 2006.

object detection, tracking, classification and gait analysis. The author [5] investigated the near shore oceanography and monitoring of coastal regions related tasks for optical remote sensing by using standard video cameras. They focused primarily on the problems of near shore remote sensing and also provided several steps to improve the Argus video sensor network's functionality to quantify the time-space characteristics of the visualization world. Their discussion in that smart application domain should apply to a wide range of video-monitoring problems. Prof. Ma [6] [7] discussed the smart world for Ubi-Kids and recommended his aspects for a ubiquitous kids care system, for helping parents in taking care of their children. [8] recommended the G-Net (based on Petri Net) methodology for object-oriented complex software system design and the EU model for object-oriented dependable system design were integrated into the unified concept of the smart object, which was an object with an associated knowledge structure that incorporated the necessary parameters of the real-time task, such as the timing constraints and reliability requirements.

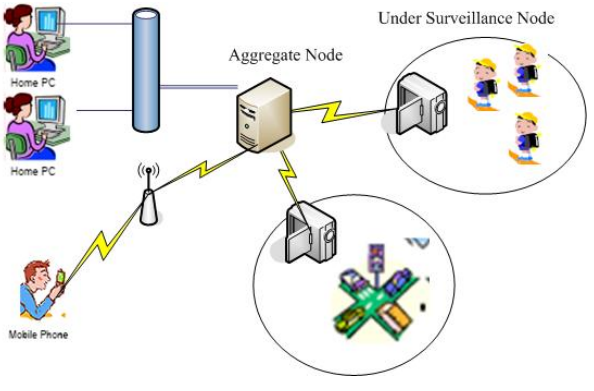


Fig. 1. A logical layout of smart application in distributed network environment

A context-aware framework [9] for an intelligent agent triggering intelligent and automatic services according to the information based on situation was proposed. They designed and implemented an intelligent home care system based on the context-awareness technology, which minimized residents' interventions and maximized functional autonomy of systems. They also suggested system architecture and its flows for the proposed system. The interaction issue about conflict and interference between various context information and service in the ubiquitous home environment had been discussed.

In this paper, we would like to address in the vision-based surveillance system in distributed environment. A scenario for our exploration was illustrated in figure 1. Remaining paragraph is organized as follow: The vision-based surveillance smart application model that is based on the Petri Net is re-defined in section 2. The implementation included the initial stage, aggregation and notification stage were presented in section 3. The discussion of variant view issues is deliberated in section 4. Finally, we made a brief summary in section 5.

2 The Vision-Based Surveillance Smart Application Model

Before we began to introduce the Extended Petri Net model for the smart application in distributed network environment, the basic concept of the Petri Net was given as follows. The Petri Net was originally proposed by C. A. Petri [10] which attempts to develop a formal methodology to describe and analyze a system behavior. The Petri Net model is a graphical and mathematical modeling tool which is especially useful to capture the synchronization characteristic among modules of a system. With the help of the netted representation by the Petri Net, the researcher can easily discover the potential problem of a running system and adjust its design to maintain the validity of this system. Petri Net and workflow both support graphics representation, nesting structure, verification, and simulation. Petri Net can also be evaluated and analyzed by a simulation tool. A Petri Net model can be formally denoted as a 5-tuple, PN = (P, T, F, M_o) where:

P = {P1, P2, …, Pm} is a finite set of places.
T = {T1, T2, ..., Tn} is a finite set of transitions.
Most importantly, P and T must satisfy the properties of
$P \cap T = \varnothing$ and $P \cup T \neq \varnothing$.
That is, at least one of these two sets P and T must be nonempty.

$F \rightarrow (P \times T) \cup (T \times P)$ is a set of arcs (flow relation) that network places and transitions. That is, $(P \times T)$ represents the set of arcs that flow from places to transitions whereas $(T \times P)$ is the set of arcs flowing in opposite directions.

Mo: P→ {M0, M1, M2, …, Mm} is the set of initial marking of each place. For the definition of the Petri Net model, Mij represents the token number on place Pj at time i and a token can be a resource of a system or control of a program.

According to the definition of the Petri Net, The generic components of a Petri Net include a finite set of places and a finite set of transitions. A Petri Net is a finite bipartite graph that places are netted with transitions, which in turn are connected to output places. The distribution of tokens over places is called a marking of the net. A transition may enable or fire when each of its input places contains at least one token. The firing of a transition results in removing tokens form their input places and adding to output places via transition. A marking represents the state of a system, which is removed from its place when a transition fired and a new marking is then generated to the output places of this transition.

We defined vision-based smart application model based on the characteristics of the Petri net. As a graphical tool of Petri net, the followings are basic properties of a Petri net and the description of learning objects:

Definition 1: An Extended Petri Net (EPN) is an 8-tuples, PN = (P, T, A, L_{ID}, C_Z, D_t, S_t, M_o) where:

- $P = \{ P_{US}, P_{SA}, P_{IA}, P_{RN} \}$ is a finite set of places,
  where, $P_{US} = \{P_{US1}, P_{US2}, \ldots, P_{USm} \}$ is a finite set of under surveillance places,
  $P_{SA} = \{ P_{SA1}, P_{SA2}, \ldots, P_{San} \}$ is a finite set of self awareness places,
  $P_{IA} = \{ P_{IA1}, P_{IA2}, \ldots, P_{IAq} \}$ is a set of information aggregation places,
  $P_R = \{ P_{R1}, P_{R2}, \ldots, P_{Rs} \}$ is a set of receiver places,
- $T = \{T_1, T_2, \ldots\ldots\ldots, T_i\}$ is a finite set and a sequence of transitions,

- $A \subseteq (P \times T) \cup (T \times P)$ is a set of arcs,
- $L_{ID} = \{0, 1, 2, \ldots\} \in$ is a set of number for location identifier,
- $C_Z = \{0, 1, 2, \ldots\}$ is a set of number for critical zone information,
- $D_t = \{0, 1, 2, \ldots\}$ is the duration of time interval,
- $S_t = \{\alpha, \beta, \ldots\ldots\zeta\} \in$ String is a set of situations at time t,
- M_o: $P_{US} \rightarrow \{M_0, M_1, M_2, \ldots, M_m\}$ is the set of initial marking of each place.

For the definition of the Petri Net model, M_{ij} represents the token number on place P_{USj} at time i and a token can be the under surveillance place within different range of visibility value (vision-based surveillance must consider circumstances change with the passage of time).

- $P \cap T = \varnothing$ and $P \cup T \neq \varnothing$.

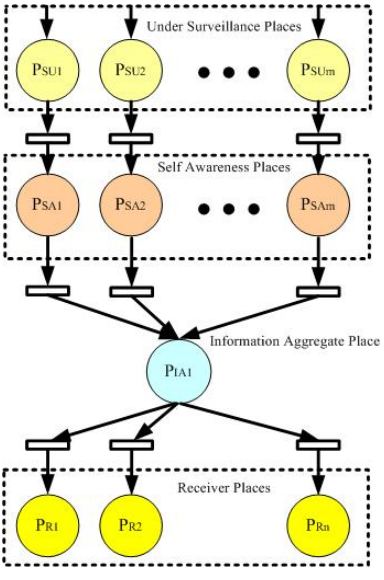


Fig. 2. A representation of Extended Petri Net model for the vision-based surveillance smart application process

3 Construction of Vision-Based Surveillance Smart Application

3.1 The Initial Stage

Under surveillance places could be realized by vision-based surveillance devices (e.g. video camera) and also provided the visual archiving through hyperspace as an essential security utility for ensuring the physical protection of homes, kindergarten, nursery school, offices or public areas. For the essential care/security reason, the most facilities should provide a self awareness mechanism to manage it. An initial under surveillance place M_{oi} can be composed from an under surveillance place P_{SUi} and a self awareness place P_{SAi}; and the whole contexts are a collection of values and can be represented as M_{oi}: $\{P_{SUi}, P_{SAi}\} = \{L_{ID}, C_Z, D_i, S_i\}$

Where L_{ID} denote the related location, C_Z is the critical zones that are configured by user, D_i is the duration of time interval, S_i is the situation at time i. Figure 3 showed the processes of an example of some kid playing on the slide. User can configure the several critical zones in monitor view as figure 3(b) firstly. If some kid appears in the critical zone 1, the self awareness program will find and change the situation in time, as figure 3(c) illustrated (the result of figure 3(c) can be computed as the difference between image 3(c) and image 3(b)).

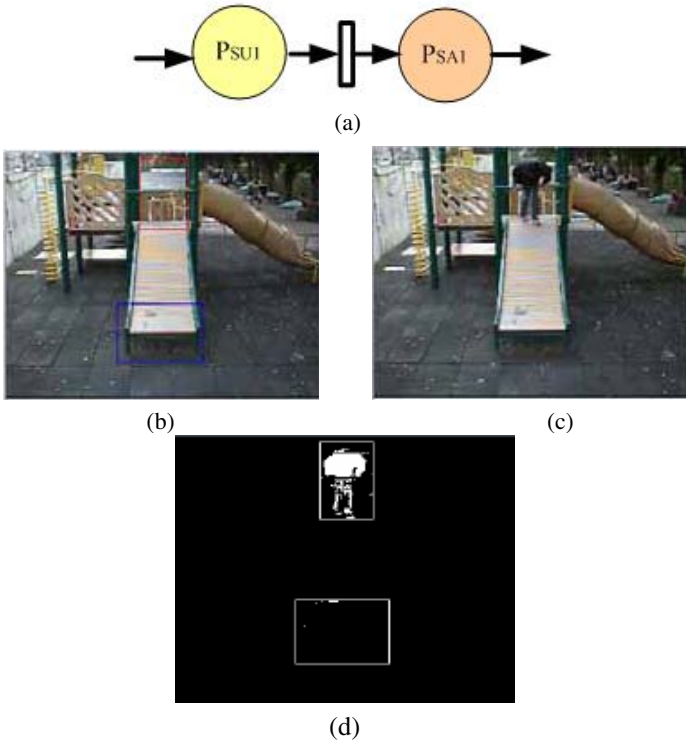


Fig. 3. (a) a representation of initial under surveillance stage M_{oi}, (b) the testbed view in an under surveillance place and the critical zones were configured as red and blue rectangular, (c) a kid play on a slide, and (d) self awareness function can be triggered and react to the situation change

The vision-based surveillance system must also consider circumstances change with the passage of time (different range of visibility value). It is necessary to refine the background reference image in a period time automatically. Figure 4(a) showed the day time view of a surveillance testbed and the difference rate is set to near 80% and the dim time view of surveillance testbed and the difference rate is downed to 60%. In self awareness operation, we defined the dynamic attributes "difference rate factor"ΔZ_i to achieve real world complexity. Difference rate factor can be remarked by the average color from the background image in time interval D_i. However, it is not yet unclear that what results of the fault detection even in a low rate means when

applying the activity detection to the real systems and using them in practical. If incomplete and uncertain contexts and mis-judged situations are common and ubiquitous, we should keep these incompletion, uncertainty and misjudgment as intrinsic features in smart u-things' research [11].

Definition 2: The self awareness operation, $\alpha\Delta Z_i\ (PN\{P_1, P_2, \ldots, P_n\})$ can compares all under surveillance place P_i with ΔZ_i in the time interval D_i .

Let the set of difference rate factors $\Delta Z_1 \in \{D_1, C_{Z1}\}$, $\Delta Z_2 \in \{D_2, C_{Z2}\}$, ….. , $\Delta Z_n \in \{D_n, C_{Zn}\}$, where $\{D_i, C_{Zi}\} \in P_i \in EPN$.

$\alpha\Delta Z(EPN\{\alpha\Delta Z_1, \alpha\Delta Z_2, \ldots, \alpha\Delta Zn\})$

$= \text{EPN}\{\alpha\Delta Z'_1, \alpha\Delta Z'_2, \ldots, \alpha\Delta Z'm\}$

$\rightarrow \alpha\Delta Z\ (EPN\{P_1, P_2, \ldots, P_n\}) = EPN\{P'_1, P'_2, \ldots, P'_m\}$

where the $\Delta Z'_i$ of P'_i in $EPN\{P'_1, P'_2, \ldots, P'_m\}$ is equal to or greater than ΔZ.

(a) (b)

Fig. 4. Different range of visibility value will influence the difference rate. (a) The day time view of a surveillance testbed and the difference rate is close to 80%, (b) The dim time view of a surveillance testbed and the difference rate is downed to 60%.

3.2 The Aggregation and Notification Stage

Definition 3: The aggregation and notification operation, $N_{select}\ (EPN\{P_{SA1}, P_{SA2}, \ldots, P_{SAn}\})$ select all the self awareness place P_i into a new $EPN'\ \{P'_1, P'_2, \ldots, P'm\}$. The designated *select* is evaluated from the personal preference profiles such as status, goal, preference, feeling, etc., and current situations S_t and past situations $S_{t-\,Di}$ to a receiver respectively.

Let the set of dynamic attribute *select* $S_1 \in P_{SA1}$, $s_2 \in P_{SA2}$,….. ,$S_n \in P_{SAn}$, where $P_{SAi} \in$ EPN, $S_i \in$ user adaptive personal preference profiles.

Process:
  FOR i=1 TO I <= n DO
    IF (S_i= = *TRUE) THEN*
  N_{select} *(* $EPN\{P_{SA1}, P_{SA2}, \ldots, P_{SAn}\}$ *)* = $EPN'\{P'_1, P'_2, \ldots, P'm\}$,
  END IF
 End FOR
End Process

Where the $P_{SA1}, P_{SA2}, \ldots, P_{SAm} \in EPN$.

4 Discussion

Single angle of vision may occur the hiding of view problem that will raise the uncertain/misjudged situations. Multi-camera configurations could solve this problem. Contemporaneously, the self awareness places should provide the concurrent mechanism and to keep track of the changes in configured critical zones. The concurrent mechanism was defined as follow:

Definition 4: The aggregation and notification operation with multi-camera, Co_N_{select} $(EPN\{P_{SA1}, P_{SA2}, \ldots, P_{SAn}\})$ select all the self awareness place P_i into a new $EPN'\{P'_1, P'_2, \ldots, P'm\}$. The designated *Co_Select* is evaluated the several concurrent $S_{i'}$, $S_{i''}$,... from the same location L_{ID} . Let the set of dynamic attribute Co_*Select* S_1, $S_{1'}$, $S_{1''}$,...$\in P_{SA1}$, S_2 $S_{2'}$, $S_{2''}$,... $\in P_{SA2}$,..... ,S_n , $S_{n'}$, $S_{n''}$,...$\in P_{SAn}$, where P_{SAi} $\in$ EPN, $S_i \in$ user adaptive personal preference profiles, and $\{S_i, S_{i'}, S_{i''},\ldots\}, \in L_{IDi}$..

Co_Process:
FOR i=1 TO I <= n DO
IF (S_i= = *TRUE and* $S_{i'}$==*TRUE and* $S_{i''}$ ==*TRUE....) THEN*
N_{select} *(* $EPN\{P_{SA1}, P_{SA2}, \ldots, P_{SAn}\}$ *) = EPN'*$\{P'_1, P'_2, \ldots, P'm\}$,
END IF
End FOR
End Process
Where the $P_{SA1}, P_{SA2}, \ldots, P_{SAm} \in EPN$ *and* $\{S_i, S_{i'}, S_{i''},\ldots\}, \in L_{IDi}$.

In our surveillance swing testbed, we set up two cameras in the same location. One was set in the front view and the other one was set on the right side; in this case, we avoided the uncertain situation and got more completed swing routes both from two cameras. Figure 5(a) showed the front configured camera 1 and with three critical zones in figure 5(b). The number of critical zones setting between is independent. The configurations of number of critical zones are dependent on the real situation. Figure 6(a) illustrated the right side configured camera 2, and also with three critical zones

(a) (b)

Fig. 5. (a)A view from the front camera 1 and (b) the three configured critical zones

setting. The camera 1 and camera 2 have the same location identification L_{ID} but may not the same situation alarm (S_i may not equal to the $S_{i'}$) by the self awareness places.If both of the situation alarm S_i and $S_{i'}$ were "TRUE" then we will set the situation alarm is "TRUE". The figure 7 illustrated the detection result from camera 1 in day/bright time. The x-axis presents the number of video frames; the y-axis presents the difference rate. The period from frame number 129 to 673 showed the situation alarm was "TRUE". At the same time, figure 8 (camera 2) showed the situation alarming "TRUE" in the period from frame number 473 to 768. So we can get the actual situation alarm by the "AND" operation and get the period from 473 to 673.

The figure 9 illustrated the detection result from camera 1 in dim time. The period from frame number 353 to 485 showed the situation alarm was "TRUE". At the same time, figure 10 (camera 2) showed the situation alarming "TRUE" in the period from frame number 375 to 507. So we can get the actual situation alarm by the "AND" operation and get the period from 375 to 485.

(a) (b)

Fig. 6. (a) A view from the right camera 2 and (b) the another three configured critical zones

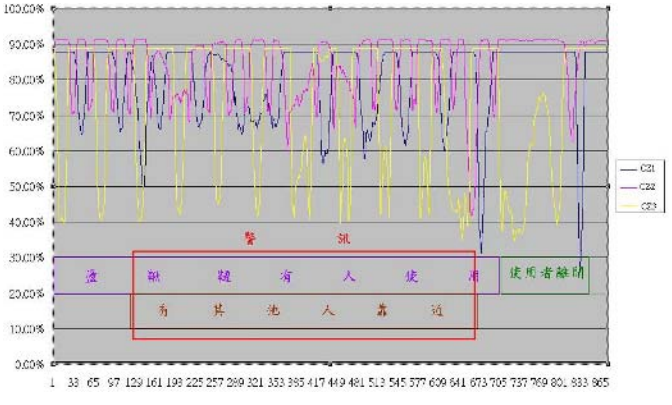


Fig. 7. The detection result from camera 1 in the day/bight time

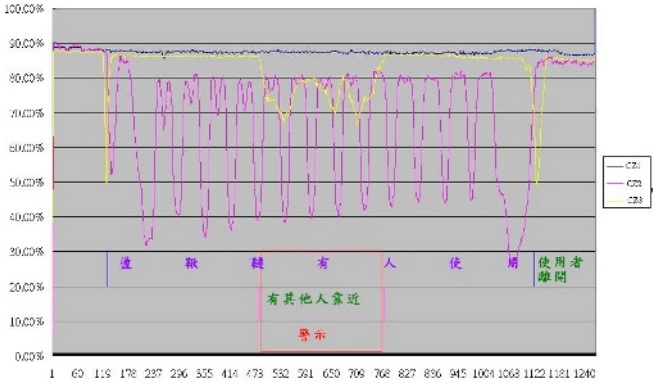


Fig. 8. The detection result from camera 2 in the day/bright time

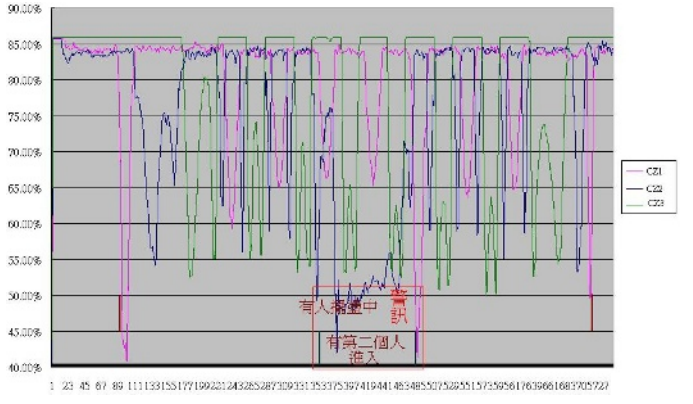


Fig. 9. The detection result from camera 1 in the dim time

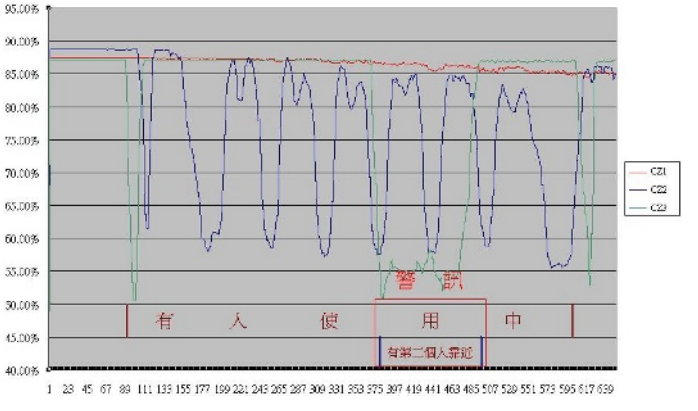


Fig. 10. The detection result from camera 1 in the dim time

5 Conclusion

This paper has presented a vision-based surveillance model that performed self awareness, aggregation and automatic sending the aspect situations to the receiver. The visualization understanding was deployed that were detected from the configuration of critical zones. The desired critical zone setting can decrease the computation load and to give the certain "real time" demand for smart application. As our experiences, there are some more issues need to investigate: (a) how to improve the more precise and sufficient from the context that could be characterized a real situation actually. (b) how to decrease the fault judgments, situation alarms and to find the adaptive information aggregation for personal aspects.

References

1. Jianhua Ma, Laurence T. Yang, Bernady O. Apduhan, Runhe Huang, Leonard Barolli and Makoto Takizawa: Towards a Smart World and Ubiquitous Intelligence: A Walkthrough from Smart Things to Smart Hyperspaces and UbicKids, in Journal of Pervasive Computing and Communication vol. 1, (2005), 53-68
2. Jianhua Ma: Ubiquitous Intelligence - The Intelligence Revolution, ID People Magazine, (2005).
3. Liyu You; Jamshaid, K.: Novel applications for 802.11x enabled wireless networked home, Consumer Communications and Networking Conference, CCNC 2004. First IEEE, 5-8 Jan. (2004) 684 – 686.
4. Collins, R.T., Lipton, A.J., Fujiyoshi, H., and Kanade, T.: Algorithms for Cooperative Multi-sensor Surveillance, Proceedings of the IEEE, Vol. 89, No.10, (2001) 1456-1477.
5. Holman, R.; Stanley, J.; Ozkan-Haller, T.: Applying video sensor networks to nearshore environment monitoring, Pervasive Computing, IEEE, Volume 2, Issue 4, (2003) 14 – 21.
6. Jianhua Ma, Laurence T. Yang, Bernady O. Apduhan, Runhe Huang, Leonard Barolli, Makoto Takizawa and Timothy K. Shih: A Walkthrough from Smart Spaces to Smart Hyperspaces towards a Smart World with Ubiquitous Intelligence, in IEEE CS Proceedings of the 11th IEEE International Conference on Parallel and Distributed Systems, Japan, (2005).
7. Katsuhiro Takata, Jianhua Ma, and Bernady O. Apduhan: A Context Based Architecture for Ubiquitous Kid□s Safety Care Using Space-oriented Model, in IEEE CS Proceedings of the 11th IEEE International Conference on Parallel and Distributed Systems, Japan, (2005).
8. Shi-Kuo Chang; Yeong-Jia Chen; Mosse, D.; Object-Oriented Real-Time Dependable Systems, 1994. Proceedings of WORDS 94, (1994) 10 – 17.
9. Seungho Baek; Hyunjeong Lee; Shinyoung Lim; Jaedoo Huh: Managing mechanism for service compatibility and interaction issues in context-aware ubiquitous home, IEEE Transactions on Consumer Electronics, Volume 51, Issue 2, (2005) 524 – 528
10. Peterson, J. L.: Petri Net Theory and the Modeling of Systems, Englewood Cliffs, NJ: Prentice-Hall, Inc., (1981).
11. A.G. Ganek, T.A.Corbi: The Dawning of the Autonomic Computing Era, IBM Systems Journal, Vol.42, No.1, (2003).

Handling Heterogeneous Device Interaction in Smart Spaces

Daqing Zhang[1], Manli Zhu[1], Hengseng Cheng[1], Yenkai Koh[1], and Mounir Mokhtari[2]

[1] Institute for Infocomm Research, Singapore
{daqing, mlzhu, hscheng, ykkoh}@i2r.a-star.edu.sg
[2] GET/INT Institut National des Télécommunications, France
mohamedali.fki@int-evry.fr

Abstract. Smart spaces pose significant technical challenges in heterogeneous device interaction/integration, user/environment perception, as well as system interoperability. As devices become more powerful and connected, the users are required to understand complex device functionalities in order to carry out simple tasks, and thus experience more and more frustration with the increase of device types and complexity. In this paper, we first examine some key issues and challenges in smart spaces, and then we propose a lightweight middleware which enables the spontaneous device interaction and can hide the complexity of heterogeneous device connection from the end users. Implementation details of the middleware prototypes are presented.

1 Introduction

Ubiquitous computing envision the future physical spaces such as homes, cars, hospitals etc. augmented with stationary and mobile devices/sensors/actuators. Those physical spaces which can provide us with a wealth of environmental information and thus empower the occupants to intelligently interact with the environment are often called *smart spaces*. Generally speaking, the information about the user and the environment is defined as *context*, e.g. user location and activity, environment temperature and ambient light, and applications that use context information are said to be *context-aware*. An example of smart space with great research potential is smart home for elderly and disabled, where the aim is to assist them for independent living and to improve their quality of life. The context-aware applications may include recognizing a crisis situation such as elderly fall, supporting everyday activities and providing awareness of daily life and long-term trends [1].

Smart spaces have posed a number of significant challenges for the system architecture. The first challenge is how to enable the heterogeneous devices interact with each other, yet hide the complexity of device interaction from the end users. The second challenge is how to acquire and understand the context information, and use context to tailor services and human-computer interaction. The typical context information is about who, what, when and where of the entities in smart spaces [2]. The

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 250 – 259, 2006.

third one is how to represent different entities in smart spaces such as people, devices, things and software functions so that a unified programming model can be based on. The fourth one is how to understand user's goals in terms of high-level tasks (what), and then let the system take care of how to achieve those tasks by composing the available services and resources.

Numerous software architectures have been proposed for smart spaces. Oxygen of MIT focuses on human-machine interaction, self-configuration, system programmability [3]. Gaia from UIUC applies the resource management approach in operating system for Active Spaces, it focuses mainly on context-awareness, security and programmability of smart space [4]. Aura at CMU adopts a task-driven approach for smart space support, the focus is on user mobility and context-awareness [5]. At UC Berkeley, the Ninja project deals with distributed Internet services, it presents an architecture for secure service discovery [6]. Context Fabric develops a service-oriented context-aware infrastructure with event and query services, it mainly addresses the privacy issues [7]. Cooltown in HP applies web technology in smart spaces and enhances physical objects with web content [8]. Gator House from University of Florida uses service-oriented framework to handle device self-integration, context-awareness and knowledge management in smart home [9]. However, none of the architectures have addressed context-awareness, spontaneous interaction and task-oriented user interface at the same time.In this paper, we intend to develop a system architecture for smart spaces, where spontaneous device interaction, context-awareness and task-oriented user interface will be supported.

2 Challenges in Building Smart Spaces

Fig. 1 is a four-layer model which logically describes the system architecture for smart spaces. The bottom layer handles the heterogeneous device interaction in a spontaneous manner. In order to offer the right services to the users in right time/place/manner, the second layer considers and processes the context information from the human as well as the environment. The human-centric interface designed in the third layer enables simple and intuitive control from the users to the smart spaces and hide the complexity of device interaction from the upper layer. The applications in the top-most layer leverage on the device interaction, context processing and user friendly interface. In the sequel, we will examine several technical challenges that we feel must be overcome before the vision of smart spaces becomes a reality. Those challenges are by no means an exhausted list of issues posed; however, they do represent some of the key issues for building smart spaces. Those challenges include:

- Device interaction and integration
- Context processing and management
- Interoperability of heterogeneous entities
- Task-oriented and Human-centric user interface

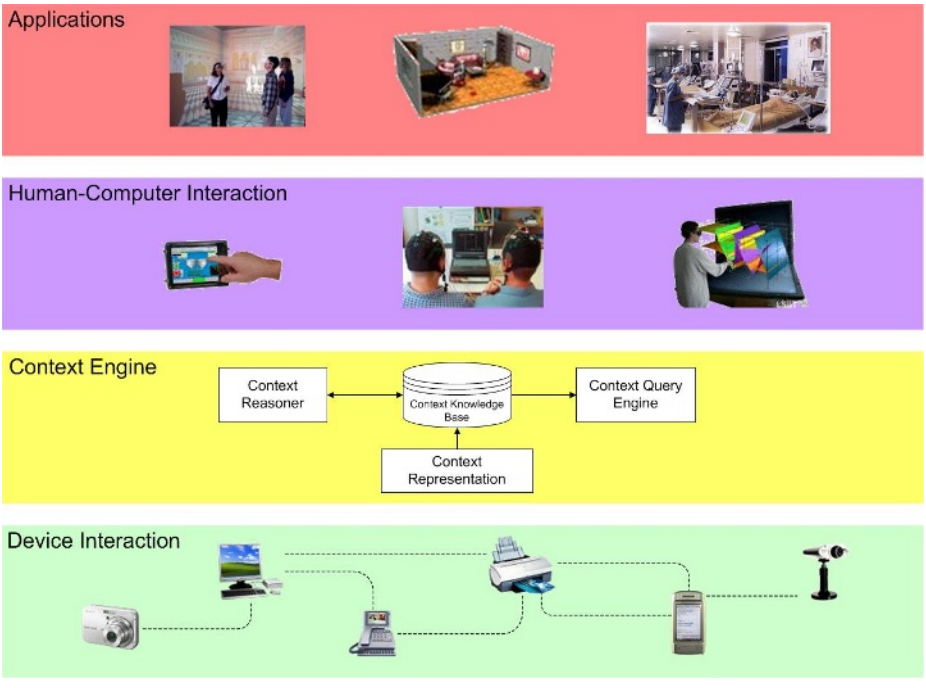


Fig. 1. Four-layer system architecture model for smart spaces

2.1 Device Interaction and Integration

Smart spaces are expected to contain large number of devices which interact with one other to achieve different goals. The interactions are characterized by a number of challenges: device interaction can be ad-hoc and spontaneous; devices and environments are heterogeneous; and context of user, device and environment is dynamic.

Currently, the developers must pre-program devices to recognize the specific protocols, standards, data formats and operations of all the peer device type they expect to encounter in order to talk to each other. With the rapid increase in numbers, types and operating domains of the devices and the services they may provide, it is unreasonable to expect that every device will have prior knowledge of every other type of device. Consider, for example, a Bluetooth adapter bought by a user for a laptop computer. It may be reasonable for the user to expect the laptop to work with Bluetooth printers that she discovers around her. But, while the presence of Bluetooth in both devices allows the laptop to discover the presence of such a printer, the laptop is unable to use it without installing the specified device driver.

There are two approaches to achieve spontaneous interaction between devices: centralized control and peer-to-peer collaboration. The centralized approach utilizes servers to aggregate information for all the devices registered in a local environment which will facilitate the communication and interaction between the client devices and the environment. The key challenge for the centralized approach is developing an open service architecture that allows the heterogeneous client devices to control the devices in a new environment [10], and yet makes minimal assumption about standard

interfaces and control protocols. A data-centric scheme provides the solution to the challenge. It utilizes an interface definition language enabling exported object interfaces to be mapped to client devices control interface. The control messages are thus generated as the RPC command sent from the client user interface to the corresponding service daemon. The centralized approach is preferable in a relatively static environment, however, for the scenarios where most devices tend to join/exit the environment freely, frequent registering of the devices and updates of the information bring excessive burden to the system. Therefore the peer-to-peer collaboration approach which directly enables the interaction between two peer devices is more desirable in dynamic environments [11].

With peer-to-peer collaboration ability, a device can connect to another device, provide metadata about itself, be controlled and provide references to other devices. In order to accommodate enormous heterogeneous types of devices in the world, the infrastructure must provide a generic approach for the interactions between devices. Instead of specifying a detailed, continuously evolving communication protocol, it defines a simple, fixed set of interfaces which allow the two devices exchange capabilities, communicate with each other and use whatever communication protocols are appropriate for the information transferring. In other words, the approach establishes the minimal set of development-time agreements to defer all other agreements required until runtime. It then delivers these agreements in the form of mobile code, which can extend a recipient's behavior to make it compatible with a new peer.

2.2 Context Processing and Management

Context information plays an important role in making the physical spaces 'smart'. Users and applications in smart spaces often need to be aware of their surrounding context and adapt their behaviors to context changes. We believe an appropriate context management framework requires the following support:

- A common context model shared by all devices, services and spaces

Understanding context information is the basis for context sharing among devices and services in one smart space or across different spaces. An appropriate model should address different characteristics of context information such as dependency and uncertainty. In our earlier work, we have proposed an ontology-based context model [12] to describe context information in a semantic way, which exhibits features such as expressiveness, extensibility, ease of sharing and reuse, and logic reasoning support.

- Context acquisition, context lookup and context interpretation

These services are essential for building applications with context-awareness in smart spaces. Context acquisition is closely coupled with sensors to acquire context data from physical or virtual sensors. Context lookup provides user and applications both synchronous context query service and asynchronous context event notification service. Considering that context information is widely spread over wide-area networks across multiple smart spaces, a robust lookup service can be challenging. Such challenges can be, for example, building an underlying lookup mechanism to allow context lookup from anywhere in the system considering the temporal characteristics of context information. Context interpretation provides the support of deriving high-level context from low-level context. Different interpretation techniques can be applied such as logical reasoning and machine learning. Our earlier work such as

Semantic Space [13], SOCAM [14] provided the set of such services to build our middleware for smart spaces.

2.3 Interoperability of Heterogeneous Entities

Within a smart space environment, entities can range from sensors, objects, devices to software functions. Those heterogeneous entities interact and service one another to complete different tasks. This sounds fine except that these entities are likely to originate from different sources and therefore use different ways to present their capabilities and connectivity requirements. As a result, entities within a smart space will not be able to interoperate with one another. We present an architecture which combines two mechanisms, i.e., Service-Oriented Architecture (SOA) and mobile code to solve the interoperability issue between a device and any devices within a Smart Space.

Service-oriented architecture is a software architectural concept that defines the use of services to support the requirements of entities. In a SOA framework, entities in the environment are represented in form of services and made available to other entities in a standardized way. As the functions of every entity are described using common convention, entities can thus understand each other and collaborate to achieve a certain goal. The utilization of SOA provides the solution to enable interoperability, however two problems remain i) how to facilitate an entity to present its functions as a service; ii) how to enable a new device to access the services in the smart space.

The developer can manually program an entity in accordance with the service description requirements of the smart space. Apparently, this approach suffers from the problem of scalability. Therefore, we design various wrappers to conform entity interface to the service entry of SOA, in other words, a unified interface understandable to all the entities. A service entry of SOA includes service announcement, service discovery and service invocation.

The second problem is solved by utilization of mobile code. When a new device enters into the smart space, it can download the mobile code which contains the abstraction of the smart space. With running the mobile code, the new device can automatically discover the services available and invoke the services.

2.4 Task-Oriented and Human-Centric User Interface

As devices and services become more and more prevalent in smart spaces, it's no longer an easy job for users to understand all the functionalities of the devices/services to carry out even simple tasks. For example, in order for an elderly to play a DVD on TV, he/she needs to use several remote controllers to accomplish this job.

In most scenarios, home users may not be interested in knowing how the devices and services work whereas what they need is just to carry out certain tasks. The possible solution to this problem is to encapsulate complex control interfaces of the individual device and present a simple and human-centric user interface to the users. The human-centric interface enables the technology-poor users to easily identify the tasks which he/she intends to finish and accomplish the tasks by simple operation, such as select one activity or click one icon. Furthermore, the user interface is desirable to adaptive in accordance with the users' preference and context information. For

example, the user moves from living room to bedroom, the interface should change accordingly. The challenges of designing a task-oriented and human-centric user interface lie in the following aspects:

- Capture users' intentions and form them into tasks. A task may involve users, data and services. For example, a typical task could be "Play the "Godfather" stored in Brown's PDA in the TV set of the living room".
- Associate the services provided by individual devices to different tasks. Two approaches are open to investigate: bottom-up and top-down.
- Present the tasks to the users as an intuitive manner. And when a new device joins the smart space, the service composition is dynamic and autonomous.
- The tasks representation is adaptive to the users' context.

3 A Lightweight Middleware to Enable Spontaneous Interaction in Smart Spaces

Section 2 studies the problems and challenges in designing smart spaces. In this section, we provide a solution to one of the critical issues aforementioned, i.e., the infrastructure to enable spontaneous device interaction in smart spaces. The problem of device interaction occurs when a mobile device intends to interact with the residential devices in an unfamiliar smart space, where the smart space could be homogeneous or heterogeneous, and the mobile device and the smart space have little a prior knowledge of each other. Our middleware solution separates the software infrastructure support into two parts, i.e., a TinyMiddleware (TM) which resides on the mobile device as well as the smart space, and a smart-space-specific mobile code for the smart space which can be discovered and run by the TinyMiddleware.

The smart-space-specific mobile code is the abstraction of a smart space which encapsulates its features. The features include how the residential devices are presented and invoked in form of services. The smart-space-specific mobile code is supposed to be dynamically updated when the residential devices join and exit the smart space. The TinyMidldleware is a platform which can be fit into mobile devices and enable the connection between the mobile device and the smart space. By discovering, downloading and executing the smart-space-specific mobile code, the mobile device can be extended with the ability to interoperate with the smart space as it can discover and access the available services offered by the residential devices.

Fig. 2 depicts two application scenarios which deploy the TinyMiddleware to achieve spontaneous interaction between a handheld device and the devices in one smart space. In Fig. 2(a), the devices in Smart Space 1 (SS1) are all UPnP-enabled which implies that they can interact with an UPnP-enabled mobile device. The handheld device intending to interact with the smart space, i.e., the PDA, is pre-installed with the TinyMiddleware. The TinyMiddleware enables the PDA to discover the devices in SS1 which are also TinyMiddleware-enabled, i.e., the media storage device. Upon the discovery of the media storage device, the TinyMiddleware will set up the connection between the PDA and the media storage device in order to download the SS1 specific mobile code. The mobile code extends the UPnP capability on the

PDA so that the PDA can control the UPnP-enabled media storage device and the plasma TV via the UPnP interface encapsulated in the mobile code.

Fig. 2(b) shows another scenario where the media storage device and the plasma TV are heterogeneous devices and the media storage device is TinyMiddleware-enabled. In this case, the TinyMiddleware contains a simple service registry and invocation interface that is implemented in the form of mobile code. Similarly, the PDA which is TinyMiddleware-enabled can download the mobile code via the media storage device. Any TinyMiddleware-enabled device in the SS2 can act as a proxy receiving the service invocation message from the PDA. The message will be forwarded to the target device and invoke the corresponding service.

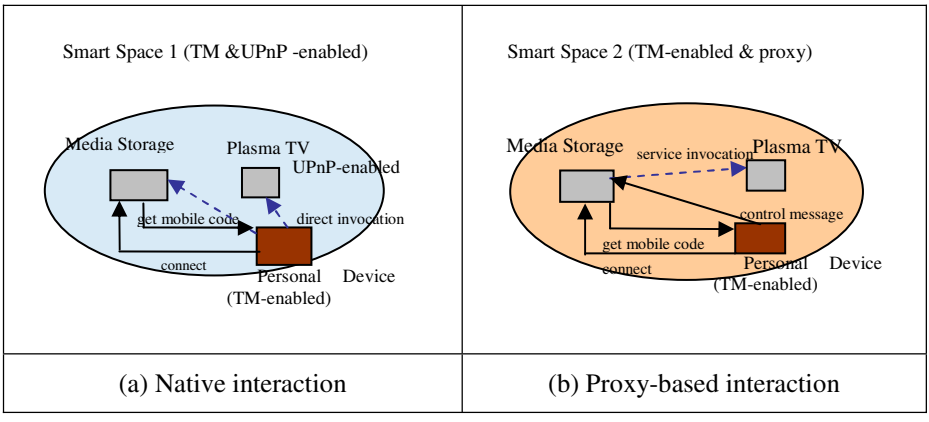


Fig. 2. Two Application Scenarios

4 Implementation of Prototypes

In this section, we will show three prototypes implemented for smart home, each of which tackles a specific issue on devices interaction in smart space. We have developed a service-oriented context-aware middleware for device interaction in smart homes. The service framework adopted is the OSGi core from ProSyst, Germany, it runs on a Residential Gateway developed by our group. Fig. 3 shows the smart home prototype leveraging on the OSGi-based Full Service Residential Gateway.

Various sensors, devices and computers have been integrated into the prototype through two solutions. One solution is using UPnP across the platform to enable the device advisement and discovery. UPnP handles the device interoperability and scalability quite well, however, the limitation is that all the devices should follow the same standard and there is no way to enable the interoperability between devices with different UPnP versions, let alone with the devices using other protocols. In order to tackle the issue of accommodating future devices, the mobile code has been used as a second solution to enable spontaneous interaction between ad-hoc devices.

Fig. 4 shows the running demo of spontaneous interaction with task-oriented and human-centric user interface. The leftmost notebook presents the personal device which is new to the smart home comprising a media server and a TV in living room.

Once the personal device is brought into the smart home, it will be prompted an intuitive user interface indicating the available tasks in the smart home. The user can perform the preferable task from the interface without knowing any details of the devices such as the communication protocols, control methods and data storage. The middleware consists of two components: a TinyMiddleware residing on the personal device and a smart home-specific mobile code to present the capability of individual smart space such as service registration, service discovery and service invocation. When the personal device enters the smart home, the mobile code will be transferred to the personal device and automatically run on top of the TinyMiddleware. Thus the personal device is enabled to interact with devices which it has no prior knowledge about in the smart home.

Fig. 3. OSGi-based Full Service Residential Gateway for Smart Home

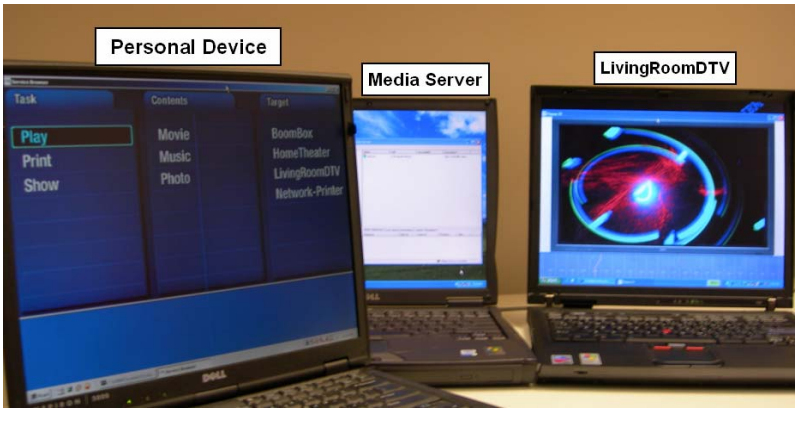


Fig. 4. Running Demo of Spontaneous Interaction with Human-centric User Interface

In order to make the use of the context information during the course of interactions, a context processing and management engine called Semantic Space has been implemented to manage the context in a single space [13][14]. Fig. 5 shows the experimental set-up for Semantic Space. It captures user's context as well as environmental context from various hardware and software sources. Semantic Web technologies are applied to support explicit representation, expressive querying, and flexible reasoning of contexts in smart spaces. Any context-aware application built on top of Semantic Space can pose context queries to the context engine and embed the Event-Condition-Action rules into its own logic. A context query can be "what's the state of TV and telephone in the living room" and a context-aware application could be "The volume of TV is lowered down when the telephone rings in the living room".

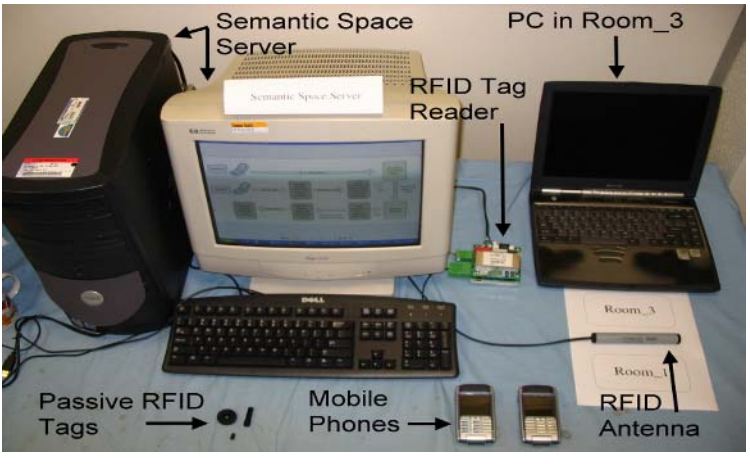


Fig. 5. Experimental set-up of Semantic Space

5 Summary

In this paper, we examined some key issues and challenges in designing smart spaces. Based on the discussions, we have proposed and implemented a lightweight middleware which handles the spontaneous device interaction in smart spaces. Furthermore, the middleware enables the applications to provide the task-oriented and human-centric user interface, which allows users to simply select what tasks they want to perform rather than how to perform. While the middleware developed by us showed some encouraging results, we plan to further validate its effectiveness by carrying out more trials using different application scenarios and performing trials of larger scale.

References

[1] E.D. Mynatt, I. Essa, and W. Rogers. Increasing the opportunities for ageing in place. *In Proceedings on the 2000 conference on universal usability*, pages 65-71, ACM Press, 2000.

[2] A.Dey, G. Abowd. *Towards a Better Understanding of Context and Context-Awareness*. Workshop on the What, Who, Where, When and How of Context-Awareness at CHI 2000.

[3] Massachusetts Institute of Technology, OXYGEN Project Overview. http://www.oxygen.lcs.mit.edu/Overview.html
[4] University of Illinois at Urbana-Champaign, Gaia: Actives Spaces for Ubiquitous Computing. http://gaia.cs.uiuc.edu/
[5] Carnegie Mellon University, Project Aura. http://www.cs.cmu.edu/~aura/
[6] Steven Czerwinski, Ben Y. Zhao, Todd Hodes, Anthony Joseph, and Randy Katz. An Architecture for a Secure Service Discovery Service. MOBICOM'99, Seattle Washington, USA, 1999.
[7] J.I. Hong, and J.A. Landy. An Infrastructure Approach to Context-Aware Computing. Human-Computer Interaction, Vol. 16, 2001.
[8] John J. Barton and Tim Kindberg. The Cooltown User Experience. Technical Reports HPL-2001-22, 2001. http://www.hpl.hp.com/techreports/2001/HPL-2001-22.pdf
[9] S. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, and E. Jansen, The Gate Tech Smart House: A Programmable Pervasive Space, IEEE Computer Magazine, pp. 50–60, March 2005.
[10] T. D. Hodes, R. H. Katz, E. Servan-Schreiber and L. A. Rowe. Composable ad-hoc Mobile Services for Universal Interaction. Mobile Computing and Networking. Mobile Computing and Networking, pp 1- 12, 1997.
[11] W. K. Edwards, M. W. Newman, J. Z. Sedivy, T. F. Smith and S. Izadi.Challenge: Recombinant Computing and the Speakeasy Approach," Proceedings of the Eighth ACM International Conference on Mobile Computing and Networking (MobiCom 2002). Atlanta, GA. September 23-28, 2002.
[12] T. Gu, X. H. Wang, H. K. Pung, D. Q. Zhang. An Ontology-based Context Model in Intelligent Environments. In Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2004), pp. 270-275. San Diego, California, USA, January 2004.
[13] X.H.Wang, D. Q. Zhang, J.Dong, C.Y. Chin and S. R. Hettiarachchi. Semantic Space: A Semantic Web Infrastructure for Smart Spaces. IEEE Pervasive Computing, Vol. 3, No. 2, 2004
[14] T. Gu, H. K. Pung, D. Q. Zhang. Towards an OSGi-Based Infrastructure for Context-Aware Applications in Smart Homes. IEEE Pervasive Computing, Vol. 3, Issue 4, 2004

A New Model to Optimize the Cost Efficiency of Broadcast in Mobile Ad Hoc Networks*

Xin Li[1], Shanzhi Chen[1], Zhen Qin[2], and Bo Hu[1]

[1] Broadband Network Research Center, State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing, 100876, China
lixin@bupt.edu.cn, hubo@bupt.edu.cn, chenshanzhi@yahoo.com.cn
[2] Institute of China Electronic System Engineering Corporation, Beijing, 100039, China
qinzhen_2005@hotmail.com

Abstract. In this paper an analytical model is established to optimize the cost efficiency of broadcast operations while guarantee required RE and delay. To facilitate analysis on cost, a new metric—WCOF (Wireless Channel Occupation Frequency), is introduced to measure the average cost per node incurred by a round of broadcast. For simplicity delay is measured by the least number of broadcasts to achieve required RE. Performance of Counter based and distance based schemes are investigated in detail using the new model. Analytical results show the relation between the cost efficiency and delay of broadcast schemes and configuration of network and broadcast parameters. It is proved that the new model can be used to choose proper broadcast scheme and parameters for optimal performance.

1 Introduction

Network-wide broadcast is a common operation in ad hoc networks. But unnecessary retransmissions incurred by full scale flooding could result in severe broadcast storm problem [1]. A considerable number of innovated broadcast schemes [2-4] have been established to address this problem. SRB and RE are two widely used metrics to evaluate [5,6] the performance of broadcast mechanisms. RE indicates the coverage ability of broadcast scheme. It is the number of mobile nodes receiving the broadcast message divided by the total number of mobile nodes. SRB is used to measure to what extent the broadcast problem is suppressed. It is the proportion between the number of nodes suppressed from retransmitting the broadcast and the total number of nodes. In most of the related studies [7, 8], SRB is generally used as the cost criterion for broadcasting and RE is often assumed to be 100%.

But such methodology is not reliable in optimizing the expense of broadcast operations. First, in wireless networks both transmission and reception of broadcast message will consume bandwidth resource, SRB only focuses on the former but entirely neglects the later. Second, RE is closely related with network scenario and not always 100%. Third, different operations may require diverse broadcast capacity, such as RE, delay and so on. So optimization of broadcast efficiency should be application oriented.

* This work is supported by the state 863 high-tech program of China (the Project Num is 2003AA121530 and 2005AA121630).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 260 – 269, 2006.

Our goal in this paper is to establish a rational system to optimize the cost efficiency of broadcast operations while guarantee required RE and delay. To facilitate analysis on cost, a new metric—WCOF (Wireless Channel Occupation Frequency), is introduced to measure the average cost per node incurred in a round of broadcast. For simplicity WCOF is measured by the number of receptions and transmissions of the same broadcast message. We also take multiple rounds of broadcast which could guarantee the required RE into account. At the expense of longer delay, multiple rounds of low coverage broadcast could also effectively alleviate broadcast problem. For simplicity delay is measured by the least number of broadcast to achieve required RE. Counter based and distance based schemes are investigated in detail. The analytical results show that the new model can be used to choose proper broadcast scheme and parameters for optimal broadcast performance.

2 The Analytical Model

2.1 General Optimization Model

Theorem 1. The RE oriented model for optimizing the cost efficiency of broadcast in mobile Ad Hoc networks is:

$$\begin{cases} N_B = ceil(\log_{1-RE}^{1-RE_R}) & RE < RE_R \\ N_B = 1 & RE \geq RE_R \\ WCOF_N = N_B \cdot WCOF \end{cases} \quad (1)$$

RE_R is the required reachability. RE is the reachability of the selected scheme in one round of broadcast. N_B is the least number of broadcasts needed to achieve RE_R. $WCOF$ is the cost incurred in one round of broadcast. $WCOF_N$ is the cost of N_B round of broadcast. It is clear that for certain broadcast scheme lower RE would result in higher N_B. But whether it will lead to poorer cost efficiency than the other configurations is of great interest to us. Our goal in this paper is to provide a new method but not a complete solution for optimization of broadcast operation. So only counter based and distance based schemes are analyzed in detail.

2.2 Assumptions

Brad Williams[9] has established models to predict the probability for a node to retransmit the broadcast. A few of his techniques have been refered in our models. Differences lies in that we take the effect of MAC layer into account and RE is not assumed to be 100%.

Some assumptions in our model are taken and enumerated below:

- All the nodes are randomly and uniformly distributed. IEEE 802.11[10] is used as the MAC layer protocol. All the nodes have the same transmission range R.
- The broadcast scheme is carried out in a “silent” network environment. There is no concurrent data traffic or broadcast initiated by the other sources.
- Source nodes of the broadcast packets are randomly chosen.
- Each node in the network should rebroadcast with the same probability P_{RT}.

2.3 Optimization Model for Counter Based Scheme

In counter based scheme, a node initiates a counter with a value of one and sets a *RAD* (randomly chosen between *0* and T_{max} seconds) upon reception of a new broadcast message. During the *RAD*, the counter is incremented by one every time it receives a redundant message. If the counter is not less than a threshold C ($C \in Z^+$, $C \geq 2$), when the *RAD* timer expires, the node should drop this message and decline to transmit it. Otherwise the message should be retransmitted.

Theorem 2. In counter based scheme the probability that a random node *u* retransmits a broadcast packet is:

$$P_{RT} = RE.\left[\frac{\pi R^2}{A_{net}}.\sum_{i=0}^{C-2} P_H{}^i.\left(1-\frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-2-i} + \left(1-\frac{\pi R^2}{A_{net}}\right).\sum_{i=0}^{C-2}\left(\frac{\pi R^2 P_{RT}}{2A_{net}}\right)^i.\left(1-\frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-3-i}\right] \quad (2)$$

Proof: in counter based scheme, for a random node u to retransmit the broadcast packet, two preconditions must be satisfied:

A. *u* must receive the broadcast packet.
B. At most *C-2* nodes within the transmission range of *u* retransmit the packet after u's reception of the broadcast but before *u*'s *RAD* expire.
The probability that u is covered by a broadcast is equal to *RE*. So:
$P_{RT} = P(A \cap B) = P(B/A)P_A = P_A \cdot P_B = RE \cdot P_B$
For a random node *v* to be able to suppress the retransmission of *u* (let this event be H), 3 events must occur:
H1. *v* must lie within the transmission range of *u*.
H2. *v* must retransmit.
H3. *RAD* of *v* must expire before *u*.

Let the area of network be A_{net}, the probability that *u* and *v* are neighbors should be:

$$P(H1) = \frac{\pi R^2}{A_{net}} \quad (2\text{-}1)$$

It is also assumed that each node has the same probability of retransmitting the broadcast packet, so we can achieve:

$$P(H2/H1) = P_{RT} \quad (2\text{-}2)$$

We do not assume that *u* and *v* receive the broadcast at the same time. But doubtless *u* and *v* do not receive their first message from each other. Suppose *u* initiates its *RADu* at *T*, because they are adjacent neighbors, the event that *v* rebroadcast can only occur within (*T, T+ RADu + RADv*). Expectations of *RADu* and *RADv* are the same. The probability that *v*'s *RAD* expires before *u*'s is 1/2. Thus

$$P(H3/H2 \cap H1) = \frac{1}{2} \quad (2\text{-}3)$$

With (1-1) to (1-3) the probability for event H to occur should be $\frac{\pi R^2 P_{RT}}{2A_{net}}$.

Considering the position of *u* to source node *S*, *B* could be divided into 2 independent events:

B1: *u* receives the broadcast packet from *S*.
B2: *u* receives the broadcast packet from node other than *S*.

In event B1 only *N-2* nodes could have chance to suppress retransmission of *u*, but in event B2 the number should be *N-3*. Thus:

$$P_{B1} = \frac{\pi R^2}{A_{net}} \sum_{i=0}^{C-2} {P_H}^i \left(1 - P_H\right)^{N-2-i} \tag{2-4}$$

$$P_{B2} = \left(1 - \frac{\pi R^2}{A_{net}}\right) \sum_{i=0}^{C-2} {P_H}^i \left(1 - P_H\right)^{N-3-i} \tag{2-5}$$

Now we could achieve:

$$P_{RT} = RE.P_B = RE.\left[\frac{\pi R^2}{A_{net}} \sum_{i=0}^{C-2} {P_H}^i \left(1 - \frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-2-i} + \left(1 - \frac{\pi R^2}{A_{net}}\right) \sum_{i=0}^{C-2} \left(\frac{\pi R^2 P_{RT}}{2A_{net}}\right)^i \left(1 - \frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-3-i}\right]$$

Theorem 3. Reachability of counter based broadcast scheme is:

$$RE = \frac{\pi R^2}{A_{net}} + \left[1 - \left(1 - \frac{\pi R^2}{A_{net}} P_{RT}\right)^{N-2}\right]\left(1 - \frac{\pi R^2}{A_{net}}\right) \tag{3}$$

Proof: For a random node *u* to receive a broadcast, it must lie within the transmission range of *S* (with probability $\frac{\pi R^2}{A_{net}}$) or receive the broadcast packet from other node than *S*. Because the probability for a random node *v* (except *S* and *u*) to be within *u*'s transmission range and retransmit the broadcast packet is $\frac{\pi R^2}{A_{net}} P_{RT}$, the probability that at least one of *u*'s neighbors retransmit the broadcast should be $1 - \left(1 - \frac{\pi R^2}{A_{net}} P_{RT}\right)^{N-2}$.

So theorem 3 is proved.

Theorem 4. For counter based scheme when *N*, A_{net}, *R* and *C* is determined *RE, SRB* and *WCOF* could be achieved by solving the system of equations:

$$\begin{cases} P_{RT} = RE.\left[\frac{\pi R^2}{A_{net}} \sum_{i=0}^{C-2} \left(P_H\right)^i \left(1 - \frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-2-i} + \left(1 - \frac{\pi R^2}{A_{net}}\right) \sum_{i=0}^{C-2} \left(\frac{\pi R^2 P_{RT}}{2A_{net}}\right)^i \left(1 - \frac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-3-i}\right] \\ RE = \frac{\pi R^2}{A_{net}} + \left[1 - \left(1 - \frac{\pi R^2}{A_{net}}.P_{RT}\right)^{N-2}\right]\left(1 - \frac{\pi R^2}{A_{net}}\right) \\ SRB = 1 - P_{RT} \\ WCOF = \frac{\pi R^2}{A_{net}} P_{RT} N \end{cases} \tag{4}$$

With (1) and (4) we can achieve the optimization model for counter based scheme with given *RE*:

$$\begin{cases} P_{RT} = RE.\left[\dfrac{\pi R^2}{A_{net}}\sum_{i=0}^{C-2}(P_H)^i\left(1-\dfrac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-2-i} + \left(1-\dfrac{\pi R^2}{A_{net}}\right)\sum_{i=0}^{C-2}\left(\dfrac{\pi R^2 P_{RT}}{2A_{net}}\right)^i\left(1-\dfrac{\pi R^2 P_{RT}}{2A_{net}}\right)^{N-3-i}\right] \\ RE = \dfrac{\pi R^2}{A_{net}} + \left[1-\left(1-\dfrac{\pi R^2}{A_{net}}.P_{RT}\right)^{N-2}\right]\left(1-\dfrac{\pi R^2}{A_{net}}\right) \\ SRB_N = (1-P_{RT})^N \\ WCOF = \dfrac{\pi R^2}{A_{net}}P_{RT}N \\ N_B = ceil(\log_{1-RE}^{1-RE_R}) \qquad RE < RE_R \\ N_B = 1 \qquad RE \geq RE_R \\ WCOF_N = N_B \cdot WCOF \end{cases} \tag{5}$$

2.4 Optimization Model for Distance Based Scheme

In the distance based scheme, upon reception of a previously unseen packet, a *RAD* is initiated and duplicate packets received are cached. When the *RAD* expires, the distances between all source nodes locations and the assessing node location are examined to see if any source node is closer than a threshold distance value *D* ($D \in R^+$, $R>D\geq 0$). If any source node is within the threshold distance, the assessing node will not rebroadcast the packet.

Theorem 5. In distance based scheme the probability that a random node *u* retransmit the broadcast packet is:

$$P_{RT} = RE\left(1-\frac{d^2}{R^2}\right)\left[\left(1-\frac{1}{2}P_{RT}\frac{\pi d^2}{A_{net}}\right)^{N-2}\frac{\pi R^2}{A_{net}} + \left(1-\frac{1}{2}P_{RT}\frac{\pi d^2}{A_{net}}\right)^{N-3}\left(1-\frac{\pi R^2}{A_{net}}\right)\right] \tag{6}$$

Proof: In the distance based scheme, for a random node u to retransmit a broadcast packet, three events must occur:

A. *u* has received the broadcast packet.
B. Sender *v* lies within the annulus area centered at *u*, with the radius of inner circle equal to *D* and the radius of the outer circle equal to *R*.
C. No node within the circular with radius *D* centered at u retransmits the broadcast packet before *u*'s *RAD* expires.

So the probability that *u* retransmit the broadcast packet is equal to:

$$P_{RT} = P(A \cap B \cap C) = P(C / A \cap B)\cdot P(B / A)\cdot P_A$$

The probability for a random node to be within *u*'s distance *D* and retransmit is $P_{RT}\dfrac{\pi D^2}{A_{net}}$. By definition of distance based scheme, if a node retransmits, all of its

neighbors within distance D would be suppressed from rebroadcast, no matter whether they have received the broadcast packet from the same source or at the same time. In a round of broadcast the probability that a random neighbor within u's distance of D retransmits before u's RAD expires is 1/2. If u receives the broadcast packet form S, the probability that u is not suppressed from rebroadcast should be $\left(1-\frac{1}{2}P_{RT}\frac{\pi D^2}{A_{net}}\right)^{N-2}\frac{\pi R^2}{A_{net}}$. If u is out of the transmission range of S, this probability should be $\left(1-\frac{1}{2}P_{RT}\frac{\pi d^2}{A_{net}}\right)^{N-3}\left(1-\frac{\pi R^2}{A_{net}}\right)$. Thus,

$$\mathrm{P}(\mathrm{C/B}\cap\mathrm{A})=\left(1-\frac{1}{2}P_{RT}\frac{\pi d^2}{A_{net}}\right)^{N-2}\frac{\pi R^2}{A_{net}}+\left(1-\frac{1}{2}P_{RT}\frac{\pi d^2}{A_{net}}\right)^{N-3}\left(1-\frac{\pi R^2}{A_{net}}\right)$$

It is known that P_A is equal to RE, and

$$P(B/A)=\frac{\pi R^2-\pi D^2}{\pi R^2}=1-\frac{D^2}{R^2}$$

Because $P_{RT}=P(C/A\cap B)\cdot P(B/A)\cdot P_A$, so the theorem is proved.

Theorem 6. In distance based scheme the reachability of broadcast should be:

$$RE=\frac{\pi R^2}{A_{net}}+\left[1-\left(1-\frac{\pi R^2}{A_{net}}P_{RT}\right)^{N-2}\right]\left(1-\frac{\pi R^2}{A_{net}}\right) \tag{7}$$

The proof of theorem 6 could refer to that of theorem 2.

Theorem 7. For distance based scheme when N, A_{net}, R and D are determined, RE, SRB and $WCOF$ can be achieved by solving the system of nonlinear equations:

$$\begin{cases} RE=\dfrac{\pi R^2}{A_{net}}+\left[1-\left(1-\dfrac{\pi R^2}{A_{net}}.P_{RT}\right)^{N-2}\right].\left(1-\dfrac{\pi R^2}{A_{net}}\right) \\ P_{RT}=RE.\left(1-\dfrac{d^2}{R^2}\right).\left[\left(1-\dfrac{1}{2}.P_{RT}.\dfrac{\pi d^2}{A_{net}}\right)^{N-2}.\dfrac{\pi R^2}{A_{net}}+\left(1-\dfrac{1}{2}.P_{RT}.\dfrac{\pi d^2}{A_{net}}\right)^{N-3}.\left(1-\dfrac{\pi R^2}{A_{net}}\right)\right] \\ SRB=1-P_{RT} \\ WCOF=\dfrac{\pi R^2}{A_{net}}P_{RT}N \end{cases} \tag{8}$$

The proof of theorem 7 is just the same as that of theorem 4.

With (1), (6), (7) and (8) we can achieve the optimization model for distance based scheme with given RE:

$$
\begin{cases}
P_{RT} = RE.\left(1-\dfrac{d^2}{R^2}\right).\left[\left(1-\dfrac{1}{2}.P_{RT}.\dfrac{\pi d^2}{A_{net}}\right)^{N-2}.\dfrac{\pi R^2}{A_{net}}+\left(1-\dfrac{1}{2}.P_{RT}.\dfrac{\pi d^2}{A_{net}}\right)^{N-3}.\left(1-\dfrac{\pi R^2}{A_{net}}\right)\right] \\
RE = \dfrac{\pi R^2}{A_{net}}+\left[1-\left(1-\dfrac{\pi R^2}{A_{net}}.P_{RT}\right)^{N-2}\right].\left(1-\dfrac{\pi R^2}{A_{net}}\right) \\
SRB_N = (1-P_{RT})^N \\
WCOF = \dfrac{\pi R^2}{A_{net}} P_{RT} N \\
N_B = ceil(\log_{1-RE}^{1-RE_R}) \qquad RE < RE_R \\
N_B = 1 \qquad RE \geq RE_R \\
WCOF_N = N_B \cdot WCOF
\end{cases}
\tag{9}
$$

2.5 Analytical Results

In this section counter based and distance based schemes are investigated in detail. Thirteen counter thresholds ranging from 2 to 14 are adopted in the analysis of counter based scheme. For distance based scheme 9 thresholds ranging from 10 to 90 meters are selected. The number of nodes in the network varies from 20 to 140. All nodes have the same transmission range of 100 meters. They are randomly and uniformly distributed within a 500m*500m square field.

Fig.1 and Fig.2 show the analytical results in one round of broadcast. Fig.1a demonstrates that the coverage ability of counter based scheme. While in fig.2a *RE* of distance based scheme drops sharply when *N* is less than 60 or *D* exceeds 50 meters. In such circumstances the assumption of 100% RE is far from reality.

Fig1.b and fig1.c show that when *N* is given *SRB* could be used as an indication to the cost efficiency. But when *C* is fixed, higher *N* could lead to sharp improvement in *SRB* and slight increment in *WCOF*. That means although the proportion of retransmitting nodes decreases, the average number of retransmitting neighbors per node still increases and lead more resources to be consumed. In such scenarios *SRB* could not be used to indicate the cost efficiency of broadcast.

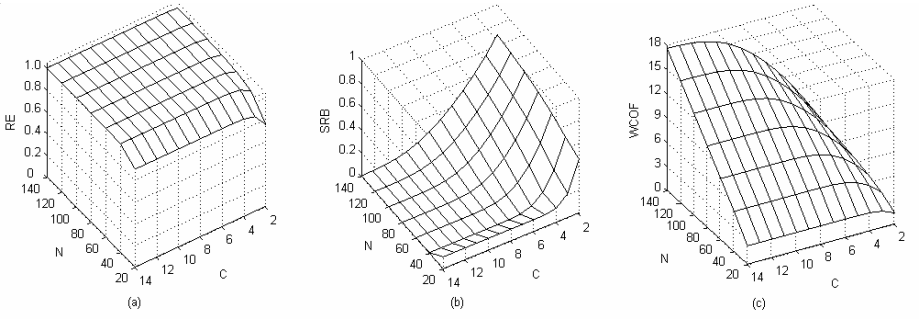


Fig. 1. Analytical results of counter based scheme in one round of broadcast

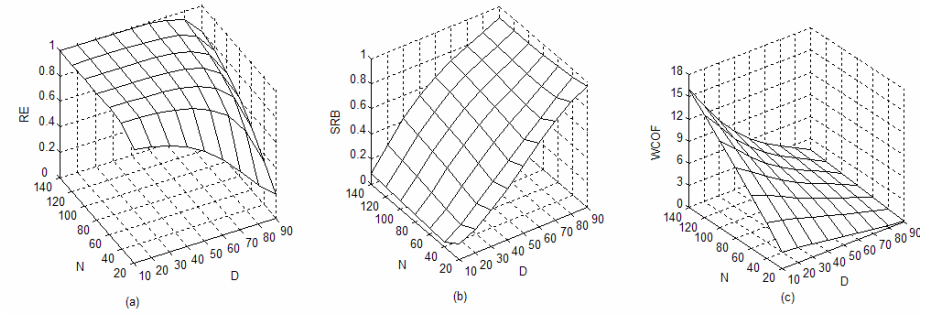


Fig. 2. Analytical results of distance based scheme in one round of broadcast

To facilitate the analysis of distance based scheme we name the annulus surrounded by the circles with radius *R* and *D* centered at a node as retransmitting zone, or *RTZ* in brief. Fig2.a demonstrates that in connected but sparse networks *RE* still drops sharply with increment in *D*. That is because larger *D* leads to smaller *RTZ* and decrease the probability that at least one node within RTZ will retransmit. Fig.2b and fig.2c show that both *SRB* and cost efficiency are proportional to *D* in networks with fixed number of nodes. But it is also noticed that the high cost efficiency and *SRB* with large *D* are partially achieved at the expense of broadcast coverage. When *D* is given SRB would keep rather stable while *WCOF* still be proportional to node density. That is because SRB is close related to the proportion between *RTZ* and πR^2, while higher node density will no doubt increase the number of retransmitting nodes within *RTZ*.

It is also observed in Fig1 and Fig2 that there is a surge of *SRB* when *N* is equal to 20. According to the conclusions of [11], when *N* is 20 some nodes may be partitioned from the other part of the network. But that should not be regarded as an improvement in broadcast performance.

Fig.3 and fig.4 show the analytical results when 99% RE is guaranteed. In some scenarios the source node may need to broadcast several times and incur higher delay. For simplicity we N_B is taken as the reference of delay.

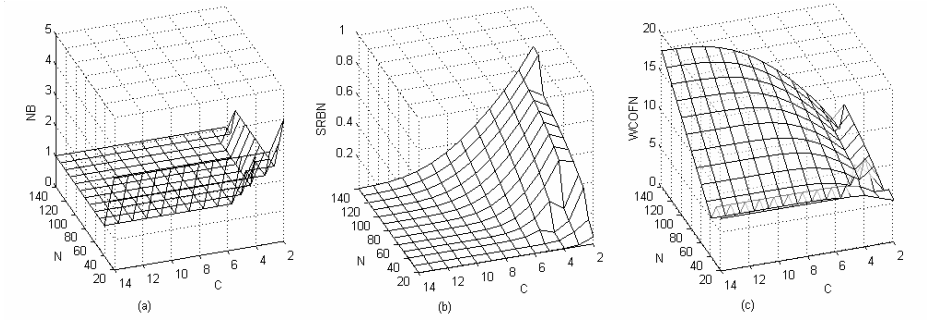


Fig. 3. Analytical results of counter based scheme with 99% *RE*

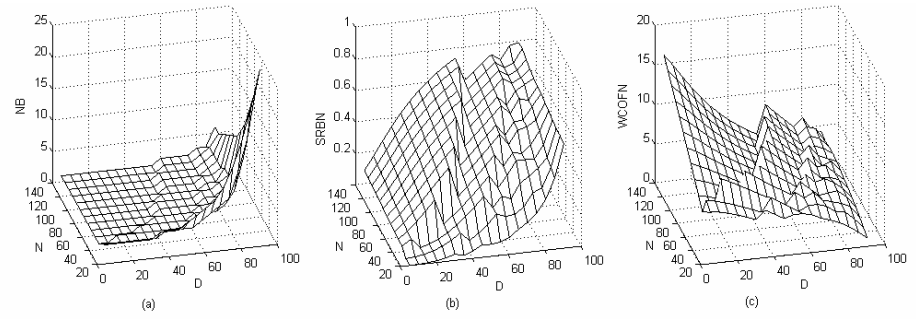


Fig. 4. Analytical results of distance based scheme with 99% *RE*

Fig.3a and fig.4a show that N_B is inversely proportional to RE in fig.1a and fig.2a. For delay constraint applications they could be used to confine the scope of available broadcast parameters.

Multiple rounds of broadcast will result in higher proportion of retransmitting nodes and consume more bandwidth resource, which is confirmed by the waves of SRB and WCOF in fig.3 and fig.4. These fluctuated curves are more valuable and helpful than their counterparts in fig.1 and fig.2 when selecting broadcast schemes and parameters to achieve optimized performance. For example, to minimize the cost of broadcast operation which require 99% RE and at most 2 rounds of broadcast in a network with 100 nodes, distance based scheme with D set to 70 meters would be the best choice.

It is also observed that multiple rounds of broadcast are not always costly. Fig.4c and fig.2a illustrate that although configuring distance threshold to 90 meters always leads to the maximum number of rebroadcasts, it is still the most cost efficient configuration.

If the requirement on delay is not so strict or very loose, we may have multiple choices to optimize the cost efficiency of broadcast. For example, the broadcast may need to be carried out in one round or multiple rounds depending on its configuration. The final decision just depends. In heavy loaded networks sharing the burden of broadcast into several rounds could effectively alleviate its interference to the other applications and improve the macro-performance of network. But in silent networks complete the broadcast in one round may be preferred.

3 Conclusions

In this paper a theoretical model is established to facilitate optimization on the cost efficiency of broadcast in mobile Ad Hoc networks while guaranteeing the required RE and delay. A new metric—WCOF, is introduced to indicate the cost per node in one round of broadcast. Using the new model performance of counter based and distance based schemes with and without requirement on RE are investigated in detail. Some important conclusions are drawn from the analytical results. First, SRB is not an appropriate metric for evaluating the cost efficiency of broadcast. Second, for applications not sensible to delay, to achieve the required RE multiple rounds of broadcast with low coverage may be more cost efficient than one round of broadcast

with high coverage. Third, choice of broadcast scheme and parameters is a tradeoff between reachability, delay, cost efficiency and other factors. So the improvement on the existed broadcast schemes and invention of new mechanisms should also be application oriented.

References

1. Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu. The Broadcast Storm Problem in a Mobile Ad Hoc Network. In Wireless Networks, pages 153-167. 2002.
2. M.T.Sun, W.Feng, T.H.Lai, Location Aided Broadcast in Wireless Ad hoc Networks. Proc. IEEE WCNC 2002, pp.597-602, Orlando, FL. March 2002.
3. YSasson, D.Cavin and ASchiper, Probabilistic Broadcast for Flooding in Wireless Mobile Ad hoc Networks. Technical Report. IC/2002/54.
4. J.Sucec and LManic, An efficient distributed network-wide broadcast algorithm for mobile ad hoc network. CAP Technical Report 248 - Rutgns University, September 2000.
5. P. Yao, E. Krohne, and T. Camp, Performance Comparison of Geocast Routing Protocols for a MANET, Proceedings of the 13th IEEE International Conference on Computer Communications and Networks (IC^3N), pp. 213-220, 2004.
6. B. Williams and T. Camp. Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks, Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '02), pp.194-205, 2002.
7. Hao Zhang, Zhong-Ping Jiang, Analysis of two ad hoc broadcasting protocols, Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE Volume 2, 21-25 March 2004 Page(s):808 - 812 Vol.2
8. Hao Zhang, Zhong-Ping Jiang, Performance analysis of broadcasting schemes in mobile ad hoc networks, Communications Letters, IEEE Volume 8, Issue 12, Dec. 2004 Page(s):718 - 720 Digital Object Identifier 10.1109/LCOMM.2004.837658
9. Williams. B; Mehta. D.P; Camp, T; Navidi, W, Predictive models to rebroadcast in mobile ad hoc networks, Mobile Computing, IEEE Transactions on Volume 3, Issue 3, July-Aug. 2004 Page(s):295 - 303 Digital Object Identifier 10.1109/TMC.2004.25
10. IEEE, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11-1997, 1997.
11. G. Ferrari and O.K. Tonguz, Minimum number of neighbors for fully connected uniform ad hoc wireless networks, IEEE International Conference on Communications (ICC'04), Paris, France, 2004

Joint Power Control and Channel Assignment Algorithms for Wireless Ad Hoc Networks*

Yuan Zhang and Shouning Qu

School of Information Science and Engineering, Jinan University, Jinan 250022, China
yzhang@ujn.edu.cn

Abstract. In this paper joint power control and channel allocation algorithms are proposed for wireless ad hoc networks. The performances of the algorithms are evaluated in both FIFO queue system and Priority queue system. To increase network capacity with time-sensitive data transmission, we incorporate the back off scheme into these systems. Through in-depth simulations, the network throughput and the probability of successful transmission of time-sensitive traffic are examined by varying several parameters. We conclude by presenting the advantages and disadvantages of different scenarios.

Keywords: power control, channel assignment, time-sensitive data, back off algorithm.

1 Introduction

In most ad hoc networks, performance depends heavily on power consumption associated with the transmitting signal. Efficient use of power at each node can prolong the lifetime of overall network and reduce the co-channel interference between links. Together with the power control scheme, optimal channel allocation can maximize the utilization of the communication channel.

Various studies have been done regarding the implementation of power control in ad hoc networks in the single channel environment [1, 2, 3]. Also channel assignment algorithms have been studied in several papers [4, 5]. Most of the research focused on continuous traffic, which is dominant in current generation wireless networks.

In ad hoc networks there are usually more than one non-interfering channels available. Therefore, proper channel allocation and power control should be considered at the same time. Both power control and channel allocation have to be implemented in a distributed manner.

Next generation wireless networks are being designed to support packetized data traffic instead of continuous traffic. For time-sensitive applications such as streaming audio or multimedia, the packets have time constraints concerning delivery deadline.

In this paper, power control with channel assignment algorithms are proposed for ad hoc networks with packetized data traffic. The performance of the algorithms are evaluated in terms of the total throughput, which is represented by the number of packets per power usage, total number of time-sensitive data packets with deadline, the

* This work is supported by China National 863 Program under Grant No. 2002AA4Z3240.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 270 – 279, 2006.

probability of successful time-sensitive traffic transmission, and the fairness of the algorithms. The data traffic with arrival rate of λ are stored in the queue of each transmitter node and transmitted in each time slot.

Two types of queues are considered here, FIFO and Priority queues. FIFO queue tries to transmit the earliest data packet in the queue, whereas Priority queue tries to transmit data packet with Priority first. In this paper time-sensitive data packet has priority over normal data packets. The performance for each case is compared to each other.

For channel selection, the least interference method in [6] is used. When the transmitter observes high interference in the channel, it recognizes that it will have to consume much power to overcome the interference and transmit a packet successfully to the receiver. Therefore, it might be better to back off if the data is not time-sensitive and the number of packets in the buffer is below the average of neighboring nodes. This keeps the interference level low, enabling the network to admit a new link with time-sensitive data packets later. This back-off algorithm is also incorporated in both FIFO and Priority based data packet selection algorithms.

The rest of the paper is organized as follows. We set up the system model in the next section. In Section 3 we propose and analyze our combination algorithm in detail. Simulation results are presented and discussed in Section 4. Finally, we conclude our work in Section 5.

2 Model Framework

The design objective is to minimize the power consumption and increase network capacity with tolerable time-sensitive data transmission while maintaining link QoS. In this model, the minimum required SINR in each link represents QoS. The data packets in the queue are transmitted over time slot of duration T. The arrival of packets follows the Poisson distribution with arrival rate λ. In each time slot the transmit power of each node is determined by the distributed autonomous power control algorithm in [2]. We assume that the convergence time of this scheme is short enough compared to the time slot T. So the time dynamics of the transmit power of each node is ignored. Only the values after convergence are used in each time slot.

2.1 Node

There are 10 transmitter-receiver pairs. The locations of the nodes are as shown in figure 1. The traffic is unidirectional from the transmitter to the receiver.

2.2 Data Traffic and Buffer

There are two types of data packets. One is time-insensitive data packet without any time constraints. The other is time-sensitive packet with deadline T. This data packet should be delivered within T time slots from the time it has arrived. After that this packet will be useless (not counted in the throughput calculation). Time-insensitive data packet will arrive at the transmitter node with arrival rate of $(1-f)\lambda$, here f means the proportion of time-sensitive traffic. Time-sensitive packet arrives with the arrival rate of $f\lambda$ (The arrival rate of λ means λ packets are expected to arrive in each time slot).

There is an infinite length buffer at each transmitter node. This buffer is implemented based on either FIFO or Priority. In FIFO data packet that has arrived first

will be selected first for transmission, whereas in Priority queue the data with priority will be selected for the transmission first. The two algorithms using FIFO and Priority buffer are compared.

2.3 Wireless Channel Model

There are K non-interfering channels available. We assume that the channel gain is determined by the simple path loss model in [7] and suffers Rayleigh fading. The power loss exponent of simple path loss model is assumed to be 4. We experiment both cases when there is an independent Rayleigh fading in each time slot and when the channel is stationary for fairness simulation.

2.4 Power Control

There are maximum transmit power constraints in the transmitters, $P_{\max}$. Each link must maintain minimum required SINR for reliable data transmission, which is denoted by γ_i. There is a noise at each receiver, η, which is assumed to be the same for all links.

For better spectrum utilization, the back-off algorithm is used in both simulations using FIFO and Priority buffers. Then the performance is compared to when the back-off algorithm is not used in both cases. In each time slot every transmitter tries to send data packet in the buffer according to the least interference algorithm in [1].

2.5 Objective of the Algorithm

The goal of the algorithms is to maximize the throughput of the networks while maintaining SINR requirements on every link. The throughput is represented by the ratio of the total number of packets transmitted to the total transmit power. Also time-sensitive data packet should be delivered within T time slots after arrival. The ratio of successful transmission of the time-sensitive data to the failed transmission of time-sensitive data should be high. This ratio is one of the factors determining QoS of the networks.

The value function is as follows

$$V = \frac{\sum_i packets}{\sum_i transmitPW} + \alpha \frac{\sum_i transmittedTSpackets}{\sum_i TSpackets} \tag{1}$$

Here α denotes the weighting factor for the probability of successful time-sensitive (TS) traffic. The number of packets for each i^{th} node is added together to get the total number of packets in the numerator and denominator.

3 Problem Formulation

3.1 Power Control Algorithm

The feasibility and the optimal power levels for N transmitter-receiver pair links within a single channel are exactly described in [2]. When there are more than one

non-interfering channels, this algorithm is applied for the links sharing the same channel. The feasibility test is used as the admission control method.

Let G be the gain matrix: G_{ij}^{l} is the power gain from jth link transmitter to ith link receiver of channel l. G_{ij} is the product of path loss, modeled by $\frac{P_r}{P_t} = k\left[\frac{d_0}{d}\right]^{\gamma}$ [7] and Rayleigh fading. ρ is a N×1 vector containing required SINR for the links. P^l is a power vector using channel *l*. P_i^l is the transmitter power of the i[th] link. *N* is noise to the receiver of the link. Then SINR of the i[th] link of channel *l* is given by:

$$SINR_i = \frac{G_{ii}^l P_i^l}{N + \sum_{j \neq i} G_{ij}^l P_j^l} \tag{2}$$

So to maintain QoS, SINR should satisfy the following:

$$SINR_i \geq \rho_i,\ i = 1,2,\ldots,N \tag{3}$$

Rewriting the inequality of Eq. (3) in the matrix form, we get $\left(I - F^l\right)P^l \geq U^l$ (component wise), where $U^l = \left(\frac{\rho_1 N}{G_{11}^l}, \frac{\rho_2 N}{G_{22}^l}, \ldots, \frac{\rho_N N}{G_{NN}^l}\right)$ is the column vector of normalized noise powers, and *F* is the matrix with entries $F_{ij} = \frac{\rho_i G_{ij}^l}{G_{ii}} I,\ \{i \neq j\}$, where I{ } is indicator function [1].

It is shown in [8] that positive P^l exists if the maximum given value of F^l is less than 1. In this case we can get this P^l by iteration of

$$P^l(k+l) = F^l P^l(k) + U^l \tag{4}$$

which can be achieved by autonomous control. Under this circumstance, the power vector $P^{l^*} = \left(I - F^l\right)^{-1} U^l$ can be shown to be the Pareto optimal power levels [9].

3.2 Channel Allocation

In ad hoc networks, the channel allocation should be implemented in a distributed way. The optimal solution shown in [6] minimizes the total transmit power, but it is possible only when there is central control with perfect knowledge of the channel and the power level of every node. The algorithm based on the least interference criterion is used here to allocate channels. The idea behind this is that selecting the channel with the least interference level would require the least transmission power to maintain the same SINR. The transmission power of the current link is determined by both the interference level and channel gain of the current link. The required power level of that link for channel *l* can be represented by the parameter

$$L\_Criteria_i^l = \frac{G_{ii}}{I_i + N_0} \tag{5}$$

In the channel probing stage, these values are compared, and the channel with the maximum value is selected for admitting new link to the network. As the number of channel increases, the throughput of the network (the number of links divided by total power usage) approaches to the optimal solution in [6].

3.3 Data Selection from the Queue

In most research FIFO queue is used. In this paper the effect of Priority queue is also examined for tolerable transmission of time-sensitive (TS) packets. The packet arrives with Poisson distribution. In each time slot the transmitter checks the data packets which have arrived but not yet process from the time slot 1 up to the current time slot. Then it selects the most imminent time-sensitive data for the transmission. This will reduce the probability of TS packet transmission failure. The data packets from selected links by power control algorithm are removed from the queue.

3.4 Back Off Algorithm

When the transmitter observes high interference in the channel on admitting a new link into the network, it will spend a lot of power to overcome the interference. Even though it is feasible to admit this link, it might be better to back off and yield the channel to other remaining links waiting for the interference to subside. This approached is introduced in [10] based on the number of packets in the buffer.

When there are TS data packets with deadlines, the back-off of the link which has time-insensitive packet can increase the successful transmission of TS packets within deadlines. We incorporate this algorithm in both FIFO and Priority queue cases.

The criteria for backing off from channel access should be determined cautiously. If it is too loose, then the number of link in the network will decrease because several links will back off even though it is admissible without causing much interference to the TS data packets. Thus total network throughput will reduce. If it is too strict, then almost no link will back off, and then the performance will be the same as when there is no back-off algorithm. Sometimes backing off the current link will not increase the chance of admitting links with TS packets in the future.

If we know the overall channel gain matrix, then it is possible to choose the back off time. That is, to back off links when the remaining links have small G_{jj} and the current link i has large G_{ji}. This means if link i increases its transmit power, then link j will not be feasible as the cross gain G_{ji} is big and the channel gain of link j, G_{jj}, is small. But this is not a proper algorithm for the distributed system with no perfect knowledge of the channel gain matrix.

The next best thing is, if the current link i requires large power and this admission increases the power levels of other interfering links too much, then this will definitely reduces the chance of future admission of remaining links. (Here we do not know the gain of other links.) Therefore, we propose the following criteria,

$$\delta \cdot I_i\left(i \notin activelink\right) \le I_i\left(i \in activelink\right) \tag{6}$$

Here I_i means the interference at receiver of link i. On the left is the initial interference power to the receiver of link i. On the right is the interference power to the receiver of

link i when link i is admitted. This interference is used to represent the power level increase of the links in this channel. δ is the designed parameter determined after several simulations. When the power levels of the links using current channel increase too much after admitting the current link, this link is backed off.

There are two more conditions for backing off admission of a new link. First, the current packet for the transmission is not time-sensitive. Second, the number of packets in the buffer is much below the average of the neighboring transmitter nodes. This condition will guarantee the fairness of the link admission in the network in case of stationary channel (without independent Rayleigh Fading). Here we assume that the transmitter broadcasts the information about the number of packets in the buffer. The autonomous back-off algorithm proposed here combines these three conditions.

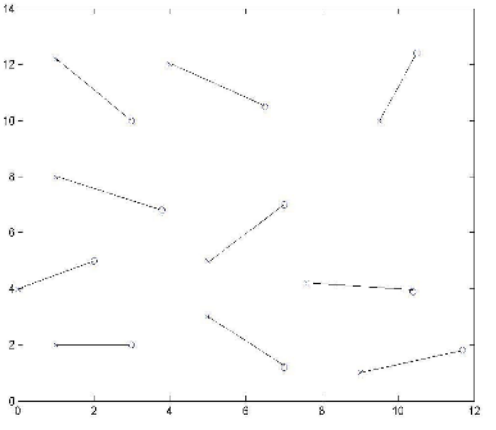


Fig. 1. The location of 10 transmitter-receiver pairs. x: transmitter, o: receiver

4 Simulation

There are 10 transmitter-receiver pairs as shown in figure 1. The target SINR for each link is the same with 10dB. The number of time slot is set to 100. The life span of time-sensitive data packets is 3 time slots. All data packets arrive with the arrival rate of λ. The fraction of time-sensitive data occupies 10% of the total load. The number of available channels is denoted by K. Thermal noise is the same for all receiver nodes with 1e-4mW. The maximum transmit power is 1W. The appropriate value of δ in equation (6) is found to be around 40 – 60 from the simulation. We set δ to be 60 in the simulation.

4.1 FIFO Queue vs. Priority Queue Without Back-Off Algorithm

We compare the two systems, one implementing FIFO queue and the other implementing Priority queue, without the back-off algorithm with 3 non-interfering channels. First we change the traffic load from arrival rate of 0.4 to 1.6. Then we observe three parameters as illustrated below. Figure 2 shows the total number of

transmitted packets. Both results approach nearly 1000 packets as the load increases. (1000 is the maximum value as it comes from 10 links * 100 time slots.) Both FIFO and Priority queue systems show similar behavior because there is no difference in power control with the least interference channel allocation scheme.

However, in figure 3, the probability of TS packet transmission gap increases as the traffic load increases. This shows the superiority of the Priority queue system in heavy load. As the load increases, the packets are accumulated in the transmitter buffer. In priority queue system TS packets are processed first. So even in the heavy load, all the TS packets are put to transmission first, making the probability almost 1. But in FIFO queue system, TS packets stored in the buffer die out. It is because the deadline constraints make the probability of successful transmission very low.

In the second simulation (see figure 5), performances are compared by varying the number of channels in both systems from 1 to 3. We assume the traffic load with arrival rate of 1. In figure 4, the probability of successful TS packet transmission is much better in Priority queue system. The probability for both systems increases as the number of channel increases with constant gap, which can also be seen in figure 3 at the arrival rate of 1. As the channel number increases, more links are activated in each time slot, giving more chance to transmit TS packets within deadline.

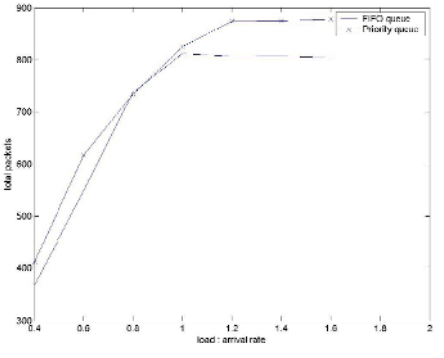


Fig. 2. Comparison of total number of transmitted packets by increasing the traffic load

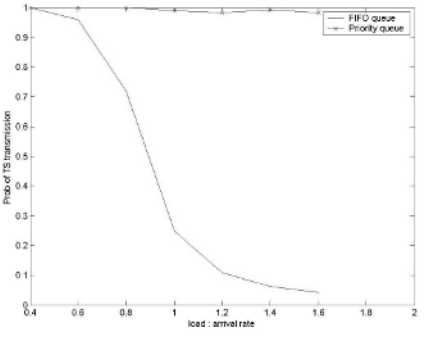


Fig. 3. Comparison of the probability of successful transmission of TS data packets by increasing the traffic load

4.2 FIFO Queue System with Back-Off Algorithm

FIFO queue systems with and without back-off algorithm are compared by varying the traffic load. The number of channels is set to 2. If there are many channels available, most of the links are admissible. In this case, backing off a new link will only decrease the total number of packets transmitted and increase the delay of the TS packets. The back-off algorithm works best when backing off high interference node gives chances for other nodes to enter the network, which have not been admitted if the current node entered the network with high power level. This happens when there are no enough channels.

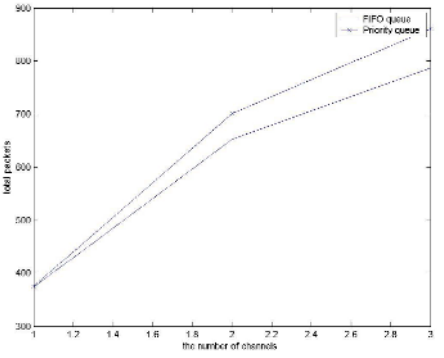


Fig. 4. Comparison of total number of transmitted packets by changing the number of channels

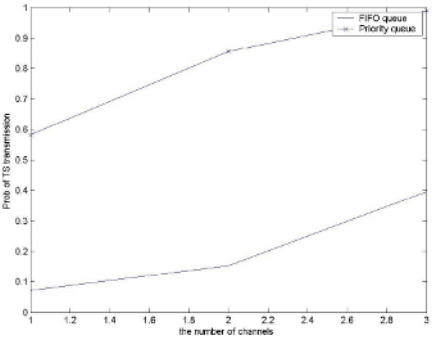


Fig. 5. Comparison of the probability of successful transmission of TS data packets by changing the number of channels

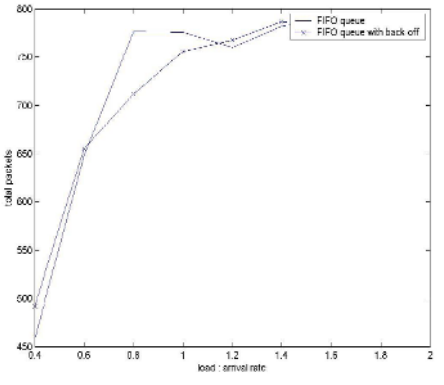


Fig. 6. Comparison of total number of transmitted packets in FIFO queue systems

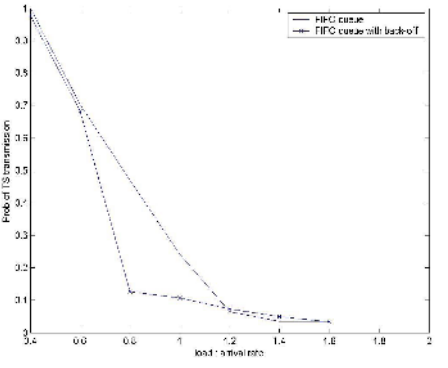


Fig. 7. Comparison of the probability of successful transmission of TS data packets in FIFO queue systems

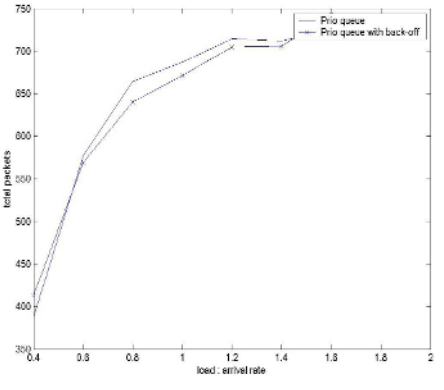


Fig. 8. Comparison of total number of transmitted packets in Priority queue systems

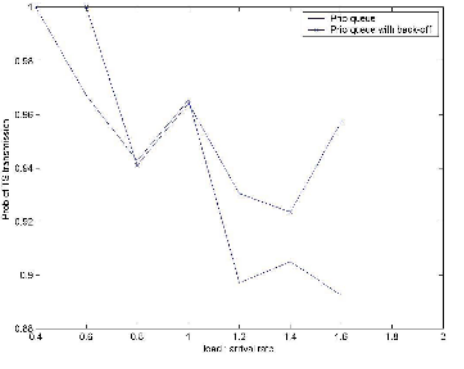


Fig. 9. Comparison of the probability of successful transmission of time-sensitive data packets in Priority queue systems

As shown in figure 6, there is not much difference between the number of total packets transmitted. But the probability of TS packet transmission is different (see figure 7). The FIFO queue system without back-off algorithm shows better performance. In FIFO queue system, the delay of TS packets increases because of backing off those high interference links. The TS packets in these node buffers suffer a starvation. So the probability is lower. But with back-off algorithm, the link which requires high power level to maintain the target SINR doesn't enter the network. The total power usage will be less than that of the system without back-off algorithm.

4.3 Priority Queue System with Back-Off Algorithm

The number of channel is assumed to be 2 as in 4.2. The traffic load varies from 0.4 to 1.6. In figure 8, the total number of packets transmitted with back-off algorithm is slightly less than that without back-off algorithm.

As the load increases, the interference gets larger. This causes voluntary back off of the links. As the number of active link approaches 7 (7 out of 10 links), this back off plays an important role in reducing the total number of transmitted packets. But when the occupation of the active links is less than 50%, system with back-off algorithm can transmit more packets as seen in figure 8.

In the FIFO case, there is not much difference in the probability of successful TS packet transmission. But in priority queue systems, back-off algorithm improves the probability as can be seen in figure 9. This improvement gets better when the number of channels is reduced to 1. This algorithm almost blocks the transmission of time-insensitive data packets when the interference is high due to insufficient number of channels.

5 Conclusion and Future Work

In this paper, the distributed power control with the least interference channel assignment algorithm is implemented. This is combined with FIFO and Priority queue systems. When there is a class of traffic which is time-sensitive, the Priority queue system works better than the FIFO queue system. When the number of channel is 3, successful transmission of TS data packets is almost guaranteed (the probability was close to 1 in Priority queue system).

To increase the transmission of TS data packets, back-off algorithm for TS packets are proposed. In FIFO queue system this algorithm doesn't show apparent improvement. But in Priority queue system, this algorithm demonstrates clear improvement in both the probability of successful transmission of TS data packets and the overall performance described by Eq.(1). This is because all the TS data packets in the buffer are brought to the head of the buffer in Priority queue system. Therefore backing off the transmission of time-insensitive packets gives more chance to the transmission of TS data packets.

The back-off algorithm is aimed to increase the total number of transmitted packets by keeping the interference level low. More sophisticated back-off criteria should be devised to achieve this. The following needs to be studied further.

- The autonomous way of finding δ in Eq.(6).
- The probability of successful transmission of TS data packets over the multi-hop routes.

References

1. Bambos N.: Toward power-sensitive network architectures in wireless communications: concepts, issues, and design aspects. *IEEE Personal Commu.*, Vol. 5 (1998), 50-59
2. Foschini G. J., Miljanic Z.: A simple distributed autonomous power control algorithm and its convergence. IEEE Transactions on Vehicular Technology, vol. 42 (1993), 641-646
3. Muqattash A., Krunz M.: A single-channel solution for transmission power control in wireless ad hoc networks. Proceedings of the IEEE INFOCOM Conference, (2004), 210-221
4. Cheng M. M.-L., Chuang J. C.-L: Performance Evaluation of Ditributed Measurement-Based Dynamic Channel Assignment in Local Wireless Communication. IEEE Journal on Selected Areas in Commun., Vol. 14 (1996), 698-710
5. Bao L. C., Garcia-Luna-Aceves J. J.: Distributed dynamic channel access scheduling for ad hoc networks. Journal of Parallel and Distributed Computing, Vol. 63 (2003), 3-14
6. Kulkarni G., Srivastava M.: A channel assignment scheme for FDMA based wireless ad hoc networks in Rayleigh fading environments. Proc. of IEEE VTC, Vol. 2 (2002), 1082-1085
7. Gibilisco S.: Handbook of Radio and Wireless Technology. McGraw-Hill, (1998)
8. Strang G.: Linear Algebra and its application. 3rd Edition, Harcourt, (1988)
9. Mitra D.: An asynchronous distributed algorithm for power control in cellular radio systems. Proc. of 4th WINLAB Workshop, (1993) 249-257
10. Bambos N., Kandukuri S.: Power Controlled Multiple Access (PCMA) in Wireless Communication Networks. Proc. IEEE INFOCOM, (2000) 386-395
11. Chiang D. J. M., Neil D. O., Boyd S.: Recource allocation for QoS provisioning in wireless ad hoc networks. Proc. of IEEE GlobCom. (2001), 2911-2915

Fast IPv6 Addressing Technique for Mobile Ad Hoc Networks*

Dongkeun Lee and Keecheon Kim**

Dept. of Computer Science & Engineering, Konkuk University
Gwang-Jin Gu, Seoul, Korea
{dklee, kckim}@konkuk.ac.kr

Abstract. Major researches in MANET have emphasized on the design of efficient routing protocols such as DSDV, AODV, etc. The majority of routing protocols assume that mobile nodes in ad hoc networks are configured with IP addresses before they begin communications in the network. Thus, Auto-configuration is desirable in implementing MANET. In this paper, we investigate a MANET topology that all the nodes in MANET want to connect to the Internet through a special node called the Internet Gateway and we propose a new IPv6 address auto-configuration mechanism in MANET. In order to validate the scheme, we present several simulation results.

1 Introduction

Most of the researches in MANET have focused on the design of efficient routing protocols such as DSDV [1], AODV [2], etc. The majority of routing protocols assume that mobile nodes in ad hoc networks are configured with IP addresses before they begin communications in the network. Thus, Auto-configuration is desirable in implementing MANET[3].

Moreover, another challenging issue in MANET is connecting to the Internet through the nodes of an ad-hoc network [6]. In order to connect to the Internet, a node in ad-hoc network can be configured with global routable IP address by address auto-configuration scheme.

In IPv6, stateless address auto-configuration[4] can be used in MANET. This mechanism allows a node to pick a tentative address randomly and then use a Duplicate Address Detection(DAD) procedure to detect duplicate addresses. The DAD procedures use timeouts. In ad-hoc networks, message delay cannot be bounded. Hence the use of timeouts cannot reliably detect the absence of a message. Such unreliability can lead to a situation in which the duplicate addresses go undetected [5].

In this paper, we investigate a MANET topology that all the nodes in MANET want to connect to the Internet through a special node called the Internet Gateway and we propose a new IPv6 address auto-configuration mechanism in MANET. Our solution uses a stateful approach. For the auto-configuration, we use new IPv6 addressing architecture and proxy approach.

---

* This research was supported by the Brain Korea 21 project.
** Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 280 – 289, 2006.

This paper is organized as follows. In section 2, related works are presented. Our stateful IPv6 address auto-configuration scheme is described in section 3. Performance analysis is presented in section 4. Finally, conclusions with future research works are presented in section 5.

2 Related Works

IETF zeroconf working group has already standardized a stateless auto-configuration mechanism for IPv6[4]. However, this protocol was not designed for mobile ad hoc networks. In the wireless ad-hoc network, we cannot determine how many nodes will exist, and how long will it take for DAD messages to be returned to the originate node. Hence, DAD waiting time is a big problem for this standard.

DAD based on the stateless auto-configuration in MANET is presented in [3], in which addresses are randomly selected. Duplicate Address Detection(DAD) is performed by each node to guarantee the uniqueness of the selected address. However, this approach uses timeouts, so it has DAD timeout bound problem. This scheme supports both IPv4 and IPv6.

There is another DAD process to compensate the DAD time. Weak Duplicate Address Detection[5] aims at lowering the overhead needed for the DAD by integrating it with the routing protocol. Weak DAD is an approach to prevent a packet from being routed to a wrong destination, even if duplicate addresses exist. In weak DAD, the nodes do the DAD process with direct linkable nodes first. After that, with the aids of routing protocol, node can detect duplicate address with itself from others. This system is based on the use of a single key that is assigned to each node. Nodes in a network are identified not only by the IP address, but also additionally by a key. If two nodes with the same address choose the same key, a conflict is not detectable because the key is only generated once by each node.

The MANETconf[7] presents an address assignment scheme for ad hoc networks based on a distributed mutual exclusion algorithm that treats IP addresses as a shared resource. In this work, each node maintains a list of all IP addresses in use in the network. A new node obtains an IP address through an existing node in the network. MANETconf produces complex and bandwidth-consuming process by maintaining common address pool information depending on the mobility parameters and also requires the use of timeouts for several operations. In contrast to this approach, our solution does not require each node to maintain and exchange a list of all IP addresses in much simpler way. Our solution can work in the presence of partitions without requiring any special procedure to detect the partitions or merging partitions.

3 New Stateful IPv6 Address Auto-configuration in MANET

3.1 Basic Idea

Our proposal makes a node in MANET auto-configure a unique IPv6 global address. The proposed approach is flexible enough to be integrated with many different routing protocols and can be used to auto-configure an IPv6 site-local address.

We consider a MANET topology in which all the nodes in MANET want to be connected to the Internet through a special node called Internet Gateway. The Internet Gateway acts as a default router for MANET. Internet Gateway allocates the addresses to other ad-hoc nodes with its network prefix and manages the allocated addresses.

When a new node enters into an ad-hoc network, it first creates its IPv6 link-local address like in [4]. And then, in order to auto-configure its global IPv6 address, it requests an available global IPv6 address to the Internet Gateway.

But without a global address, if it isn't directly linked with the Gateway, the new node cannot send an address request message to the Internet Gateway using ad-hoc routing protocol with multi-hops. Hence, a new node chooses a reachable MANET node that can perform an address allocation for itself.

For example, the new node *i* selects a neighboring node *j* as an initiator and then sends an Address_Request(AREQ) message to the initiator node j. Upon receiving the AREQ message from i, node j forwards this message to the Internet Gateway and receives an Address_Reply(AREP) message including the available global address from the Gateway. Finally node j delivers the received AREP message to node i, and node i configures the address included in the received AREP message as its global address.

In order to obtain information of Internet Gateway and neighboring nodes, we use Ad-hoc Node Advertisement(ANA) messages. ANA message contains received Internet Gateway's advertisement message. When a node receives an ANA from its neighbors, it extracts the Router Advertisement(RA) message from the received message and then broadcasts the RA to its neighbors using its ANA.

All nodes in ad-hoc network broadcast ANA messages periodically or non-periodically. When a node receives a Router Solicitation message from its neighbor nodes, it sends its Ad-hoc Node Advertisement immediately in order to inform Internet Gateway's information to neighbor nodes. ANA is restricted within one hop distance.

A new node selects an initiator by ANA messages received from its neighboring nodes. We explained more about ANA message and link-local auto-configuration in [10].

3.2 Proxy Nodes

Ad-hoc networks have many hops between a node and the Internet Gateway. This is why the problem of waiting time for address allocation is occurred. For a rapid address allocation, we introduce proxy nodes that execute the address allocation on behalf of the Internet Gateway. With proxy nodes, a new node can send an AREQ message to the proxy node, and then the proxy node replies with a new IPv6 global address to that node. In this procedure, the distance between the proxy node and the mobile node is one hop, so the new node can get a global IPv6 address more rapidly.

Proxy nodes should send periodic Proxy_Advertisement messages to its neighboring nodes. This message is limited by one-hop range. Actually, Proxy_Advertisement can be used instead of Ad-hoc Node Advertisement message.

3.3 Address Allocation

In order to allocate the addresses in ad-hoc network, we use new IPv6 addressing format that has three parts, Global Network Prefix, Ad-hoc Prefix and Host ID. Network Prefix is identical to the general IPv6 prefix[8]. Interface ID field in normal IPv6 address is divided into two parts, Ad-hoc Prefix and Host ID. Ad-hoc Prefix is allocated by the Internet Gateway and is used as proxy id. Host ID is used as free space and the proxy can allocate addresses within this space to other nodes. Fig. 1 represents the addressing architecture for ad-hoc networks.

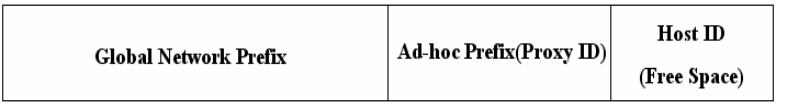


Fig. 1. Addressing architecture for ad-hoc networks

For example, if we assume that the prefix length of the Internet Gateway is 64bits and the length of Ad-hoc Prefix is 48 bits, Host ID can have 16 bits length.

When the Internet Gateway allocates the new address, it selects an unused value as Ad-hoc Prefix and sets the value of Host ID as 0. Thus, if the address that is allocated from the Gateway is 3ffe:2e01:2b:1111:2222:2222:2222:0000, the first 64bits, 3ffe:2e01:2b:1111, is a global network prefix, and the second 48bits, 2222:2222:2222, is ad-hoc prefix. These two parts are fixed by the Gateway. At last, the last 16bits, 0000, is the free space. The new node which receives this address from the Gateway becomes a proxy node, and it can allocate an address to another node by using free space of Host ID from 3ffe:2e01:2b:1111:2222:2222:2222:0001 to 3ffe:2e01:2b:1111:2222:2222:2222:FFFF.

As usual, the proxy node uses the first free address as its own node address. The address, which all bits of Host ID are set to 0, must be used as a proxy-scope multicast address. In above example, 3ffe:2e01:2b:1111:2222:2222:2222:0001 is used as a proxy's address and 3ffe:2e01:2b:1111:2222:2222:2222:0000 is used as a proxy-scope multicast address. In above example, the Gateway can allocate addresses to maximum 2^{48}, and the proxy can allocate addresses to maximum 2^{16}-2, which means quite scalable as an ad-hoc network.

When a new node receives AREP message with the Host ID part of the address is set to 0 from the gateway, the node can become a proxy node and allocate the free addresses to other nodes.

When a new mobile node wants to acquire a global IPv6 address, it, at first, finds any neighboring proxy node with the received proxy advertisement. If one or more proxy nodes exist, the new node sends AREQ to the proxy node. Otherwise, the new node selects an initiator, and sends AREQ to the initiator. If the initiator knows that there is at least one proxy node within one hop area from itself from the periodic Proxy_Advertisement messages by neighbor nodes, it sends the received AREQ to one of proxy nodes that exist within one hop area from itself.

If there is no proxy node, the initiator sends the AREQ to the Internet Gateway with new IPv6 hop-by-hop option, Address Request Option(ARO). Only proxy nodes can process ARO. In this case, if at least one proxy exists in the path from the initiator

to the Gateway, the proxy seizes the request packet and sends a reply with an available global address to the initiator. In this way, the new mobile node can get a global IPv6 address faster than getting it from the Gateway. The address received from the proxy is a single address, thus the new node cannot become a proxy node. If there is no proxy in the path, the request packet is delivered to the Gateway, and the Gateway allocates new address set for the new node and sends a reply message. In this case, the new node becomes a proxy node.

In an ad-hoc network, mobile node can move any direction. Therefore, proxy nodes may huddle up in the edge of ad-hoc network or leave the ad-hoc network. As a result new nodes that need an IPv6 address can't find any proxy node. So, general node must be able to become a proxy node in order to achieve the high performance of this address allocation scheme.

If a general node that is already configured with IPv6 address doesn't receive any Proxy_Advertisement message during some period, it sends a Proxy_Address_Request message to the Gateway in order to become a proxy. When the Gateway receives this message, it allocates a new proxy address space to that node. This process is identical with allocating a new proxy address space to a new mobile node. And then the node becomes a proxy node and it can allocate addresses within its address space to other nodes. In this way, proxy nodes can be distributed equally in ad-hoc network.

3.4 Address Management

The Internet Gateway allocates the addresses from its address space. So, it should do the address management to prevent the loss of addresses. When the Internet Gateway allocates a new address to a new node(i.e. proxy node), it records this address with its lifetime. This lifetime is included in AREP message.

The proxy node should send Address_Refresh message to the Gateway before the end of its lifetime. Then the Gateway extends the lifetime of the proxy node and reply to the proxy node with Refresh_Reply message containing the extended lifetime. If the Gateway does not receive any refresh request message from the proxy node before the lifetime expires, the Gateway multicasts the Refresh_Request message using the proxy-scope multicast address to the ad-hoc network. In this case, all bits in Host ID of destination address are set to 0. This address indicates that all nodes that have the identical Ad-hoc prefix with the requested destination address must receive the packet and respond to the Gateway.

If the Gateway does not receive any reply message for the Refresh_Request message before timeout, it decides that there is no node using that address space, and removes the proxy address space from the table. This address space can be allocated to the other nodes later.

If a Gateway receives reply messages from any node, it knows that there is no proxy node but some nodes are using the proxy address space. In order to reduce the request and reply messages, the Gateway may select a tentative proxy node among the responding nodes and sends Refresh_Reply message to the tentative node. If a node is selected as a tentative proxy node, it cannot allocate addresses to other nodes, but it must send periodic Address_Refresh messages to the Gateway in order to prevent the gateway from sending additional Refresh_Request messages.

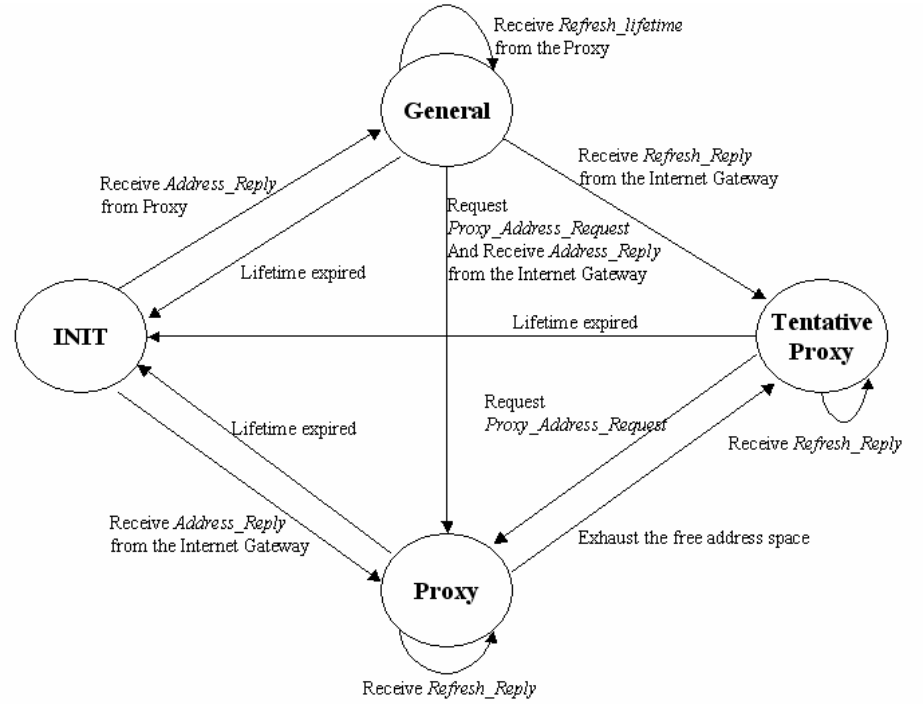


Fig. 2. State transition diagram

The state transition diagram of an ad-hoc node is depicted in Fig. 2. When a new mobile node enters in ad-hoc network and does not configure global address yet, it is in 'INIT' state. And then the node is in 'General' state if it receives a global address from a proxy. 'Proxy' state represents the proxy node and 'Tentative Proxy' state represents the tentative proxy node.

4 Performance Evaluation

4.1 Simulation Setup

The primary goal of the simulation is to gather statistics regarding delays in getting global address. Especially, we focus on effectiveness of using proxies. We assume that nodes are moving according to a random waypoint mobility model[9]. In our simulation environment, we set the pause time to be 10 seconds and the node speeds to be randomly distributed between 2 and 10m/s. Nodes move in a square area. Networks of the maximum size of 30, 50 and 100 nodes were investigated.

Our simulation scenario starts with only one static node, Internet Gateway, and adds other mobile nodes in the simulation area at 2, 100 and 200 seconds in simulation time. Every nodes can move to anywhere in the test area so that the network partition and merger may occur often during the simulation. Parameters used in the

Table 1. Simulation parameters

Parameters	Values
Total Number of Nodes	30, 50, 100, 150, 200
Simulation Area	500m x 500m, 750m x 750m, 1000m x 1000m, 1200m x 1200m, 1500m x 1500m
Advertisement Interval	3000ms
Ad-hoc Routing Protocols	AODV, DSDV
MAC	802.11
Transmission Range	100m

simulation are displayed in Table 1. We used ns-2 simulator to perform simulation and analyze the performance.

4.2 Simulation Results and Analysis

When a new node recognize ad-hoc network by receiving an advertisement message, it sends AREQ and receive AREP. We evaluated this round trip time as global address allocation latency. In order to verify the efficiency of proxy, we compared the performance of proxy scheme with the performance of non-proxy scheme.

Without a proxy, every new node must receive an available global address from the Internet Gateway. Non-proxy scheme, however, doesn't broadcast address request message and doesn't wait address reply messages from all other nodes. Thus, the performance of non-proxy scheme may be better than other protocols using broadcasting for address auto-configuration. Non-proxy scheme can be a good substitution for DHCPv6 or other ad-hoc auto-configuration schemes.

Fig. 3 shows the address allocation latency of each node in the ad-hoc network. Figure (a) and (b) shows comparison between AODV and DSDV. Figure (a) shows address allocation latency until a new node which receives a global address from it sends an AREQ message. Figure (b) shows latency when a new node which receives a global address from it comes into the simulation area. When a new node enters into the simulation area, it may participate in ad-hoc network immediately or not. Fig. 3 (a) and (b) represent our proxy scheme can be applied in different ad-hoc routing protocol and produce good performance.

Fig. 3 (c) and (d) show address allocation latency comparison between proxy scheme and non-proxy scheme using AODV when the total number of nodes in simulation area is 50 and 100. At first, when there are few proxy nodes in ad-hoc network, the effect of having proxies is small. But when there are some proxy nodes in ad-hoc network, the effect of proxies in allocating address to new mobile nodes is increasing. When a proxy node is located uniformly and if more nodes exist in an ad-hoc network, we get better performance.

In fig. 3. (c) and (d), some nodes take a great deal of time to receive addresses. It means that network partitioning now happen. After the network partitions, ad-hoc routing delay may increase. Thus, the allocation latency may increase too. However, in the proxy scheme, a new node can acquire an address from the proxy nodes which exists in the same partition without additional routing delay.

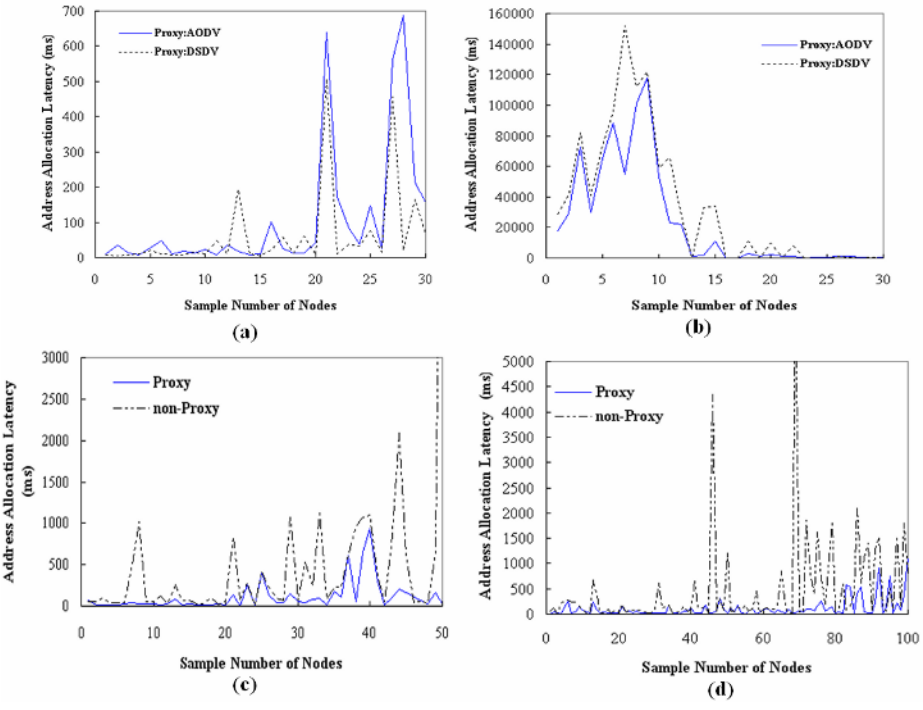


Fig. 3. Address allocation latency per node

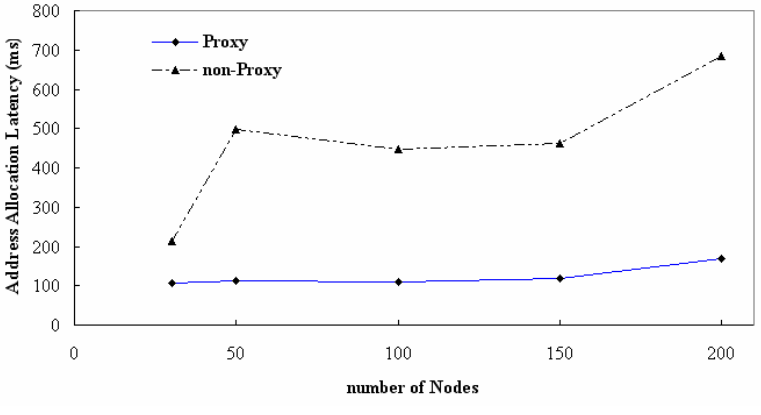


Fig. 4. Average address allocation latency

Some nodes using proxy scheme take longer time compared with non-proxy scheme. By analyzing the log data, we can explain the reason. First, there are some delays in ad-hoc routing and MAC 802.11 because of congestion in proxy. Second, the information of proxy node in initiator is expired. So hop counts between proxy

node and the initiator are more than one hop. This situation occurs because lifetime of proxy advertisement is longer than the interval of proxy advertisement.

Fig. 4 shows the average address allocation latency of the networks using AODV. In general, as more nodes are included in the network, the average node density of one node increases too. So, the allocation latency does not increase dramatically. In proxy scheme, we can see the average allocation increases more slowly. As the number of nodes increase, the number of proxies increases too. Thus, proxy nodes are distributed equally in ad-hoc network. As a result, the address allocation latency does not increase a lot.

According to the simulation results, we know that the proposed solution in this paper gets more flexible and scalable as the size of ad-hoc network increases.

5 Conclusion and Future Works

This paper proposes a distributed IPv6 address auto-configuration protocol for mobile ad-hoc network in which all the nodes in MANET want to connect to the Internet through an Internet Gateway. Since a new node gets a global IP address from a proxy node, which is placed near the new node, we can improve the performance of auto-configuration in ad-hoc-network as shown in our simulations.

The proposed protocol is flexible enough to be integrated with different routing protocols. It uses no broadcasting packets when performing an address configuration. There are only a few multicasting packets for address management. It seriously reduces the number of the necessary packets in address management.

In this paper, we assume that the nodes follow the random waypoint mobility model without a malicious movement. Proposed protocol requires the existence of a centralized server and the new nodes need initiators for auto-configuration. In this situation, some security problems may occur. Hence, we need to specify the possible security issues that may be related to the proposed protocol. Finding a solution to these security issues will be the focus of our future research.

Acknowledgement

This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

References

1. C. Perkins, P. Bhagwat: Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In Proceedings of the ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications (1994)
2. Charles Perkins, Elizabeth Royer: Ad Hoc On-Demand Distance Vector Routing. In Proceedings of the 2nd IEEEWorkshop on Selected Areas in Communication (1999) 90–100
3. C.E. Perkins, J.T. Malinen, R. Wakikawa, E.M. Belding-Royer, and Y. Sun: IP Address Autoconfiguration for Ad Hoc Networks. Internet Draft. Internet Engineering Task Force (Work in Progress) (2001)

4. S. Thomson, T. Narten: IPv6 Stateless Address Autoconfiguration. Request for Comments 2462. Internet Engineering Task Force (1998)
5. N. H. Vaidya: Weak Duplicate Address Detection in Mobile Ad Hoc Networks. In Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02) (2002) 206–216
6. Y.Sun, E. M. Belding-Royer, C. Perkins: Internet Connectivity for Ad hoc Mobile Networks. Intenational Joulnal of Wireless Information Networks special issue on Mobile Ad hoc Networks (2002)
7. S. Mesargi, R. Prakash: MANETconf: Configuration of Hosts in a Mobile Ad Hoc Network. In Proceedings of the IEEE Conference on Computer Communications (INFOCOM) (2002)
8. R. Hinden, S. Deering,: IP Version 6 Addressing Architecture. Request for Comments 3513, Internet Engineering Task Force (2003)
9. J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, J. Jetcheva: A Performance Comparison of Multihop Wireless Ad Hoc Network Routing Protocols. In Proceedings of the 4th ACM/IEEE International Conference on Mobile Computing and Networking (MobiCOM'98) (1998) 85–97
10. Dongkeun Lee, Jaepil You, Keecheon Kim, Kyunglim Kang: IPv6 Stateless Address Auto-configuration in Mobile Ad-Hoc Network. Advanced Web and Network Technologies, and Applications(APWeb 2006). Lecture Notes in Computer Science, Vol. 3842. Springer-Verlag, Berlin Heidelberg New York (2006) 360–367

A Distributed Fairness Support Scheduling Algorithm in Wireless Ad Hoc Networks*

Yong-Qian Chen, Kwen-Mun Roh, and Sang-Jo Yoo

Multimedia Network Lab., Graduate School of Information Technology
& Telecommunications, Inha University
chen.yongqian@gmail.com, rkm21@path.pe.kr, sjyoo@inha.ac.kr
http://multinet.inha.ac.kr

Abstract. Fairness is an important design purpose for shared channel contention based MAC protocols, such as IEEE 802.11 in wireless networks. It is a complex problem due to its many dimensions that include consideration of location-dependent contention, spatial reuse of channels, and desire to achieve fully distributed scheduling in the wireless communication systems. This paper presents a distributed fairness supported scheduling algorithm (DFSS), which accounts for the received service of each flow and adjusts it's backoff time to ensure getting a fair service. In DFSS, each node exchanges the flow's average service index with neighbors as one parameter of determining backoff time. The main advantage of DFSS is that nodes only need to exchange little information to achieve global and local fairness.

1 Introduction

Providing quality of service (QoS) in ad hoc networks is challenging, which needs all nodes to make scheduling decisions. For scheduling disciplines, fair distribution of bandwidth and maximization of resource utilization have been identified as two important design goals. In ad hoc networks, contention-based MAC protocols, such as IEEE 802.11, are popular since they are easy to implement. However, it is widely known that the 802.11 distributed coordination function (DCF) MAC suffers from the problem of fairness in wireless ad hoc networks. As an alternative, scheduled schemes can provide much better guarantees on fairness in ad hoc networks due to their deterministic and explicit scheduling. However, existing algorithms, e.g., generalized processor sharing (GPS), weighted fairness queueing (WFQ)[1] do not directly apply global fairness in wireless ad hoc network due to location-dependent channel contention. In this paper, we propose an approach DFSS to achieve not only local fairness but also global fairness scheduling in wireless ad-hoc networks. For local fairness, DFSS uses weighted fairness queueing (WFQ) to schedule the packets in each node. According the service index, DFSS can determine a suitable backoff value for each flow. Excessive transmitting node will have big backoff time and less probability

* This work was supported by INHA UNIVERSITY Research Grant.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 290 – 300, 2006.

to access to channel. Thus, we can achieve both global fairness and local fairness by exchanging some messages between nodes.

The paper is organized as follows: Section II describes the related work for fair scheduling in ad hoc networks. In Section III, the detail description of the proposed DFSS is presented. In Section IV, we present the simulation result of DFSS and conclude the paper in Section V.

2 Related Work

There are some papers which have proposed distributed scheduling algorithms to provide global fairness in wireless ad hoc network[2][3][4][5][6]. In [2] and [3], a flow graph must first be generated. Each vertex in the flow graph represents a flow. They use a node graph to represent the network topology. Each node is represented by a vertex in the node graph and any two nodes within the transmission range of each other are connected with an edge. Each flow is marked with an arrow from its sender to its receiver. Depending on the scheduling discipline, one of the two tags is chosen as the service tag. The packet with the smallest service tag is sent first. The mechanisms in [2] and [3] differ only in how they achieve spatial channel reuse.

In [4], the author proposed a hop-by-hop backward-pressure scheduling. The forwarding nodes as well as the source are notified of the congestion and then are restrained to send packets to their next hops. This efficiently reduces the MAC layer contentions due to intra-flow contentions and inter-flow contentions on those congested nodes. In [5], the author devised a distributed fair scheduling (DFS) protocol which is based on the IEEE 802.11 MAC and self-clocked Fair queuing SCFQ. The DFS protocol borrows on SCFQ's idea of transmitting the packet whose finish tag is smallest, as well as SCFQ's mechanism for updating the virtual time. The essential idea of DFS is to choose a backoff interval that is proportional to the minimum finish tag of packet to be transmitted. In [6], the author devised a new model to achieve the fairness and maximum allocation of channel bandwidth. This model requires that every node must maintain up-to-date information of all flow in the network.

These timestamp-based algorithms suffer two problems which our algorithm can lighten or avoid. First, because the above scheduling model requires every node know the flow scheduling sequence (generated by WFQ algorithms) globally or locally, each node needs to exchange and maintains all neighbor node flow’s service tags which will give too much overload to wireless networks. It will decrease the throughput of network too much. Second, is that the virtual clock cannot be reinitialized to zero unless the system is empty [7].

3 Distributed Fairness Supported Scheduling algorithm (DFSS)

The objective of our approach is to develop a distributed fairness algorithm that is simple to implement and uses local information for its operation. It must improve over the shortcomings of previous approaches discussed earlier.

3.1 Local and Global Fairness

To achieve our fairness objective, each node combines local fairness scheduling algorithm and global fairness scheduling algorithm. We employ WFQ to acknowledge each flow's service turn and use adjustment of backoff time to get the global fairness.

At first we need to introduce the concept of "flow service index" that keeps track of a flow received service in the interval *T* according it's weight in wireless networks. We define a service index $SI_m^i(t_1,t_2)$ which indicates the total amount of received service from t_1 to t_2 time interval of the flow *i* of node *m* , a node service index $SI_m(t_1,t_2)$ which indicates the average service index of all flows of node *m*. for node *m*, a minimum neighbor service index $minNSI_m(t_1,t_2)$ which indicates the minimum node service index of all neighboring nodes of node *m*. Three parameters are showed in equation 1.

$$
\begin{aligned}
&SI_m^i(t_1,t_2) = \frac{W_i(t_1-t_2)}{\phi_i \times (t_1-t_2)} \\
&SI_m(t_1,t_2) = \frac{\sum_{i=0}^{f(m)} SI_m^i(t_1,t_2)}{k}, \\
&f(m) = the\ number\ of\ \ flows\ of\ node\ m \\
&minSI_m(t_1,t_2) = \min_{\forall n \in N_m} \{SI_n\}, \\
&N_m = set\ of\ \ neighbor\ nodes\ of\ \ node\ m
\end{aligned}
\tag{1}
$$

So in the interval time between t_1 and t_2, the local fairness means the value of $\left|SI_m^i - SI_m^j\right|$ should be near zero for every two flows of node *m*, and global fairness means $\left|SI_m^i - SI_n^j\right|$ should be near zero for every two flows in networks.

3.2 Backoff Time Determining in DFSS

In DFSS, if a flow *i* has bigger SI_m^i than other node's flow average SI_n, it should adjust its backoff timer to reduce its chances to transmit. We also give higher priority to those flows that have higher contention because they have small SI_m^i. Because Local fairness is already used, each flow in the same node has similar service index which means $\left|SI_m^i - SI_m^j\right|$ is near zero, and each node's average SI_m can show the measurement of global fairness in the wireless networks. Thus in our proposed algorithm, we compare the flow's SI_m^i with neighbor node's SI_n and compute the suitable backoff value. In the following, we will explain the procedure.

1) Each node maintains a table which includesflaw's ID, SI_m^i and node's $minSI_m$.
2) The backoff time for a flow is determined first by the difference between the finish tag $F(p_i^k)$ and the start tag $S(p_i^k)$ in WFQ, expressed as equation (2):

$$backoff = [F(p_i^k) - S(p_i^k)] \times sf \times \rho \tag{2}$$

where *sf* is a scale factor to keep the initial backoff value less than a chosen constant to ensure some degree of efficiency in terms of channel utilization. It is a constant value and is the same for all nodes. ρ is a random variable with mean 1. Usually, ρ is uniformly distributed in [0.9, 1.1].

3) Each node transmits the SI_m to neighbors.
4) All neighbors of node receive this information and update their $minSI_n$.
5) After finishing its transmission, each flow calculates the difference index DI_{mi}^i between SI_m^i and $minSI_m$ in its table by equation (3).

$$DI_m^i = \frac{SI_m^i - minSI_m}{SI_m^i} \tag{3}$$

The expression in following algorithm is used to obtain the backoff timer for flow.

$temp = 2 \times backoff$
`If` ($temp > collision\text{-}threshold$)
$new\ backoff = random\ value\ between$ [0, $temp$]

---

6) For collision resolution, we use the method as proposed in the IEEE 802.11. When a collision occurs, the new backoff time is computed as following algorithm:

`If` ($-1 \le DI_m^i \le 1$)
  $backoff = backoff \times (1 + DI_m^i)$
`Elseif` ($DI_m^i > 1$)
  $backoff = backoff \times 2$
`Elseif` ($DI_m^i < -1$)

The collision threshold value was chosen to be 800, as suggested in [2].

3.3 Maximizing Channel Utilization

In ad hoc networks, any two flows that do not interfere with each other can potentially transmit data packets over the physical channel simultaneously, yet the transmission of a flow in one region still affects the transmission of the other flow in the rest of the network. The selection of simultaneous transmitters thus determines the aggregate channel utilization, hence the "global" nature of packet scheduling requires effective selection of such simultaneous transmissions, while taking into account fairness considerations across flows [8].

The goal of maximizing channel utilization is to maximize channel utilization, while achieving fairness. The proposed scheme uses WFQ and adjusts the backoff time to achieve local fairness and global fairness. At the same time, this decreases the channel utilization. For example, as shown in Figure 1, at time t, if node 2 determines flow f1 will be transmitted by the WFQ algorithm and flow f1 will access the medium after the backoff time, then at time, node 4 determines flow f3 should be transmitted by the WFQ algorithm, flows f1 and f3 can not be transmitted simultaneously due to collision. However, if node 4 chooses to transmit flow f4, there will be no collision between flows f1 and f4. Therefore, in this situation, the WFQ wastes channel resource. In [2][6][9], this problem is solved by finding the maximum independent flow set in which each flow can transmit simultaneously.

For the same problem, the present study proposes the following mechanism in DFSS: each node's scheduler can determine whether the packet with the minimum finish service tag should be transmitted or not. For example, in Figure 4, if flow f1 is being transmitted by node 2, while node 4 determines f3 should be transmitted by WFQ, node 4 can not transmit flow f3 data, because node 4 can not receive a CTS packet from node 3 after it sends an RTS packet to node 3 due to a packet collision in node 3. So the node 4 scheduler just holds this packet and schedules the next flow (flow f4), thereby improving the throughput. However, this approach causes the problem that the packet with the minimum finish service tag may not be transmitted first and cause some delay. Thus, there is a tradeoff between delay and throughput. In DFSS, a threshold is predefined for different finish service tag values between each flow in a node. In a node, if the maximum difference of each flow's finish service tag is below the threshold, the next suitable flow can be scheduled to maximize the throughput, otherwise the packet with minimum finished service tag needs to be transmitted, even though it will waste the channel resource. Thus, the delay is bounded in equation (4). Figure 2 shows the procedure for the maximum channel utilization.

$$
\begin{aligned}
&maxF_m - minF_m \le Threthold \\
&maxF_m = \max_{\forall i}[F_m^i] \\
&minF_m = \min_{\forall i}[F_m^i] \\
&F_m^i = virtual\ \ finished\ time\ of\ \ head\ of\ \ line \\
&\qquad packet\ of\ \ flow\ i\ of\ \ node\ m
\end{aligned}
\tag{4}
$$

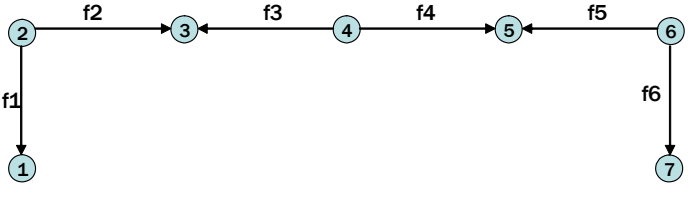


Fig. 1. Example of scheduling

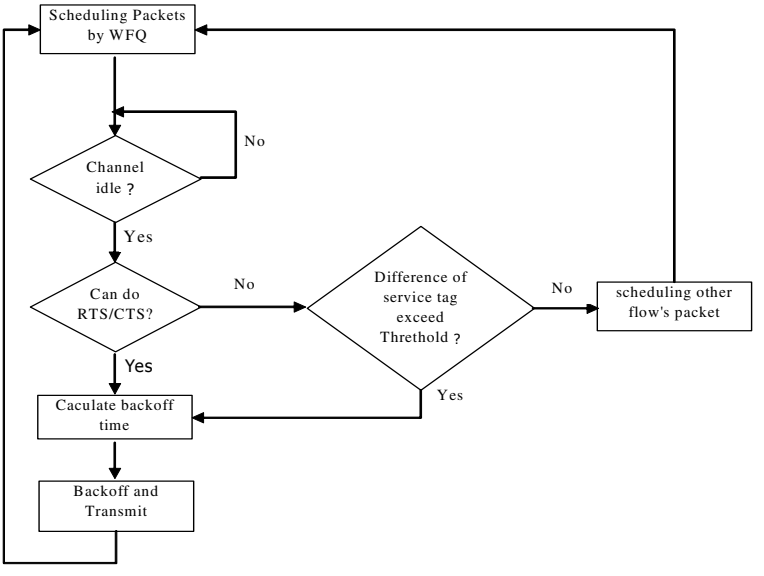


Fig. 2. Maximizing channel utilization

3.4 Implementation Consideration

To implement DFSS, we need to take more consideration for following issues.

3.4.1 Flow Service Index Calculation

We adjust each flow's backoff timer according the difference index DI_m^i between SI_m^i and $minNSI_m$. We calculate each flow's SI_m^i periodic by following

$$SI_m^i(K+t)=\alpha\times SI_m^i(K)+(1-\alpha)\times SI_m^i(K,K+t) \quad \alpha\in(0,1) \tag{5}$$

Where $SI_m^i(K+t)$ denote value SI_m^i in time $(K+t)$, and α denote the weight of past value SI_m^i in new determined SI_m^i, t denotes the interval of calculating SI_m^i, $SI_m^i(K, K+t)$ denote the SI_m^i during time $(K,K+t)$ which calculated by equation (6).

$$SI_m^i(K,K+t)=\frac{W_i(K,K+t)}{\phi_i\times t} \tag{6}$$

According above equations (5) (6), we can get each flow's SI_m^i and node's SI_m.

3.4.2 Service Index SI_m Exchanging and Updating

Each node m transmits the SI_m to all its neighbor nodes as part of a data packet instead of as frame header in order to reduce the overhead of our algorithm. Otherwise, the SI_m field need be added in each frame if we transmit it as part of frame header. To make the overhead minimum, we let the sender decide whether transmits the SI_m or not by following:

- If the value of SI_m is varied to last time transmitted value exceeded a threshold θ, for example 5%, then transmits the SI_m to neighbors.
- If a new node access to the medium to content resource, then transmits the SI_m to neighbors.
- If node has not transmitted it for a period of T (expired interval T), then transmits the SI_m to neighbors**.**

3.4.3 Scheduling Algorithms of DFSS

Here, we will show the whole scheduling algorithm by following algorithm.

Init: For node m, let the set neighbors of m be $\Re$.

$\Re = \{N_n \mid$ *neighbor nodes of node* $m\}$

Procedure:

(a) Node m uses algorithm WFQ to schedule outcoming packets.

(b) Node m counts each flow service and calculate the flow's SI_m^i.

(c) Calculate the SI_m in periodic time t.

(d) If { $\left|\frac{SI_m(K+t)-SI_m(K)}{SI_m(K)}\right| \geq \theta$ or (last transmitted expired T)

or (new node joins and content the medium)} then

Sender transmits the SI_m by DATA MAC header.

Receiver transmits the same SI_m by ACK MAC header.

(e) If $\forall N_n \in \Re$, then update the $minNSI_n$ with received SI_m .

---

4 Simulation Study

This section presents simulation results when using the proposed algorithm and evaluates them. The proposed algorithm is compared with the IEEE 802.11 MAC protocol (no scheduling) and IEEE 802.11 MAC protocol (with WFQ local fairness scheduling). The proposed algorithm is implemented using an ns-2 simulator, and it is assumed that the channel is error-free, hence any error in transmission is due to the collision of packets.

4.1 Simulation Parameters and Evaluation Criteria

In the simulation, a destination-sequence distance vector (DSDV) was used as the routing protocol and UDP as the transport layer protocol. The data was simulated to transmit at a constant bit rate (CBR), and the packet size was fixed.The analysis used a fairness index and the throughput as the two evaluation criteria. The throughput was defined as the number of packets transmitted during the simulation time, while the fairness index (FI) was expressed as the following equation (7):

$$FI = \frac{(\sum_{i=1}^{n} x_i)^2}{n\sum_{i=1}^{n} x_i^2} \tag{7}$$

where x_i is flow rate the flow i, and n denotes the number of flows in the network.

4.2 Comparison with IEEE 802.11 With and Without WFQ

We simulate with following topology as shown in figure 3, and 12 flows are set same weight and different weights respectively. Furthermore, we don't think about the node's mobility. We set *scale factor*= 0.3, α = 0.5, θ = 5%. The important parameter is set as Table 1.

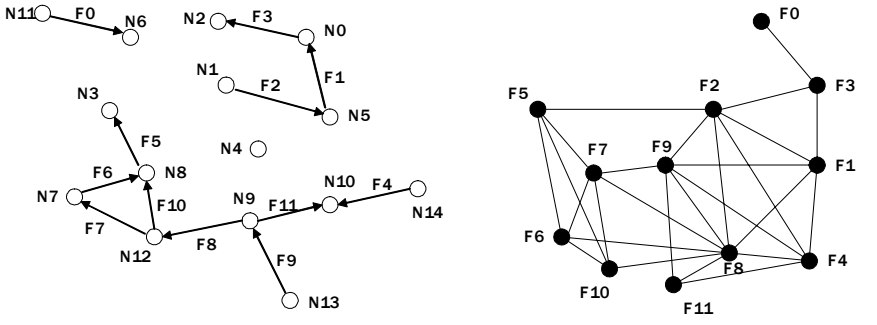


Fig. 3. Node graph and flow graph of simulation

Table 1. Simulation parameters of DFSS

Name	Value
Number of flows:	12 flows
Flow rate:	800kbps
Packet size:	512bytes
Simulation time:	500Sec
Routing protocol:	DSDV
Channnel bandwidth:	2Mbps

The result of simulation is showed in following Table 2:

Table 2. Throughput and fairness of simulation

Flow	Weight	802.11	802.11+ WFQ	DFSS	Flow	Weight	802.11	802.11+ WFQ	DFSS
F0	1	22157	21816	20354	F0	1	22157	19781	10742
F1	1	6853	7903	16542	F1	1	6853	6932	7046
F2	1	16547	13794	9675	F2	1	16547	14684	7463
F3	1	24658	23583	27361	F3	1	24658	22173	15632
F4	1	5631	7846	11527	F4	2	5631	6749	11739
F5	1	31591	31284	13235	F5	2	31591	28541	15748
F6	1	29134	28453	13173	F6	2	29134	29438	14825
F7	1	6795	7151	8651	F7	2	6795	5294	10816
F8	1	4528	5842	7316	F8	4	4528	6035	9673
F9	1	11356	9035	9752	F9	4	11356	11965	14738
F10	1	7653	7275	9941	F10	4	7653	10452	20562
F11	1	3648	3932	7168	F11	4	3648	4138	15348
Throughput		170551	167914	154695	Throughput		170551	166182	154332
		100%	98%	91%			100%	97%	90%
FI		0.68	0.70	0.84	FI		0.53	0.57	0.81

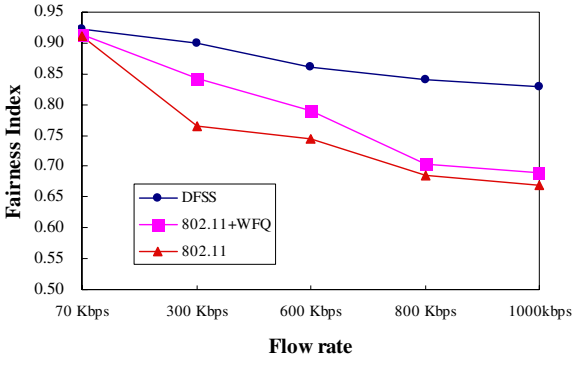


Fig. 4. Fairness index with same weight and different flow bit rate

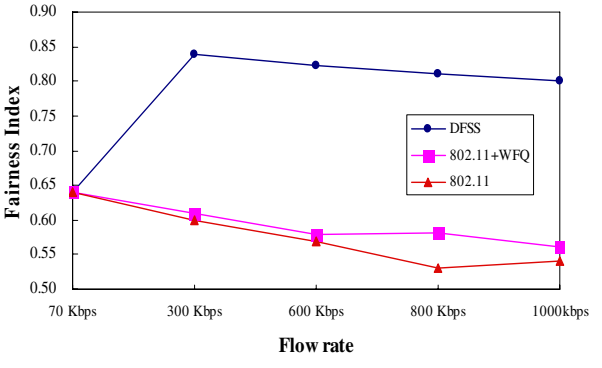


Fig. 5. Fairness index with different weight and different flow bit rate

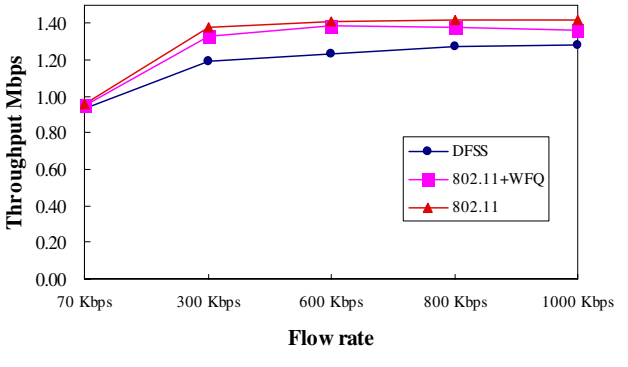


Fig. 6. Throughput with same weight and different flow bit rate flow with same weight

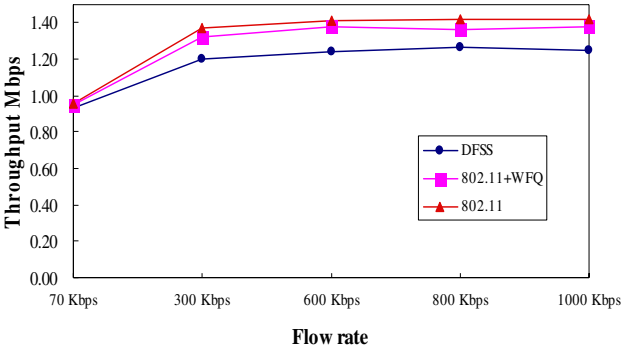


Fig. 7. Throughput with different weight and different flow bit rate

Additionally, we varied the network loading by giving each flow's different bit rate in simulation. As Figure 4 and Figure 5 show, the flow fairness will become bad when the networks load increase. But our algorithm's fairness index is also better than 802.11 and 802.11+WFQ.

Table 2 shows that our scheduling algorithm decreases the networks throughput only 10 percents compared with 802.11 networks and gets better global fairness service. Fig.6 and 7 show the networks throughput with different flow rate.

5 Conclusion

In this paper, we have proposed a distributed scheduling algorithm DFSS for ad hoc networks which combines local fairness and global fairness. DFSS counts each flow's received service from the network and calculates the node service SI_m in each node. The SI_m will be exchanged between neighbor nodes. Based this information, we can determine the suitable backoff time for each flow. Thus, combining with WFQ, we can provide global and local fairness for each flow. Comparing with other distributed scheduling algorithm(e.g. in [2][3]), our algorithm needs to neither generate flow graph nor exchange each flow's service tags which will give too much overload to wireless networks. Furthermore, we don't suffer from the problem of virtual clock cannot be reinitialized to zero unless all the system is empty in [2][3]. After implementing our proposed algorithm in ns-2 and simulate it, we find the fairness index is improved obviously (about 10%~20%), especially when the network load is high. We demonstrate the effectiveness of our proposed design through both simulations and analysis.

References

1. Jinran Chen and Arun K.Somani: Fair Scheduling in Wireless Ad-Hoc Networks of Location Dependent Channel Errors. IEEE International Conference of Performance, Computing and Communications (2003) 103-110
2. N. Vaidya, P. Bahl and S. Gupta: A Topology independent Fair Queueing Model in Ad Hoc Wireless Networks. IEEE Int. Conf. Network Protocols (2000) 325-335

3. H. Luo, P. Medvedy, J. Cheng and S. Lu: A Self-Coordinating Approach to Distributed Fair Queuing in Ad Hoc Wireless Networks. IEEE INFOCOM (2001) 1370-1379
4. H. Zhai, J. Wang and Y. Fang: Distributed packet scheduling for multihop flows in ad hoc networks. IEEE Wireless Communications and Networking Conference (2004) 1081-1086
5. N. Vaidya and P. Bahl: Distributed Fair Scheduling in a Wireless LAN. IEEE Transactions on Mobile Computing (2005) 616 – 629
6. H. Luo, S. Lu, V. Bharghavan: A New Model for Packet. Scheduling in Multihop Wireless Networks. ACM Mobicom (2000) 76-86
7. Hsi-Lu Chao, Wanjiun Liao: Fair Scheduling With QoS Support in Wireless Ad Hoc Networks. IEEE Transactions on Wireless Communications 3(6) (2004) 2119-2128
8. Haiyun Luo, Cheng J, Songwu Lu: Self-coordinating Localized Fair Queuing in Wireless Ad Hoc Networks. IEEE Transactions on Mobile Computing (2004) 86-98.
9. Xinran Wu, Clement Yuen, Yan Gao, Hong Wu and Baochun Li: Fair Scheduling With Bottleneck Consideration in Wireless Ad-hoc Networks. International Conference on Computer Communications and Networks (2001) 568-572

QoS Model for Improving End-to-End Service in 802.11e-Based Wireless Ad Hoc Networks

Joo-Sang Youn[1], Seung-Joon Seok[2], and Chul-Hee Kang[1]

[1]Department of Electronics and Computer Engineering, Korea University
5-1ga, Anam-dong, Sungbuk-gu, Seoul, Korea
{ssrman, chkang}@widecomm.korea.ac.kr
[2]Dept. of Computer Engineering, Kyungnam University, Kyungnam, Korea
sjseok@kyungnam.ac.kr

Abstract. In order to obtain better performance in wireless ad hoc networks, the majority of existing work focuses on the multi-priority based packet scheduling mechanism under multiple service classes, admission control and rate policing, according to application requirements and network conditions. In addition, no studies have reported on the feasibility and scalability of adapting the IEEE 802.11 EDCA scheme in such networks. This paper presents a new QoS provisioning model called Dynamic Hop Service Differentiation (DHSD), which performs dynamic class selection based on estimation of each service class (AC) in 802.11e based ad hoc networks. This model is designed for Soft QoS provisioning. Performance evaluation is performed by simulation using OPNET. It is demonstrated that this QoS model outperforms existing service models in wireless ad hoc environments.

1 Introduction

In wireless ad hoc networks, QoS provisioning for packet forwarding is a challenging task [1], because decentralized access to share a medium often results in unacceptable variations in QoS of flows. In addition, many other problems in wireless ad hoc networks further impose challenges on QoS support. Much advanced work [2, 3, 4] for QoS provisioning has been reported. There have been numerous mechanisms that affect QoS provisioning across the communication protocol stack. Queuing disciplines, admission control policies, resource reservation protocol, priority access control policies and QoS support orientated media access control protocol, can affect the perceived QoS. However, the effectiveness of these mechanisms is very sensitive to exterior environments such as the variation in traffic conditions, dynamic topology and suitability, which are dependent on the QoS requirements of the application. Suitable QoS provisioning for supporting diverse applications in such networks currently represents a significant technical challenge. In existing work, QoS schemes are proposed in 802.11-based ad hoc networks. 802.11 type MAC protocols do not offer any service differentiation for QoS support. Therefore, in order to a provide priority-based service model that guarantees real-time traffic over best-effort traffic in such networks,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 301–311, 2006.

differentiated scheduling and medium access algorithms are proposed [11]. These solutions still face the many overheads for QoS guarantee. Recently, Extension of IEEE 802.11 has been proposed to support service differentiation. Such extensions include IEEE 802.11e [6]. These extensions divide traffic into different classes and accept different contention related parameters (e.g. minimum contention window size, maximum MAC frame size, and so on) However, these extensions still have problems in feasibility and scalability for adapting IEEE 802.11EDCA in wireless ad hoc networks. In this approach, packets placed in different queues in a node, according to application's type and service differentiation between different queues, depends on adjusting the related contention parameter. If collisions frequently occur in networks, it is demonstrated that the effect on service differentiation is less obvious and less stable [7]. In addition, this scheme provides static priority based service differentiation. Through this approach, the probability that queue overflow occurs in the class increases, as packets using a particular class increase under congested networks. Also, the reversed phenomenon of per-hop delay between a class with high priority and a class with low priority occurs in simulation, this is a result of the increased queue delay of a class with high priority. In this paper, an attempt is made to minimize the functions for QoS provisioning, to develop Soft QoS provisioning [5]. For example, without estimating available bandwidth for data transmissions as channel bandwidth, the QoS model is designed to operate only in a monitoring queue state.

In order to provide stable QoS support under 802.11e-based ad hoc networks, especially low delay, high throughput and low loss, a new QoS model called dynamic hop service differentiation (DHSD), is proposed. The DHSD model is designed to support three types of service; low delay, high throughput and low loss service. In the DHSD model, Access Category (AC) in 802.11 EDCA is used to support service classes ordered in different weight values for packet scheduling. Dynamic class selection (DCS) algorithms are proposed, which selects a service class that meets QoS requirements for a specific service at a node. Therefore, the DHSD model based on DCS algorithms can maintain QoS requirements of applications independent of a node's dynamic bandwidth, drop rate and traffic arrival at all times. A set of mechanisms to realize the DHSD model in IEEE 802.11e-based ad hoc networks is developed, in the presence of traffic variations, and network disturbances due to topology dependent contention (hidden and exposed nodes), fluctuating queuing delays and loss rate includes both collision loss rate and buffer loss. The remainder of the paper is organized as follows. Section 2 describes related work. Section 3 describes the proposed QoS model. Section 4 describes the DCS algorithms for each service. Section 5 summarizes the results of simulations. This paper concludes in Section 6.

2 Related Works of MANET

Performance results in [8] reveal that EDCA is capable of providing excellent service differentiation. However, other work decorates the necessity of adding new

features to overcome a few limitations in the performance of the EDCA mechanism. Only the studies relevant to the work in this paper are addressed. In [9] the authors point out the necessity of local data control and admission control to guarantee service for real-time traffic in high traffic load conditions. In this model, each node maps the measured traffic-load condition into backoff parameters locally and dynamically. Further, in [8, 10] performance analysis is shown through setting the TXOP limit, differentiating both the minimum backoff window size and the backoff window increasing factor, and limiting retransmission attempts. Adaptive CW adjustment, according to application requirements and network conditions, has been analyzed in [11]. This scheme reveals that overall goodput is obtained up to 25 percent higher than EDCA. In contrast with existing work which dynamically adapts CW values according to QoS requirements based on application requirements and network conditions in one-hop wireless environments, only the 802.11e EDCA mechanism is used to implement several forwarding classes and the different weighted forwarding packet rate. The solutions of end-to-end QoS provisioning, using IEEE 802.11e, have recently been investigated in multi-hop environments. However, some solutions have been proposed in literature, attempting to improve performance of the 802.11 MAC in multi-hop networks. The proposed QoS model does not overcome problems related to EDCA in wireless ad hoc environments, but rather presents a novel service model, using the EDCA to obtain better performance.

3 Proposed QoS Model

In this section, a new QoS model, called Dynamic Hop Service Differentiation (DHSD) is proposed to achieve service differentiation in 802.11e based wireless ad hoc networks. The proposed model support three types of service; low delay, high throughput and low loss. In the following subsections, more details are given, illustrating the functional behavior of the proposed model.

3.1 The DHSD Model

In the DHSD model, four types of services are defined to support the QoS requirements of diverse applications in wireless ad hoc networks.

- Low delay guaranteed service: the service provides lower per-hop delay than other services at each node.
- High throughput guaranteed service: the service provides both lower per-hop delay and higher per-hop throughput than other services at each node.
- Low loss guaranteed data service: the service provides lower per-hop loss rate than other services at each node.
- Default service: the service provides no guaranteed QoS.

The main concept behind the DHSD model dynamically assigns each service to a service class per hop according to current service state and loading rate of each AC in a node. Applications with end-to-end flows between arbitrary pairs

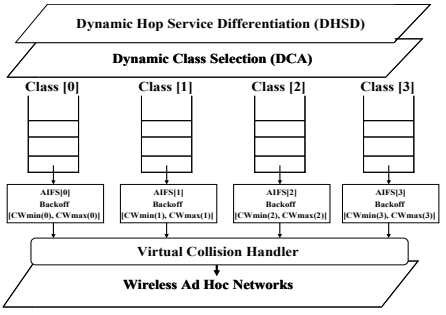


Fig. 1. DHSD-based guaranteed service framework

of nodes have its specific QoS requirement. Then each flow requests based on the QoS requirement of application. The DHSD model supports applications classified four services with four service classes (ACs) in 802.11 EDCA. Fig. 1 presents the DHSD based guaranteed service framework at a node. With DHSD as the common basis, an application selects a service such that its QoS objective is satisfied. Delay-sensitive applications such as voice applications select low delay guaranteed service. The applications with QoS requirements of both high end-to-end throughput and low end-to-end delay, such as multimedia streaming, select high throughput guaranteed service. Loss-sensitive applications such as TCP-based FTP select low loss guaranteed service. Finally, other applications select default service. And the specific selected service in the sender is fixed at overall nodes along the route over the duration of the connection. After service decisions of the application in DHSD, an application can effectively be provided best service for packets with DCS algorithms in each node. DCS algorithms perform class selection based on the current state of a class. For low delay guaranteed service, the predicted waiting service delay based DCS algorithm performs class selection. For high throughput guaranteed service, an average number of backlogged packet based DCS algorithms perform class selection. Ed - Please check, original meaning unclear. For low loss guaranteed service, the loss rate based DCS algorithm is proposed with an average loss rate of a class. For default service, the DCS algorithm selects class 3. Thus, DHSD concurrently supports delay-sensitive, throughput-sensitive and loss-sensitive applications with no way of distinguishing between the two. Fig. 2 shows the implemented mechanisms at each node. As shown, each packet is marked with its service by the DHSD marker. After service decision, in order to select a proper service class among the four classes (ACs) of the service, the class marker requests a class for this service from the DCS and then the DCS returns the best class that satisfies service-requirements such as per-hop low delay, high throughput and low loss rate measured per class. Class marker marks the packet with class i returned by the DCS. Thus, the marked packet is serviced at class i in 802.11EDCA.

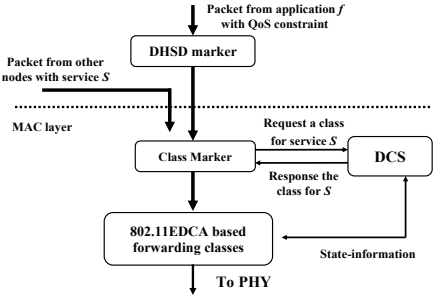


Fig. 2. The operation of DHSD at a node

Packets are dynamically marked with the class i returned per node by the DCS. The DHSD model is realized with the DHSD marker, class marker and DCS. The following sections describe the DCS mechanisms at each node.

4 DCS Mechanisms

In this section, DCS mechanisms for each service are described. The QoS parameters for DCS are first defined to select a service class for each service and then DCS algorithms for three services are described.

4.1 Measurement-Based State-Information of Each Class

In order to perform class selection for each service, the DCS observes the current state of each class in 802.11EDCA. The DCS uses the three types of state information of each class as QoS parameters to select a proper service class according to service type. The three types of state information are the expected waiting service delay, average number of backlogged packets, and average drop rate. Methods are discussed for measuring the three types of state information.

Expected waiting service delay estimation. The expected waiting service delay is defined as the expected time until a packet is transmitted to the physical layer. The expected waiting service delay is then calculated through multiplying the current number of backlogged packets by the average value of contention delay consumed for all packets transmitted by a class. Here, if this exchange is used for that packet, contention delay includes the period for successful RTS/CTS exchange. Similarly, if the initial transmission of the packet is delayed due to one or more internal collisions, it may also include multiple numbers of backoff periods. The model for the measurement of the expected waiting service delay is shown in fig. 3. Let N_i be the current number of backlogged packets in class i and $\overline{ct}_{i,k}$be the average contention delay of class i belonging to the kth packet at time t when the class marker requests a class from the PHCA. The expected waiting service delay estimation w_i of class i is computed as follows:

$$w_i = N_i(t) \times \overline{ct}_{i,k} \tag{1}$$

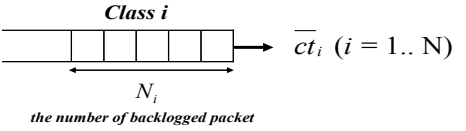


Fig. 3. The model of the expected waiting service delay estimation

The weighted moving average is used to smooth the estimated expected waiting service time. At each sampling time, the average contention delay is updated as follows:

$$\overline{ct}_{i,k} = \alpha\ \overline{ct}_{i,k-1} + (1-\alpha)n_{i,k} \tag{2}$$

where α is the weighting factor ($\alpha < 1$), whose best value has been computed to be 0.8 following a comprehensive simulation under traffic conditions and $n_{i,k}$ is the contention delay achieved by the kth packet. The initial value $ct_{i,0}$ is set to the value adding the slot-time of AIFS[i] to the slot-time of middle value between $\mathrm{CW}_{min}(i)$ and $\mathrm{CW}_{max}(i)$.

A average number of backlogged packet estimation. In order to estimate the average number of backlogged packets $\overline{N}_{i.k}$, the average number of backlogged packets of each class is taken by measuring the current number of backlogged packet of each class every at short-term interval τ. Let $n_{i,k}$ be the current number of backlogged packet of class $i$at the kth interval. The $\overline{N}_{i.k}$value at the kth times is estimated as follows:

$$\overline{N}_{i,k} = \alpha\ \overline{N}_{i,k-1} + (1-\alpha)n_{i,k} \tag{3}$$

where the parameter α is the weighting factor and $\alpha < 1$, whose best value has been computed to be 0.8 following a comprehensive simulation under traffic conditions.

The average loss rate estimation. The average loss rate estimation is based on a packet drop of class i. Here, packets are dropped at two places in a queue: queue drop and collision drop. In the case of a queue drop, a packet in each forwarding class is dropped due to a) full buffer for transmission, or b) the size of packet in the higher layer, which is greater than the maximum allowed data size defined in the IEEE 802.11 standard. In the case of collision drop, a packet is dropped due to consistently failing retransmissions. This drop includes the number of packets that are discarded because the MAC could not receive any ACKs for (re)transmissions of those packets or their fragments, and the packets' short or long retry counts reached the MAC's short or long retry limit, respectively. For convenience, the average loss rate $\overline{D}_i$ is computed in a straightforward manner as follows:

$$D_i = \frac{\overline{D}_i^{queue} + \overline{D}_i^{collision}}{L_i} \tag{4}$$

where $\overline{D}_i^{queue}$ and $\overline{D}_i^{collision}$ are the queue drop rate and collision loss rate of class i , respectively and $\overline{L}_i$is the number of transmission of class i from a high

layer to the LLC layer. Each value for every internal value is estimated and then the $\overline{D}_i$ value is computed with the estimated value. The weighted moving average is also used to smooth the estimated value. At each sampling time, each average value is updated as follows:

$$\overline{D}_i^{queue}(k) = \alpha\ \overline{D}_i^{queue}(k) + (1-\alpha)\overline{D}_i^{queue}(k-1) \tag{5}$$

$$\overline{D}_i^{collsion}(k) = \alpha\ \overline{D}_i^{collsion}(k) + (1-\alpha)\overline{D}_i^{collsion}(k-1) \tag{6}$$

$$\overline{L}_i(k) = \alpha\ \overline{L}_i(k) + (1-\alpha)\overline{L}_i(k-1) \tag{7}$$

where α represents a suitable smoothing factor and $\alpha \quad < 1$, whose best value has been computed to be 0.8, following a comprehensive simulation under traffic conditions. Both average queue drop and average collision drop are independent of each other.

4.2 DCS Algorithms

In this subsection, three DCS algorithms are described in terms of three types of service; low delay guaranteed service DCS, high throughput guaranteed service DCS and low loss guaranteed service DCS.

DCS algorithm for low delay guaranteed service. Low delay guaranteed service aims to achieve low delay service at a node. Hence, for this service all nodes in the network must assign the class with the lowest waiting service delay when a packet marked with low delay guaranteed service arrives at a node. The DCS calculates the expected waiting service time per class, according to both the current number of backlogged packets of a class and the average contention delay of a class. Thus, it determines a class as shown in equation (8).

$$DCS_{delay}(t) = \text{class } i \textit{ of minimum waiting service time} \tag{8}$$

where time t is the instantaneous time a packet arrives. After the class decision finds the class, the class marker marks a packet with class $i$returned by the DCS.

DCS algorithm for high throughput guaranteed service. High throughput guaranteed service achieves both high throughput and low delay service at a node. In order to search for a forwarding class achieving highest throughput among all forwarding classes, the DCS uses the average number of backlogged packets of each forwarding class. This value is used because the queue achieving high throughput always maintains a minimum number of backlogged packets in saturation conditions. Therefore, for high throughput guaranteed service The DCS assigns a service class with a minimum value among all classes. Then, it determines a class as in equation (9).

$$DCS_{throughput}(t) = \textit{class } i \textit{ of minimum number of backlogged packet} \tag{9}$$

where time t is the instantaneous time when a packet arrives.

DCS algorithm for low loss guaranteed service. Low loss guaranteed service achieves a low loss, in terms of packet drop characteristics. For this service, the DCS uses the average loss rate of each class. Hence, the DCS selects a service class with the lowest average loss rate among all classes, as in equation (10).

$$DCS_{loss}(t) = class\ i\ of\ minimum\ average\ low\ rate \tag{10}$$

where time t is the instantaneous time a packet arrives.

5 Simulation Studies

In order to illustrate the effectiveness of the DHSD model, a comprehensive performance study is presented and the model is compared with a strict priority service model in two scenarios: one-hop ad hoc environments, and peer-to-peer multi-hop environments. The strict priority service represents the QoS model with fixed priority, based on the 802.11e EDCA scheme. The priority for each flow in the strict priority service model is statically mapped to the mac priorities as follows: low delay guaranteed traffic → AC 0, high throughput guaranteed traffic → AC 1, low loss guaranteed traffic → AC 2 and default traffic → AC 3. Simulations are performed in OPNET v11.5 [12]. Dynamic Source Routing (DSR) is employed in the routing protocol. The RTS/CTS mechanism is used. A heterogeneous traffic scenario with three types of traffic flows is considered, i.e., voice, video, and data. The traffic flow is characterized by a packet arrival pattern and payload statistics as follows: Voice traffic for low delay guaranteed traffic is characterized as a two state Markov ON/OFF [13]. The ITU-T G.711(silence) speech codec is selected to model good-quality voice calls. Voice traffic is approximately the mean on-time arrival rate of 40kbit/s. The video source rate is modeled by the first-order autoregressive markov model [14]. The following represents the constant bit-rate of a video source during the generating period of the video frame. It is assumed that both incoming stream frame size and outgoing stream frame size is 1728 byte/pixel, the frame size of the video source is 128x120 pixels, and frame inter-arrival time is 10frames/sec. Data packets arrive from the high layer in a Poisson sequence, with exponentially distributed packet length. A mean packet length of data traffic for loss guaranteed traffic is 1024 bytes and mean packet length of default traffic is 512 bytes. Thus, the average data throughput is 40 Kb/s and 20 Kb/s, respectively. The flow varies according to simulation requirements.

5.1 One-Hop Ad Hoc Environments

In this scenario, one-hop ad hoc networks are considered using topologies where 20 static nodes are located randomly in 500m x 500m square regions. Each flow randomly chooses a node as sources and destinations in the networks. All service flows are increased with the same ratio as the number of flows (1:1:1:1) and the number of each flows varies from 2 to 10. The performances of the DHSD model are compared with strict priority service. The simulation continues for 200s.

Table 1. The performance of each service in one-hop ad hoc networks

Scheme	Strict Priority / DHSD			
Number of traffic flows	Delay of low delay service (second)	Throughput of high throughput service (ratio × 100)	Throughput of low loss service (ratio × 100)	Throughput of default service (ratio × 100)
8	0.1003 / 0.1012	98.0 / 99	97.3 / 98.2	96.8 / 97.4
16	0.1006 / 0.1010	96.8 / 97.9	94.3 / 88.7	94.7 /77.5
24	0.1017 / 0.1027	96.5 / 98.4	74.2 / 83.9	71.2 / 48.7
32	0.7554 / 0.1096	91.2 / 97.2	41.7 / 74.5	48.7 / 32.1
40	3.7719 / 2.1600	75.2 / 86.3	24.5 / 66.6	19.5 / 9.2

The simulation results are presented in table. 1. It has been proven from these results that the proposed DHSD model outperforms the strict priority service model in both end-to-end delay and normalized average end-to-end throughput. The normalized end-to-end throughput of high throughput guaranteed service is above 90 % in congested networks. This is because queue drop decreases using the dynamical forwarding class. Instead, the throughput of default service in the DHSD model is lower than that of the default strict priority service. It is verified that the DHSD model almost provides optimal end-to-end service.

5.2 Peer-to-Peer Multi-hop Ad Hoc Environments

In this scenario, multihop ad hoc networks are considered, where 25 nodes are placed in a 5 x 5 grid topology over 800m x 800m square regions. Each flow randomly chooses nodes as sources and destinations in the networks. All service flows are increased with the same ratio as the number of flows (1:1:1:1), and the number of each flows varies from 1 to 5. The simulation continues for 400s. Fig. 4 shows the results for the strict priority service and DHSD model. There is an improvement in the end-to-end service of the DHSD model compared with that of the other model. Fig. 4 (a) reveals that when the number of total flows is greater than 16, the performance of low delay guaranteed service using the DHSD model is increased more than that of the strict priority service. This occurs because, as the number of flows of high throughput guaranteed service and low loss guaranteed service increases, queue delay of all classes in a node increases. On the other hand, as shown in fig. 4 (b), (c), the normalized end-to-end throughput achieved by high throughput guaranteed service and low loss guaranteed service is a great improvement. Here, the normalized end-to-end throughput of high throughput guaranteed service is greater than 0.8 in congested networks. This is because the queue drop decreases when using the dynamic service class (AC). The result also reveals that the class with the average minimum number of backlogged packets generates high throughput. In contrast, as shown in fig 4(c), the throughput of default service in the DHSD model shows lower than that of the default service in the other model. It is verified that the DHSD model almost provides optimal end-to-end service.

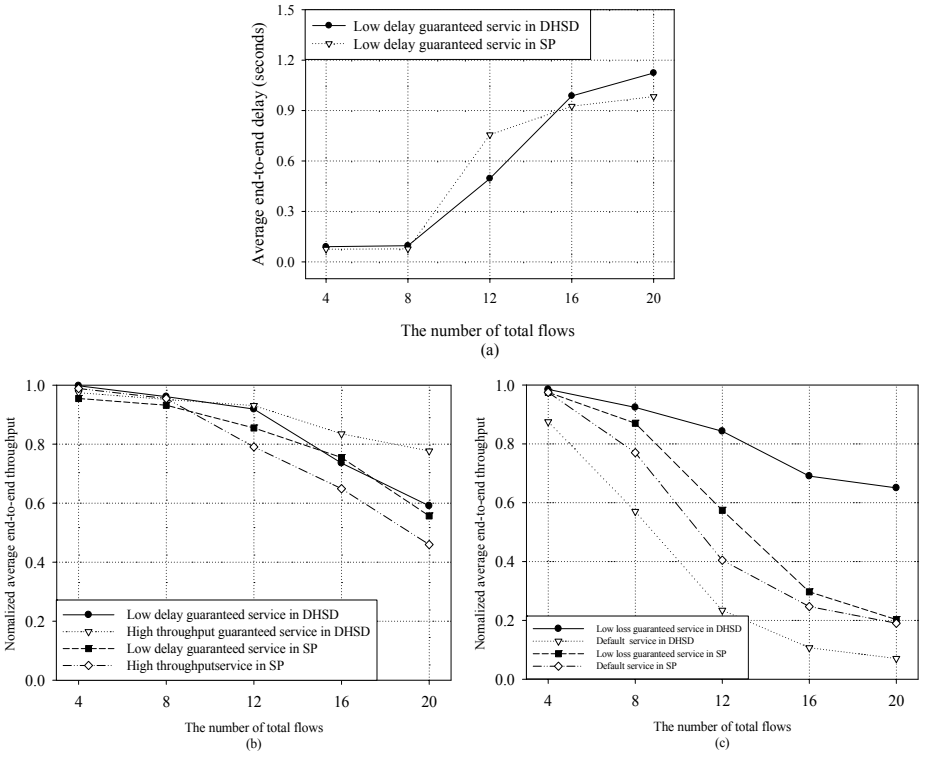


Fig. 4. The performance of each service in the multi-hop environments. (a) average end-to-end delay of low delay guaranteed and high throughput service, (b) normalized average end-to-end throughput of low delay guaranteed and high throughput service and (c) normalized average end-to-end throughput of low loss guaranteed and default service.

6 Conclusion and Future Work

In this paper, a mechanism of providing end-to-end QoS for low delay, high throughput and low loss in 802.11e based wireless ad hoc networks, is presented. In considering the traffic characteristics of the ad hoc network, four services are defined as follows: low delay guaranteed service, high throughput guaranteed service, low loss guaranteed service and default service. The proposed model supports the four services through DCS algorithms which select a service class with measurement-based current service state of each AC at each node. In simulation, the proposed solution for end-to-end QoS provisioning is shown to effectively achieve low delay, high throughput and low loss. Future studies will present the end-to-end performance of the DHSD model in wireless ad hoc environments, with random arrivals of mobile nodes.

References

1. S.B. Lee, G.S. Ahn, and A. T. Campbell.: Improving UDP and TCP performance in mobile ad hoc networks with INSIGNIA. In IEEE Commun. Mag. Vol. 39. No. 6. (2001)156-165
2. A. Lo, H.Xiao and KC Chua.: A Flexible Quality of Service Model for Mobile Ad hoc networks. In IEEE Vehicular Technology Conference. (2000)445-449
3. A. Veres G.Ahn, A.T Campbell and L.Sun.: SWAN: Service Differentiation in Stateless Wirelass Ad hoc network. In Conference on Computer Communications. (2002)
4. K.C. Wang and P. Ramanathan.: QoS assurances through class selection and proportional differentiation in wireless networks. In IEEE J. Sel. Areas Commun. Vol. 23. No. 3. (2005)573-584
5. C. Lei and W.B Heinzelman.: QoS-aware routing based on bandwidth estimation for mobile ad hoc networks. In IEEE J. Sel. Areas Commun. Vol. 23, No. 3. (2005)561 - 572
6. The IEEE P802.11 Task Group E. The IEEE 802.11e. [Online]. Available: http://grouper. ieee.org/groups/802/11/.
7. I. Aad, C. casterlluccia.: Differentiation mechanisms for IEEE 802.11. In ieee inforcom. (2001)
8. A. Iera, G. Ruggeri and D. Tripodi.: An Algorithm for Dynamic Priority Assignment in 802.11e WLAN MAC Protocols. In LNCS. Vol. 3124. (2004)1267 - 1273
9. Y. Xiao and H. Li.: Local data control and admission control for QoS support in wireless ad hoc networks. In Vehicular Technology. IEEE Transaction on. Vol. 53. No. 5. (2004)1558–1572
10. J. del Prado Pavón and S. Shankar.: Impact of Frame Size, Number of Stations and Mobility on the Throughput Performance of IEEE 802.11e. In IEEE Wireless Commun. and Net. Conf. (2004)
11. L. Romdhani, Q. Ni, and T. Turletti.: Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad-Hoc Networks. In IEEE Wireless Commun. and Net. Conf. (2003)
12. The OPNET Modeler. In http://www.opnet.com/products/modeler/home.html.
13. C. Coutras, S. Gupta, and N. B. Shroff.: Scheduling of real-time traffic in IEEE 802.11 wireless LANs. In Wireless Netw. Vol. 6. (2000)457–466
14. A. Banchs and X. P'erez.: Providing throughput guarantees in IEEE 802.11 wireless LAN. In Wireless Communications Networking Conf. Vol. 1. (2002)130-138

Transmission Range Designation Broadcasting Methods for Wireless Ad Hoc Networks

Jian-Feng Huang, Sheng-Yan Chuang, and Sheng-De Wang

Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan
sdwang@ntu.edu.tw

Abstract. The broadcast operation in ad hoc wireless networks is essential, but expensive in terms of power consumption. In DP (dominant pruning) algorithms, the number of forward nodes is treated as a criterion to measure the consumed power. However, this principle is not suitable anymore if the power-adaptive characteristic is supported and each node can detect the strength of a received packet and tune its transmission power level. Thus, in this paper, we propose a modified version of the DP algorithm with the objective being to minimize the total power consumption of forward nodes. Comparing with original DP algorithms, the sender in our algorithm not only chooses its forward nodes, but also designates the transmission ranges of its forward nodes. As a result, the designated transmission ranges not only maintain the same coverage as DP algorithms, but also reduce the total power consumption. We propose a new forward nodes selection process where the weight of a node is dependent on the incremental transmission cost rather than the effective degree. Although the resultant number of forward nodes in the proposed approach is often greater than DP algorithms, the power consumption shown in simulations is less than DP algorithms.

1 Introduction

A mobile wireless ad hoc network (MANET) is a set of mobile nodes that communicates with each other without infrastructures [14], [17]. Broadcasting is an essential operation in ad hoc wireless networks which is used to send an alarm or the routing requests [18], [19], [20] to all other nodes. Due to the mobile property of nodes in MANET, broadcasting is a basic way to provide information to or gather information from other nodes. Thus designing a good broadcasting algorithm is important in MANET. A straightforward method to do a broadcast is the blind flooding, i.e. forwarding a packet whenever a broadcast packet is received. Nevertheless, this kind of broadcasting wastes too much transmission energy because every node would forward a broadcast packet no matter it has been forwarded or not. In addition, the collision problem introduced by the blind flooding is considerable. Many nodes sending a packet at the same time cause packet collisions. After the collision occurred, a retransmission is preformed and then causes another collision. In such a chained reaction (collision and then retransmission and vice versa), the total

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 312 – 321, 2006.

consumed energy increases considerably and the traffic is also misused. It hence leads to a serious problem, called the broadcast storm problem, in the wireless ad hoc network.

To alleviate this problem, it is important to choose intermediate nodes to forward packets in a broadcast algorithm. If there is a good selection algorithm to choose intermediate nodes appropriately, it not only reduces redundant packets but also consumes less energy of the network [13]. The intermediate nodes can be chosen by either probabilistic or deterministic approaches. In [13], each node was determined as a forward/non-forward node by using a probabilistic method. As a result, there is no guarantee of receiving the broadcast packets even when the network is connected. In [1], [2], [3], [4], [7] and [15], the intermediate nodes are chosen by using deterministic methods. We focus on the deterministic algorithms in this paper. Selecting the forward nodes in a deterministic broadcast algorithm needs some extra information. The information could be local (k-hop neighborhood information) or global (the entire network topology). Algorithms that use global information are not suitable for wireless mobile ad hoc networks due to the high maintenance cost of global information in an ad hoc network.

In [7], H. Lim and C. Kim treat the broadcast cost as the number of forward nodes. In other words, their goal is to reduce redundant transmissions. Based on this strategy, they proposed two broadcasting methods: self pruning and dominant pruning. The performance analysis shows that both methods improve a lot when comparing with the blind flooding. Especially, dominant pruning outperforms self pruning due to the extent of node information where dominant pruning utilizes two-hop neighborhood information and self pruning only utilizes one-hop neighborhood information to decide a node to be a forward node or not.

In [1], Wei Lou and Jie Wu propose two enhanced dominant pruning algorithms: the TDP (Total dominant pruning) and the PDP (Partial dominant pruning). In [9], Peng and Lu proposed a CDS-based broadcast algorithm (CDSB). In addition to the previous sender, the forward nodes with lower node Ids in their algorithm is used to reduce the uncovered two-hop neighbor set. The difference between PDP (or TDP) and CDSB is that in PDP (or TDP) a designated forward node must perform forwarding, while in CDSB it can abort the forwarding if the uncover set is empty.

Another way to save energy in MANET is by adjusting the transmission power level. For saving energy, nodes can adjust the transmission power to an adequate level which is able to communicate with all the neighbors. Hence in power-aware broadcasting approaches, each node in an ad hoc wireless network assumes that it is able to adjust its power level and detect the signal strength of other nodes via receiving packets. Cartigny et al. in [4], proposed a localized approach for power-adaptive broadcasting. In their approach, any node determines its transmission range based on the RNG (relative neighborhood graph) [6]. Each node adjusts its transmission range according to the distance between itself and the farthest neighbors in RNG. Although the new graph produced by the adaptive transmission range is different to the original graph, the strong connectivity is still guaranteed [4]. Their simulation results show that it is a better method to save energy when compares with BIP (broadcast incremental power) [3] in a high density network. However, there is an assumption that in order to evaluate the distance between nodes the parameter used to model the power attenuation with distance must be known in advance. Based on the

RNG, the average node degree reduces significantly and hence the average hop diameter decreases.

In [2], Xiaohu Chen et al. considered the above situation and proposed an approach using a decreasing transmission range to reduce the power consumption in terms of local broadcasts. In local broadcasts, the redundant transmissions problem still exists. Self pruning can reduce the redundant transmissions by using a back off time to wait rather than forwarding the packet immediately. The forwarding is then eliminated if all neighbor nodes have received the packet. In most case, the back off time is chosen according to the node degree or the distance between the node and the sender.

In this paper, we make use only the localized neighborhood information and assume the transmission power level is adaptable. We also assume that the signal strength could be detected via receiving packets, just as other power-aware algorithms do. Based on a DP (dominant pruning) algorithm [1] [7], we enhance it with an adaptable transmission power level in forwarding packets. For selecting forward nodes, a different strategy is proposed in this paper. A node is marked as forward or non-forward node by the previous relaying node or the source node. A node marked as non-forward node does nothing when received a broadcast packet. And a node that is marked as a forward node relays each broadcast packet and designates which are the next forward or non-forward nodes among its neighbors, just like a domino effect. Finally, a greedy and heuristic method is proposed to reduce the total transmission cost of its forward nodes while the full coverage of entire network is still held.

2 The Transmission Range Designation Broadcast Algorithm

In our wireless communication model, we assume that each node has a default maximal power level, and this value is the same for all nodes. Without loss of generality, all antennas are assumed to be omni-directional. At beginning, each node communicates with its neighbors using the default maximal power level and then adjusts the transmission power level according to the neighbors' location. Let graph $G = (V, E)$ represent an ad hoc network, where V is the set of nodes and E is a set of edges. We say an edge (u, v) exists if u can communicate with v and vice versa. It is easy to understand that any edge in this graph is symmetric because their transmission power levels are the same. Such a graph is usually called unit disk graph and the strong connectivity of the network is guaranteed.

The medium channel model used in this paper is: $P_{rec} = P_{tx} / r^{\alpha}$, where P_{rec} is the received power, P_{tx} is the transmission power; r is the transmission range and α is a parameter that ranges from 2 to 4, depending on the characteristics of the communication medium [2] [3] [4] [5] etc.

We use a HELLO message to exchange the needed transmission power level between nodes. Thus, receivers can determine the attenuation of energy during the packet transmission. Since the wireless channel are reciprocal [8], receivers can therefore evaluate the minimum power that it needs to reach the senders. And the result would be appended to a HELLO message. After several rounds of the exchange of HELLO messages, each node knows its one- and two-hop neighbors and the required transmission power level to reach them.

2.1 Basic Idea

Dominant pruning algorithms save energy by reducing the number of forward nodes in broadcasting. Furthermore, if we can adjust the transmission power level as well as reduce the number of forward nodes in broadcasting we get even better results in energy saving. We give an example to illustrate this property. We assume that the power required to communicate between any two nodes is proportional to the square of the distance between them. In Fig. 1(a) node S is the source and node A is its only forward node by using the DP algorithm, while in Fig. 1(b), nodes A and B are forward nodes by using adaptive range designation. The total power consumption in Fig. 1(a) and Fig. 1(b) are proportional to 2.25^2=6.2 and $1.2^2+1.5^2$=3.39, respectively. In this case, it is obviously that the transmission range designation approach results in lower power consumption.

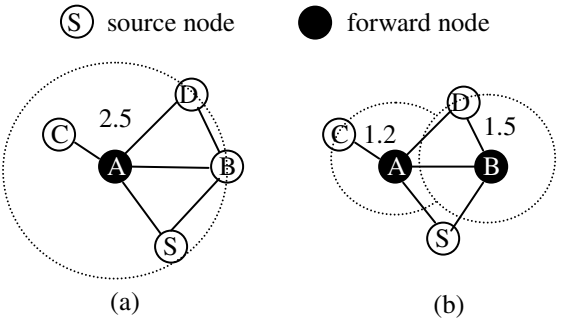


Fig. 1. The forward nodes of (a) dominant-pruning and (b) transmission range designation

In order to let each forward node know its transmission range, the source node (or the intermediate node) appends the range information of its forward nodes to a broadcast packet besides the forward nodes' Ids. Of course, the full coverage of a network for a broadcast operation must be guaranteed, where the full coverage of the entire network means that all other nodes in the network will receive the same broadcast message from the sender. For example, in Fig. 2, each neighbor that is within two-hop neighborhood will receive a broadcast packet from node *S*, where node *S* is source node, nodes *A* and *B* are designated forward nodes, and their transmission ranges are *d(A,C)* and *d(B,E)*, respectively. As shown in Fig. 2, the uncovered two-hop neighbor set of node *S* are {*C,D,E*}, the range covered by the forward node *A* contains {*C,D*}, and the range covered by another forward node *B* contains {*E*}. All nodes included in the uncovered two-hop neighbor set of node *S* are covered by forward nodes *A* or *B*.

It can be seen from Fig. 2 that there exists an edge between nodes *B* and *D,* meaning that if the transmission power level is fixed, nodes *B* and *D* will be interfered by each other. In the DP algorithm, the node *D* would receive the same broadcast packet from nodes *A* and *B*. Furthermore, the transmission contentions and collisions

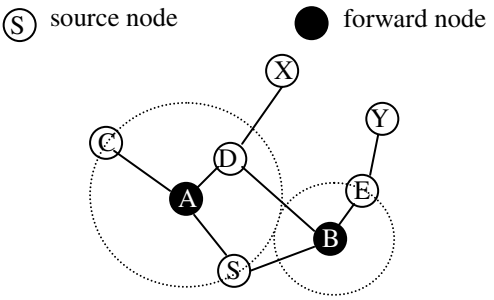


Fig. 2. The source node S designates the transmission ranges for its forward nodes

may occur due to the transmission range of node *B*. For instance, if both nodes *A* and *B* need to forward a broadcast packet, a transmission collision may occur at node *D* since nodes *A* and *B* transmit simultaneously. In another case, nodes *B* and *D* contend for the channel when doing the forward operation. However, if their transmission power levels are adjusted appropriately, there is no interference when nodes *D* and *B* transmit to node *X* and *E* respectively.

2.2 The Transmission Range Designation Broadcast Algorithm

In this section, we propose the transmission range designation (TRD) algorithm with a forward node selection process. Our goal is that any neighbor u within the range of two-hop is covered by the neighbor v within the range of one-hop and the required power of node v to reach node u has a minimum incremental power in terms of the total power consumption. Based on BIP (Broadcast Incremental Power) [3], we propose a heuristic node selection algorithm that considers the incremental cost on determining the next forward nodes. The notations used in the algorithm are given as follows:

N(v): The one-hop neighbor set of node v.
N2(v): The two-hop neighbor set of node v.
R(v): The transmission range designated by a previous sender.
NR(v): The neighbors of node v within the designated transmission range R(v).
U(v): The two-hop neighbor set whose members have not received the broadcast packet yet.
X(v): The set of candidate for forward nodes.
F(v): The set of forward nodes.

We describe the proposed transmission range designation (TRD) algorithm with a forward node selection process as follows:

Step 0: Compute the uncover set $U(v)$ and candidate set $X(v)$ for node v.
If the node v is a forward node and the node u is a previous sender, we have
$U(v) = N(NR(v)) - N(v) - N(u)$
$X(v) = NR(v) - N(u)$

If the node v is a source node, we have

$U(v) = N_2(v) - N(v)$

$X(v) = N(v)$

Step 1: For a node v, find a node $w \in U(v)$ that has only one neighbor $n \in X(v)$, remove this node w from the set $U(v)$ and set the transmission range of the node n to the node w so that the two-hop away neighbor w is covered by n, which is one-hop neighbor of the node v.

Step 2: Repeat Step 1 until there is no such node $w \in U(v)$.

Step 3: Determine the incremental cost $cost(n,w)$ of each node $w \in U(v)$ by using the following equation:

$$cost(n,w) = Preq(n,w) - Ptx(n),$$

where $cost(n,w)$ is the incremental cost for neighbor $n \in X(v)$ to reach its neighbor $w \in U(v)$, $Preq(n,w)$ is the required transmission power for node n to reach node w, and $Ptx(n)$ is the transmission power of node n, which is computed by the designated transmission range. If the $cost(n,w)$ is less than zero, just remove the node w from $U(v)$, because the node w has already been covered.

Step 4: Find out the minimum cost from $cost(n,w)$ (which is greater than zero), and then remove this node w from the set $U(v)$ and re-adjust the transmission power of the node n.

Step 5: Repeat the Steps 3 and 4 until the set $U(v)$ is empty.

An example of the selection process is shown in Fig. 3. Initially the node s is the source node and other nodes (a to k) are receivers. And Table 1 represents the required power P_{ij} for node i to reach another node j in Fig. 3. The process is described in detail as follows:

Fig. 3(a) is the original graph where no nodes are determined as forward nodes by node s and the set $U(s) = \{g, h, i, j, k\}$. After executing the TRD algorithm, the node g, in Fig. 3(b), is removed from the set $U(s)$ firstly because only one one-hop away node has connection with it. At the same time, the node a is marked as a forward node and then its transmission power P_a is determined temporally as P_{ag}, the minimal required transmission power to communicate with the node g. As the node a's forward status and designated transmission range are modified, the variable $cost(a, .)$ of other nodes may need to be recalculated. For example, $cost(a,j)$ is recalculated as $P_{aj} - P_{ag}$ rather than P_a. We perform this step continually until there are no such uni-parental nodes in the set $U(s)$. After removing all uni-parental nodes from the set $U(s)$ and then recalculating all affected $cost(a, .)$, we go to the next step where the node with minimum incremental cost is selected firstly. Hence, in Fig. 3(c), $cost(a,j)$ is selected and then we readjust node a's transmission power as P_{aj} and remove node j from $U(s)$. Although we can discover that the required power P_{aj} is greater than P_{fj}, we assign node a to transmit a broadcast packet to node j rather than assigning node f because of the requirement of the full coverage of two-hop neighbors. In the same way, nodes k, i, and h are removed from $U(s)$ in turn, and nodes e and b are marked as forward nodes and then their transmission ranges are readjusted.

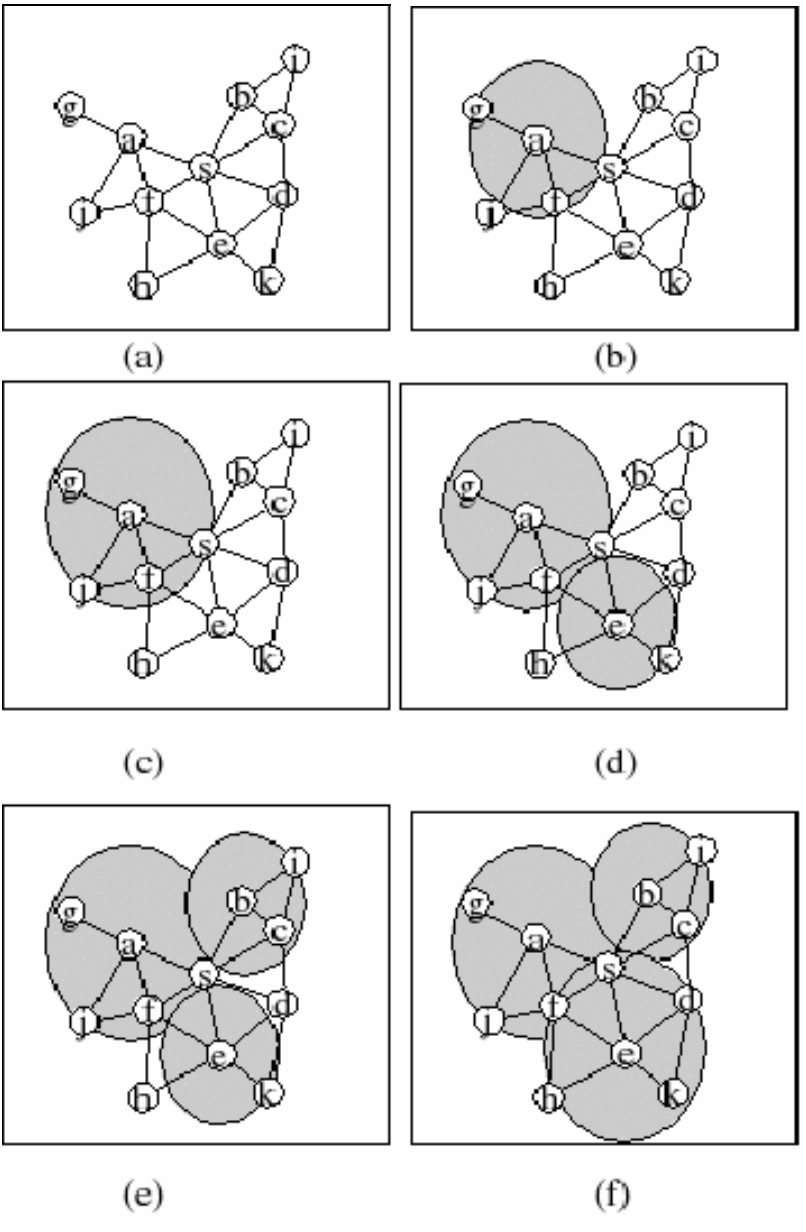


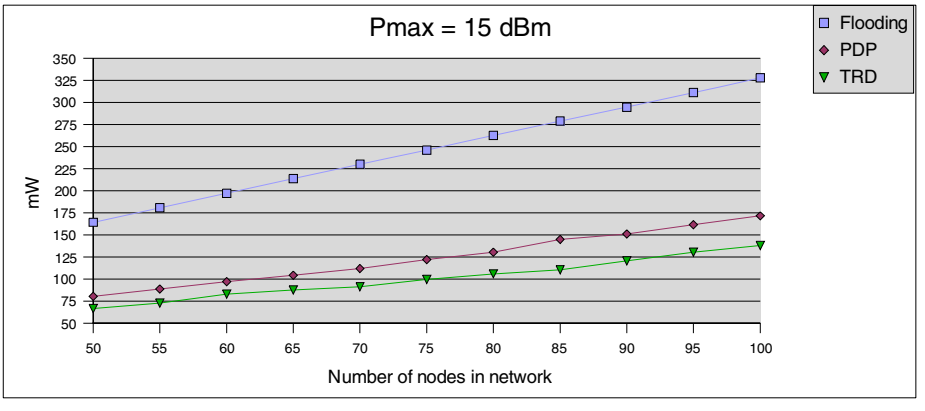
Fig. 3. The process that node s chooses its forward nodes and determines their transmission range

Table 1. The required power in Fig. 3

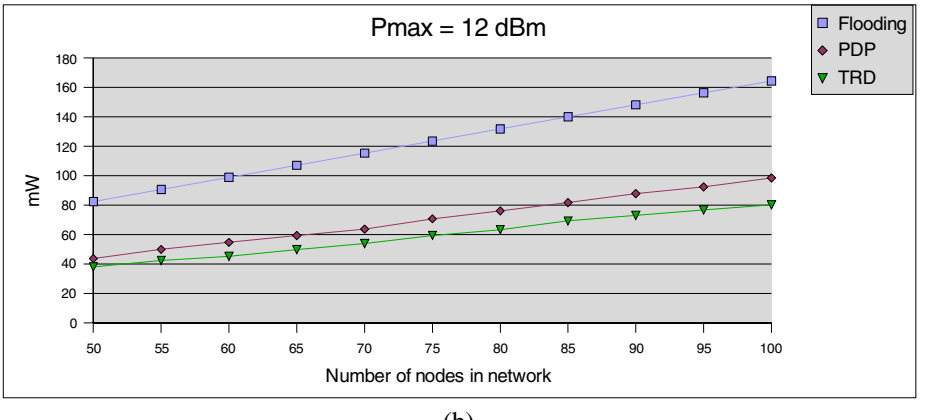
power	P_{ag}	P_{aj}	P_{fj}	P_{fh}	P_{eh}	P_{ek}	P_{dk}	P_{bi}	P_{ci}
value	7.84	12.25	8.41	10.24	13.69	6.25	11.56	6.76	7.29

3 Simulations

We use GloMoSim [12] as the simulation tool to simulate the performance of the three algorithms: DP algorithm, blind flooding and TRD (transmission range designation) algorithm. The DP algorithm is based on PDP, but the difference is that we use the adaptive transmission power level instead of the fixed transmission power level. The adaptive transmission power level is determined by the distance from the farthest neighbor. In blind flooding, each node always performs forwarding once it received the first copy of the packet and the transmission power level is fixed to default value. The last method (TRD) is the proposed algorithm and we use the common neighbors to reduce the uncovered two-hop neighbor set *U* (based on the principle of PDP).



(a)



(b)

Fig. 4. The total power consumption of a broadcast process where (a) Pmax= 15 dBm (b) Pmax = 12 dBm

In our simulation environment, we consider a wireless ad hoc network with 50 ~ 100 random static nodes. The simulation area is 1000m x 1000m. For each node, the default maximum transmission power level (P_{max}) is the same, and we use three levels (15, 12 and 9 dBm) to simulate respectively. For each given number of participating nodes, we perform our experience 20 times. Those 20 random network topologies are generated by setting the random seed. The radio bandwidth is 2Mb/s. The consumed transmission energy (CTE) is calculated as: CTE = (the time to transmit a packet) x (the transmission power level). We use the TWO-RAY as our Transmission-PathLoss Mode. It uses free space path loss (path loss exponent, sigma) = (2.0, 0.0) for near sight and plane earth (4.0, 0.0) for far sight. The length of data to be broadcasted is 1024 bytes. The Mac Layer is 802.11, and we use the broadcast mode without the support of RTS/CTS/ACK. In order to reduce the rate of the packet loss due to the transmission collisions and contentions, each node wait a random time before forwarding a packet.

Fig. 4(a) and (b) shows the total power consumption of a broadcast process. When the P_{max} is 15 dBm, our power consumption is about 81.45% of the power consumption of PDP; When the P_{max} is 12 dBm, our power consumption is about 83.83% of the power consumption of PDP; From the simulation results, we can observe that the proposed algorithm outperforms the other algorithms and the percentage of energy saving is even better when the default transmission range is larger.

The goal of our algorithm is to reduce the total power consumption during a broadcast process. The additional benefit of the adjusting of the transmission power level is the increase of spatial reuse. The increase of spatial reuse means the decrease of the interference. We use the duplicate packet ratio to evaluate our interference in the simulations. The simulation result shows that our algorithm has less duplicate packet ratio than PDP.

4 Conclusions

In this paper, we propose a DP-based broadcast algorithm for ad hoc wireless networks. The proposed algorithm, called TRD, make use of a forward nodes selection process and the concept of transmission range designation. In the existing DP-based broadcast algorithm, the number of forward nodes is the primary concern, while in the proposed strategy we consider the incremental cost. Our algorithm designates some neighbors as forward nodes, and determines their transmission ranges to reduce the power consumption. The incremental cost is used and regarded as two-hop away node's weight. The two-hop away node with minimum weight is chosen and then removed from the uncovered two-hop neighbor set U. In addition, the forward nodes should be responsible for the coverage of two-hop away nodes, so the sender determines their transmission ranges in order to cover it. The new forward nodes selection process is continued until the uncovered two-hop neighbor set U is empty. The simulation results show that the power consumption of the proposed algorithm is less than partial dominant pruning algorithms.

Acknowledgments

The work was partially supported by MediaTek Inc. Taiwan and partially supported by National Science Council, Taiwan, under the grant no. NSC 94-2213-E-002-042.

References

1. Wei Lou and Jie Wu,"On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks," *IEEE Trans. On Mobile Computing*, Volume: 1, Issue: 2, Apr-Jun 2002.
2. Xiaohu Chen, Michalis Faloutsos, and Srikanth V. Krishnamurthy, "Power adaptive broadcasting with local information in ad hoc networks," *Proc. International Conference on Network Protocols* (ICNP'03), 2003.
3. J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "On the construction of energy-efficient broadcast and multicast trees in wireless networks," *Proc. nineteenth Annual Joint Conference of the IEEE Computer and Communication Societies (INFOCOM),* 2000.
4. J. Cartigny, D. Simplot, and I. Stojmenovic, "Localized minimum-energy broadcasting in ad-hoc networks," *Proc. 22nd Annual Joint Conference of the IEEE Computer and Communication Societies (INFOCOM),* 2003.
5. T.S Rappaport, *Wireless Communications, Principles and Practices*, Prentice Hall,1996
6. G.Toussaint, "The relative neighborhood graph of finite planar set," *Pattern Recognition*, vol. 12, no. 4, pp. 261-268, 1980
7. H. Lim and C. Kim, "Muticast Tree Construction and Flooding in Wireless ad hoc networks," *Proc. ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, 2000.
8. J.P. Linnartz, *Narrow Band Land-Mobile Radio Networks*, Artech House, 1993.
9. W. Peng and X. Lu, "Efficient broadcast in mobile ad hoc networks using connected dominating sets," *Journal of software*, vol. 12, no. 4, pp. 529-536, 2001.
10. Wei Lou and Jie Wu, Localized broadcasting in mobile ad hoc networks using neighbor designation, accepted to appear in *Handbook of Mobile Computing,* I. Maghoub (ed.), CRC Press, 2004.
11. L. Lovasz, "On the ration of optimal integral and fractional covers," *Discrete mathematics* 13 (1975) 383-390.
12. GloMoSim website, http://pcl.cs.ucla.edu/projects/glomosim/.
13. S. Ni, Y. Tsereng, Y. Chen and J. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Proc. IEEE/ACM International Conference on Mobile Computing and Networking (MOBICOM),* 1999.
14. Charles E. Perkins, *Ad Hoc Networking*, Addison Wesley, 2000.
15. I. Stojmenovic, S. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks," IEEE Trans. Parallel and Distributed Systems, 13(1):14-25, Jan. 2002.
16. Andrew S. Tanenbaum, *Computer Networks*, 4th Edition, Prentice Hall, 2003.
17. E. M. Royer and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications*, 6(2):46-55, 1999.
18. C. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA), pp. 90-100, Feb. 1999.
19. D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networks," Mobile computing, T. Imielinski and H. Korth, Eds., Kluwer Academic Publ. 1996.
20. V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Protocol for Mobile Wireless Networks," *Proc. Annual Joint Conference of the IEEE Computer and Communication (INFOCOM),* 1997.

Bandwidth-Aware Multipath Routing Protocol for Mobile Ad Hoc Networks

Zhi Zhang, Guanzhong Dai, and Dejun Mu

College of Automation, Northwestern Polytechnical University, Xi'an, 710072, China
zhangzhi@mail.nwpu.edu.cn, {daigz, mudejun}@nwpu.edu.cn

Abstract. Real-time video transmission in mobile ad hoc networks is an important yet challenging problem. A Bandwidth-aware Multi-path Routing Protocol is proposed to support QoS for real-time video delivery under ad hoc network environments. We incorporate the on-demand node-disjoint multi-path routing scheme with the bandwidth estimation method. In this approach, multiple node-disjoint paths are formed during the route discovery process and are actively maintained. The detector packets measure the available bandwidth of each hop along the paths. The approximate bandwidth of a node is estimated based on the bandwidth consumption information indicated in the modified HELLO messages. The estimated bandwidth is used as the metric to choose the primary route. Simulation results show that packet delivery rate increases a lot, and end to end delay and jitter decrease significantly, while the overhead is not increase too much, compared with original single path routing protocol.

1 Introduction

A mobile ad hoc network (MANET) is a collection of wireless nodes that self-organized to form a network. This network depends on no infrastructure and nodes communicate with each other in a peer-to-peer fashion. In such kind of networks, providing support for video delivery is an important yet challenging goal. The real time video delivery applications have strict demands on high bandwidth, low latency and low jitter. Single path routing protocols (such as DSDV [1], DSR [2]and AODV[3]) fail to fulfill those requirement. However, the topology of mobile ad hoc networks provides the existence of multiple routes between two nodes which can be utilized to transmit the packet to better support video transmission. In case of path broken, alternative path still can be used to send the packets to reduce the delay and jitter of video streaming. Multi-path routing can also be used to balance load [8] by forwarding video packets on multiple paths at the same time to avoid the network congestion.

Additionally, several image compression techniques such as MPEG4 [4], H.263/H.263+[5], H.264/AVC[6] and multiple description coding[7], are designed to meet various channel conditions. The current network state, such as the available bandwidth, is desirable to feedback to the application to appropriately adjust the amount of compression.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 322 – 330, 2006.

In this paper, we propose a Bandwidth-aware Multi-path Routing Protocol (BMRP) based on AODV for mobile ad hoc networks to meet the requirements of real-time video applications. Multiple node-disjoint paths are formed during the route discovery process and are maintained by unicasting update packets along each path periodically. These update packets measure the available bandwidth of each hop along the paths. The approximate bandwidth of a node is estimated based on the bandwidth consumption information indicated in the HELLO messages from its two hop neighbors. The evaluated bandwidth is used as the metric to choose the primary path. If the difference between the current path and the best path is above some threshold value the source immediately switches to the best in contrast to waits for its current path to break.

The rest of this paper is organized as follows. In section 2, we review the related work. We present the proposed Bandwidth-aware Multi-path Routing Protocol in Section 3. In Section 4 we verify the performance of the proposed protocol. Finally, Section 5 provides our conclusions and future works.

2 Related Work

AODV is an on-demand single path routing protocol for ad hoc network. In AODV protocol, the source broadcasts a RREQ packet in the network to search for a route to the destination when it needs a route to a particular node. When a RREQ reaches either the destination or an intermediate node that has a route to the destination, a RREP packet is sent back to the source. A path between the source and the destination is established by this way. Data is transferred along this path until one of the links in the path breaks and can't be local repaired. The source is informed of this link failure by means of a RERR packet from the node upstream of the failed link. The source node then restarts a route discovery process to find a new route to the destination.

In AODV, though the source actually discovers multiple paths during the route discovery process, it chooses only the least hops route and discards the rest. Since no alternate paths are available, packets have to be buffered or dropped when route breaks. This could reduce the overall packet delivery ratio. Moreover, the frequent route discoveries can increase the average end-to-end delay and jitter when frequent link breaks in high mobile environment.

On-demand multi-path routing approach is believed to achieve better performance than the single path one [8] and a number of solutions [9, 10, 14] for multi-path routing in ad hoc networks have been proposed. In these protocols, the alternate paths are computed during route discovery and are rarely maintained during the course of data transfer. Thus the paths could become outdated by the time they are actually utilized. These routing protocols usually use the hop count as the metric which often leads to less capacity than the existing best path [11]. The path with the shortest hop count is chosen as the primary path while other paths are used only when the primary path breaks.

Some routing protocol, such as CEDAR[12], AQDR[13]disseminate the available bandwidth information to support QoS in ad hoc network. Those approaches are single path routing protocol and fail to fulfill the requirement of real time video

applications. AODV is an on-demand single path routing protocol for ad hoc network. In AODV protocol, the source broadcasts a RREQ packet in the network to search for a route to the destination when it needs a route to a particular node. When a RREQ reaches either the destination or an intermediate node that has a route to the destination, a RREP packet is sent back to the source. A path between the source and the destination is established by this way. Data is transferred along this path until one of the links in the path breaks and can't be local repaired. The source is informed of this link failure by means of a RERR packet from the node upstream of the failed link. The source node then restarts a route discovery process to find a new route to the destination.

In AODV, though the source actually discovers multiple paths during the route discovery process, it chooses only the least hops route and discards the rest. Since no alternate paths are available, packets have to be buffered or dropped when route breaks. This could reduce the overall packet delivery ratio. Moreover, the frequent route discoveries can increase the average end-to-end delay and jitter when frequent link breaks in high mobile environment.

On-demand multi-path routing approach is believed to achieve better performance than the single path one [8] and a number of solutions [9, 10, 14] for multi-path routing in ad hoc networks have been proposed. In these protocols, the alternate paths are computed during route discovery and are rarely maintained during the course of data transfer. Thus the paths could become outdated by the time they are actually utilized. These routing protocols usually use the hop count as the metric which often leads to less capacity than the existing best path [11]. The path with the shortest hop count is chosen as the primary path while other paths are used only when the primary path breaks.

Some routing protocol, such as CEDAR[12], AQDR[13]disseminate the available bandwidth information to support QoS in ad hoc network. Those approaches are single path routing protocol and fail to fulfill the requirement of real time video applications.

3 Bandwidth-Aware Multi-path Routing Protocol

In this section, we propose the Bandwidth-aware Multi-path Routing Protocol (BMRP) for mobile ad hoc network, which is built off AODV. We now describe the two main components of BMRP, including node-disjoint multi-path routing and bandwidth estimation.

3.1 Node-Disjoint Multi-path Routing

There are two kinds of multiple disjoint paths: link-disjoint and node-disjoint. Link-disjoint paths do not have any common link, but may have common nodes. Node-disjoint paths do not have any nodes in common except the source and destination. The node-disjoint paths are more stable than link-disjoint path according to the previous analysis [14] and are utilized in our approach.

3.1.1 Discovery of Multiple Node-Disjoint Paths

We modify the route discovery mechanism of AODV. The RREQ packet is modified to contain the address of the neighbor of the source through which it has been forwarded. The other intermediate nodes forward the first received RREQ packet with the same source address and sequence number. The destination node uses this information to reply to only those RREQs that forwarded from different neighbors of the source. Since every intermediate node forwards only one of RREQs which have the same source address and sequence number toward the destination, each RREQ arriving at the destination has gone through along a unique path from source to destination. When the destination replies only to RREQs from distinct neighbors of the source, these RREPs arrive at the source via node disjoint paths. The source node then stores multiple next hops for each destination in its route table. The maximum number of next hops for each destination is limited to 3 according to previous studies [15] which have shown 3 to be the optimal number of routes for multi-path routing. Furthermore, intermediate relay nodes are excluded from sending an RREP packet directly to the source.

The RREQ and RREP packets are also modified to carry the evaluated bandwidth (abbreviated as EBW) value during the route discovery process. The source is able to learn the bandwidth of the multiple paths during the route discovery as described in section 3.1.2. Once the source receives the RREPs, it stores its next hop information and chooses the path with the greatest EBW as its primary path for data transmission.

3.1.2 Maintenance of Alternate Paths

To keep the alternate paths stored at each source node adaptive to the frequently changes in the network topology, we introduce following maintain mechanism. The source node periodically sends “detector” packet to the destination along each of its alternate paths. The detector packets contain a special field for collecting the EBW along the path. Every node along the path updates this field when the detector packets propagate through the alternate paths. The destination records the EBW in the detector packet and sends a new detector packet back to the source along the same path. We use the Maximum-Minimum approach to measure the quality of the path. In this approach the EBW of the entire path is just the EBW of the weakest link. The source node chooses the path with the maximum EBW for routing. The source node will switch from its current primary path to an alternate path if the difference in theirs EBW is higher than the predefined threshold in contrast to waits for its primary path to break.

3.1.3 Disable Local Repair Mechanism

AODV makes use of a local repair mechanism that allows intermediate nodes detecting link failures to queue packets temporally while it tries to repair the route. But we do not use the local repair mechanism of AODV in case of real-time video application. First, the local repair process takes too much time which will cause great congestion for delivering real-time video packets. Second, repaired routes tend to be longer than the original. Additionally, Local repair mechanism will degrade the performance of the packet lost rate as well as the latency and jitter according to the previous simulation work [16].

3.2 Bandwidth Estimation

In the IEEE 802.11 MAC, nodes are allowed to access the wireless channel when no other nodes are transmitting packets within the interference range. Normally, the interference range is twice the transmission range. In this paper, we take the two-hops as the interference range and use the raw channel bandwidth in the estimation to approximate the bandwidth usage. We can take the bandwidth calculation problem to determining the residual bandwidth within the two-hop neighborhood range. Each node can approximate its residual bandwidth information based on information from nodes within two-hops.

We modified the Hello packet to piggyback the sender's current bandwidth usage and the sender's one-hop neighbors' current bandwidth usage. Each node estimates its available bandwidth based on the information provided in the Hello messages. The one-hop neighboring nodes' information can be obtained directly, but it's hard to get the two-hop neighboring nodes' bandwidth information directly. We use the one-hop relay to disseminate the second neighboring nodes' information. The Hello packet used in AODV only keeps the address of the node who initiates this packet. We modify the Hello packet to add two fields. The first field includes <node ID, consumed bandwidth, timestamp>and the second field includes <neighbors ID, consumed bandwidth, timestamp>. Each node determines its consumed bandwidth by monitoring the packets it feeds into the network. This value is recorded in the bandwidth consumption register at the node and updated periodically.

Once a node knows the bandwidth consumption of its one-hop neighbors and its two-hop neighbors, the residual bandwidth can be estimated as formula (1), the raw channel bandwidth minus the overall consumed bandwidth, multiply by a weight factor. We need to multiply the residual bandwidth by a weight factor α due to overhead of IEEE 802.11 MAC, overhead of routing protocol and the following situation. If a node is in sender's interference range but it isn't in any of sender's neighbors' transmission range. In this situation, the sender will never know this node bandwidth usage. However, these instances do not happen frequently since it has to meet strict requirements. So we use weight factor to overcome this situation effect.

$$Bandwidth_{available} = \alpha(Bandwidth_{raw} - Bandwidth_{all-consumed}). \tag{1}$$

$$0 < \alpha < 0.8865$$

$$\frac{DATA}{RTS + CTS + (DATA + MACHead + IPHead) + ACK} = \frac{1500}{44 + 38 + (1500 + 52 + 20) + 38} = 0.8865 \tag{2}$$

In consider the bandwidth used in RTS, CTS, ACK and the protocol head, the weight factor α should less than 0.8865 as calculated in expression (2). In addition, other factors will effect the estimation of bandwidth and α should be taken carefully.

4 Simulations and Discussions

4.1 Simulation Environment

To evaluate the performance of our bandwidth-aware multi-path routing protocol, we ran simulations using QualNet [17].We compare the performance of our solutions with that of AODV. The IEEE 802.11 MAC protocol with a channel data rate of 2Mbps was used. The traffic load used in the simulations was Constant Bit Rate (CBR) between 10 different pairs of nodes. Each data session consisted of 1500 packets of 512 bytes each sent at the rate of 10 packets per second. The other simulation parameters are as show in Table 1. The threshold used for switching paths is defined as 25%. The weight factor α is defined as 0.65. The results are averages of 5 simulation runs with different random seeds.

Table 1. Simulation Parameters

Parameter	Value	Parameter	Value
Network range	1000m*1000m	Mobility model	Random Waypoint
Number of nodes	50	Node speed	1,5,10,15,20mps
Simulation time	300 second	Pause time	10s
MAC	IEEE 802.11	Node placement	Random

4.2 Simulation Result

We evaluate the packet delivery ratio, end-to-end delay and jitter for increasing node speed (from 1mps to 20mps) in a random waypoint model.

4.2.1 Packet Delivery Ratio

Figure 1 compares the packet delivery ratio of BMRP and AODV in varying mobility conditions. The graph demonstrates that BMRP performs better than the AODV at nearly all speeds. The AODV perform well at low speeds but degrade at high speed, while the BMRP do not degrade too much. Higher packet delivery rate of BMRP is because of the availability of alternate path to forward the packets when one path fails. AODV has to resort to a new discovery when the only path fails. This is also cause the significant reduction in route discovery frequency with BMRP.

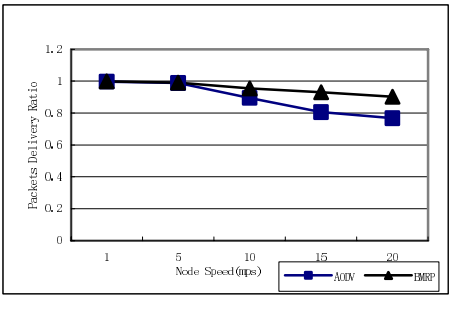


Fig. 1. Packet Delivery Ratio

4.2.2 Delay and Jitter

Figure 2 and Figure 3 presents the average delay and jitter as a function of the mobility speed respectively. BMRP outperforms the AODV at all speeds. The regular maintenance of the paths in BMRP leads to an increased availability of valid alternate paths when the primary path breaks. The frequent rediscovery of route of AODV in high speed increase the end-to-end delay and jitter.

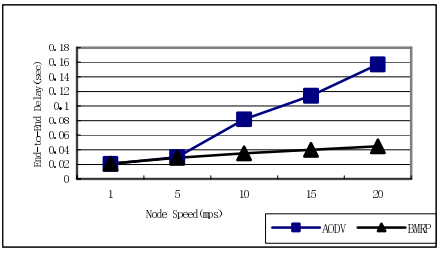


Fig. 2. End to End Delay

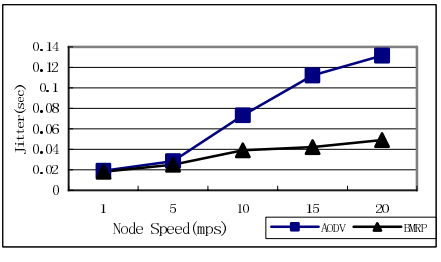


Fig. 3. Jitter

4.2.3 Overhead

Even though BMRP significantly reduces the number of route discoveries, it has more overhead per route discovery and route maintenance. This is because of the use of additional RREPs to form multiple forward paths to the destination and the use of

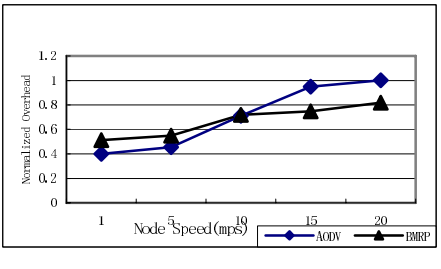


Fig. 4. Normalized Overhead

HELLOs to evaluate the available bandwidth. Figure 4 shows the overall normalized overhead of routing, BMRP is slightly higher than AODV in low speed but is lower in high speed.

5 Conclusion and Future Work

In this paper, we proposed an approach for multi-path routing in mobile ad hoc networks and introduced bandwidth estimation by disseminating bandwidth information through HELLO packets. The main characteristic of this approach is that it can adapt to dynamical network topology by proactively estimating the available bandwidth of each path to the destination and always using the best path. Simulation results show that the protocol's performance is superior to the AODV in all most scenarios.

Future research will focus on optimally distributing traffic over multiple paths to upgrade the protocol's performance.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments. This research was supported by Aviation Science Fund (05F53029) and Innovation and Technology Fund of Northwestern Polytechnical University.

References

1. C. Perkins, P. Bhagwat: Highly dynamic destination sequenced distance vector routing for mobile computers, in Proceedings of ACM SIGCOMM'94, (1994)
2. D. Johnson , D. Maltz: Dynamic source routing in ad hoc wireless networks, in Mobile Computing, Kluwer Academic Publishers (1996) 153–181
3. C. E. Perkins , E. M. Royer: Ad hoc on-demand distance vector routing, in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, (1999) 90–100
4. IEC 14496-2, Coding of Audio-visual Objects: Visual, Final Draft International Standard lSO/IEC JTCI/SC29/WG11 N2502 (1998)
5. G. CotC, B. Erol, M. Gallant, F. Kossentini: H.263+:Video Coding at Low Bit Rates, IEEE Trans. on C&S for Video Tech., vol. 8, no. 7, (1998)
6. 2 ITU-T and ISO/IEC JTC 1: Advanced Video Coding for Generic Audiovisual services, ITU-T Recommendation H.264 - ISO/IEC 14496-10(AVC), (2003)
7. S. Servetto, K. Ramchandran, V. Vaishampayan, K. Nahrstedt: Multiple-Description wavelet based image coding, in Proceedings of the IEEE International Conference on Image Processing, (1998) 659–663.
8. S. Mueller, R. P. Tsang, D. Ghosal: Multipath routing in mobile ad hoc networks: issues and challenges, Invited paper in Lecture Note in Computer Science, (2004)
9. Mahesh K. Marina, Samir R. Das: On-demand multipath distance vector routing in ad hoc networks in Proceedings of the IEEE International Conference on Network Protocols (ICNP), Riverside, California, (2001) 14–23.

10. S. J. Lee and M. Gerla: Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks, Proceedings of the IEEE ICC, (2001) 3201-3205
11. D. S. J. De Couto, D. Aguayo, B. A. Chambers, R. Morris: Performance of Multihop Wireless Networks: Shortest Path is Not Enough, Proceedings of the First Workshop on Hot Topics in Networking, Princeton, New Jersey, (2002)
12. P. Sinha, R. Sivakumar, V. Bharghavan: CEDAR: a Core-ExtractionDistributed Ad hoc Routing algorithm," in IEEE Infocom99, New York, NY, (1999)
13. Q. Xue , A. Ganz: Ad hoc QoS On-demand Routing in Mobile Ad hoc Networks, Journal of Parallel and Distributed Computing, (2003)
14. Xuefei Li, Laurie Cuthbert, Stable Node-Disjoint Multipath Routing with Low Overhead in Mobile Ad hoc Networks, in Proceedings of the IEEE MASCOTS04, (2004)
15. Nasipuri, R. Castaneda, S. R. Das: Performance of multipath routing for on-demand protocols in mobile ad hoc networks, in ACM Mobile Networks and Applications (MONET) Journal, vol. 6, (2001) 339–349.
16. D. Dubke, K. Farkas, B. Plattner, Real-Time Multiplayer Game Support Using QoS Mechanisms in Mobile Ad Hoc Networks, WONS06, (2006)
17. SNT, www.scalable-network.com

Adaptive Power-Aware Clustering and Multicasting Protocol for Mobile Ad Hoc Networks

James Jiunn Yin Leu[1], Ming-Hui Tsai[1], Tzu-Chiang Chiang[2], and Yueh-Min Huang[1]

[1] Department of Engineering Science, National Cheng-Kung University, Taiwan, ROC
[2] Department of Information Management, Hsing-Kuo University of Management, Taiwan
huang@mail.ncku.edu.tw

Abstract. One of the most critical issues in wireless ad hoc networks is represented by the limited availability of energy within network nodes. Most of the researches focused on the problem of routing issues rather than energy efficiency or prolongation of network lifetime. In this paper, we proposed a multicast power greedy clustering algorithm (termed as MPGC) with the mesh scheme in the multicasting protocol of ad hoc wireless networks. The greedy heuristic clustering partitions a large-scale ad hoc network into a hierarchical cluster structure. Nodes in a cluster determine adaptively their power levels so as to be power efficient. The clusterheads acting as the agents of transmitters/receivers can reduce efficiently bandwidth consumption and complexity of mesh structures. Besides, the mechanism of cluster maintenance can remarkably prolong the network lifetime. The power aware multicasting protocol based on ODMRP executes suitably on the super-nodes topology formed by clusterheads. The results of the simulation show that our scheme achieves better performance for ad hoc networks, in terms of network lifetime and network scalability.

1 Introduction

An ad hoc network is a dynamic wireless network established by a group of mobile nodes on a shared wireless channel without any infrastructure. A communication session is achieved either through single-hop transmission if the recipient is within the transmission range of the source node, or by relaying through intermediate nodes otherwise. For this reason, ad hoc networks are also called multi-hop packet radio networks. However, nodes are usually powered by batteries of limited capacity in this circumstance. Once the nodes are deployed, it is difficult or even impossible to recharge or replace their batteries in many application scenarios. Hence, reducing power consumption seems to be the only way to prolong network lifetime. For the purpose of energy conservation, each node can dynamically adjust its transmitting power based on the distance to the receiving node and the background noise.

Recently, multicasting has emerged as one of the most focused areas in the field of networking, e.g., video conference, distance learning and video on-demand. Some different multicast protocols have been proposed in mobile ad-hoc networks, but most of them do not take the power consumption into account which can prolong

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 331 – 340, 2006.

efficiently the whole network lifetime. In this paper we focus on the multicast protocol with power awareness and cluster issues. We expect to improve scalability [3] of the network and prolong the whole network lifetime. However, the most of proposed multicasting algorithms, ODMRP [1], MAODV [2], AMRoute [5], and CAMP [6], did not come from the consideration of power consumption. Even that Caimu Tang et al. [7] proposed a combined clustering and multicasting protocol which has explicitly proved the property of "*a larger cluster always advantageous*" but no thorough description about clustering and multicasting scheme in detail.

In this regard, we propose greedy heuristic clustering algorithm with the mesh scheme in the multicasting protocol of ad-hoc wireless networks, expecting to make the multicasting protocol of ODMRP operate suitably in the super-nodes topology formed by clusterheads. Super-nodes acting as the agents of transmitters/receivers can reduce efficiently bandwidth consumption and complexity of mesh structure. Though those nodes will deplete their energy in a faster rate, a suitable "take turns" scheme will overcome the drawbacks.

2 Related Works

Recently, several routing protocols propose energy efficient schemes. Singh et al. [9] first raised the power awareness issue in ad hoc routing and introduced new metrics for path selection, which include the energy consumed per packet, network connectivity duration (i.e., the time before network partitions), node power variance, cost per packet, and maximum node cost. The following represents several power aware algorithms on different consideration:

2.1 Minimum Battery Cost Routing (MBCR)

In [9] it calculates the sum of the power residual of mobile nodes and chooses a route with the maximum sum of power residual from all possible routes. Let C_i^t be the battery capacity of a node n_i at time t ranging between 0 to 100. The battery cost function of a node n_i is defined as the following:

$$f_i\left(c_i^t\right) = 1/c_i^t \tag{1}$$

As the battery capacity decreases, the value of cost function for node n_i will increase. The battery cost R_j for route i, comprising D nodes, is

$$R_j = \sum_{i=0}^{D_j-1} f_i\left(c_i^t\right) \tag{2}$$

Hence, to choose a route with the maximum remaining battery capacity, we select a route i which has the minimum battery cost.

$$R_i = \min\left\{R_j \mid j \in A\right\} \text{ Where } A \text{ is the set containing all possible routes.} \tag{3}$$

Due to the above route selection criterion in MBCR, it is a possible problem which the route has the minimum battery cost along with almost energy depletion on some

intermediate nodes. Those nodes may die quickly and result in the selected route broken more rapidly than other routes.

2.2 Min-Max Battery Cost Routing (MMBCR)

C.K. Toh et al. [8] proposed a different metric for route selection to improve the shortcoming and ensure that no node will be overused. The above function (Eq. 3) can be modified as the underlying equation.

$$R_j = \max_{i \in route\_j} f_i\left(c_i^t\right) \tag{4}$$

Battery cost R*j* for route *j* is redefined. Similarly, the criterion of route selection is modified as the following equation.

$$R_i = \min\left\{R_j \mid j \in A\right\} \tag{5}$$

The lifetime of the route with nodes having the most battery cost is shorter than all other possible routes. The key node effecting on the route lifetime is the first node run out of energy. On this criterion, it tries to avoid the route with nodes having the least battery capacity among all nodes in all possible routes. Though it can't be promised that the total power consumption of the selected route is the most efficient, the route lifetime is distinctly elongated.

As shown in Fig. 1, the source node A has two routes to reach the destination node H. Comparing between the most battery cost of node (i.e., 1/3 at node C) in Route1 and the most battery cost of node (i.e., 1/1 at node F) in Route2, the route (Route 2) is selected according to the Eq. 5.

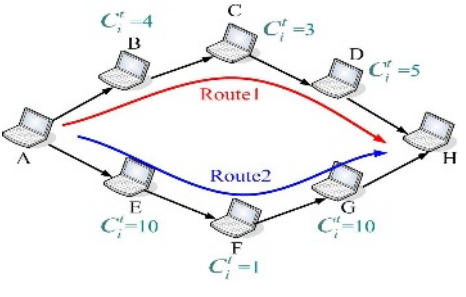


Fig. 1. MMBCR route selection

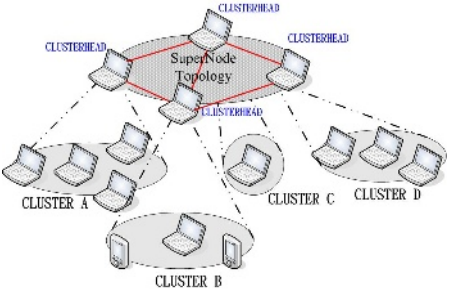


Fig. 2. Super-nodes Topology

2.3 Power Aware Clustering Algorithms

In recent decade, there are many clustering algorithms proposed successively including a variety of categories on the specifically desired [**13**] or combined (hybrid) metrics, such as mobility awareness based [4], loading balance based [12], and low maintenance based [11]. Here, we focus on the category of power awareness based [10]. Clustering has two main processes: formation and maintenance. The power aware should be adopted into the two processes for energy efficiency or network lifetime prolongation.

2.3.1 Power awareness in Forming Process

The kind of clustering formation algorithm is mainly in accordance with the assumption that mobile nodes can control adaptively its power level while transferring. Caimu Tang et al. [7] used a simple greedy heuristic algorithm to determine the power levels of all nodes. It grew clusters with successively increasing the power levels in distributed self-organized approach. The lack of precious description about clusterheads' alternation and nodal mobility is its weakness; especially when some nodes (clusterheads) bear excess tasks.

2.3.2 Power Awareness in Maintaining Process

To avoid the rapid energy depletion of clusterheads, a "take turns" scheme is must to prolong the whole network lifetime. In IDLBC [14] a virtual ID (VID) is set as its ID at first and is regarded as the priority for clusterhead selection. Whenever a node serves as a clusterhead continuously for the duration, it resets its VID to 0 and become a non-clusterhead node. Sequentially, a non-clusterhead with the largest VID in the neighborhood can resume the clusterhead function. However, this kind of new clusterhead selection may introduce ripple effect of re-clustering over the whole network.

3 Adaptive Power-Aware Clustering and Multicasting

We base the design of our power-greedy cluster multicast protocol on adaptive power control and the adaptation of On Demand Multicast Routing Protocol. Importantly, to concentrate on reduction of energy consumption and prolongation of the whole network lifetime, we first form the hierarchical cluster structures with greedy power control where each node can adjust flexibly its transmission power to fit individual geographical location. We use the refined version of ODMRP, called as Power aware ODMRP (PODMRP), whose super-nodes topology composes of the above chosen clusterheads, shown as Fig 2.

3.1 Clustering on Greedy Heuristic

Clustering partitions an ad hoc network into an amount of clusters and chooses some nodes as clusterhead. These nodes will comprise super-nodes topology and act as forwarding agents for their cluster members. Packets from multicast server will be delivered to all its multicast members via the clusterhead along those super-nodes. Before clustering, we assume that each node can control its power intensity as jth level within m levels for radio transmission, where $\{j, m | j, m \in$ positive integer, $1 \leq j \leq m$ and m is a limited and small value$\}$. Adaptive power level setting for individual nodes can retrench the dispensable energy consumption. Though the stronger intensity of power brings the more nodes within the same cluster, it results in the more energy cost of transmitter. Even that the more clusters in the same ad hoc network, the more involved clusterheads consuming their energy among super-nodes. Therefore, we adopt greedy heuristic clustering in a distributed manner to accomplish energy efficiency which is described as the underlying three phases based on [7]:

Beacon phase
Each node sends a beacon signal with highest power for being aware of its neighbors. Each node will collect the information (i.e. ID, battery residual and so on) from its neighbors on receiving this beacon signal. An assumption is that none node will not receive any beacon from its neighbor.

Greedy phase
Each node sends the clusterhead declaration with the necessarily sufficient power level to reach its closest neighbor(s) and then it increases its power level step by step until no more new neighbors seen or the specified power level (*m*th level) reached. In other words, the cluster grows up only if the number of communicable neighbors increases along with incremental power level. In this phase, each node will determine its adaptive power level on this greedy heuristic; simultaneously, the required power level to reach the certain neighbors will be stored. Therefore, this topology snapshot is unalike the topology in which each node does not restrict its power level (full power), see Fig 3.

Recruiting phase
Because of the previous beacon phase, each node has the information accompanied with beacon signal including power residual of its neighbors. Hence, we choose a node as clusterhead only if its power residual is the maximum between its neighbors and itself. Note that "*its neighbors*" means the neighbors could be connected while the clusterhead using its adaptive power level determined in the greedy phase. The cluster formation snapshot is also shown as Fig 3(c) and (d). The gray nodes and the circles stand for clusterheads and their transmission range, respectively.

$$Clusterhead = \left\{ i \mid \max_{i \in n} [power\_residual_i] \right\} \tag{6}$$

$Power\_residual_i$: the power residual of node i.
n: the set of a node's neighbors and itself.

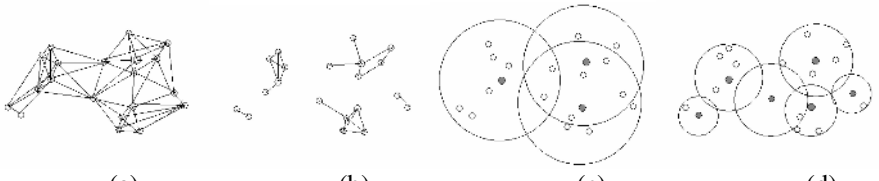


Fig. 3. Topology (a) with full power and (b) with restricted power, Cluster formation (c) and (d) derived from (a) and (b), respectively

3.2 PODMRP

Whereas the absence of energy efficiency of the original ODMRP, we propose a significant routing selection criterion for energy efficiency to prolong the whole network lifetime. The criterion is prior to maximum battery residual other than the original metric such as minimum delay despite of the route not being the shortest

path. Therefore, our PODMRP chooses its multicast paths with careful consideration about lifetime of all candidate paths such as MMBCR dose. Moreover, the original ODMRP is poverty-stricken while considering scalability due to multiple multicast sources comprising of complex mesh construction. Our PODMRP executing only on the super-nodes topology formed by the above hierarchical architecture provides significant assist to defeat the distress of scalability.

The mesh topology consists of all super-nodes which is executing PODMRP (reactive routing). In addition, the other nodes in the same cluster are running table-driven (proactive) routing protocol, i.e. DSDV. We represent the reactive routing and the proactive routing between super-nodes (inter-cluster) and between nodes in the same cluster (intra-cluster) as the underlying description, respectively.

3.2.1 Reactive Routing Between Inter-clusters

We assume that each super-node can adjust adaptively its power level to communicate with at least one other super-node for ensuring the connectivity of the entire network. Here the power level between different super-nodes can be different depending on actual situation, i.e. distance between the two super-nodes. For accurately executing PODMRP in super-nodes topology, we have to revise the Join Query and Join Reply with an additional field, maximum battery residual (MBR). The routing process consists of three steps as the following:

1. A super-node as a multicast source sends a Join Query with its MBR filled with an appropriate large value. We do not necessarily consider the battery residual of source super-node and destination super-node by reason of their necessary presence in actual routing.
2. Continuously, a super-node receives the above flooding dissemination and compares itself batter residual with the MBR in the packet. The MBR field within the dissemination will be overwritten by the smaller value and the dissemination be forwarded.
3. After a destination super-node received this dissemination, waiting an appropriate period, the most stable path (meaning, maximum lifetime) will be chosen as the multicast path and the Join Reply will be advertised along the path to source super-node.

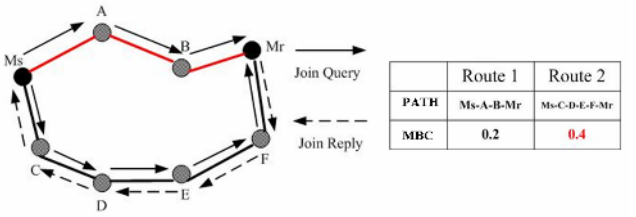
The condition function of route selection follows as:

$$R_{PODMRP} = \max_{R \in M} \left[\min_{i \in R} \left(battery\_residual_i \right) \right] \tag{7}$$

M: be denoted as the set of all possible multicast routes to a certain destination super-node.

i : be denoted as the subscription of a super-node in a multicast route except for both of source and destination super-nodes.

While the Join Query disseminated from multicast source via intermediate super-nodes to the receivers, there might be multiple paths. As Fig 4 shown, Ms and Mr denote as a multicast source and receiver, respectively. There is two distinct paths between Ms and Mr, route 1 and route 2 is (Ms-A-B-Mr) and (Ms-C-D-E-F-Mr), respectively. However, the battery residual of intermediate super-nodes is A=0.2, B=0.6, C=0.4, D=0.8, E=0.7, and F=0.6. On (7), Mr chooses the route 2 to reply the Join Reply to Ms and this route will be its data packet route.



	Route 1	Route 2
PATH	Ms-A-B-Mr	Ms-C-D-E-F-Mr
MBC	0.2	0.4

Fig. 4. Route selection on MBC criterion

3.2.2 Proactive Routing Between Intra-cluster

This kind of routing is no difference with the previously most proposed clustering algorithms. Nodes first propagate packets to their clusterhead (super-node) from one hop away. The clusterhead possesses all information of its cluster members in the table-driven routing scheme. Although in the original ODMRP, no explicit control packets need to be sent to join or leave the multicast group. However, in our revised version, a cluster member wants to join or leave a certain multicast group, it has to send a signal to update the table maintained by their clusterhead. Those clusterheads then in super-nodes topology act as the node in original ODMRP.

Since the super-nodes topology will be used for all multicast data forwarding, the energy consumption of super-nodes will by far exceed the other non-super-nodes. For avoiding the power drain of some super-nodes and reducing the network lifetime, nodes in a cluster can take turns to become clusterheads with some energy threshold or some scheduling scheme. As for the execution of our PODMRP on super-nodes, the routes with maximum power residual will be chosen specially based on power aware criterion retain the characteristic of longer lifetime.

3.3 Cluster Maintenance

The dynamic nature of the mobile nodes in mobile ad hoc networks causes the mobile nodes to join and leave the clusters, thereby disturbing the membership of the clusters. Therefore cluster maintenance schemes are required to handle new admissions and releases of non-clusterhead nodes, to take clusterheads turns, and to absorb clusters to be a larger one. Whenever the role of clusterheads is taken over by other nodes, the tables (i.e. cluster members, multicast groups and members) kept in the original clusterhead should be transfer to the new one. We give two categories about the maintenance as the underlying:

Clusterhead related maintenance

This kind of maintenance is also regard as **super-nodes topology related.** Whenever a clusterhead drains its energy approaching to a thread hold ρ (i.e., 1/4 maximum battery capacity), the "take turns" scheme is excited to maintain this cluster. The cluster member with the most battery capacity in this cluster will take over the role of clusterhead from the original one and leave its determined power level unchanged. Therefore, depending on their neighborhood information, they (including the old clusterhead) join the nearest clusterhead and readjust their power level. Besides, there is another situation that two clusterheads are close to each other within their

transmission range. The clusterhead with the less battery capacity relinquishes its clusterhead role and joins the cluster, in the meantime its cluster members which lose their cluster affiliation will join a cluster as the above described. Last, there is an interest situation that a non-clusterhead moves/leaves into the non-covered region. Because a node can not hear any clusterhead in the region, it declares itself as a clusterhead. The clusterhead dissemination is the same as those in recruiting phase.

This kind of scheme can eliminate the ripple effect or the whole network re-clustering which are the main clustering costs. The rapid convergence of localized re-clustering without the requirement of excess neighborhood information exchange supports strongly the execution of PODMRP on the super-nodes topology, especially on demand protocol. Only the new clusterheads and the relinquished clusterheads result in the super-nodes topology changed.

Non-clusterhead related maintenance

Different to the clusterhead related maintenance, this kind of maintenance is easier that the above. It does not affect the super-nodes topology and means that no clusterheads will be relinquished or raised. Due to the mutual movement of non-clusterheads and clusterheads, a non-clusterhead may moves/leaves into the covered region. Because a node in the region can hear at least one clusterhead, there is no newly raising cluster. If it moves/leaves beyond the range of its original cluster, it will join the nearest clusterhead and adaptively adjust its power level.

4 Performance Evaluation

In MPGC, only the clusterheads have the opportunity to be the multicast forwarding group nodes and they have adaptive “take turns” scheme to avoid out of energy. On the contrary, in ODMRP all nodes participate the multicast routing and without the adaptive power control. This section describes the simulations conducted to evaluate the performance of MPGC. We compare MPGC with ODMRP in terms of network lifetime and network scalability. In the simulations, we focus on the impact of the battery capacity of nodes and assume that the energy depletion is free to node movements. These simulations were carried out using the software simulator.

Let there be 200, 400, and 600 nodes randomly distributed in a square of 600m by 600m, respectively. The mobility model used in each of the simulations is in a random direction and each node then moves at an arbitrary speed between 0 and 20 meter/second. The transmission range of each node can reach 144m while using full power level. The nodal power control functions as the maximum transmission range reaching 64m within 8 levels (i.e., m=8 and a node using 5th power level can reach the 25m transmission range). The average time of a node draining out of its battery is 2 hours while using continuously full power in a 2Mbits/sec data transmitting rate. There are 10, 20, and 30 multicast groups executing in the scenarios respectively and each group has 10 receivers initially. For instance, MPGC-20 means that there are 20 distinct multicast groups independently and simultaneously executing the MPGC protocol.

In Figure 5, we use the metric of “the percentage of survival nodes” to evaluate the performance of protocols. The larger percentage of survival nodes demonstrates the better performance in terms of prolongation of network lifetime and balance among

nodal energy consumption. Obviously, the results using MGGC are better than ODMRP after executing for 3 hours. Note that all of the multicast servers are still alive. Moreover, the more multicast groups participate, the more forwarding group nodes be needed. Observing the results, all of the MPGC protocols go beyond 85% in this metric and present the excellent performance than ODMRP.

In addition, we evaluate carefully the network lifetime, i.e., the duration from the start to the time that any node dies. As the Figure 6 shown, our MPGC protocols have longer network lifetime than ODMRP. Intuitively, the nodal mobility may contribute possibly the similar scheme of "take turns" to prevent the overuse of some nodes. But without the supports of hierarchical clustering structure, adaptive power control, and cluster maintenance, the first dead node in ODMRP appears extremely earlier than in MPGC. With regard to scalability, there are more route selections in the larger ad hoc networks within the same square so as the opportunity of the "take turns" scheme increases obviously. So the performance of the lager networks represent better than the smaller one.

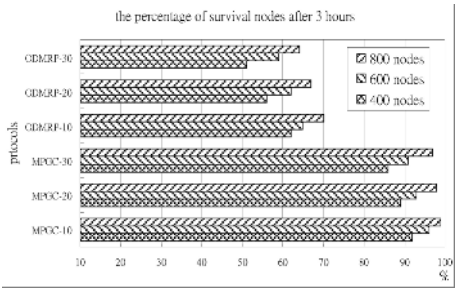


Fig. 5. The percentage of survival nodes

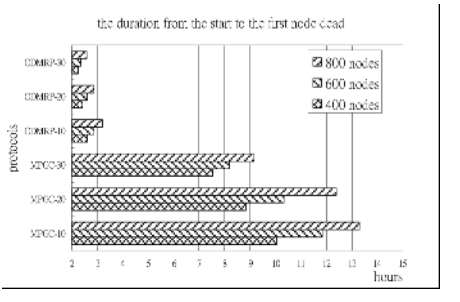


Fig. 6. The duration from the start to the first node dead

5 Conclusion

This paper described greedy heuristic clustering, power aware multicasting and clustering maintenance that try to be energy efficient and prolong the network lifetime. We assume that each node has multiple power levels for transmission and any clusterhead among the super-nodes can connect directly at least one of the other clusterheads for guarantee of strong connection. The greedy heuristic clustering tries to partition a large scale ad hoc network into clusters. Simultaneously, it adjusts all nodes' power level for the purpose of power conservation. The selected clusterheads comprise the super-nodes topology which our power aware multicast (PODMRP) can execute on. Importantly, the cluster structure could be disturbed due to the mobility of nodes. The outstanding maintenance scheme provides enough stability and rapid convergence for the super-nodes topology without the need of excessive neighborhood information exchange. As the simulation results presented, our algorithm has shown the better performance than other protocols in terms of network lifetime and network scalability.

References

1. C.C. Chiang, M. Gerla, and S.J. Lee :On-demand multicast routing protocol in multihop wireless. Mobile Networks and Apllications, 2002, pp.441-453.
2. E. M. Royer and C. E. Perkins. :Multicast operation of the ad-hoc on-demand distance vector routing protocol. in Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking, Aug.1999,pp.207–218.
3. Ming-Hui Tsai, Tzu-Chiang Chiang and Yueh-Min Huang : On Scalability and Mobility Management of Hierarchical Large-Scale Ad Hoc Networks. Lecture Notes in Computer Science,Volume 3823 ,pp. 714 - 723,12/2005.
4. Beongku An, Symeon Papavassiliou :A mobility-based hybrid multicast routing in mobile ad-hoc wireless networks. IEEE Military Communications Conference, no. 1, October 2001 pp. 316-320
5. J. Xie, R. R. Talpade, A. Mcauley, and M. Liu :AMRoute: ad hoc multicast outing protocol. Mobile Networks and Applications, Dec. 2002.
6. J. J. Garcia-Luna-Aceves and E. L. Madruga : The core-assisted mesh protocol. IEEE Journal Selected Area in Communications, Vol. 17, No. 8,Aug. 1999,pp.1380–1394.
7. C. S. Raghavendra, and C. Tang : Energy efficient adaptation of multicast protocols in power controlled wireless ad hoc networks. Mobile Networks and Applications, 2004, pp.311-317
8. C.-K. Toh, H. Cobb, and D. A. Scott : Performance evaluation of battery-life-aware routing schemes for wireless ad hoc networks. IEEE International Conference on Communications, vol.9,2001, pp.2824-2829
9. S. Singh, M. Woo, and C. S. Raghavendra: Power-aware routing in mobile ad hoc networks. in Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, Oct. 1998.
10. C. F. Chiasserini, I. Chlamtac, P. Monti, and A. Nucci: An Energy-Efficient Method for Nodes Assignment in Cluster-Based Ad Hoc Networks. Wireless Networks 10, 223–231, 2004, pp 223-231.
11. J. Y. Yu and P. H. J. chong: 3hBAC (3-hop between adjacent clusterheads) a Novel Non-overlapping Clustering Algorithm for Mobile Ad Hoc Networks. in Proc. IEEE Pacrim'03, vol. 1, Aug. 2003, pp. 318-321.
12. Christian Bettstetter : The Cluster Density of a Distributed Clustering Algorithm in Ad Hoc Networks. IEEE Communications Society, 2004, pp 4336-4340.
13. Tzu-Chiang Chiang, Ming-Hui Tsai and Yueh-Min Huang: Adaptive Clustering with Virtual Subnets Support in Ad Hoc Networks. Lecture Notes in Computer Science,Volume 3992 ,pp. 1008–1015,5/2006.
14. A. D. Amis and R. Prakash: Load-balancing clusters in wireless ad hoc networks. in Proc. 3rd IEEE ASSET'00, Mar. 2000, pp.25-32.

Backtracking Based Handoff Rerouting Algorithm for WiMAX Mesh Mode

Wenfeng Du, Weijia Jia, and Wenyan Lu

Department of Computer Science, City University of Hong Kong,
Kowloon, Hong Kong SAR China
{Wenfeng, itjia, Wenyanlu}@cityu.edu.hk
http://www.cityu.edu.hk

Abstract. Reconstruct a new route for network services during the handoff process is a fundamental issue of wireless communication. This paper proposes a new rerouting algorithm to achieve a fast handoff based on *k*-hop backtracking mechanism. The algorithm can dynamically decide the backtracking hops according to the velocity of wireless devices and the current network bandwidth through iterative strategy. During the backtracking process, our algorithm is able to find out an optimized route for the handoff network services and require all intermediate nodes which has received the *Location Update* information forward all received packets to Mobile Terminal with their optimal route to the destination subscriber station. This will greatly reduce the cost of packet forwarding during the handoff process.

1 Introduction

With the quick development of wireless network, the IEEE 802.16 protocol, also known as World Interoperability for Microwave Access (WiMAX), for wireless metropolitan networks (Wireless MAN) has been proposed and standardized. It has provided two modes to share the wireless medium: PMP (Point to Multipoint) and Mesh mode [1]. Wireless Mobile Terminal (MT) can access the Internet through Subscriber Station (SS) in WiMAX network (see Fig. 1).

Due to the mobility of MT, once MT leaves current SS's service area and enters into destination SS's service area, the route of ongoing application on MT must be reconstructed in order not to incur interruption. Meanwhile, all packets sent to MT must be forwarded to the destination SS with short delay and lossless. Handoff represents the process of changing some radio parameters of a channel (e.g., frequency, time slot, and spreading code) occupied by an existing connection when MT crosses a cell boundary or the received signal on currently employed channel is in a deteriorated quality. Due to the limitation of wireless network bandwidth, the service provider always uses a large number of little residual capacity SSs to provide wireless network service, which may increase the ongoing application handoff frequency [2].

In WiMAX PMP mode, Base Station (BS) plays an important role in the communication between all subordinated SSs. SS can consult BS the best route to destination SS and all packets exchanged between SSs will be forwarded by BS. However, in

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 341 – 350, 2006.

WiMAX Mesh mode, SSs are connected by direct connection or relayed through other SSs, thus there is no central administration node. SS can only send *Location Update* message to its neighbor SS to obtain the local information about the entire network. In this mode, the new constructed route must be constructed in an iterative way on the reference of local information managed by intermediate SSs between current SS and Correspond Node (CN). Hence, how to reconstruct a new route for network services in WiMAX Mesh mode during the handoff process is a fundamental issue of wireless communication.

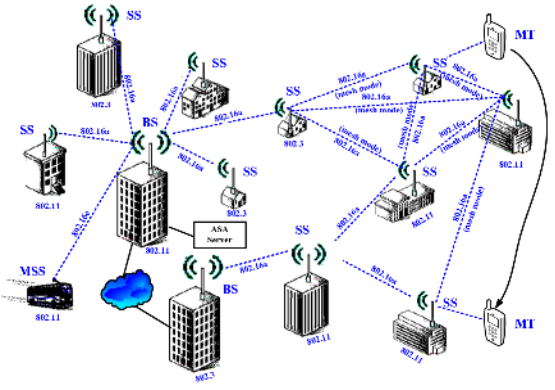


Fig. 1. The architecture of WiMAX

Recently, a number of handoff rerouting strategies have been proposed in literature in general [3-8]. These schemes can be classified as four categories: Full Connection Rerouting (FCR), Route Augmentation (RA), Partial Connection Rerouting (PCR), and Multicast Connection Rerouting (MCR) [3]. FCR maintains the connection by establishing a completely new route for each handoff just as initiating a new call [4]. RA will relay the route from current SS to destination SS and add a hop to the original connection [5]. PCR reestablishes several segments of the original connection, while preserving the remainder [6,7]. MCR combines the former three protocols but includes the maintenance of potential handoff connection routes to support the original connection, to reduce the latency of finding a new route for handoff [8]. But MCR requires more communications due to multicast. All these rerouting protocols aim to solve different handoff problems and part of them may incur packet loss during handoff process. In this paper, we propose a rerouting scheme based on *Location Update* backtracking to optimize the reconstructed route and avoid packet loss. According to the moving velocity of MT and the link bandwidth between CN and MT, our scheme dynamically selects the hop number of *Location Update* backtracking and optimizes the reconstructed route.

The rest of the paper is organized as follows. Section 2 introduces the principle of our rerouting scheme. Rerouting and packet forwarding model of the proposed scheme are provided in Section 3. Section 4 analyzes the performance of FCR, PCR, RA, MCR and our scheme. Section 5 concludes the paper and discusses future research.

2 Principle of Backtracking Based Handoff Rerouting

In WiMAX Mesh mode, SSs are connected by direct connection or multi-hop wireless relaying. Common node is the intermediate node which has connections with CN, current SS, and destination SS. An example of common node is given in Fig.2. PCR rebuilds the new route at the nearest common node of current SS and destination SS and preserves the rest routes [6]. Because the nearest common node is the first common node from current SS to CN, the reconstructed route through the nearest common node can not always be the optimal route, especially when destination SS is several hops away from current SS. Our rerouting scheme tries to find out the best common node through which the new route can achieve the lowest route cost in original link backtracking and rebuilds the communication route from this common node in an iterative approach.

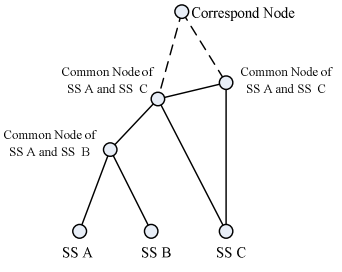


Fig. 2. Example of common node

Due to the limitation of wireless medium, the process of *Location Update* backtracking from current SS to CN may be costly and introduce longer delay, especially when MT has a high velocity or the available network bandwidth is limited. In order to improve the efficiency of handoff procedure, our scheme dynamically selects the backtracking hop of *Location Update* message, k, according to the velocity of MT and the current network bandwidth. During the *Location Update* backtracking process, our scheme finds out the best common node among the k hops link, through which the new route can achieve a optimized route. If MT moves quickly and the available bandwidth is limited, the hop of *Location Update* message backtracking, k, has a small value; otherwise, k is a large number.

Meanwhile, when common nodes received the *Location Update* message, they will forward all packets with their optimal route to destination SS. Hence, when the backtracking process is ended, the *Location Update* message has found out the optimal route and all packets have been forwarded to MT with a low forwarding cost. We assume that the *Location Update* message is consisted of the address of CN, the address of current SS, the address destination SS, the address of the common node with optimal route, the value of optimal route, and the number of backtracking hop.

When MT enters into the common area of current SS and destination SS, it will receive the signal from both of them. MT will compare the Received Signal Strength (RSS) from these two SSs and initiate the handoff process according to some special criterions, such as RSS threshold [9].

From the view point of radio properties, the signal received at MT is consisted of three components: (1) Path loss with respect to distance, which may be predicated by empirical models [10]. (2) Shadow fading or slow fading, which is attributed to shadowing caused by structures and terrain variations. It is characterized by lognormal distribution with standard deviation depending on the general properties of the propagation environment. (3) Fast fading or multi-path fading. Hence, propagation phenomenon in mobile radio is strongly affected by the particular environment surrounding MT. Due to fast fluctuations of these three components, handoff algorithms cannot be designed to respond to fast fading. We assume current SS and destination SS are away from D meters and MT is moving from current SS to destination SS along a straight line with constant speed. When MT separated from current SS with d meters, the RSS of current SS, RSS_{old}, and destination SS, RSS_{new}, can be derived as follow:

$$RSS_{old} = K_1 - K_2 \times log_{10}(d) + u(d)$$

$$RSS_{new} = K_1 - K_2 \times log_{10}(D - d) + v(d)$$

where K_1 and K_2 are the signal strength which depend on the transmitted power, antenna feature on the SS, and the transmission environments. $u(d)$ and $v(d)$ are zero mean stationary Gaussian random processes that model lognormal shadow fading [10].

Let $RSS(t)$ be the RSS_{old} received by MT at time t and $RSS(t + \triangle t)$ be the estimate value in coming $\triangle t$, then the velocity factor of MT, θ ,can be expressed as

$$\theta = \frac{dRSS(t)}{dt} = \frac{\left|RSS(t) - RSS(t + \Delta t)\right|}{\Delta t}$$

Note that the beginning and end of handoff procedure will be decided by MT according to some special criterions of RSS [9]. We assume two thresholds, TH_{up} and TH_{low} to limit the range of RSS. The handoff process will be initiated when the RSS of current SS is lower than TH_{up} and the RSS of destination SS is greater than TH_{low}. Meanwhile, the handoff process will be ended when the RSS of current SS is lower than TH_{low} and the RSS of destination SS is greater than TH_{up}. In order to describer the range of RSS allowing the handoff process to take place, the handoff window is introduced. The handoff process will only happen when the RSS of current SS and destination SS in the range of handoff window, as in Fig 3.

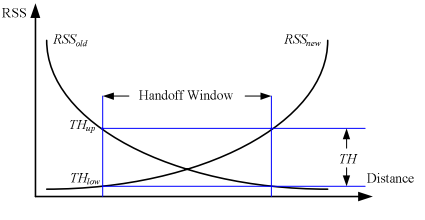


Fig. 3. The relationship of the handoff window and the value of TH_{up} and TH_{low}

Hence, the beginning and end of handoff process follow below rules:

(1) Handoff process begins when

$$RSS_{old} = K_1 - K_2 \times log_{10}(d) + u(d) < TH_{up}$$

$$RSS_{new} = K_1 - K_2 \times log_{10}(D - d) + v(d) > TH_{low}$$

(2) Handoff process ends when

$$RSS_{old} = K_1 - K_2 \times log_{10}(d) + u(d) < TH_{low}$$

$$RSS_{new} = K_1 - K_2 \times log_{10}(D - d) + v(d) > TH_{up}$$

From the above rules, it can be found that the distance, d, that MT separated from current SS plays an important role in the handoff decision making. From Fig 3 we also can find that TH_{up} and TH_{low} are the key factor to decide the range of handoff window. Meanwhile, when the transmitted power of SS is constant, the handoff window's size will also be affected by the distance between the two SSs, D. The far the distance between SSs is, the larger the size of handoff window is.

When the size of handoff window has been taken down, the velocity of MT will decide the time slot in which the handoff process will happen. At the same time, it can be easy understood that the faster MT moves, the little time that it takes MT to go through the entire handoff window. If we denote the time slot that MT goes through the handoff window as T_{cover}, it can be found that T_{cover} is a decreasing function of the velocity factor of MT, θ. Meanwhile T_{cover} is also an increasing function of the distance of current SS and destination SS, D. Hence, the handoff process can only take place during the period of [0, T_{cover}].

$$T_{cover} = \frac{TH_{up} - TH_{low}}{\theta} = \Delta t \times \frac{TH_{up} - TH_{low}}{\left| RSS(t) - RSS(t + \Delta t) \right|}$$

Our rerouting scheme computes the hop number of *Location Update* backtracking according to T_{cover} and the link bandwidth between CN and current SS. The best common node selects procedure as follow:

Algorithm. Best common node selection in k-hop Range.

Input: node set $X = \{X_0, X_1, \ldots, X_i, \ldots, X_n\}$, $i \in [0, n]$ in the link between Current SS and CN. X_0 is for Current SS and X_n is for CN; The time queue of *Location Update* transmitting between two adjacent nodes, $T = \{T_{0,1}, T_{1,2}, \ldots, T_{k-1,k}, \ldots, T_{n-1,n}\}$. // $T_{i,j}$ is the time slot during which *Location Update* transmits from node X_i to node X_j

Output: best common node

1. Initiate $SumT = 0$, $i = 0$ //i is the node flag of intermediate node;

---

2. While $SumT <= \alpha \times T_{cover} = \alpha \times \Delta t \times \frac{TH_{up} - TH_{low}}{|RSS(t) - RSS(t+\Delta t)|}$ {

3. $j = i+1$; $SumT = SumT + T_{i,j}$; $i = i+1$;}

4. $k = i+1$; $i = 0$;

5. While $k \geq 0$ do {

6. Compute the route cost from node X_n to X_0 through X_i

7. If the route cost is lower than the cost stored in the *Location Update* then replace the cost with the new cost and update the best common node address;

8. Node X_i begins forwarding all coming packets to MT with its optimal route to destination SS;

9. $i = i + 1$; $k = k-1$ }

where α is the factor of network bandwidth, it can adjust itself according to the current network situation. When the link bandwidth is enough,αcan be a greater value, our rerouting scheme can achieve a larger value of k. Otherwise, the value of k will be limited to a little value.

3 Analytical Model

Assume a network topology as $G = [X, V]$. $X_p = \{X_0, X_1, X_2, \ldots, X_n\}$, $X_p \subset X$, is the set of n+1 nodes in the link between current SS and CN. $V = \{V_{i,j}, i,j \in X\}$ is the link set which connect node of X. If $V_{i,j} > 0$, we think there is a bi-direction connection between node X_i and node X_j with bandwidth $V_{i,j}$. Meanwhile, the transmitting time of *Location Update* between two adjacent nodes forms a time queue $T = \{T_{0,1}, T_{1,2},\ldots, T_{k-1,k},\ldots, T_{n-1,n}\}$. $T_{i,j}$ is the time that *Location Update* message transmits from node X_i to node X_j will take, which includes the transmitting time and the waiting time in the queue of common node. Assume there is an optimal route from node X_i to destination SS and its route cost is F_i. The route cost between CN and node X_i is O_i. Let $M(X_k, T_{i,j})$ be the number of stored packet at node X_k after time $T_{i,j}$. The packet that node X_k received and sent during time slot $T_{i,j}$ are $P_r(X_k, T_{i,j})$ and $P_s(X_k, T_{i,j})$ respectively.

A matrix with size 6×6 is given in Fig 4 to introduce the principle of our scheme. In this figure, we use X_0 and X_{new} to denote current SS and destination SS respectively. CN sends packets to current SS along route $\{X_n, X_{n-1},\ldots, X_2, X_1, X_0\}$. According to the principle described in last section, it can be found that the *Location Update* message has been backtracked to node X_k after time slot $T_{k-1,k}$. Node X_k and node X_{k-1} must

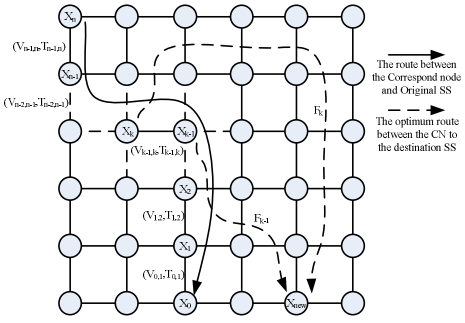


Fig. 4. Example of backtracking and packet forwarding principle

forward all packets stored or received during time slot $T_{k-1,k}$ to destination SS. The number of packet received by node X_{k-1} after time $T_{k-1,k}$ can be expressed as

$$P_r(X_{k-1}, T_{k-1,k}) = \min\{\ T_{k-1,k} \times V_{k-1,k},\ \ M(X_k, T_{k-2,k-1}) + T_{k-1,k} \times V_{k,k+1}\ \} = P_s(X_k, T_{k-1,k})$$

$$M(X_{k-1}, T_{k-1,k}) = P_r(X_{k-1}, T_{k-1,k})$$

The number of packet received and sent by node X_k during time $T_{k-1,k}$ is given as

$$P_s(X_k, T_{k-1,k}) = \min\{\ T_{k-1,k} \times V_{k-1,k},\ \ M(X_k, T_{k-2,k-1}) + T_{k-1,k} \times V_{k,k+1}\ \}$$

$$P_r(X_k, T_{k-1,k}) = \min\{\ T_{k-1,k} \times V_{k,k+1},\ \ M(X_{k+1}, T_{k-2,k-1}) + T_{k-1,k} \times V_{k+1,k+2}\ \} = P_s(X_{k+1}, T_{k-1,k})$$

$$M(X_k, T_{k-1,k}) = M(X_k, T_{k-2,k-1}) + P_r(X_k, T_{k-1,k}) - P_s(X_k, T_{k-1,k})$$

Hence, the cost of packet forwarding, $F(T_{k-1,k})$, and the number of packets forwarded, $S(T_{k-1,k})$, of our rerouting scheme during time slot $T_{k-1,k}$ are

$$F(T_{k-1,k}) = M(X_{k-1}, T_{k-1,k}) \times F_{k-1} + M(X_k, T_{k-1,k}) \times F_k$$

$$S(T_{k-1,k}) = M(X_{k-1}, T_{k-1,k}) + M(X_k, T_{k-1,k})$$

where $M(X_0, 0) = 0$

$$P_r(X_{n-1}, T_{n-1,n}) = T_{n-1,n} \times V_{n-1,n} = P_s(X_n, T_{n-1,n})$$

Our rerouting scheme requires the *Location Update* message be backtracked k hops during the handoff process. Hence, the number of packet forwarding, S, and the cost of packet forwarding, F, can be derived as follow

$$S = \sum_{i=0, j=i+1}^{k} S(T_{i,j}) = \sum_{i=0, j=i+1}^{k} \left[M(X_i, T_{i,j}) + M(X_j + T_{i,j}) \right]$$

$$F = \sum_{i=0, j=i+1}^{k} F(T_{i,j}) = \sum_{i=0, j=i+1}^{k} \left[M(X_i, T_{i,j}) \times F_i + M(X_j + T_{i,j}) \times F_j \right]$$

During the process of *Location Update* message backtracking, the optimal route and the best common node address will be updated continuous. At the end of handoff

process, the best common node and the optimized route will be found. The reconstructed route cost, R, is

$$R = \min\{O_k + F_k, O_{k-1} + F_{k-1}, \ldots, O_1 + F_1, O_0 + F_0\} = \min_{i \in [0,k]}(O_i + F_i)$$

When the value of k has a larger value, the *Location Update* message will be backtracked to CN or the common node near CN, our rerouting scheme can find out a new route just like the FCR does. At the same time, all packets will be forwarded to the destination SS with a small forwarding cost. Meanwhile, when the value of k has a small value, our rerouting scheme will reconstruct the route on the common node near current SS or on current SS, which will reduce the impaction brought by handoff process and incur less handoff delay. In this situation, our scheme rebuilds the route like a combination of PCR and RA does.

4 Performance Evaluation

In this section, some simulation results are present. A matrix with size 20×20 is employed to simulate Mesh mode topology. Each SS at most has 4 connections with its neighbor. Route cost between two adjacent SSs is generated by a Poisson process with rate 0.5. If the route cost between two SSs equals to 0, we think there is no connection between them. The time slot of transmitting *Location Update* message between two adjacent SSs is an exponential process with rate 0.3. MT follows a random moving model in which MT moves to north, south, east, west, southeast, northeast, southwest, and northwest with equal probability of 1/8. The packet stored in each SS is assigned with a value chosen in the range of [0, 2000] with uniform distribution.

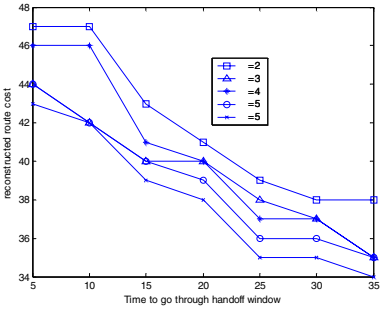


Fig. 5. Time and cost for the new route

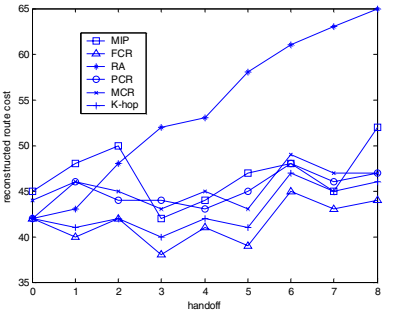


Fig. 6. The cost of new constructed routes

Fig. 5 presents the new route cost while MT is moving straightly toward destination SS with different values of T_{cover} and network bandwidths. We can see that the new route cost decrease little by little with the increment of T_{cover}. The greater the value of T_{cover} is, the more hops the *Location Update* message will be backtracked and the better route that our rerouting scheme can find out. Meanwhile, when α has a large value, which mean that the link bandwidth between CN and current SS is enough, the

backtracking hop of *Location Update* message also be assigned with a greater value. It can be found that the more bandwidth of the link is, the better the new route is.

During the handoff process, it can be found that there are different new constructed route costs for CN with different handoff rerouting schemes. Fig. 6 shows the new constructed route cost for Mobile IP [11], FCR, RA, PCA, MCR, and *k*-hop. FCR can find out the best new route for CN because it will reconstruct a new route for the ongoing application at each handoff process. RA simply extends the original connection via a hop to MT's next location. The more the handoff is, the more the new route cost reconstructed by RA is. The rerouting scheme proposed in this paper can find out a new route for CN and the new route cost is between FCR and PCR.

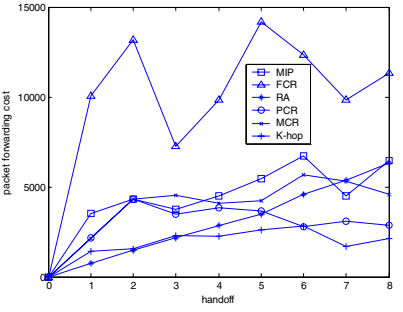


Fig. 7. Packet forwarding cost

Fig. 7 presents the packet forwarding cost of different rerouting schemes. It can be seen that the packet forwarding cost of FCR is far greater than that of the other rerouting schemes. FCR will forward all packets lost in the handoff process from CN to MT. These packets forwarded by the FCR will go through the entire new route, which make the packet forwarding cost rise greatly. At the same time, we can see from Fig. 6 that the new route rebuilt by RA from the first current SS to the last destination SS increase with the increment of handoff process. So, the packet forwarding cost of RA will increase as the handoff increase. Our rerouting scheme make the common node received *Location Update* message forward all packets received and stored during the handoff course with its optimal route to destination SS. It can be found that if MT has experienced less handoff, the packet forwarding cost of our scheme is greater than that of RA. However, the packet forwarding cost of our scheme will be lower than RA with the increase of handoff. During the simulation, we can find that our rerouting scheme can achieve a lower packet forwarding cost than PCR.

5 Conclusion

This paper proposed a rerouting scheme based on *k* hop backtracking of *Location Update* message. The number of backtracking hops, *k*, will be selected dynamically according to the velocity of MT and the link bandwidth between CN and current SS. During the backtracking course, our rerouting scheme finds out an optimized route which leads CN to destination SS through the best common node among *k* hop

backtracking range. Meanwhile, when common node has received the *Location Update* message, it will forward all packets stored and received during the handoff process to destination SS with its optimal route to destination SS, which reduce the packet forwarding cost greatly. The performance data show that our scheme can construct a better new route for MT than PCR and the packet forwarding cost are also reduced.

Reconstructing a new route for the application running on MT has become a key issue in handoff research field and IETF has organized a new group, MIPv6 Signaling and Handoff Optimization (mipshop) Working Group, to study this issue. With the development of wireless technology, the importance of handoff rerouting research will become significant.

Acknowledgment

This work is supported by Strategy Grant of City University of Hong Kong under nos 7001709 and 7001777 and partially supported by the National Grand Fundamental Research 973 Program of China under Grant No. 2003CB317003.

References

1. IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems.
2. Alexe E., Brian L.Mark. Modeling and Analysis of Fast Handoff Algorithms for Microcellular Networks. IEEE/ACM MASCOTS'2002.
3. Ian F.Akyildiz,Janise Mcnair,Joseph S.M.HO, Mobility Management in Next-Generation Wireless Systems, Vol. 87, No. 8, Proceedings of the IEEE (1999) 1347–1384.
4. Wenchao M, Yuguang F, Dynamic Hierarchical Mobility Management Strategy for Mobile IP Networks, Vol. 22, No. 4, IEEE Journal on Select Areas Commun (2004) 664–676.
5. B. Akyol, D. Cox, Rerouting for handoff in a wireless ATM network, Vol. 10, No. 3, IEEE Personal Com (1996) 26–33.
6. T. CK. A hybrid handover protocol for local area wireless ATM networks. Vol. 1, No.3, ACM-Baltzer J. Mobile Networks Applicat. (MONET) (1996) 313–334.
7. M. Veeraraghavan, M. Karol, K. Eng. Mobility and connection management in a wireless ATM LAN. Vol. 15, IEEE J. Select. Areas Commun (1997) 50–68.
8. A.S.Acampora, M.Naghshineh. An architecture and methodology for mobile-executed handoff in cellular ATM networks. Vol. 12, No. 1, IEEE J. Select. Areas Commun (1994) 1365–1375.
9. Ye Min-hua, Liu Yu, Zhang Hui-min. THE MOBILE IP HANDOFF BETWEEN HYBRID NETWORKS, IEEE PIMRC (2002).
10. Ian F.Akyildiz,Janise Mcnair,Joseph S.M.HO, Mobility Management in Next-Generation Wireless Systems, Vol. 87, No. 8, Proceedings of the IEEE (1999) 1347–1384.
11. C. E. Perkins, IP mobility support, Request for Comments (RFC) 2002-2006.

A Self-tuning Reliable Dynamic Scheme for Multicast Flow Control

Naixue Xiong[1,2], Yanxiang He[1], Laurence T. Yang[3], and Yan Yang[2]

[1] The State Key Lab of Software Engineering, Computer School, Wuhan University
[2] School of Information Science, Japan Advanced Institute of Science and Technology
[3] Department of Computer Science, St. Francis Xavier University, Canada
{n.xiong, yxhe}@whu.edu.cn, lyang@stfx.ca, y.yang@jaist.ac.jp

Abstract. This paper describes a novel control-theoretic distributed multicast congestion control scheme, called self-tuning proportional integrative plus derivative (SPID) controller. The control parameters can be designed to ensure the stability of the control loop in terms of source rate. The distributed explicit rate SPID overcomes the vulnerability that suffers from the heterogeneous multicast receivers. The SPID controller is located at the multicast source to regulate the transmission rate. Simulation results demonstrate the efficiency of the proposed scheme in terms of system stability and fast response, low packet loss, and high scalability.

1 Introduction

With ever-increasing multicast data applications recently, multicast (multipoint-to-multipoint) transmission now have considerable interests on many application services. Multicast improves the efficiency of multipoint data distribution [1,5,9,11]. Unfortunately, the widely used multicast transport protocols, which are layered on top of IP multicast, can cause congestion or even congestion collapse if they do not provide adequate congestion control. Congestion control thus plays an important role in the traffic management of multicast communications.

There are many congestion schemes handling unicast transmissions efficiently [6, 10], and they are formulated as a discrete-time feedback control problem with delays. All these methods in [7-8] are efficient in rate allocation and congestion control to unicast transmission. Unfortunately, multicast congestion control is much more sophisticated than that of unicast due to the complexity of multicasting mechanism. The papers [2] adopt a simple hop-by-hop feedback mechanism. The main merit of these methods lies in the simplicity of the hop-by-hop mechanism; however at the same time, they often lead to the so-called consolidation noise problem [3] due to incomplete feedback information. To overcome this drawback, the papers [4] proposed a method called feedback synchronization at each branch point by accumulating feedback from all downstream branches. These schemes then introduce another problem of slow transient response due to the feedback from long path. Such delayed congestion feedback can cause excessive queue build-up and packet loss at the bottleneck link.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 351–360, 2006.

In this paper, we develop a distributed ER allocation algorithm to overcome the vulnerability that suffers from the heterogeneous multicast receivers. In our scheme, congestion controllers regulate the source rate of a multicast tree, which accounts for the buffer occupancies of all destination nodes. The proposed control scheme uses a distributed Self-tuning Proportional Integrative plus Derivative (SPID) controller, where the control parameters can be designed to ensure the stability of the control loop in terms of source rate. System stability criterion is derived in presence of destination nodes with heterogeneous RTTs. Finally, we evaluate the performance of the proposed scheme by simulation.

2 The Network Configuration Model

To conveniently analyze the performance and characteristic of the proposed multicast scheme, we focus on the following network model [9] (Fig. 1). The multicast network is a connection-oriented one, which is composed of some sources and destination nodes, and time is slotted with the duration $[n, n+1]$ by the sampling period T. The associated data is transferred by a fixed size packet.

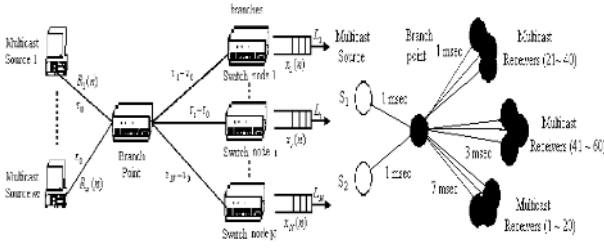


Fig. 1. A multicast model

Fig. 2. The multicast simulation model 1

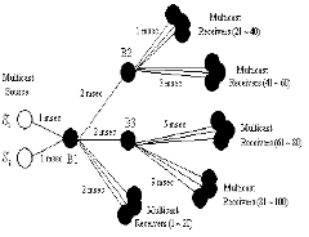


Fig. 3. The multicast system model 2

In each multicast connection and every sampling period, the multicast source issues and transmits forward control packet (FCP) to the downstream nodes (the branch node and destination nodes), and the backward control packet (BCP) is constructed by the downstream nodes and sent back to the source. After the multicast source receives the BCPs from the downstream nodes, they will take appropriate action to adjust their transmitting rates of multicast traffic based on the computed value of the SPID controller. After receiving the data packets coming from the network, the receivers construct the BCPs and send them back to the network.

The considered multicast service is described as follows: The packet number sending out by the switch node i in one interval T is denoted by L_i, the switch node i has the forward delay $\tau_i (1 \leq i \leq N)$, from sources, and the τ_0 is the delay from sources to the neighbor downstream node. Then, the round-trip delay (RTD) for the switch node i is $\tau_{Ri} = 2\tau_i$ and $\tau = max\{\tau_{R1}, \tau_{R2}, ..., \tau_{RN}\}$. We

further assume that τ_i and τ_{Ri} are integers, which are reasonable by adjusting T. The link delay is dominant compared to the other delays such as proceeding delay, queuing delay, etc. In the model, we assume $\tau_i \leq \tau_j$ when $1 \leq i \leq j \leq N$. Each router schedules the packets in a first-come-first-served way.The component $R_i(n)$ represents the receiving rate of the computed receivers i at time slot n.

Based on the above consideration, the buffer occupancy of the switch node i is determined by [5, 9]: $x_i(n+1) = Sat_{K_i}\{x_i(n) + \sum_{q_1}^{m} e_q R_q(n-\tau_i) - L_i\}$, where K_i is the buffer size, $x_i(n)$ is the buffer occupancy of the switch node i at time slot n, and $R_q(n-\tau_i)$ is the sending rate of the q^{th} $(1 \leq q \leq m)$ source to the switch node $i, (1 \leq i \leq N)$.

$$Sat_{K_i}\{x_i\} = \begin{cases} K_i, x_i > K_i; \\ x_i, 0 \leq x_i \leq K_i; \\ 0, x_i < 0. \end{cases} \qquad e_q = \begin{cases} 1; & if\ the\ q^{th}\ source\ is\ active; \\ 0; & if\ the\ q^{th}\ source\ is\ not\ active. \end{cases}$$

After lifting the saturation restriction [10], equation (1) can be written into

$$x_i(n+1) = x_i(n) + \sum_{q=1}^{m} e_q R_q(n-\tau_i) - L_i. \tag{2}$$

3 The Specific Algorithm

The router buffer occupancy is expected to stabilize at the neighborhood of the desired level. If $x(n)$ is too high, it often leads buffer overflowing and packet loss. In addition, under this circumstance long queuing delay usually results in time out and retransmission, which in turn build up the mounting of the buffer occupancy; consequently a vicious circle is formed. If $x(n)$ is too low, it increases the likelihood of link under-utilization during occasionally idle period. Thus the router buffer occupancy plays an important role in the congestion control that is chosen out to be the feedback carried in BCP. Generally, among all downstream nodes, the most congested one, defined as the worst node deserves special attention. Based on this consideration, we propose the following control scheme,

$$\begin{aligned} R_q(n) = \mu + a \cdot \sum_{i=1}^{N} (x_i(n-\tau_i) - \bar{x}_i) + \sum_{j=1}^{\tau} b_j R_q(n-j) \\ + c \cdot \textstyle\sum_{k=1}^{N} (x_k(n-\tau_k) - x_k(n-\tau_k-1), \end{aligned} \tag{3}$$

where $a, b_j, (j = 1, 2, ..., \tau)$ and c are the proportional, integral and derivative control gains respectively, which are to be determined from the stability criteria. These coefficients are used to locate all the poles of the closed-loop equations (2) and (3) within the unit circle to ensure stability. The component $\bar{x}_i$ is the target queue length, and μ is the maximum sending rate of sources. In (3) it is seen that, if the buffer occupancy of the switch node i is measured at the instances $n - \tau_i$, after the feedback delay τ_i the BCP reaches the controller located at the source $q, (q = 1, 2, ..., m)$, and the router then takes out the buffer occupancy of

the destination nodes at time $t = n$. By doing so, the designed controller can be expected to have flexibility to cope with the sharp oscillation in buffer occupancy that could cause the network to lose packets. In addition, the calculation in (3) is completely independent of virtual connections traveling through the multicast session. This means the scheme has scalability.

The branch point of the multicast tree replicates each data packet and FCP from its upstream node to all its downstream branches. The downstream nodes return their congestion information via BCPs to the parents through the backward direction of the coming path once they receive FCPs. Moreover, the branch nodes consolidate the BCPs that carry all the available rates and the relevant link bandwidth from different branches into one BCP and feedback the new BCP to their upstream node. Rate adaptors associated are located at multicast source. There is a single FIFO (first-in-first-out) queue to multiplex all flows traveling through the outgoing link. Assume that congestion never happens at the router connected with the sources, hence these two can be consolidated into one node, which is true in most cases in real networks.

Before we present the algorithm in details, we specify the following variables. The variable $multicasttree[i] = 1(0)$ means the i^{th} branch point receive (dont receive) FCP or BCP control packet ; while $receivertree[j] = 1(0)$ means the j^{th} branch point receive (dont receive) confirmations of all destination nodes. Based on the above specifications, the pseudo-code of the proposed router and source algorithms in congestion control model of multiple points to multiple points in a multicast network is as follows. **Source algorithm**: (1)Upon every T epoch (say, time k): Transmit data including FCP; (2)Upon receipt of a consolidation BCP from it's downstream: Compute the sending rate based on consolidation BCP using SPID controller; Adjust the transmitting rates based on computed sending rate. **Router algorithm**: If *multicasttree*[i]= =1 if the packet is an FCP Put the data packet in the buffer; Copy the data including FCP; Multicast them to the downstream nodes; else Construct the BCP based on the received BCPs ; Feedback it to the upstream node; if *receivedtree*[j]==1 Delete the data packets from the buffer; else Maintain the data packets in the buffer until receive all confirmations of the receivers; **Destination node algorithm**:Upon receipt of an FCP (1)Put the data packets into the buffer; (2)Construct the BCP based on the current congestion condition of the receiver nodes; (3)Feedback the BCP to the upstream branch point.

4 System Stability Analysis

The ability of a multicast tree to provide efficient heterogeneous distributed communication that can handle multiple qualities of service guarantees can only be realized by effective traffic management schemes.

In this paper, a rate-based scheme is used to achieve congestion control in MR-MCC tree. The controller parameters are designed to guarantee the stability of rate, which ensures a smooth dynamic of rate adaptation to minimize the packet loss rate.

In this section, the stability of the proposed SPI congestion control scheme is analyzed as follows. Considering the equation (2), if z-transformation is applied, one can easily arrive at

$$(z-1)X_i(z) = \sum_{q=1}^{m} e_q R_q(z) z^{-\tau_i} - L_i D(z), \tag{4}$$

where the z-transformation of $x_i(n)$, $R_j(n)$) are respectively described by $X_i(z) = \sum_{n=0}^{+\infty} x_i(n) z^{-n}$, $R_q(z) = \sum_{n=0}^{+\infty} R_q(n) z^{-n}$ and $D(z) = \sum_{n=0}^{+\infty} z^{-n} = \frac{z}{z-1}$.

Taking the z-transform of equation (3), one yields:

$$\begin{aligned} R_q(z) = \mu D(z) + a \cdot \sum_{i=1}^{M} [z^{-\tau_i} X_i(z) - \bar{x_i} D(z)] + \sum_{j=1}^{\tau} b_j z^{-j} R_q(z) \\ + c \sum_{k=1}^{N} (z^{-\tau_i} X_i(z) - z^{-\tau_i - 1} X_i(z)). \end{aligned} \tag{5}$$

From equation (4) and (5), one has $\Delta z R_q(z) = a \sum_{i=1}^{N} (-L_i D(z) z^{-\tau_{Ri}/2} - \bar{x_i} D(z)$ $(z-1)) - c \sum_{k=1}^{N} z^{-\tau_{Ri}/2} L_i D(z)(1-z^{-1}) + \mu D(z)(z-1)$, where we have denoted,

$$\Delta z = (1 - \sum_{j=1}^{\tau} b_j z^{-j})(z-1) - a \sum_{i=1}^{N} z^{-\tau_{Ri}} - c \sum_{k=1}^{N} (z^{-\tau_{R_k}} - z^{-\tau_{Rk}-1}). \tag{6}$$

The coefficients $a, b_j, (j = 1, 2, ..., \tau)$ and c are determined by the stability criteria of the control theory.

The component Δz is the *Characteristic Polynomial* (CP) of the multicast-system given by equations (2) and (3) [10]. The CP (6) is closely related to the stability of the congestion-controlled network system. From a control-theoretic view when all the zeros of (6) lie within the unit disc, the original network system (2) with the controller (3) is stable in terms of the sources sending rate. Stability is a prerequisite in congestion control to ensure that the network has no sending rate oscillation and thus minimize the packet loss rate.

There are the following computation procedures that aid us in determining if all the roots of a characteristic polynomial lie within a unit disc.

For our purpose, we use Schur-Cohn stability test here. Generally, for a polynomial $A_Q(z) = 1 + \sum_{n=1}^{Q} a_n^{(Q)} z^{-n}$, we need to know if all its zeros lie in the unit disc. The following algorithm is proposed for this purpose.

Algorithm 1. Schur-Cohn stability test

Step 1: Start with the original polynomial of degree Q. (Q coefficients);
Step 2: Generate a sequence of polynomials recursively. $A_i(z), i = Q : -1 : 0$, according to $A_{i-1}(z) = (A_i(z) - q_i z^{-i} A_i(z^{-1}))/(1 - q_i^2)$, where $q_i = a_i^{(i)}$. Note that $z^{-i} A_i(z^{-1})$ is a flipped version of $A_i(z)$.
Step 3: The zeros of the polynomial $A_Q(z)$ are inside the unit circle iff $|q_i| < 1, i = Q : -1 : 1$.

With regard to CP (6), some manipulations are needed for it to satisfy the form of the polynomial $A_Q(z)$. This is done in the following manner: $\Delta(z) = z[1-$

$(1+b_1)z^{-1}+(b_1-b_2)z^{-2}+...+(b_{\tau-1}-b_\tau)z^{-\tau}+(b_\tau-a-c)z^{-\tau-1}+cz^{-\tau-2}]$ where we denote $\Delta(z)=1-(1+b_1)z^{-1}+(b_1-b_2)z^{-2}+...+(b_{\tau-1}-b_\tau)z^{-\tau}+(b_\tau-a-c)z^{-\tau-1}+cz^{-\tau-2}$.

It is sufficient to ensure the stability if all the roots of $\Delta_1(z)$ lie inside the unit circle for $\Delta(z)$ has all roots of $\Delta_1(z)$ together with a root $z=0$, which is obvious inside the unit circle. Thus, Schur-Cohn criterion can be applied to $\Delta_1(z)$ directly with $Q=\tau+2$.

Without loss of generality, we group those nodes into one class, which have a small variation of time delays and sending rates. Thus we divide N destination nodes into M groups based on the RTTs, and in each group, the RTT is assumed to be equal, i.e., $\{\tau_{R1},\tau_{R2},\tau_{R3},...,\tau_{RN}\}=\{\tau_1,...\tau_1,\tau_2,...\tau_2,....,\tau_M,...,\tau_M\}$ and we set n_i is the number of the RTT $t_i(i=1,2,...,M\ )$ corresponding to the i^{th} group receivers, then $N=\sum_{i=1}^{q}n_i$.

And n_i is positive integers. So the CP is: $\triangle z=z[1-(b_1+1)z^{-1}+...+(b_{\tau_{R1-1}}-b_{\tau_{R1}})+(b_{\tau_{R1}}-b_{\tau_{(R1)+1}}-an_1-cn_1)z^{-(\tau_{R1}-1)}+(b_{(\tau_{R1+1})}-b_{(\tau_{R1})+2}+cn_1)z^{-(\tau_{R1}-2)}+...+(b_{(\tau_{Rn_i}-1)}-b_{\tau_{Rn_i}})z^{-\tau_{Rn_i}}+(b_{\tau_{Rn_i}}-b_{\tau_{(Rn_i)+1}}-an_i-cn_i)z^{-(\tau_{Rn_i})-1}+(b_{\tau_{(Rn_i)+1}}-b_{\tau_{(Rn_i)+2}}+cn_i)z^{-\tau_{(Rn_i)}-2}+...+(b_{\tau-1}-b_\tau)z^{-\tau}+(b_\tau-an_M-cn_m)z^{-\tau-1}+cn_Mz^{-\tau-2}]$

let $1+b_1=b_2-b_1=...=b_{\tau_{R1}}-b_{\tau_{R1}-1}=an_1+b_{\tau_{R1}+1}+cn_1-b_{\tau_{R1}}=b_{\tau_{R1}+2}-cn_1-b_{\tau_{R1}+1}=...=b_{\tau_{Rn_i}}-b_{(\tau_{Rn_i}-1)}=b_{(\tau_{Rn_i}+1)}-b_{\tau_{Rn_i}}+an_i+cn_i=b_{\tau_{Rn_i}+2}-b_{\tau_{Rn_i}+1}-cn_i=...=b_\tau-b_{\tau-1}=\varepsilon$, and $an_M+cn_M-b_\tau=\varepsilon, i=1,2,3,...,(M-1)$. We set $a=(\tau\varepsilon+2\varepsilon-1)/N, c=(-\varepsilon)/n_M$, and

$$b_j=\begin{cases} j\varepsilon-1;\ (j=1,2,3,...,\hat{o}_{R1}) \\ j\varepsilon-1-a(n_1+n_2+...+n_i)-cn_i;(j=\hat{o}_{(R1)}+1) \\ j\varepsilon-1-a(n_1+n_2+...+n_i);(j=\hat{o}_{R1}+2) \\ j\varepsilon-1-a(n_1+n_2+...+n_{i+1});(i=2,3,...,M; \\ j=\hat{o}_{Rn_{(i-1)}}+3,...,\tau_{Rn_i};and j=\hat{o}_{R1}+1,...,\hat{o}). \end{cases}$$

We can set

$$\Delta(z)=z^{-\tau_{Rj}}[z^{(\tau_{Rj}+1)}-\varepsilon(z^{\tau_{Rj}}+z^{(\tau_{Rj}-1)}+z^{(\tau_{Rj}-2)}+...+z+1)], \quad (7)$$

From [10], when $\epsilon<1/(\tau+2)$, all the zeros of (7) lie within the unit disc, and the original network system (2) with the controller (3) is stable.

5 The Simulation Results

In this section, we evaluate the performance of the proposed scheme by a number of simulations. To evaluate the performance of the studied multicast congestion control method, we focus upon the following two simulation models (Fig. 2-3) based on the mode complex degree and network dynamic behavior, we give two simulation experiments. The simulation is carried out over a wide range patterns and propagation between two different nodes can lie in LAN (Local Area Network) case or the WAN (Wide Area Network) case.

We focus on analyzing the transient behaviors of the network. In the performance analysis, the duration of response time, receiving rate of receivers and steady state of buffer occupancy are the main concerns.

Control method with short response time has following advantages: when the buffer of receiver nodes is close to the threshold, one may notice the sending node to reduce the sending rate and prevents the loss of packets as soon as possible; while when the available bandwidth increases, the sending node as soon as possible increases the sending rate and enhances the utilization rate of the bandwidth.

In the whole simulations, we assume that the link delay is dominant compared to the other delays such as processing delays and queuing delay. Furthermore, we process the nodes together, which have a small extent change of the time delay and sending rate. Then we make the time delay and sending rate to be unified respectively. Since the situation of every node in each group (about 20 receivers) is similar, we only choose one node from each group as a representative.

5.1 Simulation 1

In this simulation, we focus on the simulation model in Fig. 2, and we are mostly interested in analyzing the transient behaviors of the network. The multicast source S1 sends data packets at 0 m sec, and the multicast source S2 start to sends data packets at 1000 m sec in the simulation time, then it enhance the network dynamic behavior, and demonstrate the efficiency of the SPID scheme.

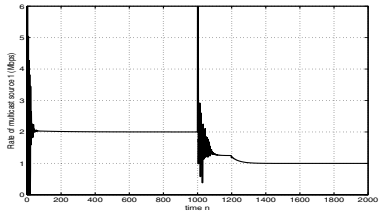


Fig. 4. The sending rate of S1

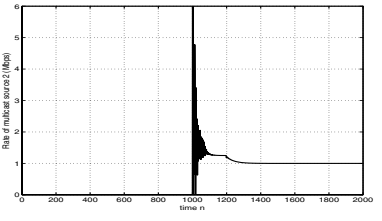


Fig. 5. The sending rate for S2

The relevant parameters for simulation 1 are as follows: for Receiver 1, Receiver 21 and Receiver 41, the target queue length $\bar{x}_i$ are 70 Kb, 80 Kb and 120 Kb respectively; the receiving rate L_i are 2 Mbps, 3 Mbps and 4 Mbps respectively; the round trip time from the source to Receiver 1, Receiver 21 and Receiver 41 are 4 msec, 8 msec and 16 msec respectively. According to the introduced stability test to select the control gains, one computes the relevant parameters a, b, c. For this case, $\tau = 16msec.$ we set ε to be $1/20$, which is stable in the system. When $\varepsilon = 1/20$, $a = -1/600$, $c = -1/400$, and $b = [b_1, b_2, b_3, ..., b_{16}] = [-19/20, -9/10, -17/20, -8/10, -2/3, -2/3, -37/60, -34/60, -26/60, -26/60, -23/60, -20/60, -17/60, -14/60, -11/60, -7/60, -8/60]$. In this section, the simulation results of simulation 1 are shown in the Fig. 4-8. The Fig. 6-8 show

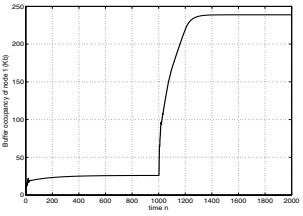


Fig. 6. The buffer occupancy of receiver node 1

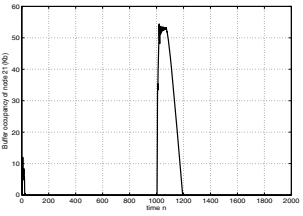


Fig. 7. The buffer occupancy of receiver node 21

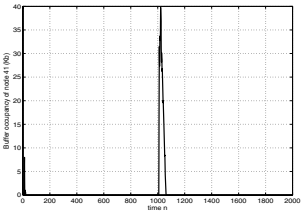


Fig. 8. The buffer occupancy of receiver node 41

the buffer transient response of the receiver node 1, node 21 and node 41 respectively. These buffer occupancies all have some fluctuation in the beginning. Then they gradually become stable. The sending rates in the sources are respectively shown in Fig. 4-5. The initial sending rate of multicast sources S1 is 6 $Mbps$. It can be seen that, although the sending rate of the multicast source S1 has some fluctuation at first. As time goes on, the sending rate is gradually adjusted and quickly stable at the value of 2 $Mbps$ during 74 $msec$. When the multicast source S2 starts to sends data packets at 1000 $msec$, the sending rate of source S1 has some fluctuation to response to the multicast source S2 joining, and quickly stabilizes at the new value of 1 $Mbps$.

These simulation results demonstrate our scheme efficiency, which the source adjusts the sending rate gradually to make the buffer occupancy and rate of sending node quickly become steady.

5.2 Simulation 2

In this simulation, we focus on the simulation model 2 (Fig. 3) and we compare the performances with different parameters, which make system stable and unstable cases. We set two different values: one enables system stable, and the other is unstable in the same system. Thus, we compare and analyze the two cases in same system.

The relevant network parameters and assumptions for model 2 are as follows: for receiver 1, receiver 21,receiver 41,receiver 61 and receiver 81,the round trip times are 6 msec, 8 msec, 12 msec, 16 msec and 24 msec respectively; and the receiving rates are 2 Mbps, 2 Mbps, 3 Mbps, 4Mbps and 5Mbps respectively; and the target queue length are set as 70 Kb, 75 Kb, 80 Kb, 120 Kb and 140 Kb respectively. According to the introduced stability test to select the control gains, one computes the relevant parameters a, b, c.

For this case, $\tau = 24msec$. we separately set ε to be 1/28 and 0.9. The first value is stable in the system, and the next one, $\varepsilon = 0.9$, enables system to be unstable. When $\varepsilon = 1/28$, $a = -2/2800$, $c = -1/560$ and $b = [b_1, b_2, ..., b_{24}]$= $[-27/28, -26/28, -25/28, -24/28, -23/28, -22/28, -196/280, -196/280, -186/280, -176/280, \ -166/280, 156/280, -132/280, -132/280, -112/280, -112/280,$

$-88/280, -88/280, -78/280, -68/280, -58/280, -48/280, -38/280, -28/280.]$
When $\varepsilon = 0.9$, $a = 0.224$, and $b = [b_1, b_2, ..., b_{24}] = [-0.1, 0.8, 1.7, 2.6, 3.5, 4.4, 1.7, 1.7, 2.6, 2.5, 3.4, 4.3, 2.64, 2.64, 3.54, 4.44, 1.76, 1.76, 2.66, 3.56, 4.46, 5.56, 6.26, 7.16]$.

In this section, as the two sources send data packets at same time, we only choose the source S1 from each group as a representative. When the control gains enable system stable, we can see the short steady response time of the sending source and buffer occupancy conflicting with the high sending rate of multicast source between the sending node and the branch nodes.

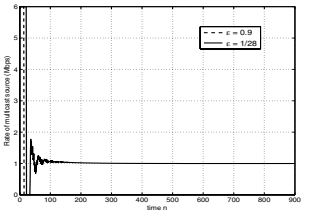


Fig. 9. The sending rate of S1

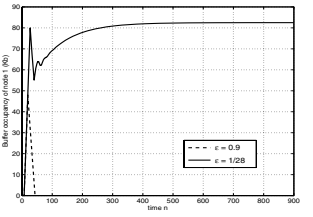


Fig. 10. The buffer occupancy of receiver node 1

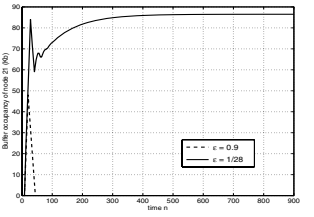


Fig. 11. The buffer occupancy of receiver node 21

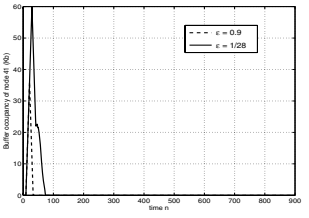


Fig. 12. The buffer occupancy of receiver node 41

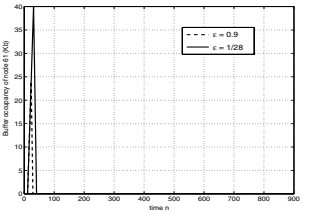


Fig. 13. The buffer occupancy of receiver node 61

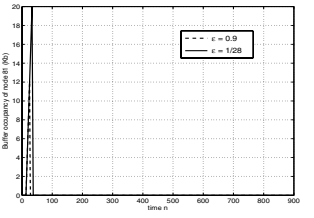


Fig. 14. The buffer occupancy of receiver node 81

The sending rate of multicast source S1 are shown in Fig. 9. After $222msec$, the sending rate in stable system is stable at the value $1Mbps$. Fig. 10-14 show the cases of buffer occupancies in Receivers 1, Receivers 21, Receivers 41, Receivers 61 and Receivers 81. In stable case, the buffer occupancies in Receivers 1 and Receivers 21 are quick respectively stable at the value $82.5Kb$ and $86.5Kb$ in Fig. 10-11. In the Fig. 12-14, some packets accumulate in the buffer of receivers in the beginning. As time goes on, the controller starts to adjust the transmission rate of the source, and the remaining packets in buffer are cleared. In unstable system, the controller fails to effectively adjust sending rate of sources to make buffer occupancy stable.

The simulation results demonstrate the efficiency of our scheme in terms of the system stability, scalability, high utilization of bottleneck link and fast response. And they also demonstrate the sources efficiently and quickly adjust the

sending rate to make the buffer occupancy of receivers and sending rate of multicast sources become excellently steady. Then the theoretical agreement with the simulations in the case of bottleneck link appearing in a multicast tree is verified. Therefore, the SPID scheme is an effective and efficient scheme.

6 Conclusions

In this paper, we present a distributed congestion control method for multicast communication networks, using an explicit rate feedback mechanism to design a SPID controller for regulating the source rates. The control parameters of the SPID controller can be designed to ensure the stability of the control loop and adjust automatically based on the network load. The simulation results clearly demonstrate the efficiency of our scheme in terms of system stability and fast response of the buffer occupancy, as well as sending rates, low packet loss, and high scalability.

References

1. R-H Gau, Z. J. Haas, B. Krishnamachari, "On Multicast Flow Control for Heterogeneous Receivers," IEEE/ACM Transactions on Networking, Volume 10, Issue 1, pp. 86-101, February 2002.
2. K. Y. Siu and H. Y. Tzeng, On max-min fair congestion control for multicast ABR services in ATM, IEEE Journal on Selected Areas in Communications, vol. 15, pp. 545-556, April, 1997.
3. X. Zhang and K. G. Shin, Statistical analysis of feedback synchronization signaling delay for multicast flow control, in Proceedigs of IEEE INFOCOM, pp. 1152-1161, April 2001.
4. W. Ren, K. Y. Siu, and H. Suzuki, On the performance of congestion control algorithms for multicast ABR service in ATM, in Proceedigs of IEEE ATM Workshop, Aug. 1996.
5. X. Zhang, K. G. Shin, D. Saha and D. D. Kandlur, Scalable Flow Control for Multicast ABR Services in ATM Networks, IEEE/ACM Transactions on Networking, vol.10, no.1, pp 67-85, 2002.
6. A. Kolarov and G. Ramamurthy, A control theoretic approach to the design of an explicit rate controller for ABR service, IEEE/ACM Transactions on Networking, vol.7, pp. 741-753, October 1999.
7. F. Blanchini, R. Lo Cigno and R. Tempo, Robust rate control for integrated services packet networks, IEEE/ACM Transactions on Networking, vol.10, no. 5, pp. 644-652, October, 2002.
8. R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, R. Viswanathan, ERICA Switch Algorithm: A Complete Description, ATM Forum-Tm 96-1172, Auguest 1996.
9. S. H. Lee and J. T. Lim, Multicast ABR service in ATM networks using a fuzzy-logic-based consolidation algorithm, IEE Proceeding Communication, Vol.148, No. 1, pp. 8-13, February 2001.
10. N. Xiong, Y. He, Y. Yang, An Efficient Flow Control Algorithm for Multi-rate Multicast Networks, 2004 IEEE International Workshop on IP Operations and Management (IPOM04), Beijing, China, pp. 69-76, October 113, 2004.
11. L. Rizzo, pgmcc: a TCP-friendly single-rate multicast congestion control scheme, In: Proceedigs of ACM SIGCOMM 2000. Stockholm, pp.17-28, 2000.

Intelligent Wireless Home Network Based on Cooperative DS-UWB System

Jee-Hoon Kim and Hyoung-Kyu Song

uT Communication Research Institute, Sejong University, 143-747 Seoul, Korea
bid4u@sdc.sejong.ac.kr, songhk@sejong.ac.kr

Abstract. Recently, wireless personal area network (WPAN) that is the main part of ubiquitous network has been spotlighted. Even though the space-time code (STC) is a good answer for overcoming the intensive fading of indoor channel, it needs larger size and higher cost for additional antennas. The cooperative scheme is another good solution which gives transmit diversity only with the existing hardware. However, classical cooperative schemes cannot provide full rate. In this paper, a new cooperate diversity scheme for code division multiple access (CDMA) system that provides full rate is presented. Moreover, because cooperative system supplies bad performance when the channel condition between the source and relay is poor, a technique that intelligently selects mode according to it is adopted. We apply the proposed scheme to the direct sequence ultra wideband (DS-UWB) system and evaluate its performance.

1 Introduction

As the concept of ubiquitous network was introduced, the importance of wireless network system was more embossed. Therefore, a number of novel techniques have been researched to improve the capacity, reliability, and efficiency for wireless communication. The space-time code (STC) is one of the most innovative works among them which use multiple antennas to acquire space diversity [1], [2]. But it needs more complex hardware and more expenses to utilize additional antennas. Furthermore, sufficient space between each antenna is required to guarantee the independence of each channel. That is to say, the devices that use multiple antennas should be enlarged. Lately, a cooperative diversity technique which has opened a new chapter in wireless communication was introduced [3]. The technique allows a device that uses single antenna to share the antenna of other device to obtain transmit diversity in a manner of constituting the virtual antenna array. Fig. 1 depicts an example of cooperative communication in wireless home network. In ubiquitous network, each device exchanges information with the other devices and knows the environment. Therefore, the following procedure of cooperative communication can be operated naturally:

Step1. Source searches the best coworker among the other devices.
Step2. Relay receives the signals from the source and executes some processes.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 361–368, 2006.

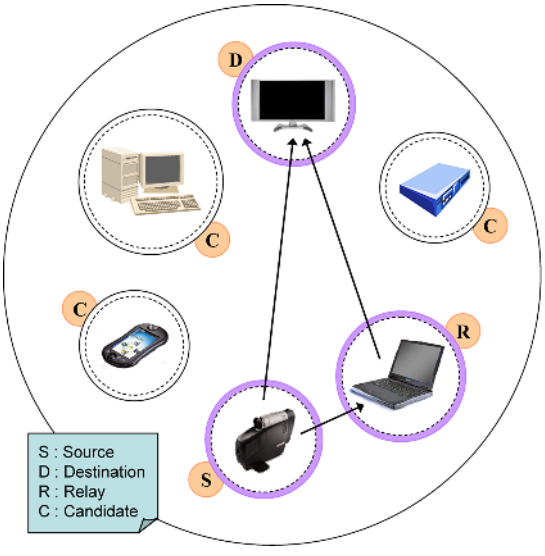


Fig. 1. Illustration of a cooperative communication in wireless home network which consists of a HDTV, a digital camcorder, a laptop, a DVD player, etc.

Step3. Processed data are retransmitted to the destination.
Step4. Destination receives two independently faded signals.

It is noted that if the environment is changed or selected relay is not available for some reason, the source searches a new relay automatically. From the above procedure, we observe that the source utilizes an antenna of the relay as its second antenna. However, the problem of conventional scheme is that the transmission-rate is decreased to separate transmitted signals from mixtures.

In this paper, we propose an enhanced cooperative scheme for code division multiple access (CDMA) system. It is noted that the transmission-rate of the proposed scheme is not decreased. Additionally, a method that intelligently selects whether or not to cooperate according to the channel condition is adopted, since the cooperative diversity technique provides bad performance in case the channel between the source and relay is poor. We apply our scheme into wireless personal area network (WPAN) system which is the main part of ubiquitous network. The direct sequence ultra wideband (DS-UWB) technology which has been considered as one of the leading candidates for high speed WPAN system is introduced to evaluate the proposed scheme.

The rest of the paper is organized as follows. The proposed technique is presented in Section 2. Section 3 gives system description. In Section 4, the performance of proposed technique for DS-UWB system is evaluated. Finally, we make a conclusion and give future works in Section 5.

2 Proposed Cooperative Communication Technique

An overview of three main cooperative communication techniques was presented in [4]. It also provided some practical issues and the limitations of existing algorithms. In this section, we introduce a classical decode-and-forward technique and compare with our new technique which is free from two limitations of classical algorithm.

In the conventional decode-and-forward technique, two devices are joined to help each other. Let X_1 be a signal of device 1 and X_2 be a signal of device 2. They are coded by following signaling block:

$$\begin{aligned} X_1 &= [\ a_{11}s_1^{(1)}c_1,\ a_{12}s_1^{(2)}c_1,\ a_{13}s_1^{(2)}c_1 + a_{14}\hat{s}_2^{(2)}c_2\ ] \\ X_2 &= [\ \underbrace{a_{21}s_2^{(1)}c_2}_{\text{Period 1}},\ \underbrace{a_{22}s_2^{(2)}c_2}_{\text{Period 2}},\ \underbrace{a_{23}s_2^{(2)}c_2 + a_{24}\hat{s}_1^{(2)}c_1}_{\text{Period 3}}\ ] \end{aligned} \tag{1}$$

where a_{ik} is the k-th signal amplitude of i-th device, $s_i^{(j)}$ indicates the j-th signal of i-th device, c_i indicates the spreading code of i-th device, and $\hat{s}$ means estimated signal. We find that each device transmits its own signals in the first and second periods. We also see that both devices transmit a linear combination of their own second signal and the coworker's second signal which are multiplied by their own spreading code in the third period. It is noticed that the signal power varies in accordance to the channel condition. The problems of above block structure are that only the second signals of each signaling block can benefit the diversity gain and transmission-rate is decreased to two-thirds. It alludes that more power are required for both devices to get the same performance as of a system which uses two real transmit antennas.

At this moment, we propose a new scheme which has the features of full cooperative diversity gain and lossless transmission-rate. The main difference between classical and proposed schemes is that proposed scheme uses two spreading codes per each device, while the classical scheme uses one spreading code per each device. In this paper, we only show single cooperative scheme for simplicity. The signaling structure of it is shown as below:

$$\begin{aligned} X_s &= [\ a_s s^{(1)}c_1,\ a_s s^{(2)}c_2,\ a_s s^{(3)}c_1,\ a_s s^{(4)}c_2,\ \cdots\ ] \\ X_r &= [\ \underbrace{0}_{\text{Period 1}},\ \underbrace{a_r\hat{s}^{(1)}c_1}_{\text{Period 2}},\ \underbrace{a_r\hat{s}^{(2)}c_2}_{\text{Period 3}},\ \underbrace{a_r\hat{s}^{(3)}c_1}_{\text{Period 4}},\ \cdots\ ] \end{aligned} \tag{2}$$

where a_s and a_r are the transmit power of source and relay, respectively; $s^{(j)}$ represents the j-th signal of the source, and $\hat{s}$ indicates the estimated source-signal at the relay. Remark that the spreading code which is used for the source is changed with each period. In other words, the spreading code c_1 is used for the odd signals and the spreading code c_2 is used for the even signals. If we assume that the fading of channel is constant for at least one signaling block, the expression of channels is simplified as follows:

$$\begin{aligned} h_{[sd]}(t) &= h_{[sd]}(t+T) = h_{[sd]} = \alpha_{[sd]}e^{j\theta_{[sd]}} \\ h_{[rd]}(t) &= h_{[rd]}(t+T) = h_{[rd]} = \alpha_{[rd]}e^{j\theta_{[rd]}} \\ h_{[sr]}(t) &= h_{[sr]}(t+T) = h_{[sr]} = \alpha_{[sr]}e^{j\theta_{[sr]}} \end{aligned} \tag{3}$$

Table 1. The composition of received signals for the proposed single cooperative scheme

Time \ Part	S-D Channel	R-D Channel	Noise
$r(t)$	$h_{[sd]}s^{(1)}c_1$	$+\ 0\ +$	$n^{(1)}$
$r(t+T)$	$h_{[sd]}s^{(2)}c_2$	$+\ h_{[rd]}\hat{s}^{(1)}c_1\ +$	$n^{(2)}$
$r(t+2T)$	$h_{[sd]}s^{(3)}c_1$	$+\ h_{[rd]}\hat{s}^{(2)}c_2\ +$	$n^{(3)}$
$r(t+3T)$	$h_{[sd]}s^{(4)}c_2$	$+\ h_{[rd]}\hat{s}^{(3)}c_1\ +$	$n^{(4)}$
$\cdots$	$\cdots$	$+\ \cdots\ +$	$\cdots$
$r(t+mT)$	$h_{[sd]}s^{(m+1)}c_x$	$+\ h_{[rd]}\hat{s}^{(m)}c_y\ +$	$n^{(m+1)}$

where $(\cdot)_{[sd]}$ is the component between the source and the destination devices, $(\cdot)_{[rd]}$ is the component between the relay and the destination devices, $(\cdot)_{[sr]}$ is the component between the source and the relay devices, α means the amplitude of channel and θ represents the phase of channel. In that case, the relay receives

$$r_r^{(k)} = h_{[sr]}s^{(k)}c_x + n_r^{(k)} \tag{4}$$

where $r_r^{(k)}$ indicates k-th received signal of the relay and $n_r^{(k)}$ denotes k-th noise factor of the relay. The relay immediately decodes the source signals as shown below.

$$\hat{s}^{(k)} = h_{[sr]}^* c_x r_r^{(k)} = \alpha_{[sr]}^2 s^{(k)} c_x c_x + c_x n_r^{(k)} \tag{5}$$

where $(\cdot)^*$ is the complex conjugate of $(\cdot)$. Since the spreading codes are mutually orthogonal, the correlation of them is calculated by following equation.

$$c_x c_y = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $x = y \in \{1, 2\}$. Therefore (5) is further simplified as

$$\hat{s}^{(k)} = \alpha_{[sr]}^2 s^{(k)} + c_x n_r^{(k)}. \tag{7}$$

After decoding the source signals, the relay multiplies decoded signals by corresponding spreading code and retransmits them to the destination. Accordingly, the destination receives the signals which consist of source symbols, two spreading codes, channel components, and noise.

Table 1 presents the composition of received signals at the destination for the proposed single cooperative scheme in case the amplitudes, a_1 and a_2, are equal to 1 for simplicity. Pay attention to that each two components are grouped by an ellipse. If it is assumed that the estimation of the relay is perfect, $\hat{s}^{(k)}$ becomes $s^{(k)}$. It means

$$\begin{aligned} h_{[sd]}^* c_1 r(t) &= \alpha_{[sd]}^2 s^{(1)} + h_{[sd]}^* c_1 n^{(1)} \\ h_{[rd]}^* c_1 r(t+T) &= \alpha_{[rd]}^2 s^{(1)} + h_{[rd]}^* c_1 n^{(2)}. \end{aligned} \tag{8}$$

The finally estimated version of $s^{(1)}$ which has cooperative diversity gain can be acquired by summing the two components in (8):

$$\begin{aligned}\tilde{s}^{(1)} &= h^*_{[sd]}c_1 r(t) + h^*_{[rd]}c_1 r(t+T) \\ &= (\alpha^2_{[sd]} + \alpha^2_{[rd]})s^{(1)} + h^*_{[sd]}c_1 n^{(1)} + h^*_{[rd]}c_1 n^{(2)}.\end{aligned} \tag{9}$$

Similarly, $s^{(2)}$ is estimated as follows:

$$\begin{aligned}\tilde{s}^{(2)} &= h^*_{[sd]}c_2 r(t+T) + h^*_{[rd]}c_2 r(t+2T) \\ &= (\alpha^2_{[sd]} + \alpha^2_{[rd]})s^{(2)} + h^*_{[sd]}c_2 n^{(2)} + h^*_{[rd]}c_2 n^{(3)}\end{aligned} \tag{10}$$

where

$$\begin{aligned}h^*_{[sd]}c_2 r(t+T) &= \alpha^2_{[sd]}s^{(2)} + h^*_{[sd]}c_2 n^{(2)} \\ h^*_{[rd]}c_2 r(t+2T) &= \alpha^2_{[rd]}s^{(2)} + h^*_{[rd]}c_2 n^{(3)}.\end{aligned} \tag{11}$$

The generalized algorithm to estimate source symbols is given now:

$$\tilde{s}^{(k)} = \begin{cases} h^*_{[sd]}c_1 r(t+(k-1)T) + h^*_{[rd]}c_1 r(t+kT) & \text{if } k \text{ is odd number} \\ h^*_{[sd]}c_2 r(t+(k-1)T) + h^*_{[rd]}c_2 r(t+kT) & \text{if } k \text{ is even number.} \end{cases} \tag{12}$$

On the other hand, comparing (9) and (10) with the equation (13) in [2],

$$\begin{aligned}\tilde{s}_0 &= (\alpha_0^2 + \alpha_1^2)s_0 + h_0^* n_0 + h_1 n_1^* \\ \tilde{s}_1 &= (\alpha_0^2 + \alpha_1^2)s_1 - h_0 n_1^* + h_1^* n_0,\end{aligned} \tag{13}$$

we find that if the relay decodes the signals from the source perfectly, our scheme provides the same performance as two transmit and one receive antennas system that uses space-time block code in [2] (2x1 STBC). It is remarkable that the full cooperative diversity is attained without the loss of code rate and without more power consumption. But, if the relay decodes the source signals incorrectly, the reliability may be diminish. Therefore, we utilize a method that intelligently select whether or not to use cooperative diversity technique according to the condition of channel between the source and relay.

3 System Description

3.1 DS-UWB System

DS-UWB technology can achieve a high data rate of up to 2 Gbps with relatively low complexity. That is agree with the purpose of IEEE 802.153a for providing low complexity, low-cost, low-power consumption, and high data rate wireless connectivity [5]. Therefore, DS-UWB has been considered as one of two weighty candidates for IEEE 802.15.3a standards [6]. There are up to 6 spreading-code sets and 12 piconets where all piconets use different center frequency, respectively. One periodic spreading code is employed per piconet to distinguish them in original DS-UWB system. However, we employ two spreading codes per each

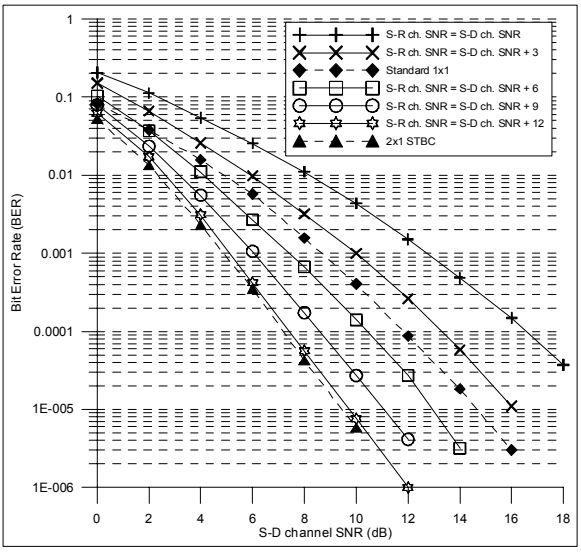


Fig. 2. BER performance comparison according to the S-D channel SNR

piconet to adopt our scheme. The correlation properties of spreading code mitigates interference [7] and provides robust multiple access capability even in the presence of strong narrowband interference [8]. DS-UWB system takes binary phase shift keying (BPSK) and quaternary biorthogonal keying (4BOK) with direct sequence spreading of UWB pulses. It also takes the RAKE receiver which is used for obtaining the path diversity in CDMA system.

3.2 Channel Model

There are three main indoor channel models which were considered for UWB system; the tap-delay line Rayleigh fading model [9], the Saleh-Valenzuela (S-V) model [10], and the Δ-K model described in [11]. We adopt Rayleigh fading WPAN model for the simulation, hence each path is faded according to independent Rayleigh fading statistics. It is assumed that each of multipath experiences a frequency-nonselective slowly fading channel, moreover channel attenuation and phase shift are essentially fixed for the duration of one signaling interval is supposed.

4 Simulation Results and Discussions

In this section, we present the benefit of introduced techniques through the bit error rate (BER) performance. It is assumed that we know each channel perfectly and the transmit power of source and relay are same. For comparison, 2x1 STBC is used. Fig. 2 displays the performance of the proposed single cooperative system in case convolutional code with constraint-length of 4 and code-rate

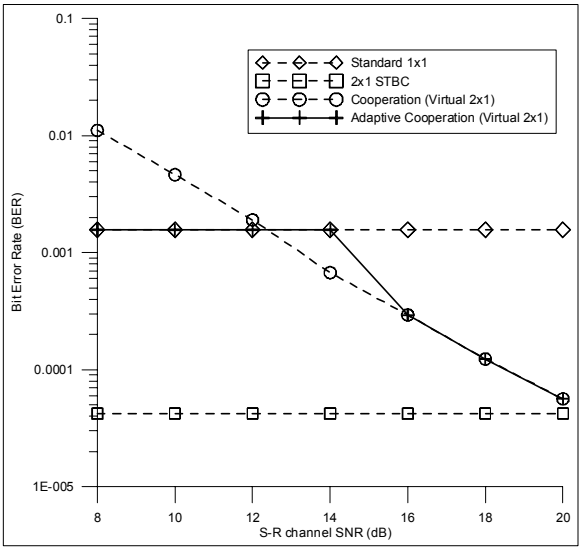


Fig. 3. Comparison of BER performanc e when S-D channel SNR is 8 dB

of 1/2 is used. It is considered that there are 8 paths between the each device and destination; there are 4 paths between the source and relay. As can be seen, the proposed scheme gives better performance than standard one transmit and receive antennas (standard 1x1) system when the channel signal to noise ratio between the source and the relay (S-R channel SNR) is higher by about 5 dB than the channel SNR between the source and destination (S-D channel SNR). Here, we verify that the S-R channel condition is very important for the cooperation. Besides, we also confirm that if the S-R channel is sufficiently good, our cooperation scheme provides the same performance as 2x1 STBC.

Fig. 3 displays the BER performance comparison of several techniques at the S-D channel SNR of 8 dB. It is shown that the cooperation technique gives worse performance when the S-R channel SNR is below about 13 dB. To avoid the worse performance, we introduce an adaptive cooperation technique. We find from the figure that the adaptive technique switches the mode from standard 1x1 to 2x1 STBC at the S-R channel SNR of 14 dB. One may think that more gain can be acquired if it switches the mode at 13 dB. However, in cooperative mode, the relay need to process some algorithm that causes power consumption. In that case, cooperation may bring out the loss for a system oppositely, since the performance gap between cooperative mode and noncooperative mode is small.

5 Conclusion and Future Works

In this paper, we have provided a new technique for single cooperative communication that can give transmit diversity of full rate without additional antennas. A simulation result has shown that the BER performance of cooperative

diversity technique according to the S-R channel condition. Since the cooperative technique gives worse performance in case of bad condition of S-R channel, we have adopted an adaptive cooperation technique. Therefore we could get better performance than standard 1x1 system for any environments.

The proposed scheme has been introduced and has been evaluated only to the personal area network system which uses DS-UWB. However, it can be employed to the other CDMA based systems such as cellular network system with its promising approach for reliability. In addition to that, we have only provided the single cooperative diversity technique. But the mutual cooperative technique which cooperative two device each other at the same time also available.

Acknowledgement

This work is financially supported by the Ministry of Education and Human Resources Development (MOE), the Ministry of Commerce, Industry and Energy (MOCIE) and the Ministry of Labor (MOLAB) through the fostering project of the Lab of Excellency.

References

1. Tarokh V., Seshadri N., and Calderbank A. R.: Space-Time Codes for High Data Rate Wireless Communication: Performance Criterion and Code Construction. IEEE Trans. Inform. Theory, Vol. 44. No. 2, (1998) 744-765
2. Alamouti S.M.: A Simple Transmitter Diversity Scheme for Wireless Communications. IEEE J, Select. Areas Commun., Vol. 16, (1998) 1451-1458
3. Nicholas Laneman J.,: Cooperative Diversity in Wireless Nerwork: Algorithm and Architectures. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, (2002)
4. Aria Nosratinia, Todd E. Hunter, and Ahmadreza Hedayat: Cooperative Communication in Wireless Networks. IEEE Communications Magazine, Vol. 42. (2004) 74-80
5. IEEE 802.15 WPAN High Rate Alternative PHY Task Group 3a (TG3a) [Online].
6. Chul-Seung Lee, Dong-Jun Cho, Young-Hwan You, Hyoung-Kyu Song, : A solution to improvement of DS-UWB system in the wireless home entertainment network. IEEE Transactions on Consumer Electronics, Vol. 51. No. 2, (2005) 529-533
7. Jones R.A., Smith D.H., and Perkins S.: Assignment of Spreading Codes in DS-CDMA UWB Systems. Ultra Wideband Systems and Technologies. IEEE Conference, (2003) 359-363
8. Qinghua Li and Leslie A. Rusch,: Multiuser detection for DS-CDMA UWB in the home environment. IEEE J. Select. Areas. Commun., Vol. 20. No. 9, (2002) 1701-1711
9. IEEE P802.15-00/110r13,"IEEE 802.15.3 (TG3) criteria definitions," Sep. (2000)
10. Saleh A. and Valenzuela R.: A Statistical Model for Indoor Multipath Propagation. IEEE JSAC, Vol. SAC-5. No. 2, (1987) 128-137
11. Hashemi H.: Impulse Response Modeling of Indoor Radio Propagation Channels. IEEE JSAC, Vol. 11. No. 7, (1993) 967-978

A New QoS Multicast Routing Model and Its Immune Optimization Algorithm

J.Q. Wang[1,2], J. Qin[2], and L.S. Kang[1]

[1] The State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, Hubei, China
[2] College of Computer Science, South-Central University for Nationalities, Wuhan 430074, Hubei, China
wjqing2000@yahoo.com.cn, wrj_qj@hotmail.com, kang_whu@yahoo.com

Abstract. The current QoS multicast routing model aims to solve a one-objective optimization problem with one or more bounded-constraints, such as delay, delay jitter, bandwidth, etc. To satisfy the individual requirement for users in multiple QoS networks, we analyze the limitation of the current model and propose a new QoS multicast routing model that supports multi-objective optimization. The new model considers the QoS guarantee as QoS optimization objectives rather than QoS constraints. It overcomes the limitations that exist in the traditional multicast routing model. Furthermore, a new routing algorithm to deal with the new model based on immune principles and Pareto concepts is given. In this algorithm, a gene library is introduced to speed up the algorithm to satisfy the real-time requirement of the routing problem. The initial experimental results have shown that the new algorithm can effectively produce more than one Pareto optimization solution compromising all QoS objectives within one single running.

1 Introduction

Multicast routing is an effective way to communicate among multiple hosts in a network. It outperforms the basic broadcast strategy by sharing resources along common links, while sending messages to a set of predefined destinations. Most multimedia applications, such as radio, TV, on-demand video and teleconferencing, have several important quality-of-service (QoS) parameters such as delay, delay jitter, bandwidth, package loss, etc. Traditionally, the multicast routing problem [1] with QoS is modeled as a single objective optimization problem with bounded constraints, i.e., the least cost Steiner tree with an upper-bounded QoS guarantee). With the development of high-speed networks, more and more kinds of multipoint applications with QoS requirements rise. Under these conditions, the traditional model for QoS multicast routing has some limitations as follows:

– Need prior knowledge to determine the upper-bound value of the related QoS parameters. It is hard to define the value suitably under a general network environment.
– Qualified solutions would be lost. For example, those solutions with a small amount of violation of QoS, but with an even smaller cost value would definitely be discarded.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 369 – 378, 2006.

– Pursuing one QoS gain will always result in another QoS loss, because individual QoS parameters are often interdependent and conflicting with others [2].

Although nowadays there are few works that consider more than one routing objective, most of these treat multiple objectives as a single objective using a weighed technique. For example, some prior knowledge [3] is used to determine the weight of each QoS parameter. One method [4] to determine the weight of the QoS parameters is to use a fuzzy set theory. In the other work [5], two objectives are considered and at last both of them are combined into one objective, too. However, this kind of method for treating multiple objectives poses some problems:

– The solution is very sensitive to the weight of the QoS parameters.
– It converges to a single solution. This removes the ability of choosing a solution that satisfies the users' individual requirements.

In this paper, we propose a new multi-objective multicast routing model. In the model, each QoS parameter is considered as an optimization objective rather than a bounded constraint, and the aim is to seek a set of Pareto optimization Steiner trees which compromises all QoS parameters. Next, we propose a new immune algorithm based on our previous research [6] called MO-IOA. The concept of Pareto is introduced in our algorithm. The main characteristic of the algorithm is that it can produce more than one solution within one single running. Every solution compromises all objectives, and users can choose any solution they want among all of the qualified candidates.

The rest of the paper is organized as follows. Section 2 gives an overview of multi-objective optimization based on the Pareto concept. Section 3 presents a multi-objective multicast routing model. Using this model, MO-IOA is provided in Section 4. Section 5 estimates the value of MO-IOA. In Section 6, we discuss conclusions and future research direction.

2 Concepts of Multi-objective Optimization Based on Pareto

Multi-objective optimization is defined as solving optimization problems simultaneously with more than two objectives (often competing), and there are usually no single optimal solution. Defining and discovering optimal solutions in such conditions is an open problem. We now give a more precise definition from the view of mathematics [7], and introduce some related definitions in order to explain our algorithm. It is noted that all of the objectives are in a minimized form in the following discussion.

Definition. Multi-Objective optimization Problem (MOP): An ordinary MOP includes m decision variables $X=(x_1, x_2, \ldots, x_m)$, n objectives $f_i(X)$($i=1,2,\ldots, n$).

$$\begin{aligned} &\min \quad F(X) = (f_1(X), f_2(X), \ldots\ldots, f_n(X)) \\ &subject\ to \quad g_i(X) \le 0 \quad i = 1, 2, \ldots\ldots, k_1 \\ &\qquad\qquad g_i(X) = 0 \quad i = k_1 + 1, \ldots\ldots, k_2 \\ &where \qquad X = (x_1, x_2, \ldots\ldots, x_m) \in \Omega \subseteq \Omega_1 \times \Omega_2 \times \ldots\ldots \times \Omega_m \\ &\qquad\qquad x_i \in \Omega_i, \forall i = 1, 2, \ldots\ldots, m \end{aligned} \tag{1}$$

where Ω is the feasible space.

Definition. Pareto Dominance: We say $u=(u_1, u_2, \ldots, u_k)$ dominates $v=(v_1, v_2, \ldots, v_k)$ ($u \prec v$) iff. $\forall i \in \{1,2,\ldots, k\}: u_i \le v_i$ and $\exists i \in \{1,2,\ldots, k\}: u_i \prec v_i$.

Definition. Pareto Optimality: Solution $X \in \Omega$ is said to be Pareto Optimality iff. $\neg \exists X' \in \Omega: F(X') \prec F(X)$.

Definition. Pareto Optimality Set: Pareto Optimality Set P is defined as:

$$P=\{X \in \Omega | \neg \exists X' \in \Omega: F(X') \prec F(X)\} . \tag{2}$$

Definition. Pareto Front: Pareto Front PF is defined as:

$$PF=\{ F(X) | X \in P\} . \tag{3}$$

In contrast to classical methods, multi-objective optimization algorithms based on the Pareto concept are used to determine the Pareto optimality set rather than one single optimal solution. The whole feasible space may be searched in an impliedly parallel manner, and multiple solutions will give users options from which to choose.

3 Multicast Routing Model Based on Multi-objective Optimization

A network is modeled as a direct graph $G = (V, E)$, where V is the set of nodes and E is the set of links. Let $|V|$ and $|E|$ be the sizes of nodes and links, and suppose that there exists no more than one link between any pair of nodes. For each link $e \in E$, let a four-tuple (Z, D, B, C) be the state of the link e, where Z is the capacity, D is the delay, B is the occupied bandwidth and C is the cost.

Definition. Multi-objective QoS Multicast Routing Optimization Problem (MQMROP): Let $s \in V$ be the source node of a multicast tree and $M \subseteq \{V-\{s\}\}$ be the destination node set, $|\{s, M\}|=m$, $S \subseteq V-\{s\}-M$ denotes the Steiner nodes. Let the multicast tree from s to all destination nodes be $T = (V_T, E_T)$, where $V_T \subseteq V$, $E_T \subseteq E$. Let $p(s, t)$ denote a path that connects the source node s with a destination node $t \in M$. Given a multicast task $O=(s, M, b, d)$, where b and d are the required maximum traffic and delay of the task O. The goal of MQMROP is to search a multicast tree set to optimize the three objectives shown in formula (4) at the same time. In other words, we need to find a multicast tree set $\Omega \subseteq \Omega_G$ (Ω_G is the multicast tree set satisfying formula (5)) satisfying: $\forall T \in \Omega$, there isn't any other multicast tree $T' \in \Omega_G: T' \prec T$.

$$\begin{cases} \min \quad f_1 = \max\limits_{i \in M} \left\{ \sum\limits_{e \in P_T(s,i)} D(e) \right\} \\ \min \quad f_2 = \sum\limits_{e \in T} C(e) \\ \min \quad f_3 = \max\limits_{e \in T} \left\{ \dfrac{b + B(e)}{Z(e)} \right\} \end{cases} \tag{4}$$

$$\begin{cases} b + B(e) \le Z(e), & \forall e \in T \\ \max\limits_{i \in M} \left\{ \sum\limits_{e \in P_T(s,i)} D(e) \right\} \le d \end{cases} \tag{5}$$

The above three optimal objectives are the end-to-end delay, the cost of multicast tree and the bandwidth utilization rate. We consider these three objectives since:

- The end-to-end delay is an important measure for real time multi-media communication.
- The cost of multicast tree (communication) is another interesting issue for users.
- The minimum bandwidth utilization rate is necessary for load-balancing the network in order to reduce the call blocking rate.

It should also be noted that it is simple to extend our model to more than three objectives.

4 Multi-objective Immune Optimization Algorithm

In this section, we propose a new routing algorithm called Multi-Objective Immune Optimization Algorithm (MO-IOA). The proposed algorithm is still a population iteration algorithm like popular multi-objective evolutionary algorithms (MOEA) [8]. But different from MOEA, we take advantage of the Artificial Immune System (AIS) [9] principle to solve the multi-objective routing problem.

The main component of MO-IOA is the clone process [10], a famous immune process in the AIS. This process is an effective search operator [11]. It is good at local search, and to some extent, global search. This capability is very important for MO-IOA to find all possible solutions successfully. The clone process is composed of two phases: clone and mutation. During the process, the amount of clone and the mutation rate conducted by an individual are inversely proportional to its fitness. That is to say, as the quality of the individual increases, the search area around it decreases. It is assumed that the area around the better solution has a greater chance of containing optimized solutions.

4.1 Gene Library

As we all know, the routing problem is a typical real time application and the speed of a routing algorithm is very vital. In order to speed up the search process of MO-IOA, a gene library is introduced here. It simulates the production of new antibodies in bone barrow [12]. In our gene library, there is more than one path from the source node to the destination nodes, and a new path constructed from a gene library has better quality than randomly constructed ones with much higher possibility. The Dijkstra *K*-th shortest path algorithm [13] is used to construct the gene library. According to the statistic [14], the optimal multicast tree includes the shortest path with a higher possibility. The details are as follows:

Suppose the paths from the source node s to the destination node $d_i \in M$ are needed to specify. Due to more than one optimization objective, the Dijkstra *K*-th shortest

path algorithm should be run three times. A different optimization objective is used each time. Then we get 3*R* paths: *R* least-cost paths, *R* least-delay paths, and *R* least bandwidth-rate paths. Among them, the paths violating the delay constraints will be discarded.

4.2 Architecture of MO-IOA

MO-IOA is an iteration algorithm based on population. Besides an antibody population, there is a memory population [15] containing the best tradeoff solution or Pareto optimized solutions found up to this point. It is noted that the antibody population size is changeable since the clone number of each antibody is related to its fitness, so a recombination mechanism is introduced to keep the size fixed between the generations. Notice that the fitness of an antibody is a vector rather than a scalar.

In MO-IOA, there are five important parameters: the rate of initial individuals constructed by the gene library $d_0\%$, the rate of deleted individuals $d_1\%$, the antibody population size d_2, the memory population size d_3, and the gene library size R. The algorithm is given below:

Input: A given network $G = (V, E)$, the state of the network, a source node s and a set of destinations.

Output: A set of Pareto-optimal multicast trees.

1. Delete the links in G violating the bandwidth requirement. If all destination nodes aren't located in the same one connected sub-graph G', it is necessary to negotiate the bandwidth requirement and then re-start the process.
2. Use the Dijkstra K-th shortest algorithm to construct the gene library, and delete the paths violating the delay constraints. If the legal path number of any destination node is 0, it is necessary to negotiate the delay requirement and then re-start the process.
3. Initialize the antibody population $P(0)$: $\lfloor d_0\% \cdot |P(0)| \rfloor$ individuals constructed by the gene library and the remaining ones by random construction. Initialize the memory population to empty. $t = 0$.
4. Detect if each individual exist any circle path. If yes, delete the existed circle.
5. Compute the three objectives' functions respectively, for each individual p_i in $P(t)$.
6. Compute fitness vector (m_1, m_2, m_3) of each p_i, m_1 is the number of individuals dominated by p_i ($p_i \prec p_j$); m_2 is the number of individuals neither dominated by p_i nor dominating p_i ($p_i \odot p_j$); m_3 is the number of individuals dominating p_i ($p_i \succ p_j$).
7. Recombination: If the size of antibody population $> d_2$, then delete $|P(t)| - d_2$ individuals which have the biggest m_3, and then delete $\lfloor d_1\% \cdot |P(t)| \rfloor$ individuals in the remaining population which have the biggest m_3.
8. Insert individuals with m_3=0 into the memory population.
9. Clone $N_0 \cdot \dfrac{m_1 + m_2 + |P(t)|}{m_3 + |P(t)|}$ copies for each individual p_i according to its fitness vector (m_1, m_2, m_3). Supposing Ψ_i the clone output set for p_i.

10. Mutate each copy in Ψ_i, supposing Ψ'_i the mutated set of Ψ_i.
11. Immune selection based on local competition: Compute fitness vector (m_1, m_2, m_3) of each individual in Ψ'_i. Choose those individuals (namely q) in it to compete with p_i: each of them satisfy (1) m_3=0 or (2) have the biggest $m_1+ m_2$. If $q \prec p$, then q substitute p_i and be inserted into $P(t)$; if $q \odot p_i$, then choose q or p_i with probability into $P(t)$.
12. Introduce $\lfloor d_1\% \cdot |P(t)| \rfloor$ new individuals into $P(t)$ by random construction.
13. $t = t+1$.
14. If the termination condition is not satisfied, goto 5; else goto 15.
15. Return Pareto-optimal individuals (multicast trees) in the memory population.

In step 9 and 10, we realize the main search process: clone search. Step 9 shows that the better the solution, the more copies it will produce. In step 10, single point mutation is used: choose a destination node v_i randomly, then substitute the path from s to v_i with another path chosen from the gene library randomly. In step 12, the introduction of new individuals will improve the diversity of the population [16] and the globe search capability of MO-IOA. The termination condition is that memory population is unchanged within consecutive five generations.

5 Simulation

In order to evaluate the benefits of MO-IOA, a simulation system is developed. This simulation system includes a network with $|V|$=14. Each link has three metrics: the delay $D(e)$(ms), the cost $C(e)$, the occupied bandwidth $B(e)$(Mbps). Suppose all links have the same 1.5Mbps bandwidth capability). Let the multicast requirement be B=0.2 Mbps, source node s=0, multicast destination set M={3,6,7,9,12,13}, maximum delay constraint 60ms. The parameters of MO-IOA are: d_0%=80%, d_1%=30%, d_2=10,

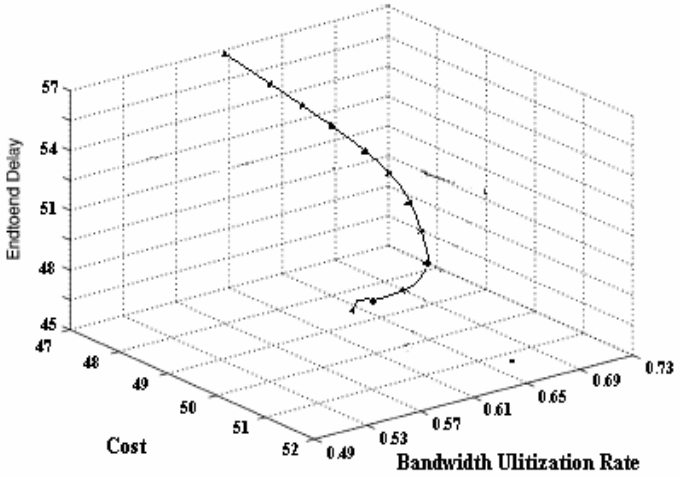
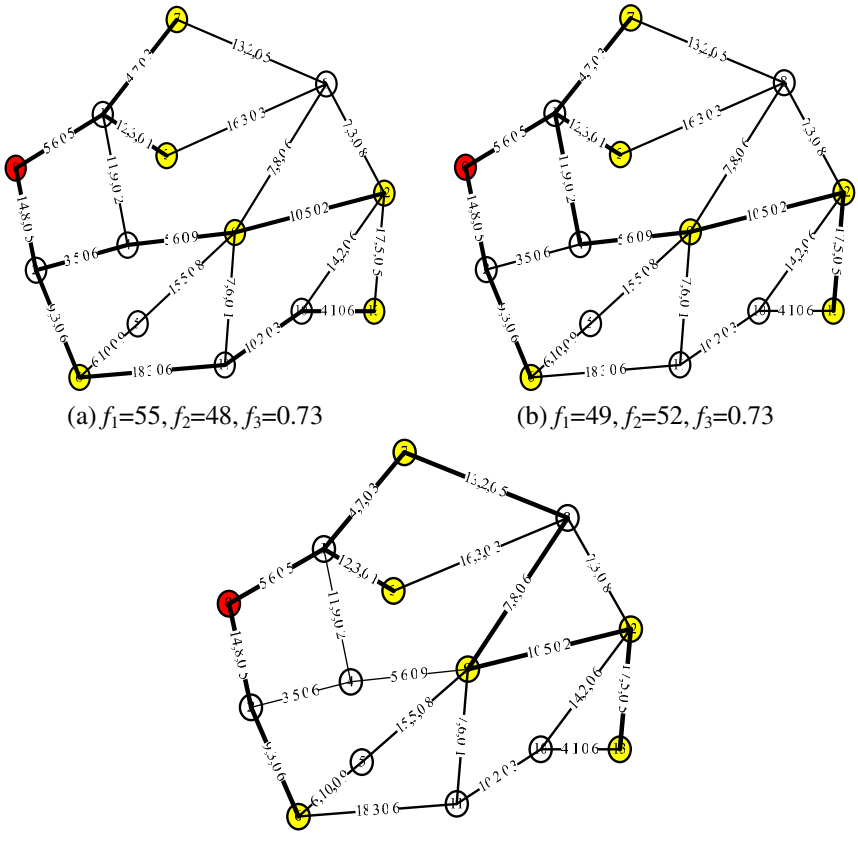


Fig. 1. Pareto Front found by MO-IOA

d_3=20, R=8. The computing platform used to perform is P4, 2.4GHz CPU, 256MB. After running MO-IOA 30 times, it can find the Pareto optimality set with a 60% possibility. The average running time is 250ms, with the longest running time being 300ms. The exhaustive search expend nearly three hours of running time.

In Fig.1, the Pareto Front found by MO-IOA is shown with dotted points, and the curve is the Pareto Front found by the exhaustive search.



(a) f_1=55, f_2=48, f_3=0.73

(b) f_1=49, f_2=52, f_3=0.73

(c) f_1=56, f_2=47, f_3=0.53

Fig. 2. Three solutions of the Pareto optimality set

Fig.2 (a) - (c) are three Pareto optimality solutions chosen randomly from the Pareto optimality set, where node number 0 is the source node and the thick links are multicast links. It is obvious that, according to the objective vector (f_1, f_2, f_3):

(55, 48, 0.73), (49, 52, 0.73), (56, 47, 0.53)

There aren't any dominated or dominating relationships between each other.

In order to compare MO-IOA with other algorithms, we choose a famous delay-constraint shortest path algorithm SP [17] as a benchmark. The network environment is the same as above. In the simulation, there are 100 multicast session requests and a 0.12Mbps bandwidth requirement. The size of multicast ranges from 4 to 10, and each

member is chosen from network randomly, including the source node. Each session's last time satisfies the exponential distribution (mean value=200s). The arriving time satisfies the Poisson distribution (λ=15). The maximum delay constraint is 1.3 times the maximum delay constraint of multicast tree constructed by SP. The comparison metrics are:

$$\text{Cost: } c_1 = \frac{C(T_{SP}) - C(T_{MO-IOA})}{C(T_{MO-IOA})}. \tag{6}$$

$$\text{Delay: } c_2 = \frac{D(T_{SP}) - D(T_{MO-IOA})}{D(T_{MO-IOA})}. \tag{7}$$

$$\text{Bandwidth utilization rate: } c_3 = \frac{R(T_{SP}) - R(T_{MO-IOA})}{R(T_{MO-IOA})}. \tag{8}$$

Fig.3 is the change curves of the three metrics during 100 sessions. We can conclude that, compared with SP, the performance of multicast trees constructed by MO-IOA has a 51% higher bandwidth utilization rate, 10% lower cost at the expense of a little longer delay than SP. Overall, when considering three objectives at the same time, MO-IOA compromises the contradiction among them and obtains an adequate tradeoff.

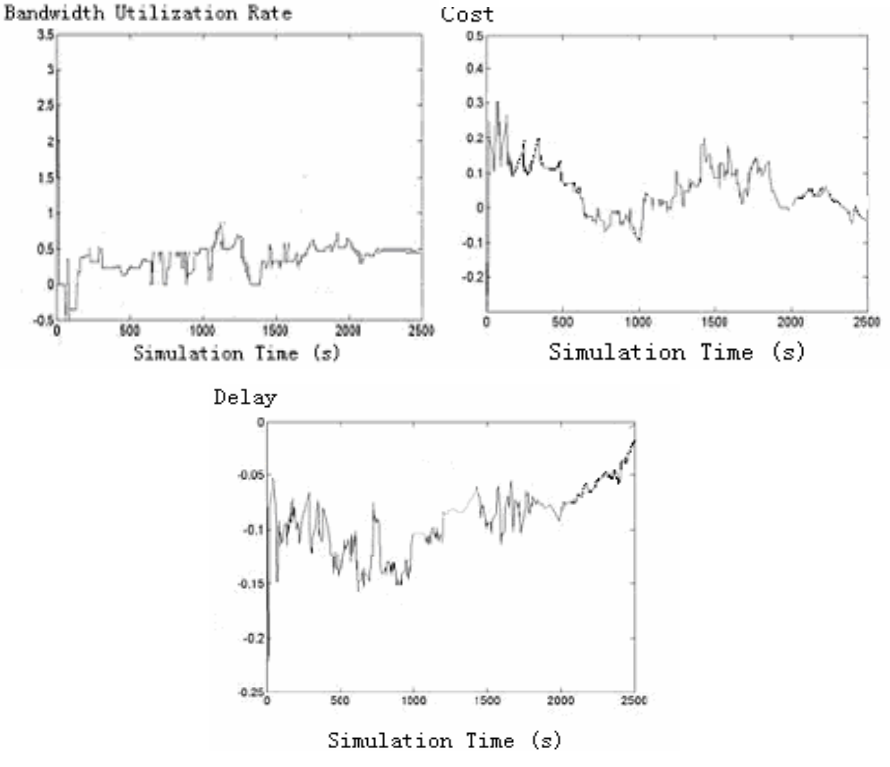


Fig. 3. The change curves of the three metrics

6 Summary

In order to overcome some limitations in the current routing model, we studied multi-objective multicast routing in this paper. Based on the Pareto concept, a new model of multicast routing and an immune algorithm named MO-IOA are proposed. MO-IOA can produce a set of Pareto solutions within a single running of the algorithm, instead of producing a single solution by combining all of these objectives together using a traditional weighed technique. The main advantage of our algorithm is that it can provide multiple choices to satisfy the individual requirements of various users. The initial experiments have shown its effectiveness and feasibility. It will be very promising to apply to routing optimization fields. Our continuing research will focus on more experiments for performance and convergence analysis of the algorithm, the function of the gene library and the analysis of parameter sensitivity.

Acknowledgments. This study was supported in part by China NSF grant 69635030, 60073043 and Hubei PSF grant 2004ABA029.

References

1. Wang, Z., Crowcroft, J.: Quality of Service for Supporting Multimedia Applications. IEEE Journal on Selected in Communications, 14(7), (1996) 1228-1234
2. Sahasrabuddhe, L. H., Mukherjee, B.: Multicast Routing Algorithms and Protocols: A Tutorial. IEEE Network, 14(1), (2000) 90-102
3. Mcquillan, J. M., Richer, I., Rosen, E.C.: The New Routing Algorithm for the ARRANET. IEEE Trans. on. Comm. 28(5), (1980) 711-719
4. Marwaha, S.: Evolutionary Fuzzy Multi-Objective Routing for Wireless Mobile Ad Hoc network. CEC, (2004) 345-355
5. Ji, Z. W.: Finding Multi-Objective Paths in Stochastic Network. CEC, (2005) 1300-1307
6. Qin, J., Kang, L. S.: A Novel Dynamic Population based Algorithm to Solve Multi-modal Function Optimization. Proceedings of World Congress on Intelligent Control and Automation, Hangzhou (2004)
7. Daid, A., Van, V., Gary B.: Multi-Objective Evolutionary Algorithms: Analyzing the State-of-the-Art. Kalyanmoy Deb, (2000)
8. Deb, K.: Multi-Objective Optimization Using Evolutionary Algorithms. Chichester: JOHN WILEY & SONS LTD (2001) 34-76
9. Dasgupta: Immune Systems and Their Applications. Springer-Verlag, (1999)
10. Ada, G. L, Nossal, G. V.: The Clonal Selection Theory. Scientific American, 257(2), (1997) 50-57
11. de Castro, L. N., Von, Z. F. J.: Artificial Immune Systems: Part I-Basic Theory and Applications. Technical Report, TR-DCA (1999)
12. Forrest, S., Perelson, A. S.: Genetic Algorithms and the Immune System. In Hans-Paul Schwefel and R. Maenner, editors. Parallel Problem Solving from Nature, Lecture Notes in Computer Science. Berlin, Springer-Verlag (1991) 320-325
13. Tanenbaum, A. S.: Computer Network.3rd ed. Prentice Hall Inc.,1996
14. Hwang, F. K., Richards, D. S.: Steiner Tree Problems, Networks 22 (1992) 55-89

15. Pan, Z. J., Kang, L. S.: Evolutionary Computation. Beijing: Qinghua university publication (1997)
16. Opera, M., Forrest, S.: How the Immune System Generates Diversity: Pathogen Space Converge with Random and Evolved Antibody Libraries. Genetic and Evolutionary Computation Conference (1999) 1651-1656
17. Doar, M., Leslie, I.: How Bad is Naive Multicast Routing. Proceedings of the IEEE INFOCOM (1993) 82-89

A Novel Collaborative Tier Scheme for Multihop Inter-Vehicle Communication Networks*

Xiaojian Xu, Li Chang, and Hanying Hu

Department of Communication Engineering, Information Engineering University, Zhengzhou, 450002, P.R. China
XuXiaojian@ieee.org, {Claymail, Huhanying}@vip.sina.com

Abstract. A novel collaborative strategy employing two interacted relays is developed for the multihop inter-vehicle communication (IVC) networks in this paper, which has great complexity degradation. Instead of the traditional terminal collaboration or distributed space-time coding (DSTC), a relay tier interaction is proposed. Considering time division multiple access (TDMA) scheme, two spatial adjacent vehicles are conjoined by an interaction radio link to create a virtual antenna array (VAA). Then the collaborative two-relay is integrated into a multihop IVC network mentioned above, which enables a single-antenna terminal to share its antenna with other terminals easily, and avoids the synchronization or other complicated requirements in terminal collaboration or DSTC systems. Furthermore, except the low complexity in hardware implementation, the proposed method also can enhance the performance of the IVC systems over both the quasi-static fading channel and time varying Rayleigh fading relay channel, which is verified through the Monte-Carlo simulations.

1 Introduction

Wireless communications technologies make a tremendous impact on intelligent transportation systems (ITS). In order to improve the safety of traffic and efficiency of road transport, ITS have been investigated intensively over the past several years [1], [2]. For example, the safety applications make the driving safer, and the driver information services intelligently inform the drivers about the congestion, the businesses, the services in the vicinity of the vehicle and other news. In addition, the mobile commerce also can be applied to the ITS, and the existing entertainments can be extended to the vehicle environments. Hence the inter-vehicle communication (IVC) capabilities play an important role in this application. Though the IVC network among cars is similar to the Ad-hoc network scenarios, the constraints and the optimizations are different, and the power efficiency is not as important for IVC as for traditional Ad-hoc networking.

By now, numerous research challenges still remain to be addressed in order for IVC to be widely deployed, and we focus on the physical layer design for two-relay

* This research has been supported in part by the National Science Foundation (NSF) of China under grant number 60472064.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 379 – 388, 2006.

collaborative multihop IVC sub-network over the quasi-static fading channel or time varying Rayleigh fading relay channel. We investigate two regenerative intermediate relays collaborated transmission and the process of distributed radios employing interaction channels within a collaborative tier. Because the intelligent antenna sharing in each tier can be viewed as virtual antenna array (VAA) [3], we design a novel relay strategy that allows the spatial distributed relays to share their antenna effectively to achieve a collaborative diversity (CD), and allocates time slots for the exchange of data between the relaying cars. Therefore the traditional traffic information center (TIC) is unnecessary to the IVC system with a capability of multihop connection. In addition, collaborative IVC can offer a number of benefits, such as overcoming dead-spots, reducing the transmitting power [4], increasing system capacity and counteracting channel fading. In [5], Toshiaki et al. apply space-time block coding (STBC) cooperative relaying scheme to multihop IVC networks, which need a perfect synchronization. Hence, we propose a simplified two-ray collaborative method which enhances the performance of the IVC systems and decreases the requirement for perfect synchronization, with the cost of additional channel resource for exchanging the information.

The remaining is organized as follows. In Sect. 2, the multihop IVC system model is introduced briefly. The two-relay collaborative strategy is proposed in Sect. 3. Then, in Sect. 4, numerical simulations for different IVC transmission protocols are compared, and some conclusions are drawn in Sect. 5.

2 Multihop IVC with Collaborative Method

A multihop IVC system with two relaying vehicles is illustrated in Fig. 1. Owing to multihop communications, data transmissions and information sharing are available

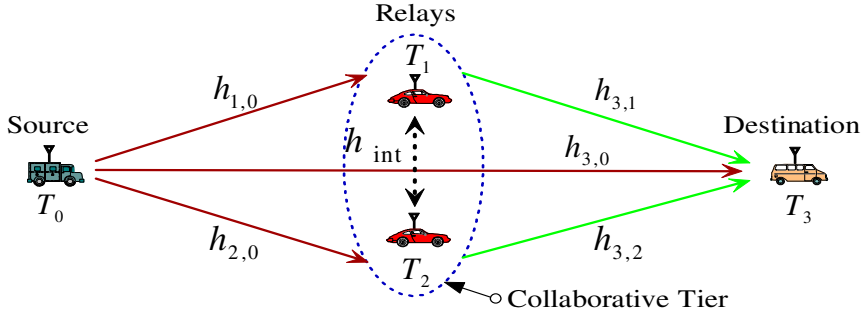


Fig. 1. Illustration of collaborative multihop IVC system (sub-network) with two-relay collaborative tier. The source vehicle broadcasts the message to the destination and multiple relaying vehicles during the first time slot. After interaction processing is completed, the relays create a collaborative tier, i.e. virtual antenna array, and retransmit signal to the destination where all received signal branch can be appropriately combined.

over an extensive area without depending on TIC [6]. In this paper, we assume peer-to-peer communications and the distance between two relays is far less than that between the source vehicle and the destination vehicle.

We denote the source vehicle as node T_0, two-relay as nodes T_1 and T_2, and the destination vehicle as node T_3. In Fig.1, narrowband transmissions suffer the effects of path loss and flat fading. Importantly, when the motorcade (composed by the nodes T_0, T_1, T_2 and T_3) is moving, all the radio links behave as a time varying Rayleigh fading channel, with Jakes models [7]. While the motorcade is static, all the links are quasi-static fading channel. With the baseband-equivalent discrete-time channel model, the directly transmitted signal of the source vehicle, denoted by x, results the received signal at the destination vehicle T_3

$$y_{3,0} = h_{3,0}x + z_{3,0}, \tag{1}$$

After the information exchanging between the two-relay T_1 and T_2 (which will be discussed in Sect. 3), the transmitted signal symbol of the collaborative tier can be denoted by $\tilde{x}$. According to the relaying strategy, the received signal at the destination vehicle T_3 can be expressed as:

$$y_{3,r} = h_{3,r}\tilde{x} + z_{3,r}, \quad r = 1, 2. \tag{2}$$

We assume that the relays and source have the same transmitting power, i.e. $E_0 = \mathrm{E}[x^{\dagger}x] = \mathrm{E}[\tilde{x}^{\dagger}\tilde{x}]$, and relays can effectively recover the data from source. In the description above, the fading coefficients $h_{i,j}$, $i \in \{1,2,3\}$, $j \in \{0,1,2\}$, $i > j$, are modeled as zero mean, mutually independent complex jointly Gaussian (ZMCSCG) random variables [8] with variance $\sigma^2_{h_{i,j}}$, and all the noises and the interferences in the radio links are modeled as ZMCSCG, $z_{i,j} \sim CN(0, N_0)$.

For the path loss, we assume a log-distance model, i.e. the received power decreases linearly with distance, on a logarithmic scale. An interesting approach based on [4], [9] suggests to model the effects of path loss into the variance of the fading variables by observing that the SNR (Signal-to-Noise Ratio) at a specific vehicle *j* obtained by transmission form vehicle *i* can be written as:

$$\gamma_{i,j} = \left[\gamma\left(\frac{d_0}{d_{i,j}}\right)^{\alpha}\right]|h_0|^2 = \gamma\left[\left(\frac{d_0}{d_{i,j}}\right)^{\alpha}|h_0|^2\right] = \gamma|h_{i,j}|^2, \tag{3}$$

where $|h_0|$ is a fading coefficient with unit variance $\sigma_0^2 = 1$, $d_{i,j}$ is the distance between transmitter and receiver vehicles, α is the path loss exponent and γ is the signal-to-noise ratio attained by the transmitter at a receiver at reference distance d_0. Without loss of generality, we define $d_0 = d_{s,d}$ and therefore $\sigma^2_{s,d} = 1$. Throughout the paper, we can now model the effect of path loss in the following way:

$$\sigma_{i,j}^2 = \left(d_{s,d}/d_{i,j}\right)^{\alpha} \triangleq r_{i,j}^{-\alpha}, \tag{4}$$

where we have introduced the variable $r_{i,j} \triangleq d_{s,d}/d_{i,j}$ as the normalised diatance between two vehicles. Based on above SNR analysis, the average error probability of multihop IVCs can be attained easily (see Appendix A). This approach allows for a convenient study of the effects of geometry on the qualities of the proposed strategy.

3 Two-Relay Collaborative Strategy

For our collaborative transmission protocol based on the considered IVC system, each relaying vehicle equipped with only one antenna, which can not to process the received and transmitted signals at the same time. Thus, to ensure half-duplex operation, we divide each channel into orthogonal sub-channels [4]. Fig.2 illustrates an example of media access control (MAC) based on time-division channel allocation for orthogonal collaboration diversity in our multihop IVC systems. In addition, some level of (block, carrier, and symbol) synchronization between the vehicles is expected to ensure the proposed collaboration method to be more effective [10-18].

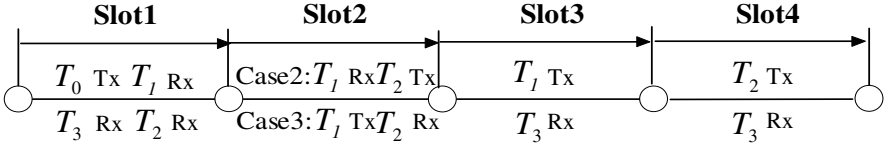


Fig. 2. TDD based time slots allocation mode for multihop IVC. This is repetition-based MAC protocol. The relaying sub-channels are allocated across time for the DF cooperative strategy.

Considering the transmission information exchanging process of two-relay in the collaboration tier at different allocated time slots, the relaying vehicles are devoting to decoder-and-forward (DF) transmitting protocols [4], [13], [14]. For the DF strategy, we note that the source vehicle transmits x, and the relay froms an estimate $\tilde{x}$ by decoding its corresponding received signal $y_{k,0}$ for k={1,2},and relays a re-encoded version of $\tilde{x}$, in (1) and (2). Without loss of generality, T_1 and T_2 are supposed to be closely spaced. The exchanging information of two relays is assumed to be entirely correct after passing through the interaction channel h_{int}, hence a repeat request scheme for failed packets is not employed. To relay the received packet, the relaying car checks the packet header information and finds the packet error. We assume that the packet error can be detected perfectly by using cyclic redundancy check (CRC) bits. If the CRC decoder output "1", the detected date is correct. Otherwise, the data is false. The four cases in the two-relay collaboration IVCs are defined as follows:

Case1: "00" means that the packet error exists at both T_1 and T_2, then T_1 and T_2 directly relay their packages to T_3.

Case2: "01" means that the packet error exists at T_1, and T_2 sends the right packet to T_1. After the interaction, relay cars T_1 and T_2 transmit their packages to T_3 at the corresponding time slots.

Case3: "10" means that the packet error occurs at T_2, and T_1 sends the right packet to T_2. After the interaction, relay cars T_1 and T_2 transmits their packages to T_3 at the corresponding time slots.

Case4: "11" means that the packets are correct at both T_1 and T_2, thus T_1 and T_2 directly relay their packages to T_3.

In the destination vehicle, T_3 makes use of maximal ratio combination (MRC) to obtain the final results. Cooperative diversity allows a collection of radio terminals that relay signals for each other to emulate a VVA and exploit spatial diversity in multihop IVC networks [13], [14]. Especially, the proposed two-relay collaborative method can also improve the performances and reliability in an asynchronous IVC networks. For a variety of processing algorithms and transmission protocols, a couple of remarks are pertinent to futher considerations as follow:

Remark 1: When channel state information (CSI) is avaiable at the ralay node, three different degrees of CSI can be condidered at the relaying strategy [17], [18]. The CSI at the relay can be only information about the first hop channel (between the source and relay) or information about the first and second hop channel (between realy and destination). The second situation is feasible for TDD systems if both destination and relay share a previous dialogue and so CSI can be obtained using the signal received from the opposite link. Finally, Opportunistic Relaying considers that the relay has knowledge about all the links involved at the coomunication, including the direct channel between source and destination [13]. Furthermore, full CSI-base relay choosing algorithm is a more unrealistic situation since it requires the destination informs the relay about the direct link. For the tradeoff between the system performance and the hardware complexity, the two-relay collaborative method has no feedback information from destination vehicle to source vehicle in this paper.

Remark 2: We have not still mentioned the DSTC systems. In [5], Toshiaki et al. applies DSTC cooperative relaying, and uses the orthogonal STBC exploting two-branch antennas, i.e. Alamouti's scheme, to improve multihop IVC system capacity without dynamic channel control schemes, which assume a perfectly synchronous IVC network. When semi-synchronous or asynchronous IVC networks are taken into account, the DSTC scheme sufferes from serious error propagation. In [16], Wei et al. analyse the asynchronous cooperation transmission protocols. For high link quality, a distributed Turbo code is integrated into the cooperative network and it enables single-antenna terminal easily to share its antenna and create a virtual array [15].

4 Simulation Result and Analysis

Some numerical results are presented in this section. The simulation parameters are shown in Table 1. We compare the performances of different collaboration transmission methods i.e. direct, one-relay (1 relay), two-relay (2 relays A) scheme without collaboration, two-relay collaborative (2 relays B) scheme, and all methods are based on DF relays for both quasi-static channel and time-varying Rayleigh fading channel, with the coordinates normalized by the distance between the source and destination vehicles.

Table 1. Simulation parameters setup for the collaborative multihop IVC systems

IVC System Parameters	Setup
Carrier frequency	2 GHz
Frame length	1024 bits
Information rate	100 Kbps
Modulation schemes	QPSK
Multiplex schemes	TDD
Max. moving speed	50 Km/h
Cooperative relaying	DF
Doppler frequency	93 Hz
Path loss coefficient	3

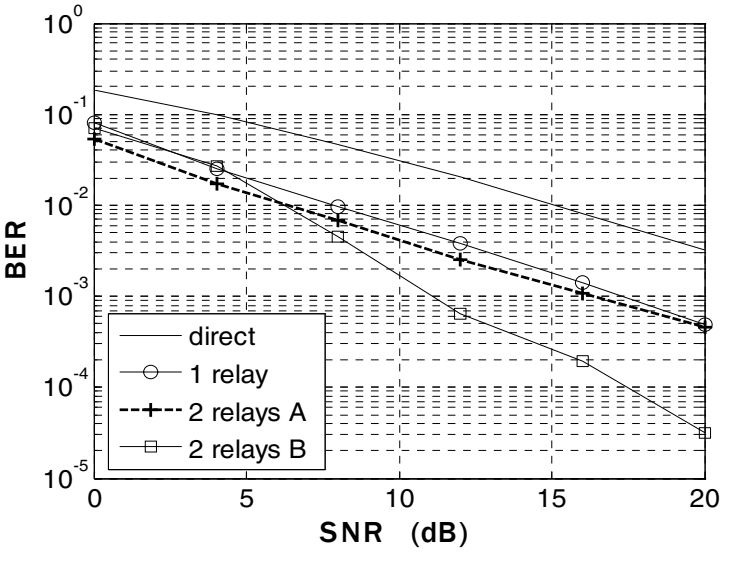


Fig. 3. The BER performance of IVC system with one or two relaying vehicles via Monte Carlo simulations. The mode has: path-loss with exponent $\alpha = 3$, time varying independent Rayleigh fading relay channel, network geometry with relays located at the midpoint between the source and destination, and uniform power allocation.

The characteristics of the channel variants can affect the bit error ratio (BER) of the proposed relay system. As shown in Fig.3 and Fig.4, when one relay is used, the required SNR is 8dB and 10dB for time-varying channel and quasi-static Rayleigh channel respectively to achieve the BER of 10^{-2}. Thus, through time diversity, the characteristics of channel variants can improve the BER performance of uncoded relay system. In addition, when the two relays are used, the corresponding required SNR is 6dB and 5.5dB respectively to achieve the BER of 10^{-2}. From the above discussions, we can find that the multihop IVC system with two relays has two times diversity gain than the system with single relay. From the two figures, we can conclude that the IVC system with the scheme "2 relays B" has the best BER performance, and all the schemes can achieve the better BER performance in quasi-static fading relay channel.

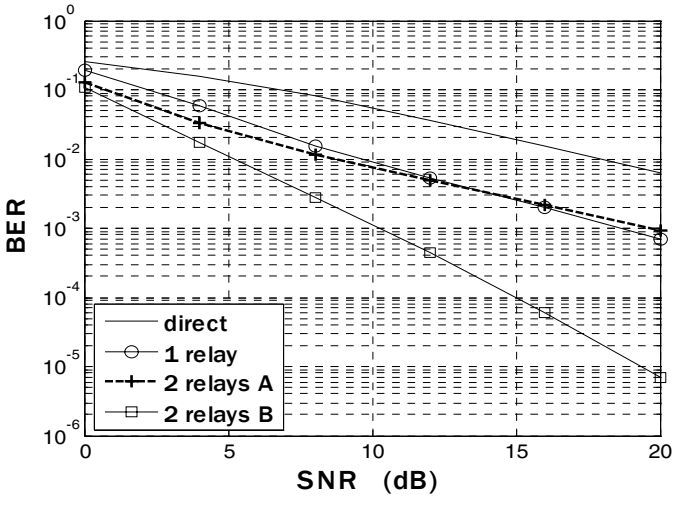


Fig. 4. The BER performance of IVC system with one or two relaying vehicles via Monte Carlo simulations. The mode has: path-loss with exponent $\alpha = 3$, quasi-static fading relay channel, network geometry with relays located at the midpoint between the source and destination, and uniform power allocation.

In Fig.5 and Fig.6, the frame error rate (FER) performance curves of the different collaboration transmission methods are shown. The curves verify that the improvement of multihop IVC with interaction in the collaborative tier is significant.

One can note that there exists an interesting phenomenon: the FER of single relay IVC system outperforms the one of the two-relay IVC system without collaboration in quasi-static fading relay channel, as illustrated in Fig.6. It can be explained that two independent relays without cooperation in 2 relays A scenario suffer from serious error propagation at a higher probability than that of one relay only and thus the schem can not achieve two times diversity gain over the single relay case at the studied scheme and system parameters. However, as shown in Fig.4 and Fig.6, the collaboration operation between the two nearbour DF relays can effectively decrease the error propagation, and results in a significant collaboration diversity gain in 2 relay B scenarios for IVC system.

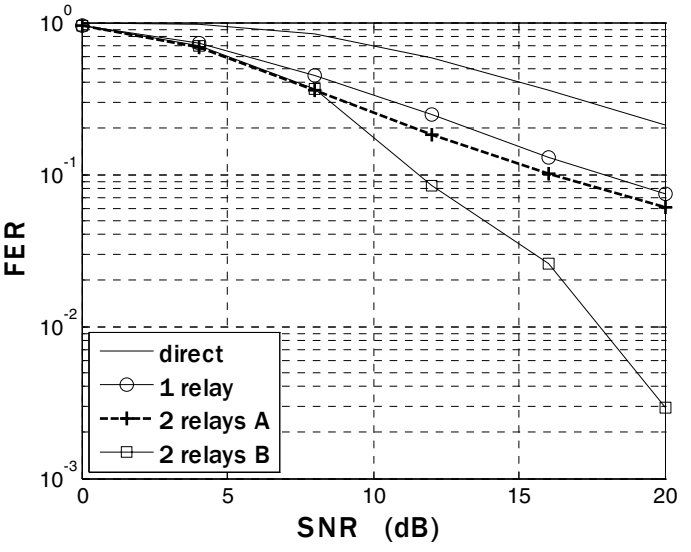


Fig. 5. The FER performance of IVC system with one or two relaying vehicles via Monte Carlo simulations. The mode has: path-loss with exponent $\alpha = 3$, time varying independent Rayleigh fading relay channel, network geometry with relays located at the midpoint between the source and destination, and uniform power allocation.

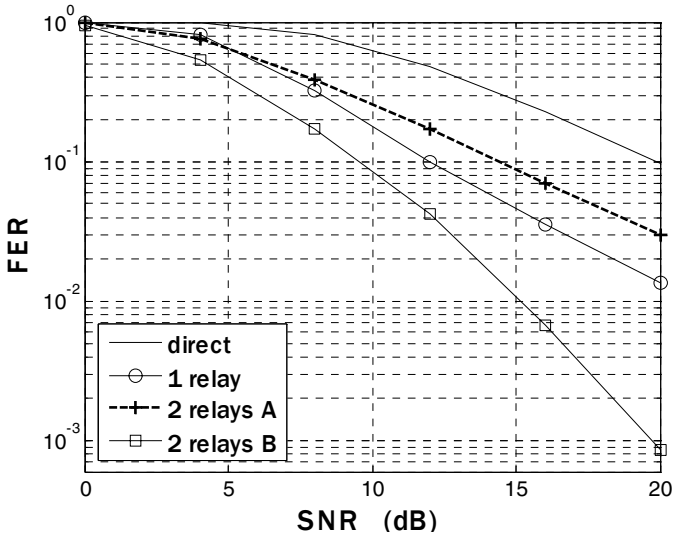


Fig. 6. The FER performance of IVC system with one or two relaying vehicles via Monte Carlo simulations. The mode has: path-loss with exponent $\alpha = 3$, quasi-static fading relay channel, network geometry with relays located at the midpoint between the source and destination, and uniform power allocation.

5 Conclusions

Two-relay collaborative multihop IVC system with TDMA is proposed in this paper. The simple collaborative scheme is used to create a virtual array which enables the single antenna vehicles to share their antennas easily over quasi-static fading channel and time varying Rayleigh fading relay channel. Simulation results prove that the proposed method has enhanced diversity performance and the BER performance can be improved greatly. Furthermore, it allows for simple implementation on the IVC systems in CDMA or FDMA system, and can be flexibly extended to the scenarios of multibranch or parallel concatenated (multistage) collaborative groups.

Acknowledgments

The author would like to thank Dr. Meng Qingmin of National Communication Research Laboratory, Southeast University (SEU) for valuable guide and thoughtful comments.

References

1. Andrisano, O., Verdone, R., Nakagawa, M.: Intelligent Transportation Systems: The Role of Third-Generation Mobile Radio Networks. IEEE Commun. Mag., Vol. 38, No. 9 (2000)
2. Franz, W.J., Hartenstein, H., Bochow, B.: Internet on the Road via Inter-Vehicle Communications. NEC Europe Ltd. and DaimlerChrysler Research & Technology, FleedNet Project Technical Report, BMBF 01AK025 (2002)
3. Dohler, M., Lefranc, E., Aghvami, H.: Space-Time Block Codes for Virtual Antenna Arrays. Proc. 13th IEEE Int. Symposium on Personal, Indoor and Mobile Radio Commun (2002)
4. Laneman, J., Tse, D.N.C., Wornell, G.: Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behavior. IEEE Trans. Inf. Theory, Vol. 50, No. 12 (2004)
5. Toshiaki KOIKE, et al.: Capacity Improvement of Multihop Inter-Vehicle Communication Networks by STBC Cooperative Relaying. IEICE Trans. Commun., Vol.E88, No. 9 (2005)
6. Pabst, R., Walke, B.H.: Relay-Based Deployment Concepts for Wireless and Mobile Broadband Radio. IEEE Commun. Magn 42 (2004)
7. Jakes, W.C.: Microwave Mobile Communications. Wiley, New York (1974)
8. Paulraj, A., Nabar, Gore: Introduction to Space-Time Wireless Communications. Cambridge University Press (2003)
9. Zimmermann, E., Herhold, P., Fettweis, G.: A novel protocol for cooperative diversity in wireless network. In 5th European Wireless Conference Mobile and Wireless Systems beyond 3G (2004)
10. Sendonaris, A., Erkip, E., and Aazhang, B.: User Cooperation Diversity, Part I: System Description. IEEE Trans. Commun.,Vol. 51, No.11 (2003)
11. Sendonaris, A., Erkip, E., and Aazhang, B.: User Cooperation iversity, Part II: Implementation Aspects and Performance Analysis. IEEE Trans. Commun. , Vol. 51, No.11 (2003)
12. Hunter, T.E., Nosratinia, A.: Diversity through Coded Cooperation. In Proc. IEEE Int. Symp. Information Theory, Chicago, IL (2004)
13. Bletsas, Aggelos.: Intelligent Antenna Sharing in Cooperative Diversity Wireless Networks. PhD thesis Massachusetts Institute of Technology. Cambridge (2005)

14. Laneman, J., Tse, D.N.C., Wornell, G.: Energy-Efficient Antenna Sharing and Relaying for Wireless Networks. In Proc. IEEE Wireless Comm. and Networking Conf., Chicago, IL (2000)
15. Meng Qing-min, Gao Xi-qi, You, X.-H.: Coded Cooperation for Wireless Two-Relay Networks. Journal of Circuits and Systems 2 (2006)
16. Wei, Shuangqing, Goeckel, D., Valenti, Matthew: Asynchronous Cooperative Diversity. IEEE Trans. Wireless Commun (2005)
17. Kramer, Gerhard, Gastpar, Michael, Gupta, Piyush: Cooperative Strategies and Capacity Theorems for Relay Networks. IEEE Trans. Inform. Theory 51 (2005)
18. Barbarossa, S., Scutari, G., Ludovici, D., Pescodolido, L.: Distributed Space-Time Coding Strategies for Wideband Multi-hop Networks: Regenerative vs. Non-Regenerative Relays. IST-2001-32549 ROMANTIK D442 (2005)

Appendix A. Multihop IVCs Error Propagation

Adopting DF relay strategy, the relay fully decodes, again encodes and retransmits message, possibly propagating decoding errors that may lead to wrong decision at the destination vehicle. Hence we derive the expression for total error propagation influence, which account for both the relay backward channels, i.e. uplink, $S \to R_k$ for k={1,2}, and the relay forward channels, i.e. downlink, $R_k \to D$ for k={1,2}. Assume that $S \to D$ and $R_k \to D$ signals are combined with MRC. Based on [9], [13], [14], [18], the multihop IVCs average error probability is given by:

$$\overline{P}(e) \le \frac{1}{4}\left(1-\mu_{SD}\right)\prod_{k}\left(1-\mu_{R_kD}\right)+\frac{1}{2}\left(1-\frac{\sum_{k}\overline{\gamma}_{R_kD}\mu_{R_kD}+\overline{\gamma}_{SD}\mu_{SD}}{\sum_{k}\overline{\gamma}_{R_kD}-\overline{\gamma}_{SD}}\right), \tag{5}$$

where $\mu(\cdot)=\left\{\kappa\overline{\gamma}(\cdot)/\left[1+\kappa\overline{\gamma}(\cdot)\right]\right\}^{1/2}$, κ is a modulation dependent constant ($\kappa=4$ for *QPSK*, and $\kappa=2$ for *BPSK*), and $\overline{\gamma}(\cdot)=\mathrm{E}\left[\gamma(\cdot)\right]$. γ_{SD}, γ_{SR_k}, and γ_{R_kD} are defined by $\gamma_{i,j}, i\in\{1,2,3\}, j\in\{0,1,2\}, i>j$, given by (3).

Performance Computation Model for IEEE 802.11e EDCF Wireless LANs

Rongbo Zhu[1,2] and Yuhang Yang[1]

[1] Department of Electronic Engineering, Shanghai Jiao Tong University,Dongchuan Road 800, Shanghai 200240, China
{rbzhu, yhyang}@sjtu.edu.cn
[2] School of Automation, Wuhan University of Technology, Luoshi Road 122, Wuhan 430070, China
rongbozhu@gmail.com

Abstract. IEEE 802.11e enhanced distributed coordination function (EDCF) provides a priority scheme by differentiating contention window sizes, medium occupancy limits and arbitrary interframe spaces (AIFS). In this paper, Gibichini's model has been extended to depict EDCF mechanism. The proposed model analyses channel throughput under finite loads, which employs two states to accurately depict the backoff scheme. The proposed model also takes packet arrival rate, contention window size, maximum backoff state and AIFS into consideration. Simulation results proved the accuracy of the proposed model and the enhancement of the EDCF scheme.

1 Introduction

Quality of service (QoS) guarantees over IEEE 802.11 wireless local area networks (WLANs) [1] have been the subject of intensive study in networking literature. A new access mechanism called enhanced distributed coordination function (EDCF) [3] developed by IEEE 802.11 task group e is an enhancement of the access mechanisms of IEEE 802.11. The emerging EDCF is designed to provide differentiated, distributed channel accesses for frames with 8 different priorities by enhancing DCF. As distinct from the legacy DCF, EDCF is a part of a single coordination function, called the Hybrid Coordination Function (HCF), of the 802.11e MAC. All the detailed aspects of the HCF [10] are beyond the scope of this paper as we focus on the HCF contention-based channel access, i.e., EDCF. An 802.11e station shall implement four access categories (ACs), eight priorities and each frame arriving at the MAC with a priority is mapped into an AC. For the priority i class, the minimum backoff contention window size is $CW_{min}[i]$, and the arbitration inter frame space (AIFS) is $AIFS[i]$. For basically, the smaller $CW_{min}[i]$ and$AIFS[i]$, the shorter the channel access delay for the corresponding priority, and the more capacity share for a given traffic condition.

There have been many performance studies and enhancement schemes proposed for IEEE 802.11e EDCF. In [4] an analytical model was proposed, but

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 389–398, 2006.

it just analyzed the throughput performance of a p-persistent version of 802.11 MAC protocol with multiple QoS traffic classes. In [6], an analytical model for EDCF was proposed that conditioned all state transitions in Bianchi's [2] Markov process on the probability that the transition is preempted by a transmission from one or more other stations. A performance model [7] for 802.11e EDCF was proposed which took the post-collision period into account. Zhen etc. z. Kong et al [8] presented a three dimensional discrete time Markov chain model to describe 802.11e EDCF, which took into account the backoff timer, freeze and virtual collision policy. However those models are all under saturation, assuming that each station always has packet to transmit. Such supposition can't reveal the essence of EDCF scheme. The maximum protocol capacity can only be achieved in the non-saturation case [9] for IEEE 802.11 and it is necessary to make clear the performance of IEEE 802.11e EDCF under both saturated case and non-saturated case.

In this paper, focusing on the 802.11e EDCF scheme, an analytical model will be introduced to study the performance of the EDCF under finite load. The paper is organized as follows. Section 2 proposes a performance analytical to evaluate the EDCF scheme. Section 3 validates the accuracy of the proposed model and efficiency of EDCF by simulations. Finally, section 4 concludes the paper.

2 Analytical Model

We assume that packets arrive to AC according to a Poisson process with rate λ packets per slot time. We also assume an ideal channel environment. Considering that each station may have up to four ACs to support eight priorities and one or more priorities are assigned to one AC, in order to simplify the model complexity we assume that only one priority is assigned to each AC in a station, so the concept of AC is equivalent to the idea of priority. Then we can use the term priority throughout our analysis. There are i $(0 \leq i \leq N-1)$ priority classes in a station, where N is the number of service class. Let n_i denote the number of active queues in the priority i class.

2.1 Proposed Analytical Model

Fig. 1 illustrates the discrete time bi-dimensional Markov process we use to model backoff and transmission for a given station in service i class. The 'Idle' state is the state in which the queue does not have any packet to transmit. The 'Ft' state represents the first transmission of a packet after the 'Idle' state if the channel is sensed idle immediately after receiving a packet. Note that the station always performs the backoff procedure after transmitting or dropping a frame, the station could be in the first backoff state while its queue is empty. Hence the post-backoff process $(-1, W_{i,j})$ is added. Let p_i be a transmission from a station in the service i class collides in any time slot and p_b denote the probability that the channel is busy.

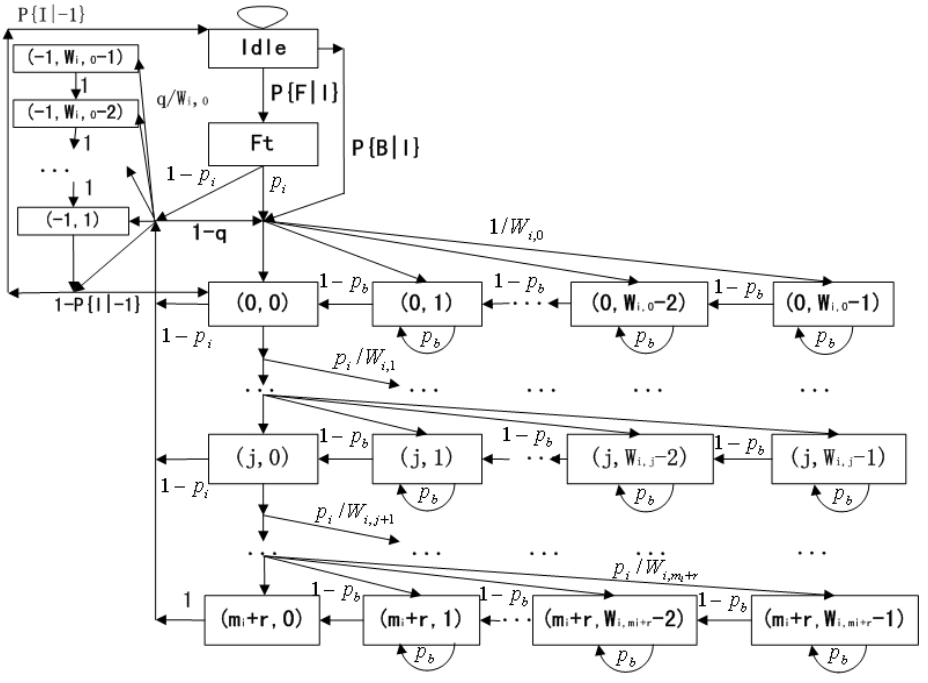


Fig. 1. Markov chain model of backoff window scheme

Following Gibichini's lead, for a given station in service i class, let $b(i,t)$ denote the state of the backoff time counter at the beginning of time slot t. Let $s(i,t)$ be a stochastic process which represents the backoff state j of a station in the service i class at time t, $j \in [0, ..., m_i + r]$, where m_i is the service i class backoff state and r is retransmissions.

We can get the model's non-zero one-step transition probabilities as follows:

$$\begin{cases} P\{j,k|j,k\} = p_b, j \in [0, m_i + r], k \in [1, W_{i,j} - 1] \\ P\{j,k|j,k+1\} = 1 - p_b, j \in [0, m_i + r], k \in [1, W_{i,j} - 1] \\ P\{-1,k|-1,k+1\} = 1, k \in [1, W_{i,0} - 1] \\ P\{0,0|-1,1\} = 1 - p\{-1|I\} \\ P\{0,k|j,0\} = (1-p_i)(1-q)/W_{i,0}, j \in [0, m_i + r - 1], k \in [0, W_{i,0} - 1] \\ P\{0,k|m_i + r, 0\} = (1-q)/W_{i,0}, k \in [0, W_{i,0} - 1] \\ P\{-1,k|j,0\} = (1-p_i)q/W_{i,0}, j \in [0, m_i + r - 1], k \in [1, W_{i,0} - 1] \\ P\{-1,k|m_i + r, 0\} = q/W_{i,0}, k \in [1, W_{i,0} - 1] \\ P\{j,k|j-1,0\} = p_i/W_{i,j}, j \in [1, m_i], k \in [0, W_{i,j} - 1] \end{cases} \quad (1)$$

where q is the probability that the station enters backoff with exactly one packet to transmit and no new packets arrive until the packet is transmitted successfully. $P\{F|I\}$ is the probability that the channel is sensed idle after receiving a packet from the 'Idle' state. $P\{B|I\}$ is the probability that the channel is sensed busy

after receiving a packet from the 'Idle' state. $P\{I|-1\}$ is the probability that station enters into 'Idle' state from the post-backoff state.

For the chain regularities, we can get the followings relations:

$$b_{i,j,k} = \begin{cases} p_i^j b_{i,0,0}, k = 0, j \in [0, m_i + r] \\ \frac{W_{i,j}-k}{W_{i,j}(1-p_b)} b_{i,j,0}, k \in [1, W_{i,j}-1], j \in [1, m_i + r] \\ \frac{W_{i,0}-k}{W_{i,0}} q[(1-p_i) \sum_{j=0}^{m_i+r-1} b_{i,j,0} + b_{i,m_i+r,0} + P_{Ft}(1-p_i)], \\ j = -1, k \in [1, W_{i,0}-1] \end{cases} \tag{2}$$

Using the normalization condition:

$$\sum_{j=0}^{m_i+r} \sum_{k=0}^{W_{i,j}-1} b_{i,j,k} + \sum_{k=1}^{W_{i,0}-1} b_{i,-1,k} + P_I + P_{Ft} = 1 \tag{3}$$

When $j \in [m_i + 1, m_i + r]$, the contention window $W_{i,j}$ does not increase but stay at the value $2^{m_i} W_{i,0}$. Taking (2) into the equation (3), we can get:

$$\begin{aligned} b_{i,0,0} = (1-p_b)(1-p_i)(1-2p_i)[2(1-P_I-P_{Ft}) - qP_{Ft}(1-p_i)(W_{i,0}-1)] \div \\ \{(1-p_b)(1-p_i)(1-2p_i)(W_{i,0}-1) + W_{i,0}(1-p_i)[1-(2p_i)^{m_i+1}] + \\ 2^{m_i} W_{i,0}(p_i^{m_i+1} - p_i^{m_i+r+1})(1-2p_i) + (1-2p_i)(1-p_i^{m_i+r+1})\} \end{aligned} \tag{4}$$

Let τ_i denote the probability that a station subscribing the service i class transmits a packet in a slot time. With (2) and (3) we have:

$$\tau_i = \sum_{j=0}^{m_i+r} b_{i,j,0} + P_{Ft}, 0 \le i \le N-1 \tag{5}$$

where $b_{i,0,0}$ can be obtained from (4).

Then we can get:

$$p_b = 1 - \prod_{j=0}^{N-1} (1-\tau_j)^{n_j}, 0 \le i \le N-1 \tag{6}$$

$$p_i = 1 - (1-\tau_i)^{n_i - 1} \prod_{j=0, j \ne i}^{N-1} (1-\tau_j)^{n_j}, 0 \le i \le N-1 \tag{7}$$

The set of equations (5) and (7) represent a nonlinear system of equations with $2N$ unknowns τ_i and p_i. Assuming knowledge of P_{Ft}, $P\{B|I\}$,$P\{I|-1\}$ and P_I, it can be proved that they have a unique solution and can be solved by using numerical techniques.

2.2 Computation of Unknown Probabilities

The probability that station enters into 'Idle' state from post-backoff state:

$$P\{I|-1\} = e^{-\lambda(W_{i,0}+1)\overline{\varepsilon}/2} \tag{8}$$

where $\overline{\varepsilon}$ is the average time between successive counter decrements:

$$\overline{\varepsilon} = (1-p_b)\sigma + p_b p_i (T_{s,i}+\sigma) + p_b(1-p_i(T_{c,i}+\sigma)) \tag{9}$$

where $T_{s,i}$ is the average length of successful slots. $T_{c,i}$ is the average lengths of failed slots. σ is one system slot time.

The probability q, as an approximation, we assume that the probability of entering backoff with one packet is one. Then we have:

$$q = e^{-\lambda E(X)} \tag{10}$$

where $E(X)$ is the expected value of the time that the packet spends in backoff before successfully transmitting the packet. We can get:

$$E(X) = \frac{\overline{\varepsilon}(\sum_{j=0}^{m_i+r} W_{i,j}p_i^j)}{2} + (\frac{2T_{C,i}p_i - \overline{\varepsilon}}{2(1-p_i)})(1-p_j^{m+r+1}) \tag{11}$$

$$P\{B|I\} = p_b p_i (1-e^{-\lambda T_{s,i}}) + p_b(1-p_i)(1-e^{-\lambda T_{c,i}}) \tag{12}$$

$$P\{F|I\} = (1-p_b)(1-e^{-\lambda\sigma}) \tag{13}$$

$$P_I = \frac{qP\{I|-1\}[(1-p_i)P_{Ft} + (1-p_i)\sum_{j=0}^{m_i-1+r} b_{i,j,0} + b_{m_i+r}]}{p_b p_i(1-e^{-\lambda T_{s,i}}) + p_b(1-p_i)(1-e^{-\lambda T_{c,i}}) + (1-p_b)(1-e^{-\lambda\sigma})} \tag{14}$$

$$\begin{aligned} P_{Ft} = {} & \{qP\{I|-1\}[(1-p_i)P_{Ft} + (1-p_i)\sum_{j=0}^{m_i-1+r} b_{i,j,0} + b_{m_i+r}]\} \div \\ & \{p_b p_i(1-e^{-\lambda T_{s,i}}) + p_b(1-p_i)(1-e^{-\lambda T_{c,i}}) + \\ & (1-p_b)(1-e^{-\lambda\sigma})\} \times (1-p_b)(1-e^{-\lambda\sigma}) \end{aligned} \tag{15}$$

2.3 Throughput and Delay

Let S_i denote the normalized throughput for the priority i class. Let $P_{tr}(i)$ be the probability that there is exactly one transmission from the tagged station in the service i class in the considered slot time. Let $p_{s,i}$ denote the probability that a successful transmission occurs in a slot time for the priority i class. We have:

$$P_{tr}(i) = \tau_i(1-\tau_i)^{n_i-1}\prod_{j=0,j\neq i}^{N-1}(1-\tau_j)^{n_j} = \frac{\tau_i}{1-\tau_i}(1-p_b) \tag{16}$$

$$p_{s,i} = n_i P_{tr}(i) \tag{17}$$

$$S_i = \frac{p_{s,i}E(L)}{(1-p_b)\sigma + p_{s,i}T_{s,i} + (p_b - p_{s,i})T_{c,i}} \tag{18}$$

where L denotes payload size.

Because the length of the transmitted frame determines the access mechanism, packets are transmitted by means of the RTS/CTS mechanism if their payload sizes exceed a given threshold. Otherwise, the basic access method is used to transmit the packets. So the values of $T_{s,i}, T_{c,i}$ can be calculated as follows:

$$\begin{cases} T_{s,i}^{bas} = H + l + ACK + 2\delta + SIFS + AIFS[i] \\ T_{s,i}^{rts} = H + l + RTS + CTS + ACK + 4\delta + 3SIFS + AIFS[i] \end{cases} \tag{19}$$

$$\begin{cases} T_{c,i}^{bas} = H + l + AIFS[i] + \delta \\ T_{c,i}^{rts} = RTS + AIFS[i] + \delta \end{cases} \tag{20}$$

where H is the time to transmit the header including MAC header and PHY header, l is the time to transmit the payload, $ACK$is the time to transmit the packet ACK , $AIFS[i]$ is the time of AIFS of priority i class , RTS is the time to transmit the packet RTS, CTS is the time to transmit the packet CTS and δ is the time of the propagation delay.

Let B_i represent the number of collisions before transmitting a frame for the priority i class. We can get:

$$E(B_i) = \frac{p_b(1-\tau_i)}{n_i\tau_i(1-p_b)} - 1 \tag{21}$$

Let NF_i represent the time interval during which the counter reaches zero without considering the case when the counter freezes. We have:

$$E(NF_i) = \sum_{j=0}^{m_i+r} \sum_{k=0}^{W_{i,j}-1} kb_{i,j,k} \tag{22}$$

Let BF_i denote the time that the backoff counter of a station freezes and C_{BF_i} denote the number of times that the backoff counter freezes respectively, for the priority i class. We can get:

$$E(C_{BF_i}) = \frac{E(NF_i)}{\max(1,(1-p_b)/p_b)} - 1 \tag{23}$$

$$E(BF_i) = E(C_{BF_i})[\frac{n_i\tau_i(1-p_b)}{p_b(1-\tau_i)}T_{s,i} + (1 - \frac{n_i\tau_i(1-p_b)}{p_b(1-\tau_i)})T_{c,i}] \tag{24}$$

For the frame delay of the priority i class D_i, we can get:

$$\begin{aligned} E(D_i) = {} & E(B_i)[T_{c,i} + SIFS + T_{ACK\_out}] + \\ & T_{s,j} + [E(NF_i) + E(BF_i)](1 + E(B_i)) \end{aligned} \tag{25}$$

3 Numerical and Simulation Results

The well-known simulation tool NS2 [5] is used to validate the proposed model and performance of EDCF scheme. In the simulation system, all stations content

Table 1. System parameters

Parameter	Value
Channel rate	11 Mbps
Frame payload	1000 bytes
MAC header	272 bits
PHY header	192 bits
ACK, CTS	112 bits + PHY header
RTS 160	bits + PHY header
RetryLimit	7
Propagation delay	1μ s
Slot time	20μ s
SIFS	1 slot time
$AIFS[3]$	SIFS + 1 slot time
$AIFS[1]$	SIFS + 2 slot time

to transmit packets to a single non-transmitting access point. Parameters used in analytical model and simulations are as follows in Table 1. Following simulation scenarios just consider the RTS/CTS mode and each station has one priority 3 class and one priority 1 class traffic.

For the limitation of space, following simulation just concentrates on the effect of packet arrival rate. Assumption each priority class has the same packet arrival rate λ. G is the offered load of all priority classes in units of data packets per slot time. It is obvious that $G = 2N\lambda$. Figs. 2 and 3 show the simulation and numerical results of the normalized throughput for priority 3 class and priority 1 class at the case station number is 10 and 20 respectively with G increasing.

From Figs. 2 and 3 we can see with the total offered load increasing, throughput of each priority class also increases at the same time. It is obvious that priority 3 class has high priority to access channel, so the throughput of priority 3 class increases more quickly than that of priority 1 class. When total offered

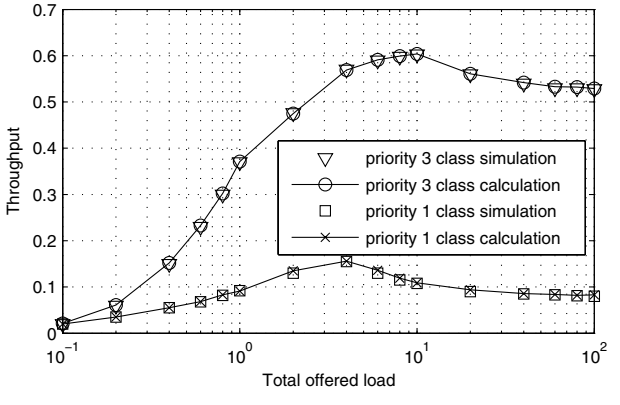


Fig. 2. Normalized throughput with 20 stations against total offered load

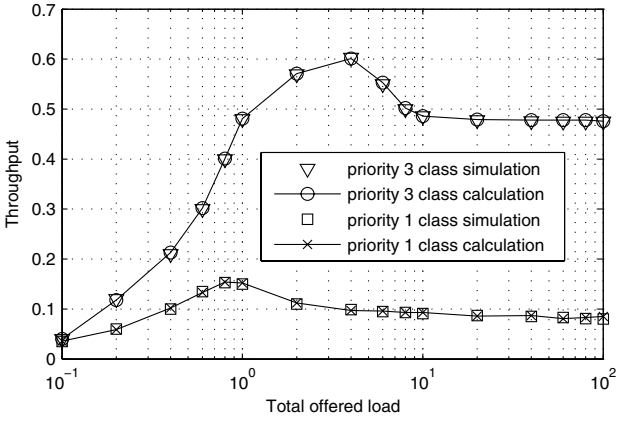


Fig. 3. Normalized throughput with 10 stations against total offered load

load is larger than certain value, throughput of each priority class does not increase but decrease, especially when the load is heavy. Then with the offered load increasing, throughput of each priority keeps constant when the system comes into saturation state. So for the system the maximal throughput appears at unsaturation case not at saturation case. At the same time, for priority 1 class, the maximal throughput appears earlier than priority 3 class, for priority 3 class has higher priority than priority 1 class. When there are packets to transmit, priority 1 class must wait, while priority 3 class gets the opportunity to transmit packet.

Figs. 4 and 5 show the numerical and simulation results for packet delay at 10 and 20 stations cases. With the total offered load increasing, the packet delay of each priority class increases slowly for the load is light. Meanwhile, the packet delay of priority 3 class increases less than that of priority 1 class for the former has higher priority and has more opportunity to access channel. When the total offered load large than certain values, the packet delay increases dramatically.

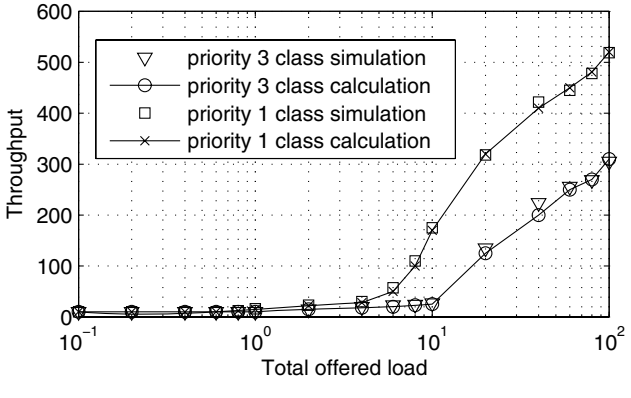


Fig. 4. Packet delay with 20 stations against total offered load

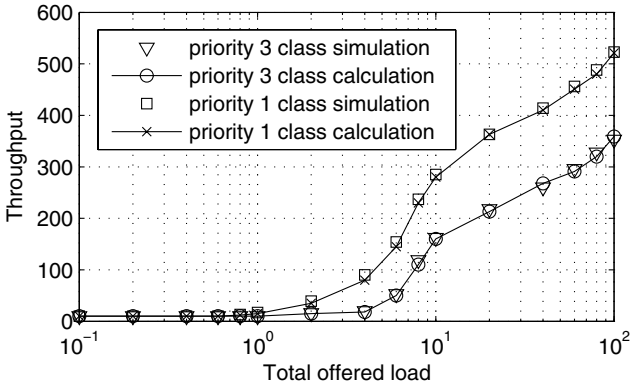


Fig. 5. Packet delay with 10 stations against total offered load

With the offered load increases, the load comes to heavy, which leads to more packet collisions and larger packet delay. The results of Fig. 4 and Fig. 5 are also coincident with those in Fig. 2 and Fig. 3.

4 Conclusion

In this paper, an analytical model is proposed to evaluate the performance of the EDCF scheme under finite load, which take the contention window size, number of backoff states, AIFS and packet arrival rate into consideration. Based on the proposed model we have studied the effects of packet arrival rate on throughput and delay performance for multiclass priority traffic. The validities of the model and EDCF scheme are shown by simulation. An important observation is that the maximal throughput appears at unsaturation case rather than saturation case. The future work will focus on studying the essence of relationship between throughput and collision probability, optimizing access parameters such as AIFS and contention window value.

References

1. IEEE Std. 802.11-1999, PART 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Reference number ISO/IEC 8802-11:1999(E), IEEE Std.802.11, 1999 edition, 1999
2. Giuseppe Bianchi: Performance analysis of the IEEE 802.11 Distributed Coordination Function. IEEE Journal on Selected Areas in Communications, vol.18, pp. 535-547, Mar. 2000
3. IEEE 802.11 WG, draft Supplement to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS), IEEE 802.11e/Draft 4.1, Februray 2003

4. Ye Ge, Jennifer Hou: An analytical Model for Service Differentiation in IEEE 802.11. IEEE International Conference on Communications, v2, 2003, pp. 1157-1163
5. NS2, URL: http://www-mash.cs.berkeley.edu/ns
6. Y. Xiao: Performance Analysis of IEEE 802.11e EDCF under Saturation Condition. IEEE International Conference on Communications, v1, 2004, pp. 170-174
7. Jeffrey W., Tejinder S.: Saturation Throughput Analysis of IEEE 802.11e Enhanced Distributed Coordination Function. IEEE Journal on Selected Areas in Communications, vol. 22, pp. 917-928, JUNE 2004
8. Z. Kong, Danny H.K.T, Brahim B.: Performance Analysis of IEEE 802.11e Contention-Based Channle Access. IEEE Journal on Selected Areas in Communications, vol.22, pp. 2095-2106, December 2004
9. H. Zhai, X. Chen, and Y. Fang: How Well Can the IEEE 802.11 Wireless LAN Support Quality of Service. IEEE Transaction on Wireless Communications, v4, n6, November 2005, pp. 3084-3094
10. IEEE Std 802.11e, specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. November 2005

Opportunistic Packet Scheduling over IEEE 802.11 WLAN

Sung Won Kim

School of Electrical Engineering and Computer Science, Yeungnam University, Gyeongsangbuk-do, 712-749, Korea
ksw@ieee.org

Abstract. This paper introduces an opportunistic packet scheduling method and medium access control (MAC) scheme for controlling the throughput in wireless local area networks (WLANs). The proposed method takes advantage of the multi-user diversity in time-varying wireless channel while the asymmetric traffic load problem between the uplink and the downlink is alleviated. The proposed method can be implemented without the modification of the deployed IEEE 802.11 nodes. The performance of the proposed method is compared with IEEE 802.11 Distributed Coordination Function (DCF) by computer simulations.

1 Introduction

Medium Access Control (MAC) protocol in the IEEE 802.11 standard [1] consists of two coordination functions: mandatory Distributed Coordination Function (DCF) and optional Point Coordination Function (PCF). In the DCF, a set of wireless nodes communicates with each other using a contention-based channel access method, namely Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). CSMA/CA is known for its inherent fairness between nodes and robustness. It is quite effective in supporting symmetric traffic loads in ad hoc networks where the traffic loads between nodes are similar.

For infrastructure wireless local area network (WLAN) applications such as hotspots [2]–[4], the system consists of N users communicating to a common entity (e.g., an access point, AP). The nature of independent time-varying channels across different users in a multi-user wireless system provides multi-user diversity. This particular form of diversity could be exploited by tracking the channel fluctuations between each user and the AP, and scheduling transmissions to users when their instantaneous channel quality is near maximum. It is observed that the probability of successful packet transfer increases significantly when the channel state information is exploited opportunistically [5].

Opportunistic scheduling, used to extract multi-user diversity gain, was first proposed in [6] and then extended to many wireless communication systems [7]. An opportunistic scheduling algorithm that exploits the inherent multi-user diversity has been implemented as the standard algorithm in the third-generation cellular system IS-856 [8] (also known as high data rate, HDR). To enable the opportunistic multi-user communications, timely channel information of each

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 399–408, 2006.

link is required for an effective scheduling. Just as all the previous schemes have assumed, the exploitation of timely channel information is possible in cellular networks where the base station acts as a central controller and control channels are available for channel state feedback.

It is difficult to utilize the multi-user diversity in WLANs. The AP cannot track the channel fluctuations of each link because of the single shared medium and the distributed CSMA/CA MAC protocol. The opportunistic packet scheduling methods for WLANs are presented in [9]–[11]. The key mechanism of the method is the use of multicast RTS (Request-To-Send) and priority-based CTS (Clear-To-Send) to probe the channel status information. Since their methods require the modification of RTS and CTS in the standard, the scheme cannot be directly applied into widely deployed IEEE 802.11 typed WLANs.

On the other hand, this form of the multi-user wireless system produces asymmetric traffic loads where most of the traffic loads converge into the AP. For example, Internet access or mobile computing uses transmission control protocol (TCP) or user datagram protocol (UDP) in which the offered traffic load is strongly biased toward the downlink (from AP to nodes) against the uplink (from nodes to AP) or the direct link (from nodes to nodes). Thus, these traffic flows for the downlink are completely blocked due to the CSMA/CA MAC protocol in distributed environments.

In this paper, we propose an enhanced WLAN MAC protocol which alleviates the bottleneck problem by using the opportunistic packet scheduling. The remainder of this paper is organized as follows. The next section reviews the background of the IEEE 802.11 system operation. Section 3 describes the proposed method. In Section 4, we investigate the enhancement of the proposed method with some numerical results. Finally, the paper is concluded in Section 5.

2 Background

2.1 Review of IEEE 802.11 DCF

The DCF achieves automatic medium sharing between compatible nodes through the use of CSMA/CA. Before initiating a transmission, a node senses the channel to determine whether or not another node is transmitting. If the medium is sensed idle for a specified time interval, called the distributed interframe space (DIFS), the node is allowed to transmit. If the medium is sensed busy, the transmission is deferred until the ongoing transmission terminates.

Each node generates a random backoff timer chosen uniformly from the range [0, w-1], where w is referred to as the contention window. At the first transmission attempt, w is set to w_{min} (minimum contention window). After the backoff timer reaches 0, the node transmits a short RTS message. If the RTS is successfully received, the receiving node responds with a CTS message. Any other node which hears either the RTS or CTS message uses the data packet length information to update its Network Allocation Vector (NAV) containing the information of the period for which the channel will remain busy. Thus, all nodes including hidden node can defer transmission appropriately to avoid the packet collision.

An acknowledgement (ACK) packet will be sent by the receiver upon successful reception of a data packet. It is only after receiving an ACK packet correctly that the transmitter assumes successful delivery of the corresponding data packet. If there is no response of ACK or CTS packet, a binary exponential backoff scheme is used. After each unsuccessful transmission, the value of w is doubled, up to the maximum value w_{max}.

Short InterFrame Space (SIFS), which is smaller than DIFS, is a time interval between RTS, CTS, data packet, and ACK. Using this small gap between transmissions within the packet exchange sequence prevents other nodes from attempting to use the medium. As a consequence, it gives priority to completion of the ongoing packet exchange sequence.

2.2 Multi-rate in IEEE 802.11

Multi-rate in IEEE 802.11 provides physical-layer mechanism to transmit at higher data rates than the basic rate if the channel conditions permit. The first commercial implementation that exploits this multi-rate capability is called Auto Rate Fallback (ARF) [12]. With ARF, transmitters use the history of previous transmission error rates to adaptively select the next transmission rate. That is, after a number of consecutive successful transmissions, the transmitter changes its modulation scheme to attempt the transmission at a higher rate, and vice versa after consecutive losses. Consequently, if a mobile user has a perpetually high-quality channel, the user will eventually transmit at higher data rates while accessing the medium according to the same IEEE 802.11 MAC.

An enhanced protocol to exploit the multi-rate capability of IEEE 802.11 named Receiver Based Auto Rate (RBAR) is proposed in [13]. The key idea of RBAR is for receivers to control the transmitter's transmission rate. RBAR uses physical-layer analysis of the received RTS message to determine the maximum possible transmission rate for a particular bit error rate. The receiver inserts this rate into a special field of the CTS message to inform the transmitter and other overhearing nodes of the potentially modified rate. This message is called reservation-subheader (RSH) and is inserted in the header of the data packet. With the RSH message, overhearing nodes can modify their NAV values to the new potentially decreased transmission time. In this way, RBAR quickly adapts to channel variations and extracts significant throughput gains as compared to ARF.

Any one of the previous multi-rate methods can be merged into the proposed method as will be explained in the next section. However, our design goal is that the deployed nodes need not to be modified. Thus, we propose that the transmitter estimates the wireless channel quality by using the SNR of the received CTS message.

3 Opportunistic Packet Scheduling

3.1 Channel Access Method

For the uplink channel, each node transmits data packets by using the DCF mechanism. For the downlink channel, we propose a collision-free channel access

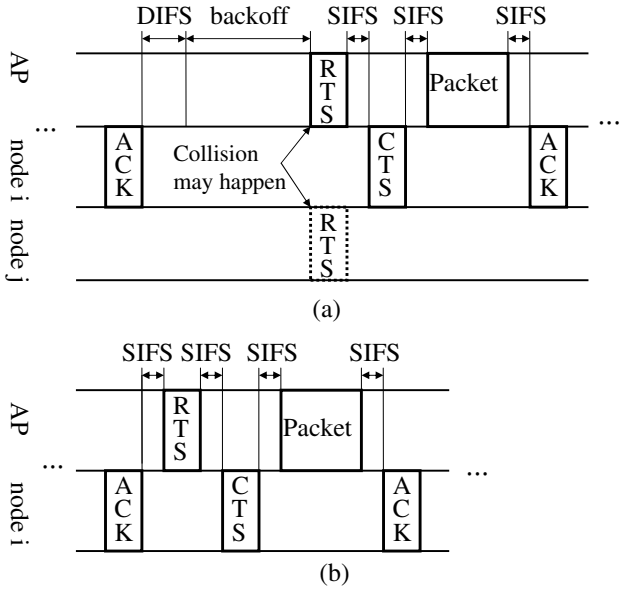


Fig. 1. Downlink channel access method (a) DCF and (b) CFCA

(CFCA) mechanism in addition to the DCF mechanism. That is, AP can select different channel access mechanism between two, DCF and CFCA, for each data packet transmission.

The DCF mechanism is illustrated in Fig. 1(a), where the next channel access should wait for DIFS and backoff window time after the previous ACK packet. A two-way handshaking technique without RTS/CTS handshaking called basic access mechanism is not considered in this paper although our proposed method can be easily extended to the basic access mechanism. The CFCA mechanism is illustraed in Fig. 1(b). In this method, the AP waits only for SIFS time instead of DIFS and backoff time. By shortening the waiting time, the AP can access the channel without collision because all other nodes should wait at least DIFS time which is longer than SIFS time. The more the AP selects the CFCA as the channel access method, the more throughput is allocated to the downlink because neither packet collision nor backoff occurs during the CFCA.

To limit the throughput superiority of the downlink channel caused by the CFCA, the selection algorithm between the two channel access mechanisms is required. That is, the frequency of the CFCA should be limited. We propose that the AP determines the channel access method depending on the history of the previous successful data packet transmissions to adaptively select the next channel access method. Let γ denote the required throughput ratio between the uplink and the downlink. For the implementation of the proposed method, AP keeps track of the successful packet transmissions. If the downlink transmits more data packets than γ, AP selects the DCF to give more throughput to the uplink. If the downlink transmits less data packets than γ, AP selects the CFCA

to get more chance of packet transmission through the downlink. The value of γ can be set to any values and we set it to one except otherwise specified.

3.2 Packet Queue Management and Scheduling in AP

Each node can directly communicate only with the AP (uplink or downlink), since we focus on the AP-coordinated wireless network. The AP manages the downlink packet queues for each node as shown in Fig. 2. During the DCF, the packet scheduling algorithm adopts the first-in first-out (FIFO) algorithm. During the CFCA, the AP schedules the packet based on the channel quality. The link with better channel quality is given higher priority in packet transmission. In order to track the latest channel quality, it is necessary to send the control packet to the node. However, this method will increase the overhead and need the modification of the IEEE 802.11 standard. Our design goal is that the scheduling method can be implemented without the modification of the nodes already deployed in the system. Thus, we propose that the AP updates the channel quality of each link after every successful packet transmissions. The channel quality is reported from the physical layer by measuring the SNR of the CTS and ACK control packets as explained in the previous section. This estimation of the channel quality may not be the timely information. However, the estimation error is in the acceptable range as will be shown in the next section. Moreover, the proposed method can be implemented without the modification of the deployed nodes.

The AP lists all the communication links according to the channel quality. When the AP selects the CFCA mechanism for the channel access method, the link that recorded the best channel quality in the previous successful packet transmission is given the first chance to transmit the packet in the queue. When there is no packet in the queue for that link, the next good-quality link is given the second chance to transmit the packet.

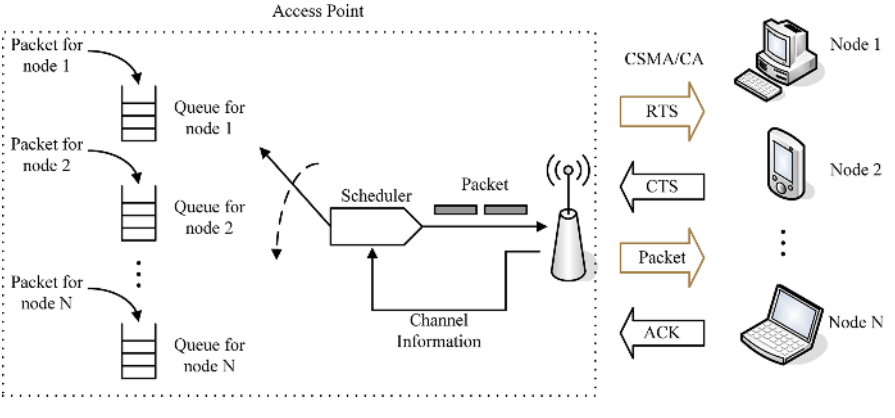


Fig. 2. Queue management in AP

Most functions of the IEEE 802.11 have been integrated in a single integrated circuit (IC) [14]. The operations of the channel access and the packet scheduling are performed by the IC. The proposed method can be implemented with a small increase of logic gates such as counters and comparators in the IC. Thus, the implementation of the proposed method does not require additional delay or computing power. The modification of the IC is required only for the AP and is not required for the nodes.

4 Numerical Results

We evaluate the performance of the proposed method, named opportunistic packet scheduling (OPS), by computer simulations. The IEEE 802.11 DCF is compared with the OPS. The parameter values used to obtain the numerical results of the simulation runs are summarized in Table 1. The values of these parameters are based on the IEEE 802.11b direct sequence spread spectrum (DSSS) standard [1].

Table 1. Parameter values

Parameter	Value
w_{min}	32
w_{max}	1024
SIFS time	10 μs
PIFS time	30 μs
DIFS time	50 μs
slot time	20 μs
MAC header	272 bits
PHY header	48 bits
Preamble	144 μs
ACK time	304 μs
RTS time	352 μs
CTS time	304 μs

To reflect the fact that the surrounding environmental clutter may be significantly different for each pair of communication nodes with the same distance separation, we use the log-normal shadowing channel model [15]. The path loss PL in dB at distance d is given as

$$PL(d) = PL(d_0) + 10n \log(d/d_0) + X_\sigma, \tag{1}$$

where d_0 is the close-in reference distance, n is the path loss exponent, and X_σ is a zero-mean Gaussian distributed random variable with standard deviation σ. We set n to 3.25 and σ to 5.2 according to the result of measurements for an office building model [15]. To estimate $PL(d_0)$, we use the Friis free space equation

$$P_r(d_0) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d_0^2 L}, \tag{2}$$

where P_t and P_r are the transmit and receive power, G_t and G_r are the antenna gains of the transmitter and receiver, λ is the carrier wavelength, and L is the system loss factor which is set to 1 in our simulation. Most of the simulation parameters are drawn from the data sheet of Cisco 350 client adapter. The received power is

$$P_r(d) = P_t - PL(d). \tag{3}$$

The minimum received power level for the carrier sensing is set to -95 dBm, which is the noise power level. The long-term SNR is

$$SNR_L = P_t - PL(d) - n + PG, \tag{4}$$

where n is the noise power set to -95 dBm and PG is the spread spectrum processing gain given by

$$PG = 10 \log_{10} \frac{C}{S}, \tag{5}$$

where C is the chip rate and S is the symbol rate. Since each symbol is chipped with an 11-chip pseudonoise code sequence in the IEEE 802.11 standard, PG is 10.4 dB. The received SNR is varied by the Ricean fading gain δ. Under this model, the SNR of the received signal is

$$SNR = 20 \log_{10} \delta + SNR_L. \tag{6}$$

For the data rate in the physical layer for each communication link, we assume that the system adapts the data rate by properly choosing one from a set of modulation scheme according to the channel condition. The set of modulation schemes used in our simulation studies are BPSK, QPSK, 16QAM, 64QAM, and 256QAM. For the simplicity, we ignore other common physical layer components such as error correction coding.

We assume that all nodes except the AP are randomly distributed in the circle area with a diameter of 150 meters and move randomly at a speed of 0.1 m/sec. The AP is located at the center of the area. To evaluate the maximum performance, traffic load is saturated and the destination addresses of the packets are the AP in each node. In the AP, there are N connections, each for one node, and packets are generated for each connection with the same distribution as those in each nodes. To make an asymmetric traffic load condition between the uplink and the downlink, the size of the downlink and the uplink packets are 1024 and 64 bytes, respectively. The number of node is set to 25. The effects of the number of nodes on the performance is evaluated by the computer simulation.

The system throughput of the proposed method is compared with the DCF in Fig. 3. In the DCF, the system throughput decreases as the number of nodes increases. This decrease of the system throughput mainly comes from the increased collision between the packet transmissions. It is noted that the probability of the packet collision increases as the number of nodes increases. On the contrary, the

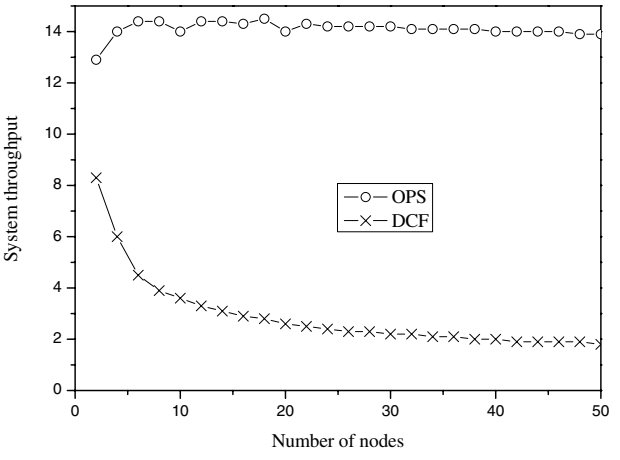


Fig. 3. System throughput versus the number of nodes

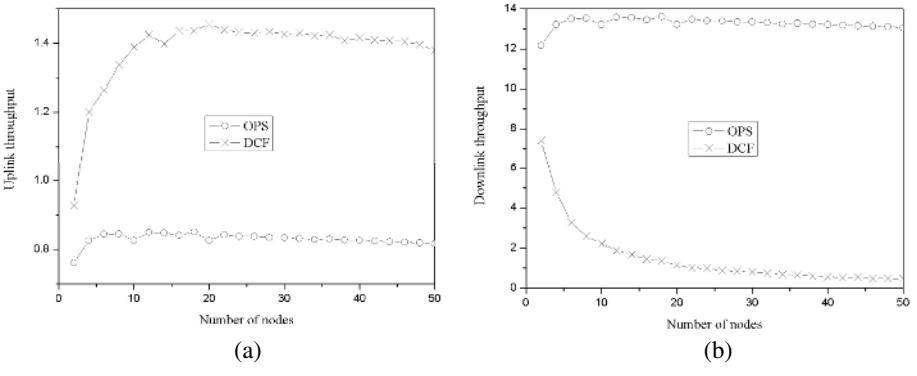


Fig. 4. (a) Uplink and (b) downlink throughput versus the number of nodes

OPS maintains a constant system throughput because it provides contention-free channel access method for the AP. Moreover, the OPS enjoys more system throughput because higher data rate is provided for the packet transmission during the CFCA.

The uplink throughput is shown in Fig. 4(a). The uplink throughput of the DCF increases as the the number of nodes increases. For a large number of nodes, the uplink throughput is saturated because of the increased number of collisions and backoff mechanism. The OPS provides constant throughput to the uplink compared with the DCF because the throughput is controlled by the selection algorithm of the two channel access methods. The downlink throughput is shown in Fig. 4(b). The decrease of the downlink throughput is proportional to the number of nodes in the DCF. Note that system throughput decreases as the number of nodes increases, which is explained in Fig. 3. It is shown

that the OPS provides larger throughput to the downlink and can mitigate the bottleneck problem in the asymmetric traffic load condition. The throughput of the OPS is constant because of the same reason as in Fig. 4(a). Because we assign asymmetric traffic load between the uplink and the downlink, the downlink is allocated more throughput than the uplink in the simulation.

5 Conclusion

We have proposed an opportunistic packet scheduling method to alleviate the throughput unbalance between the uplink and the downlink and to enhance the system throughput of the IEEE 802.11 DCF. The proposed method also reduces the probability of the data packet collision. The proposed method can be implemented without the modification of the IEEE 802.11 standard for nodes that are widely deployed.

The efficiency of the proposed system has been demonstrated by the computer simulation. The results show that the proposed method enhances the system throughput for asymmetric traffic load. This, in turn, drastically reduces the blocking probability of the multimedia data packets in the proposed systems compared with that in the IEEE 802.11 DCF.

References

1. IEEE Std 802.11b-1999: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band. (1999)
2. Honkasalo, H., Pehkonen, K., Niemi, M.T., Leino, A.T.: WCDMA and WLAN for 3G and beyond. IEEE Wireless Commun. Mag. (2002) 14–18
3. Doufexi, A., Tameh, E., Nix, A., Armour, S.: Hotspot wireless LAN to enhance the performance of 3G and beyond cellular networks. IEEE Commun. Mag. (2003) 58–65
4. Kishore, S., Greenstein, L.J., Poor, H.V., Schwartz, S.C.: Uplink user capacity in a CDMA macrocell with a hotspot microcell: Exact and approximate analyses. IEEE Trans. Wireless Commun. **2** (2003) 364–374
5. Wang, H., Mandayam, N.B.: Opportunistic file transfer over a fading channel under energy and delay constraints. IEEE Trans. Commun. **53** (2005) 632–644
6. Knopp, R., Humblet, P.A.: Information capacity and power control in single cell multiuser communications. In: Proc. IEEE ICC 1995. (1995) 331–335
7. Ajib, W., Haccoun, D.: An overview of scheduling algoriths in MIMO-based fourth-generation wireless systems. IEEE Network **19** (2005) 43–48
8. IS-856: CDMA 2000 standard: High rate packet data air interface specification. (2000)
9. Wang, J., Zhai, H., Fang, Y.: Opportunistic packet scheduling and media access control for wireless LANs and multi-hop ad hoc networks. In: Proc. IEEE WCNC'04, Atlanta, GA, USA (2004) 1234–1239
10. Cateura, A., Alonso, L., Verikoukis, C.V.: Opportunistic scheduling for WLAN systems using cross-layer techniques and a distributed MAC. In: Proc. IEEE VTC-2005-Fall, Dallas, TX, USA (2005) 221–224

11. Sadeghi, B., Kanodia, V., Sabharwal, A., Knightly, E.: OAR: An opportunistic auto-rate media access protocol for ad hoc networks. ACM Wireless Networks **11** (2005) 39–53
12. Kamerman, A., Monteban, L.: WaveLAN-II: A high-performance wireless LAN for the unlicensed band. Bell Labs Tech. J. **2** (1997) 118–133
13. Holland, G., Vaidya, N., Bahl, P.: A rate-adaptive MAC protocol for multi-hop wireless networks. In: Proc. IEEE/ACM MOBICOM'01, Boston, MA, USA (2001) 236–251
14. http://www.realtek.com.tw/: (Realtek)
15. T. S. Rappaport: Wireless communications: principles and practices, 2nd Ed. Prentice Hall (2002)

A Scalable, Efficient and Reliable Routing Protocol for Wireless Sensor Networks

Peter Kok Keong Loh

School of Computer Engineering, Nanyang Technological University,
Nanyang Avenue, Singapore 639798
askkloh@ntu.edu.sg

Abstract. The nodes in wireless sensor networks are energy-constrained and have limited bandwidth. In addition, communication links between nodes are unpredictable and unstable. These factors make the design of routing protocols for such a network very challenging. We propose a scalable, efficient and reliable routing protocol called EAR that provides reliable data delivery from nodes to hubs in a wireless sensor network. Simulation results have shown EAR outperforming existing routing protocols.

1 Introduction

Wireless sensor network (WSN) is an emerging technology that enables development of applications such as surveillance, large scale environmental monitoring and intrusion detection systems by deploying large number of nodes that are small and cost-effective [1-5]. Nodes are equipped with radio frequency (RF) transceivers for wireless communications, micro-controllers for data processing and sensors for capturing physical phenomenon. The nodes are typically battery-powered and no larger than a few cubic centimetres. In most applications, WSN nodes are deployed in an ad hoc manner with no pre-defined topology - nodes form a network with multi-hop topology and every node has to perform sensing activity and also take on the role of router by forwarding packets to hubs for other nodes.

Reliable delivery of data with low latency and minimum energy consumption are essential for WSN applications to achieve their objectives. With this in mind, we propose a scalable, efficient and reliable data-centric routing protocol (EAR) that routes packets to hubs reliably using a light weight mechanism that requires minimal control packets to handle a changing topology caused by unreliable RF links and failure of nodes. Simulation results have shown that EAR delivers more packets than some existing routing protocols with lower latency and with less energy. The remainder of this paper is organised as follows. Section 2 presents the challenges faced by routing protocols for WSN. Section 3 highlights our motivation and the contribution of our work. Section 4 describes our proposed routing protocol in details. Simulation results are presented and discussed in Section 5. Finally, Section 6 concludes this paper.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 409 – 418, 2006.

2 Challenges of WSNs

Energy Consumption: as nodes are battery-operated, the energy is limited and replacing flat batteries is not feasible in many situations. The routing protocols must consume the minimum amount of energy to extend the network lifetime [6].
Node Failures: nodes deployed in harsh environments are prone to hardware failure. The routing protocol must be able to react to the change in topology quickly due to failure of some nodes and reduce the impact on network performance to a minimum.
Unreliable RF Links: unpredictable and unstable wireless transmission is one of the major problems. The low-powered RF transceiver used by nodes worsens the problem. Consequences are high packet loss and error rates and intermittent communication disruptions. Routing protocols must operate under such harsh conditions to achieve efficient and reliable data delivery.
Low Bandwidth: RF transceivers used typically offer a bandwidth in the order of tens to hundreds of kilobits per second. This constrains the amount of traffic generated in the WSN. Hence, routing protocols must minimise the amount of control packets to leave more of the limited bandwidth for data packets.
Scalability: One prominent feature of WSN is the deployment of large numbers of nodes. Routing protocols designed for WSN must therefore be scalable by maintaining consistent performance with increase in network size.

3 Motivation

WSNs have practical benefits that will improvise the quality of life and also productivity and efficiency in various application scenarios. To realise these benefits, network nodes must be supported by an efficient and reliable communication system. At the heart of this communication system is the routing protocol responsible for the dissemination of data. Our work has two contributions to the field of WSN: Firstly, we proposed a new data-centric routing protocol for WSN that can achieve the performance requirements of a WSN even under adverse operating constraints and conditions. Our work serves as a framework for further research and study. Secondly, we analysed the simulated performance of similar routing protocols in a typical WSN and compared the performance of our routing protocol based on standard metrics.

4 Protocol Details

4.1 Framework

EAR adopts a data-centric routing strategy that routes packets according to the type of data it carries. Data generated by nodes that is of interest to network users will be forwarded to hub(s). A hub is different from other nodes. Hubs are equipped with better RF transceivers, more powerful processors and have larger energy reserves. The hub's main role is to collect data generated by the nodes and then relay these data to the remote command centre using satellite transmission or long range radio. EAR uses the IEEE 802.11 Medium Access Control (MAC) protocol [7] that provides reliable

link-to-link transmission by using Request-To-Send (RTS), Clear-To-Send (CTS) and Acknowledgement (ACK) handshaking mechanisms.

4.2 Algorithm

Packets generated by nodes are known as Node Report (NREP) packets. EAR classifies NREP packets into different types according to the data carried and will route each packet type to the appropriate hub(s). If more than two hubs are interested in the same type of NREP packet, then nodes will route the NREP packets to the nearest one to minimize latency. Alternatively, the application protocol may also have its own header field to distinguish the type of data carried and forward them to a selected hub.

Setup Phase: When a hub is powered on, the application protocol will inform EAR of the type of NREP packet that it is interested to receive. Upon receiving this info, the hub will broadcast an Advertisement (ADV) packet. In the ADV packet, there is a field to indicate the type of NREP packet that the hub is interested to receive. When neighbouring nodes around the hub receive this ADV packet, it will store this route in their routing table. Nodes will not propagate the ADV packet received. When a node is powered on, the application protocol will inform EAR of the NREP packet types that will be generated. This node will then create a routing table for each type of NREP packet. As a typical network will have only a few types of NREP packets, the total storage space for maintaining the routing tables is small. Each node begins the initialisation process by randomly broadcasting a Route Request (RREQ) packet for a route to any hub for a NREP packet type. A RREQ packet is broadcasted for each type of NREP packet in the network. When a hub receives a RREQ packet, it will broadcast a Route Reply (RREP) packet if it is interested to receive the type of NREP packet. Similarly, when a node receives a RREQ packet, it will broadcast a RREP packet if it has a route to a hub that is interested in receiving the type of NREP packet. Otherwise, it will ignore the RREQ packet. Nodes do not propagate RREQ packets. When a node receives a RREP packet, it will store the route in the corresponding routing table for that NREP packet type. When it has at least a route to the hub for each type of NREP packet, it exits the initialisation process. By introducing a random delay for each node, a portion of nodes will receive a RREP packet before they have begun their initialisation. This enables fast propagation of routes and also saves on the amount of control packets generated in the setup phase (see Fig. 1). A node may store more than one route to the hub for each NREP packet type and each route is indexed using the next hop node's ID which is a neighbour of this node. A node will only keep one route entry for a neighbour that has a route to the hub. That neighbour could, however, have multiple routes to the hub. In the route table, every entry is uniquely identified by the neighbour's ID and only the best route of that neighbour is stored.

Route Management: In a typical WSN, each node has multiple neighbours to provide for redundancy and fault tolerance while still keeping the probability of packet collisions low. As nodes have very limited memory, the size of the routing table must be restricted. In EAR, two metrics are used to admit a route into the routing table. The first metric is the number of hops a route needs to reach the hub (route length). This metric is chosen because the best path is always the shortest path, with the lowest

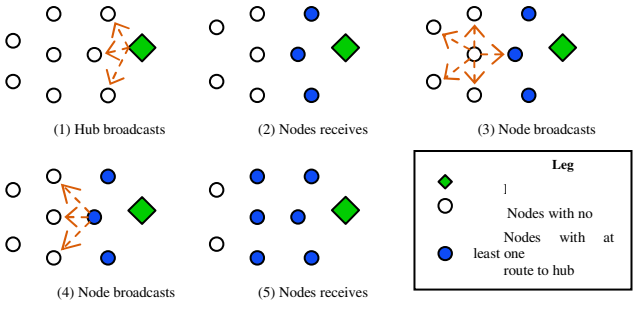


Fig. 1. Illustration of setup phase

latency and expends the least energy. But the RF link between a node and each of its neighbours will not be the same because of the difference in physical distance and the type of terrain between them. In this situation, the best route is not the shortest route because trying to forward a packet to a neighbour with shorter path but bad RF link quality will expend more energy in re-transmission and also increases packet latency than forwarding to a neighbour with a longer path but with better RF link. In EAR, we introduce "route blacklisting". Initially, routes are admitted into the routing table using their length as admission criteria to ensure that only shortest routes are chosen. As NREP packets start to flow through these routes, less desirable ones will start to exhibit high packet loss rate and will eventually be blacklisted and omitted from the route table. Omitted routes will not be re-admitted until after a period of time. Some routes are only affected by temporary external disturbances and so should be given the chance to be re-admitted after a period of time. The mechanism uses a sliding window that tracks the outcome of the last N attempts to forward NREP packets on a route. If a route fails to forward all packets in the last N consecutive attempts, it will be blacklisted and omitted from the routing table. The second metric, called energy metric, measures the energy remaining in the next hop node. This metric ensures that traffic load is evenly distributed among all the nodes. When a route is received from a new neighbour and the routing table is currently full, a route replacement strategy is carried out. Routes that are blacklisted are ignored. In the replacement algorithm, the first step is to search for the worst route in the routing table. The worst route is the route with the largest route length. If there is a tie, the energy metric is used to select the route with a lower value. In the second step, the worst route is then compared against the incoming route using the same process as in the first step. Only the best routes are stored in the routing table. This scheme supports reliable data delivery and low latency. Packets are guaranteed to travel on the best path from a node to the hub.

Data Dissemination: After the setup phase, every node will have at least one route to the hub for each type of NREP packet. Depending on the application, nodes will start generating NREP packets periodically or go into idle mode waiting for some event before generating NREP packets. An NREP packet carries two fields in its header: *ExpPathLen* and *numHopTraversed*. *ExpPathLen* defines the expected number of hops this packet must traverse before reaching the hub, according to eqn (1):

$$\text{ExpPathLen} = \text{NH} \times \alpha \,. \tag{1}$$

NH is the number of hops from this node to the hub for the route selected to forward the NREP packet. α is a weight assigned from 1.0 to 2.0. *NumHopTraversed* records the number of hops a packet has traversed and is initialised to 0. The NREP packet is then queued in the output buffer. The route selection mechanism will choose the best route (described below) for this NREP packet and forward it to the next hop in the route. When the next node receives the NREP packet, it will increment *numHop-Traversed* by one and then compare it with *ExpPathLen* that is never altered after initialisation. If *ExpPathLen* is larger than or equal to *numHopTraversed*, the routing mechanism will choose a route based on its score calculated using eqn (2):

$$\text{Score} = (P_E \times W_E + P_L \times W_L) / 2 \,. \tag{2}$$

P_E – energy level of the next hop node, W_E – assigned weight for P_E
P_L – link quality to next hop node, W_L – assigned weight for P_L

Should there be a tie in score, the shorter route is chosen. The objective is to forward the NREP packet to the next hop with minimal energy expenditure and to avoid heavily used nodes as a form of congestion control. It also selects the next hop neighbour with good link quality to reduce the probability of packet loss. If *ExpPathLen* is smaller than *numHopTraversed*, a simple selection requiring only two comparisons is used. Firstly, select the route with the shortest length. If there is a tie, select route with highest value in energy metric. If the number of hops a packet has traversed exceeds the expected number of hops, there must be some changes in the network topology affecting the RF communication. During this period, the NREP packet will take the shortest path to the hub to avoid looping. The same routing mechanism is invoked at each intermediate node until the packet reaches a hub.

Route Update: WSN nodes are prone to failure and the unpredictable RF link quality between neighbouring nodes changes frequently. Also, node energy levels will decrease according to the amount of NREP packets they receive for routing. Nodes need to maintain updated and fresh routes in the routing table at all times. EAR uses a novel solution that provides extremely fast updating of route at negligible cost. This solution uses the handshaking messages used by IEEE 802.11 MAC protocol. When *node 1* wants to send a packet to *node 2*, *node 1* sends an RTS packet and when *node 2* receives the RTS packet, it will send a CTS packet to *node 1*. The route information is piggybacked on both RTS and CTS packets. This enables the neighbours of both *nodes 1* and *2* to obtain the latest route information. RTS and CTS packets are received and processed by all nodes as part of the collision avoidance mechanism employed by the MAC protocol. On the other hand, DATA and ACK packets need not be received by all nodes and to conserve, all other neighbouring nodes can go to "sleep" except for the sender and receiver.

5 Simulation

The performance of EAR was evaluated on GloMoSim [8] against three existing reliable routing protocols for WSNs: AODV [9], DSR [10] and GRAB [11]. In GloMoSim, nodes are modelled after crossbow MICA2 mote [12] which is a popular hardware platform for WSN. Settings for the simulator are shown in Table 1.

Table 1. Simulator Settings

Frequency	**433 MHz**	**Frequency**
Bandwidth	76800 kbps	Bandwidth
Radio Range	56 m	Radio Range
Radio Model	Signal-to-Noise (SNR) Bounded	Radio Model
Propagation Model	Ground Reflection (Two-Ray)	Propagation Model
MAC protocol	IEEE 802.11 (DCF)	MAC protocol

In the simulation, all nodes generate data packets that are routed to one hub in the centre of the terrain. To show the capability of EAR to disseminate data packets to multiple hubs efficiently in a network we also simulated EAR with four hubs in the network. The three additional hubs were located in the centre of a uniform four-section partition of the terrain while keeping other factors such as topology of other nodes constant. In the entire test, the average number of neighbours per node was limited to 10. This ensured that nodes would have sufficient neighbours to elect a good one to forward packet and also prevent overcrowding which may lead to high packet loss rate due to collision. This is applied to all protocols under test. We use the following metrics to measure the performance of the routing protocols.

Packet delivery ratio (PDR): This measures the percentage of data packets generated by the nodes that are successfully routed to the hubs. It is expressed as the ratio of the total number of data packets successfully delivered to the total number of data packets generated. A core function of any good routing protocol is to route and deliver data packets with high probability.

Packet latency: This measure the average time it takes to route a data packet from the source node to the hub. It is expressed as the ratio of the sum of individual data packet latencies to the total number of data packets received. In any applications, it is desirable for data packets to be disseminated to the hub in the fastest time. Timely arrival of data packets at the hub is critically in ensuring that the network users are informed of any important events that have happened in the network.

Energy efficiency: This measure the total energy expended. We only calculate energy expended in transmitting and receiving by the nodes' RF transceiver. This parameter will show the energy-efficiency of routing protocols. Energy-efficient routing protocols are highly desired in any type of network especially in a WSN where nodes have limited energy. This directly affects the lifespan of a network.

5.1 Noise Tolerance

We used a model where every network node except the hubs has an error probability of 10% to 50% where packets were assumed to be corrupted or lost. In this setup, a source node generated 120 data packets at an interval of 1 packet every 30 second. To show how the routing protocols react under different traffic load condition, we conducted the test with 50% and 100% of nodes generating data packets. To study the

scalability of the routing protocols, we simulated 50 to 500 nodes. Results were averaged over 10 runs each with a different seed and shown in Figs. 2, 3 and 4.

AODV has the worst performance for PDR. AODV stores only one route to a hub in its routing table. When the path fails, it has to re-initiate route discovery, increasing bandwidth usage, packet collision and energy consumption. DSR performs better than AODV because it operates in promiscuous mode and thus able to obtain routing information quickly without the overhead of exchanging control packets. Although operating in promiscuous mode increases energy consumption by receiving all packets, we evaluated DSR's performance operating with and without promiscuous mode. We found that operating in promiscuous mode yields higher PDR and consumes less energy than operating without promiscuous mode. It may seem counter intuitive at first but deeper analysis leads to the explanation. Overhearing packets addressed to other nodes enables the node to obtain the latest route information even though it may not be sending any packet currently. When the node sends a packet, it is assured of a valid route in the routing table. This reduces packet latency as no route recovery is needed and using a valid route enhances successful delivery reducing the need for retransmission that incurs additional energy. These savings outweigh the cost of operating in promiscuous mode. DSR also stores multiple routes to a single hub.

At both 50% and 100% load, GRAB has higher PDR than DSR beyond 300 and 200 nodes respectively. In larger networks, the number of hops to reach the hubs increases and the changing topology due to noise overwhelms DSR's route recovery and maintenance mechanisms. GRAB uses interleaved mesh to forward packets reliably without any route recovery mechanism thus achieving a higher PDR. However, GRAB broadcasts packets and shows high packet collision resulting in lower PDR than EAR. GRAB consumes more energy than the other routing protocols because broadcast generates large amount of duplicated packets and all these use additional bandwidth, resulting in high packet latency. Full range of energy expended and packet latency of GRAB were not shown because they exceeded the scale.

EAR achieves very high PDR, over 90% with 50% and 100% loads. It delivers more packets than AODV, DSR and GRAB, with lower latency and higher energy efficiency. EARMulti is EAR operating with 4 hubs. It shows that EAR is capable of routing packets to multiple hubs at no additional cost. EAR has a lightweight mechanism that can react quickly to link failure and avoid routes with bad link quality. At

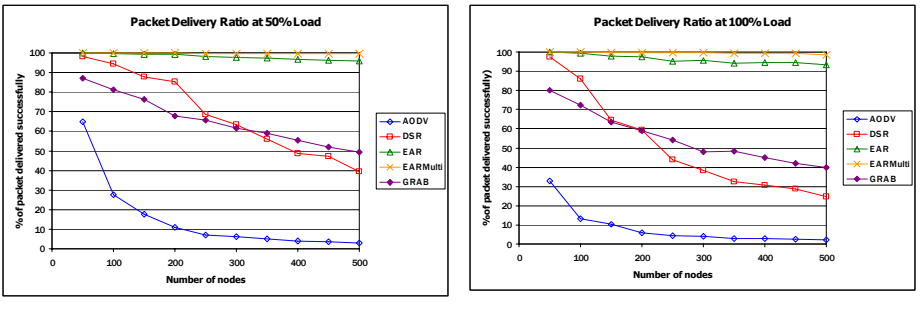


Fig. 2. Graphs showing packet delivery ratio

100% load, EAR's packet latency exceeds that of AODV and DSR with more than 400 nodes. This is caused by network capacity being exceeded resulting in high packet collision rate. But EAR still achieves over 90% PDR in a congested network as it has a load balancing mechanism that distributes traffic evenly across the network. EAR's performance is also scalable. At 100% load, EAR achieves a PDR of 100% with 50 nodes and a PDR of 93% with 500 nodes.

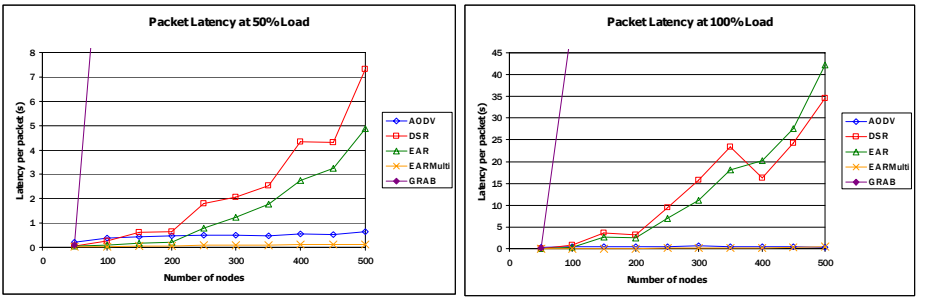


Fig. 3. Graphs showing packet latency

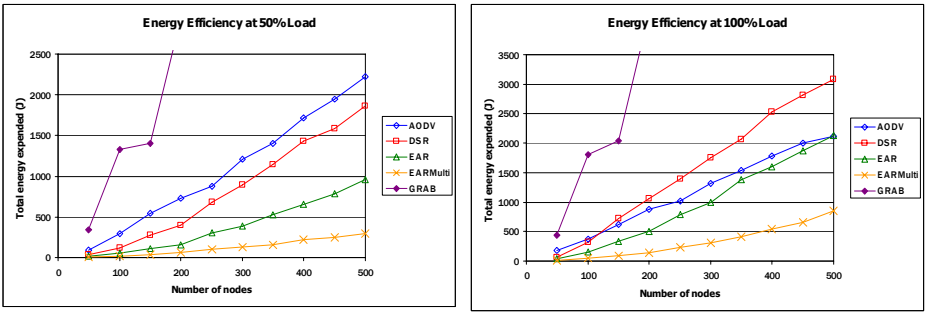


Fig. 4. Graphs showing energy efficiency

5.2 Fault Tolerance

Nodes are prone to failure in a WSN and routing protocols must be fault tolerant by achieving high and consistent PDR in a network plagued by nodes failures. In the test, 50% of the nodes fail at a random time within the simulation duration. We set 50% of the nodes to generate data packets at a rate of 1 packet every 30 second. Results are shown in Fig. 5. At 500 nodes, EAR achieves 96% PDR while DSR and AODV managed only 82% and 12% respectively. But the latency incurred by EAR is also higher. This is expected as the nodes in the network fails, there are fewer nodes for routing of packet and this leads to higher packet latency because of congestion in the remaining nodes. The higher latency can also be attributed to longer route taken by packet as they need to detour around the failed nodes. When operating with 4 hubs, EAR has the best performance credited to its ability to route packets to a hub offering the best path. EAR also expends lesser energy than AODV and DSR but delivers more packets.

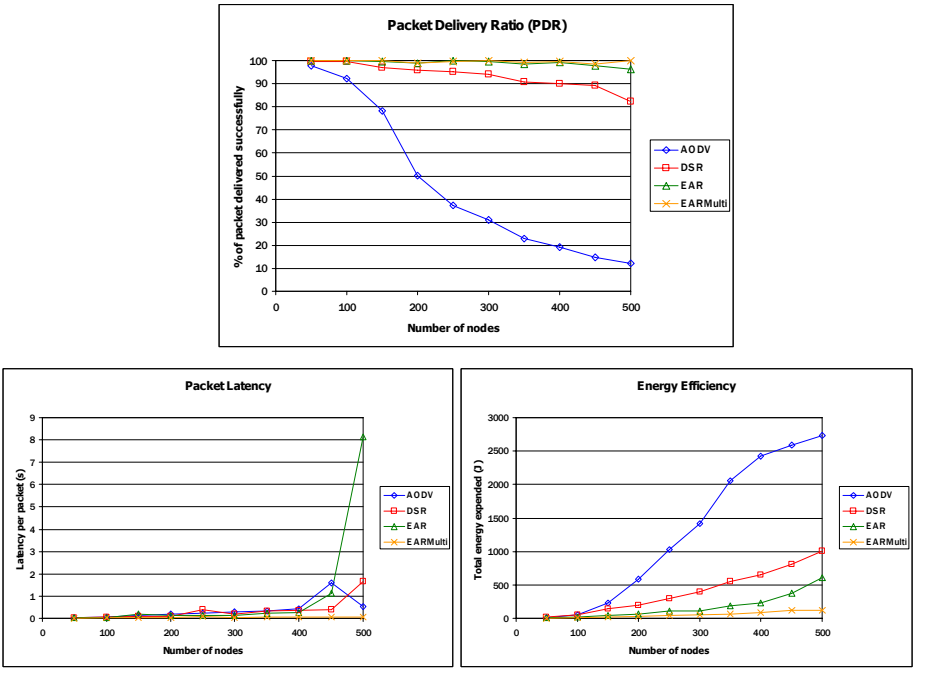


Fig. 5. Fault Tolerance results

6 Conclusion

We have proposed EAR; an efficient and reliable routing protocol for wireless sensor network. Using simulations results, we have shown that it outperforms three existing routing protocols when operating in a noisy wireless environment where node failure rate is high. Future work includes incorporating better congestion avoidance mechanism to achieve lower packet latency and higher PDR when operating in a network with high traffic volume.

References

1. Estrin D., Girod L., Pottie G., Srivastava M.: Instrumenting The World With Wireless Sensor Networks, Proceedings ICASSP 2001, USA.
2. Akyildiz I.F., Cayirci E., Sankarasubramaniam Y., Su W: A Survey on Sensor Networks, IEEE Communications, Pages 102-114, August 2002.
3. Bulusu N., Estrin D., Girod L., Heidemann J.: Scalable Coordination for Wireless Sensor Networks: Self-Configuring Localization Systems, Proceedings of 6th International Symposium on Communication Theory and Applications (ISCTA'01), Ambleside, UK, July 2001.
4. Estrin D., Govindan R., Heidemann J., Kumar S.: Next Century Challenges: Scalable Coordination in Sensor Networks, International Conference on Mobile Computing and Networks (MobiCOM'99), Seattle, Washington, United States, August 1999.

5. Kahn J. M., Katz R. H., Pister K. S. J.: Next Century Challenges: Mobile Networking for Smart Dust, Proceedings of the 5th Annual ACM/IEEE, International Conference on Mobile Computing and Networking, Pages 271-278, Seattle, Washington, USA, 1999.
6. Chandrakasan A., Cho S-H., Ickes N., Min R., Shih E., Sinha A., Wang A.: Physical Layer Driven Protocol and Algorithm Design for Energy-Efficient Wireless Sensor Networks, Proceedings of 7th Annual International Conference on Mobile Computing and Networking, pp 272-287, Rome, Italy, 2001.
7. LAN MAN Standards Committee of the IEEE Computer Society: Wireless LAN medium access control (MAC) and physical layer (PHY) specification, IEEE, New York, NY, USA, IEEE Std 802.11-1997 edition, 1997.
8. Ahuja R., Bagrodia R., Bajaj L., Gerla M., Takai M.: GloMoSim: A Scalable Network Simulation Environment, Technical report 990027, UCLA, Computer Science Department, 1999.
9. Das S., Perkins C. E., Royer E. M.: Ad Hoc On-demand Distance Vector (AODV) Routing, IETF Internet Draft, draft-ietf-manet-aodv-13.txt, February 2003 (Work in Progress).
10. Hu Y. C., Jetcheva J. G., Johnson D. B., Maltz D.A.: The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, Internet-Draft, draft-ietf-manet-dsr-09.txt April 2003. Work in Progress.
11. Lu S., Ye F., Zhang L., Zhong G.: A Robust Data Delivery Protocol for Large Scale Sensor Networks, IEEE International Workshop on Information Processing in Sensor Networks (IPSN), 2003.
12. http://www.xbow.com

ACO Based QoS Routing Algorithm for Wireless Sensor Networks

Wenyu Cai, Xinyu Jin, Yu Zhang, Kangsheng Chen, and Rui Wang

Department of Information Science & Electronic Engineering, College of Information Science & Engineering, Zhejiang University, Hangzhou, China, 310027
{suncai, jinxy, zangwill, chenks, wangr}@zju.edu.cn

Abstract. In this paper, we proposed an approach for Quality of Service (QoS) routing algorithm of Wireless Sensor Networks (WSNs) based on Ant Colony Optimization (ACO). The special characteristics of WSNs need to reduce the computational complexity and energy consumption of the QoS routing algorithm especially. We note that ACO algorithm using collective intelligence of artificial ants as intelligent agents is very appropriate to solve the combinatorial optimization problems in a fully distributed way, so in this paper we use modified ACO approach to solve Delay Constraint Maximum Energy Residual Ratio (DCMERR) QoS routing problem of WSNs. The QoS routing solution proposed in this manuscript, which is named as ACO based QoS routing algorithm (ACO-QoSR), searches for the best paths, which are satisfied with the QoS requirements with intelligent artificial ants. To overcome the problem of limited energy in WSNs, there are some modifications to enhance ACO's convergence rate. ACO-QoSR algorithm is the tradeoff between a certain guaranteed QoS requirements and acceptable computational complexity. The simulation results verify that ACO-QoSR algorithm can reduce the selected paths' delay and improve the selected paths' normalized energy residual ratio at the similar levels of routing overhead.

1 Introduction

Wireless Sensor networks (WSNs) [1] is a promising approach for a variety of applications, such as monitoring safety of buildings, monitoring body health, and tracking environmental pollution levels. In order to maximize the whole network's lifetime, recent research on routing protocols designed for wireless sensor networks mainly considered energy consumption efficiency. However, the transmission of real-time data has posed additional challenges, which require both energy efficient and QoS aware routing to ensure efficient energy usage and meet real time applications.

The Objective of QoS routing [2] is to find a feasible path satisfied by QoS requirement, so as to optimize the usage of the network. QoS routing schemes have received considerable attention over past decade but the research of QoS routing for WSNs is still seldom. It is because QoS routing for WSNs will lead to a significant amount of computation and communication overhead, especially to the sensor nodes with limited battery energy and poor computation ability. Ant Colony Optimization

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 419 – 428, 2006.

(ACO) [3] is a graph representation based evolutionary meta-heuristic algorithm and consider the ability of simple ants to solve complex problems by cooperation. ACO is inspired by the behavior of the real ant colony. It has been experimentally observed that ants in a colony can converge on moving over the shortest among different paths connecting nest and source of food. ACO has been successfully employed to solve many different combinatorial optimization problems such as traveling salesman problem (TSP), graph coloring problem and vehicle routing problem.

In this article, we mainly dedicate to improve the delivery quality of WSNs at the aspect of QoS routing algorithm. ACO-QoSR algorithm proposed in this article focuses on how to solve on-demand QoS routing problem of WSNs with ACO approach. To improve the convergence rate of ACO algorithm, some small modifications have been applied to traditional ACO.

2 Related Work

ACO routing algorithms are first designed for wired networks as in AntNet [4], ABC (Ant Based Control) [5]. The main concept of both AntNet and ABC is to deploy small ant packets to discover forwarding paths from source to destination. These algorithms exhibit a number of interesting properties such as working in a fully distributed way, being highly adaptive to network and traffic changes, and automatically taking care of data load spreading. But using periodic unicast ants to discover routers would incur a large delay and the adaptability to topology changes would be unacceptably slow. There are some ACO routing schemes for Mobile Ad-hoc Networks (MANETs) such as GPS/Ant-like routing algorithm for mobile ad-hoc networks [6], Ant-AODV [7], and ARA (Ant-Colony-Based Routing Algorithm) [8]. However these algorithms have made some modifications to fit for different network types but none of them is satisfied for special characteristics of WSNs.

Current research on routing of sensor data mostly focused on protocols that are energy aware to maximize the lifetime of the network, scalable for large number of sensor nodes and tolerant to sensor damage and battery exhaustion. However, QoS routing for WSNs considering not only above factors but also different QoS parameters has few been designed. It is complex and difficult to handle because of many disadvantages of WSNs such as limited energy, limited memory, large scale diameter, and short communication range, et al. In this manuscript we tackle these challenges into account and a novel QoS routing ACO-QoSR algorithm considering constraint delay and energy balancing for WSNs is discussed.

3 ACO Based QoS Routing Algorithm

QoS routing requires a route that satisfies the end-to-end QoS requirement, often given in terms of bandwidth, delay, loss probability, jitter or cost. The applications run in the WSNs are mostly delay-sensitive applications, so in this study QoS metrics of WSNs mainly refer to transmission delay and energy conservation ratio including energy-balancing factor.

3.1 WSNs QoS Routing Model

The objective of ACO-QoSR algorithm presented in this paper is described as Delay Constraint Maximum Energy Residual Ratio (DCMERR) Problem:

Consider a wireless sensor network *G(N,E)*, where *N* is the set of sensor nodes and *E* is the set of links. Each link $e_{ij} \in E$ is associated with a delay parameter $\mathrm{Delay}(e_{ij})$ and each node $n_i \in N$ is associated with a parameter called node's *ERR* (Energy Residual Ratio), which denotes the residual ratio of node's energy and is defined as $ERR = \frac{E_{residual}}{E_{total}}$. The QoS routing problem is defined to find a path *P* from a source sensor node to a destination sensor node such that:

$$(1)\ \mathrm{Delay}(P) \triangleq \sum_{e_{ij} \in P} \mathrm{Delay}(e_{ij}) \le D$$

$$(2)\ ERR(P) \triangleq \frac{1}{Hop_P} \sum_{n_i \in P} ERR(n_i) \ge ERR(P^*),\ \forall P^* \text{ and } P \text{ satisfying (1)}$$

where *D* denotes the maximum permitted delay value and Hop_P denotes the hop count of path *P*. *ERR*(*P*) is defined as path's normalized energy residual ratio. ACO-QoSR is a routing protocol based on probability and the probability to select path *P* satisfies:

$$(3)\ Prob(P) > Prob(P^*)\ ,\ \forall P \text{ and } P^* \text{ statisfying (1)(2)}$$

How to solve the above question is also a constrained path optimization (CPO) problem, which is proved to be NP-hard [9, 10]. So it is too expensive to resource limited sensor nodes of WSNs even with Lagrange algorithm [11]. The following demonstrates how to solve DCMERR problem using Lagrange Multiplier (LM) algorithm. The Lagrange function is described as:

$$L(P, \lambda, D_i) = ERR(P) + \lambda(D_i - \mathrm{Delay}(P))\ \ (0 \le D_i \le D\ ) \tag{1}$$

As we known, the condition of DCMERR is an inequation, so the Lagrange function is extended to Lagrange function matrix shown as below:

$$\overline{L} \triangleq \left[L(P, \lambda, D_1) \quad \ldots \quad L(P, \lambda, D_i) \quad \ldots \quad L(P, \lambda, D) \right]^{\mathrm{T}} \tag{2}$$

Therefore, the results can be obtained through calculating the partial differential of Lagrange function matrix:

$$\overline{L}_P = \left. \frac{\partial \overline{L}}{\partial P} \right|_{P=P_0} = 0\ \ \&\&\ \ \overline{L}_\lambda = \left. \frac{\partial \overline{L}}{\partial \lambda} \right|_{\lambda=\lambda_0} = 0 \tag{3}$$

In the following section we explain ACO-QoSR algorithm to solve this primary problem excellently with a fully distributed way in ACO approach.

3.2 ACO-QoSR Algorithm

The working process of ACO-QoSR algorithm is described as follows: when a source node has sensor data to send, it checks its routing table to search appropriate path. A new route probe phase will start only if there are no unexpired paths to the destination, and node needs to cache data waiting for transmit at the same time. There are *m* forward ants needed to send for route probe. After routing discovery process, cached data will be sent to destination immediately. To reduce the delay of first discovery phase, it is requested that ACO-QoSR algorithm would start a full route probe phase at the time of network initialization. The flow to deal with packet when sensor node received an ant is described in Fig.1. There are three phases in the ACO-QoSR: forward ants phase, backward ants phase and route maintenance phase.

3.2.1 Forward ANTs Phase

If source sensor finds there is no satisfied and unexpired path to packet's destination in its routing table, it will generate a certain number of forward ants to search for paths to this destination. Forward ants are agents that establish the pheromone track from source node to the destination node. Forward ants carry their born timestamp, source and destination address and forward ant IDs, collect intermediate node's local information $ERR(n_i)$ and record its path information in its travel from source to destination.

If a node receiving a forward ant for the first time, it creates a record in its routing table and select one neighbor node as the next hop randomly. If there is the record in the node's routing table, the next hop is selected according a certain probability value. Forward ants choose their next nodes at intermediate node *i* following the probability defined in the routing table:

$$p_{ij} = \left[\tau_{ij}\right]^{\alpha}\left[\eta_{ij}\right]^{\beta} \Big/ \sum_{k\in N_i}\left[\tau_{ik}\right]^{\alpha}\left[\eta_{ik}\right]^{\beta} \tag{4}$$

where τ_{ij} and η_{ij} denote the pheromone value and local heuristic value of the link e_{ij}, α and β are constants that determine the relative influence of the pheromone values and the heuristic values on the decision of the ant, N_i denotes neighbour nodes of node *i*. In the ACO-QoSR algorithm we define local heuristic information on the links e_{ij} as the ratio between the residual energy of node *j* and the summary residual energy of all the neigbour nodes of node *i* :

$$\eta_{ij} = E_{residual}(j) \Big/ \sum_{k\in N_i} E_{residual}(k) \tag{5}$$

Therefore forward ants' transition probability is the tradeoff between the heuristic and pheromone factor. For the heuristic factor, the next node with more residual energy within all the neighbor nodes should be chosen with high probability, thus implementing energy balancing. As to the pheromone factor, the link with more

pheromone should be chosen with high probability, thus implementing positive convergence to better paths.

The pair <source node address, ant id> uniquely identifies a forward ant and the nodes are able to distinguish duplicate packets on this pair to guarantee loop-free routes. When a forward ant passed through an intermediate node *i*, the energy residual ratio of node *i* will be accumulated to the field "Path_ERR" in the forward ant's packet.

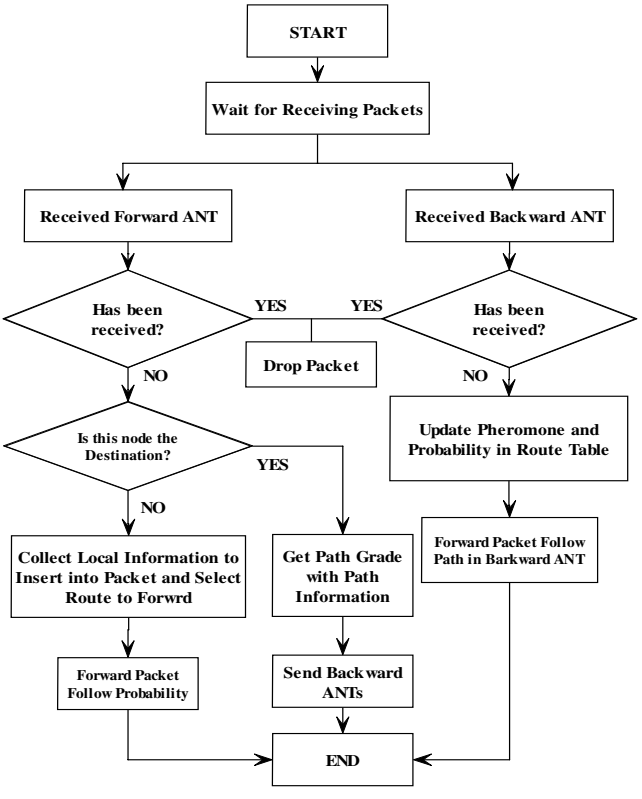


Fig. 1. Route Discovery Process of ACO-QoSR

3.2.2 Backward ANTs Phase

When a forward ant reaches its destination, the forward ant *k* will be killed and a backward ant with ant id *k*+*m* will be generated. A backward ant carries source and destination address, backward ant ID, path information copied from corresponding forward ant and pheromone update value.

The path's total energy residual ratio $ERR_k = \sum_{n_i \in P_k} ERR(n_i)$ and hop count Hop_k carried by forward ant *k* will be used to calculate the increment value of pheromone:

$$\Delta\tau^k = \begin{cases} f(ERR_k^*, \Delta\tau^{k-1}) & \text{Delay}(k) \le D \\ 0 & \text{Delay}(k) > D \end{cases} \tag{6}$$

The normalized energy residual ratio of path k is defined as:

$$ERR_k^* = ERR_k / Hop_k \tag{7}$$

where ERR_k and Hop_k denote the summary energy residual ratio and hop count of path through which forward ant k traveled, Delay(k) denotes the delay time of this path and equals to difference between the received time of forward ant and forward ant's timestamp, D denotes required maximal delay value. From Eq.(6) we find that there is no pheromone increment of path with larger delay than D. In the other word, pheromone updating is intended to allocate a greater amount of pheromone with low energy residual ratio journey path. This value is evaluated when the ant has completed a journey. The following rule is used to calculate updating amount of substance previously laid on the trail:

$$f(ERR_k^*, \Delta\tau^{k-1}) = \begin{cases} \Delta\tau^{k-1} + \lambda(ERR_k^* - ERR_{k-1}^*) & ERR_k^* > ERR_{k-1}^* \\ \Delta\tau^{k-1} & ERR_k^* \le ERR_{k-1}^* \end{cases} \tag{8}$$

where ERR_k^* and ERR_{k-1}^* denote the normalized energy residual ratio of path traveled by forward ant k and k-1.

Backward ants select their next hop node following the path information carried by them and update intermediate node's pheromone value. If ant k uses link e_{ij} in its tour, the rule to update pheromone of node i is shown as follow:

$$\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} + \rho\cdot\Delta\tau_{ij} \tag{9}$$

where $\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau^k$ and $\rho \in (0,1)$ represents the volatility degree of pheromone to reduce the effect of past experience and prevent pheromone concentration in non-optimal paths.

The pheromone updating rule was meant to the addition of new pheromone deposited by ants on the visited edges and to pheromone deposited by ants on the stops iterating either when an ant found a solution or when a maximum number of generations have been performed.

3.2.3 Route Maintenance Phase

Routing table conserved in the node memory is organized as Table1. Each column in this table corresponds to a neighbor and each row corresponds to each destination. The entries in the table are the pheromone and probabilities that the next-hop is a specific neighbor. Probabilities are used to allow forward ants to randomly explore the whole network and possibly find new and better paths. Once the routes are discovered, the next-hop probabilities are updated to reflect the new discoveries.

Table 1. Routing table of node i

Destination	NB_1		NB_j	NB_{Ni}		Expire time
D_1	τ_{D_1,NB_1}	p_{D_1,NB_1}	…	$\tau_{D_1,NB_{N_i}}$	$p_{D_1,NB_{N_i}}$	T_1
D_2	τ_{D_2,NB_1}	p_{D_2,NB_1}	…	$\tau_{D_2,NB_{N_i}}$	$p_{D_2,NB_{N_i}}$	T_2
D_k	……		…	……		T_k
D_N	p_{D_N,NB_1}	p_{D_N,NB_1}	…	$\tau_{D_N,NB_{N_i}}$	$p_{D_N,NB_{N_i}}$	T_N

$D_1 D_2 \cdots D_k \cdots D_N$: all the destination nodes in WSNs
$NB_1 \cdots NB_j \cdots NB_{Ni}$: all the neighbor nodes of node *i*

τ_{D_i,NB_j} : pheromone value between node *i* to the node D_i with next node NB_j

p_{D_i,NB_j} : probability value between node *i* to the node D_i with next node NB_j

T_k : expiration time of route from node i to destination D_k

Every entry in the routing table has an expiration time and some entries will be disabled as time goes on. When the current time exceeds the expired time, a new route discovery phase will restart. Expiration time is set according to the stabilization of network topology and the default value is 10s. If network topology changes more frequently, expired time will be set much shorter.

ACO-QoSR also uses periodic HELLO messages to maintain updated information about the connectivity of neighboring nodes. Once the next hop becomes unreachable, the node will not restart route discovery immediately. When the link breaks during data transmission phase, the intermediate node will first deactivate this link by setting the pheromone value to 0 and then search for an alternative neighbor node in its routing tables to retransmit.

3.3 Discussions

In the following simulations, the parameters used in ACO-QoSR algorithm are set as: $\alpha = \beta = 1; \rho = \gamma = 0.1; m = 50; \tau_{ij\min} = 0.1; \tau_{ij\max} = 10$, Route Expiration Time=10sec; Hello Interval=1sec; Network Diameter=30hops.

There are a few techniques about pheromone shaping to improve the performance of ACO-QoSR algorithm to alleviate stagnation.

Pheromone limiting: Placing limits on the pheromone $\tau_{ij} \in [\tau_{\min}, \tau_{\max}]$ concentration on every path coerces the probability of ants choosing a particular path to a certain range. By placing an upper and lower bound of pheromone for every edge, the preference of an ant for optimal paths over non-optimal paths is reduced. This approach prevents the situation of generating a dominant path.

Pheromone smoothing: pheromone smoothing seems to be effective in preventing the generation of dominant paths because paths with very high pheromone concentrations will be reinforced with lesser pheromone.

In this paper, we integrate pheromone evaporation, pheromone limiting and smoothing techniques together to get tradeoff between exploration and exploitation so that to improve the performance of ACO-QoSR algorithm. The pheromone update rule is shown below:

$$\tau_{ij}(t) \leftarrow \tau_{ij}(t) + \gamma \cdot (\tau_{ij\max} - \tau_{ij}(t)) \quad \gamma \in (0,1) \tag{10}$$

where $\tau_{ij\max} = 10\ (\forall i, j \in N)$ and $\gamma = 0.1$. The initial pheromone value of link e_{ij} is set as: $\tau_{ij}(0) = 0.1\ (\forall i, j \in N)$.

Furthermore, how to select the maximal and minimal pheromone value perfectly will be discussed as follows. The minimal pheromone value of ACO-QoSR algorithm is set as zero for simple propose. The theorem shown in the below is used to decide the maximal pheromone value.

Theorem: $$\lim_{t\to\infty} \tau_{ij}(t) \le \Delta\tau_{\max} \quad (\Delta\tau_{\max} = \max\{\Delta\tau_{ij}\}) \tag{11}$$

4 Simulation and Experiments

Network Simulator (NS-2) [12] is used to simulate ACO-QoSR algorithm. The well-known reactive routing protocols for wireless multi-hop ad hoc networks including AODV [13], DSDV [14] have been implemented. The simulation models a network of 100 sensor nodes migrating within an area of 1000m×1000m. All the sensor nodes are regarded as static and the node 99 is the base station. The link layer is implemented using IEEE802.11 Media Access Control (MAC) protocol. The interface queue gives priority to routing packets being served. We select random twenty CBR (Constant Bit Rate) flows from twenty sensor nodes to the base station, the CBR rate is varied with Rate (packets/s) parameter and the packet's size is fixed as 512bytes. The whole simulation scenario is run for 300 simulated seconds. The same scenario is repeated several times and the average results are described as below.

We use the wireless communication path loss model to calculate transmission energy consumption [15]. In this model, the received signal power is dependent on the distance between sensor nodes. In order to verify the energy balancing and maximum effect, the initial energy of each sensor node is set as a very small value 100.0 Joules. The power units used by transmission, receiver and idle process are set as: txPower=0.660, rxPower=0.395, idlePower=0.035.

Four metrics are selected to measure the performance of ACO-QoSR versus AODV and DSDV: mean end-to-end packet delay, packet delivery ratio, routing overhead and path's normalized energy residual ratio. The detailed results are shown in Fig.2-Fig.5. Each simulation has been repeated several times with different rate values.

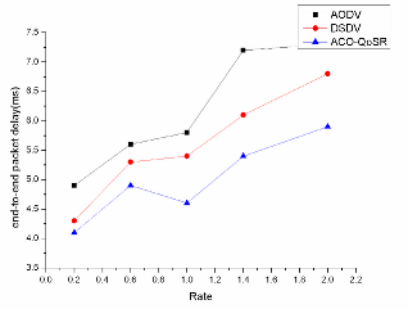


Fig. 2. Mean end-to-end packet delay

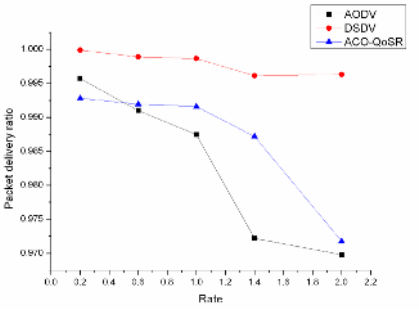


Fig. 3. Packet delivery ratio

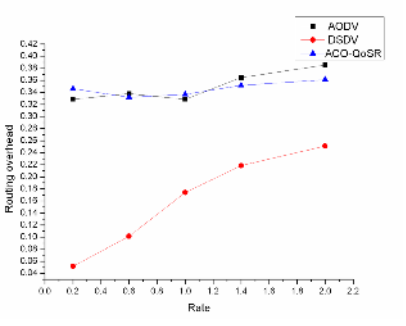


Fig. 4. Routing overhead

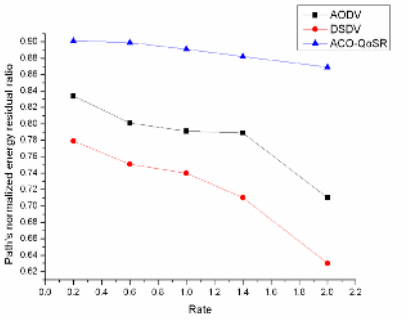


Fig. 5. Path's normalized energy residual ratio

Fig.2 illustrates the mean end-to-end delay comparison and the average delay of ACO-QoSR is less than AODV and DSDV and the average delay is restricted below scheduled delay constraint, which is set as 8 msec in the simulation. In Fig.3 we can find that ACO-QoSR shows a comparable average packet delivery ratio with AODV and DSDV. The routing overhead of ACO-QoSR is very small in the Fig.4, because there are no routing tables which are interchanged between the nodes, and the forward ants and backward ants are smaller than data packets. Thus they do not have to transmit additional routing information. Fig.5 illustrates that ACO-QoSR improves the path's normalized energy residual ratio much larger than AODV and DSDV, as a result, the energy balancing effect is achieved. From above simulation results we noticed that the overhead of network traffic is still too light and the change of parameter's values with Rate value is inconspicuous, it is because that the sensor nodes in the scene are all not mobile, but the difference between ACO-QoSR with AODV and DSDV is evident.

5 Conclusions

In this paper we present a new on-demand routing algorithm named ACO-QoSR, which uses modified ACO algorithm to deal with QoS routing problem of WSNs. ACO-QoSR can solve Delay Constraint Maximum Energy Residual Ratio (DCMERR)

problem of WSNs with ACO approach in a fully distributed way. ACO-QoSR algorithm is scalable to the more sophisticated QoS requirements in WSNs. Furthermore, the state-of-the-art approaches for mitigating stagnation of ACO approach were analyzed. By comparing network's performance from simulation results, the effectiveness of the ACO-QoSR algorithm has been verified.

References

1. Akyildiz, F., Su, W., Sankarasubramaniam, Y., et al.: A survey on sensor networks. IEEE communication Magazine, Aug 2002:102-114
2. Baoxian Zhang, Mouftah, H.T.: QoS routing for wireless ad hoc networks: problems, algorithms, and protocols. IEEE Communications Magazine 43 (10), Oct. 2005: 110-117
3. Colorni, A., Dorigo, M., Maniezzo, V., et al.: Distributed optimization by ant colonies. Proceedings of ECAL'91 (European Conference on Artificial Life). Paris, France, 1991:134-142
4. Caro G.D., Dorigo, M.: AntNet: A Mobile Agents Approach to Adaptive Routing. University Libre de Bruxelles, Belgium, Technical report IRIDIA/97-12, 1997
5. Schoondenvoerd, R., Holland, O., Bruten, J., et al.: Ant-based load balancing in telecommunications networks. Adaptive Behavior, pp.169-207, May 1997
6. Camap, D., Loureiro, A.A.F.: A GPS/Ant-Like Routing Algorithm for Ad Hoc Networks. in IEEE Wireless Communications and Networking Conference (WCNC'00), Chicago, IL, September 2000
7. Marwaha, S., Tham, C.K., Srinivasan, D.: Mobile Agents based Routing Protocol for Mobile Ad hoc Networks. In IEEE Global Telecommunications Conference (GLOBECOM'02), Taipei, Taiwan, November 17-21 2002
8. Gunes, M., Sorges, U., Bouazizi, I.: ARA-The Ant-Colony Based Routing Algorithm for MANETs. In International Conference on Parollel Processing Workshops (ICPPW'02), Vancouver, B.C., Canada, pp.79-85, August 2002
9. Guin, R., Orda, A.: QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms. Proceedings of the INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution, p.75, April 09-11, 1997
10. Feng G., et al.: Performance Evaluation of Delay-Constrained Least-Cost Routing Algorithms Based on Linear and Nonlinear Lagrange Relaxation. Proc. of ICC'2002, New York, 2002
11. Bertsekas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Academic Press, 1982
12. VINT. The Network Simulator ns-2 [CP/OL] (2003-06-10). http://www.isi.edu/nsnam/ns/
13. Perkins, C., Royer, E.: Ad-hoc On-Demand Distance Vector Routing. Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, February 1999
14. Perkins, C.E., Bhagwat, P.: Highly Dynamic Destination -Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. Proc. of the SIGCOMM'94 Conference on Communications, Architectures Protocols and Applications, August. 1994, pp. 234-244
15. Rappaport, T.S.: Wireless communication: Principles and Practice. pp.69-122. Prentice Hall, 1996

Cluster Number Variability Problem in LEACH

Huafeng Liu, Liang Li, and Shiyao Jin

National Lab for Parallel and Distributed Processing,
Changsha 410073, Hunan, China
liuhuafeng01@gmail.com

Abstract. LEACH is one of the most popular hierarchical routing algorithms for sensor networks. In LEACH, the cluster number per round takes an important effect on the network lifetime. Thus the goal of the cluster head selection algorithm is to ensure that the expected number of clusters per round equals a pre-determined optimal value *k*. However, a slight inaccuracy is spotted in the computation of the node self-selected probability per round. As a result, the number of clusters produced by the algorithm is distributed in a large range around the target value. We propose an improved clustering scheme (I-LEACH) to correct this inaccuracy. The experimental results show that this scheme can ensure a more stable number of clusters.

1 Introduction

Hierarchical or cluster based routing methods, originally proposed in wireline networks, are well-known techniques with special advantages related to scalability and efficient communication. As such, the concept of hierarchical routing is also utilized to perform energy-efficient routing in wireless sensor networks. [1] Heinzelman, *et al*. [2] [3] introduced a hierarchical clustering algorithm for sensor networks, called Low Energy Adaptive Clustering Hierarchy (LEACH). LEACH randomly selects a few sensor nodes as cluster heads (CHs) and rotates this role to evenly distribute the energy load among the sensors in the network. The cluster head selection algorithm was created to ensure that the expected number of clusters per round is *k*, a pre-determined system parameter, which can lead to the minimum energy dissipation of sensor networks.

However, the number of CHs produced by LEACH is distributed in a very large range around the optimal value. Fig. 1 shows the simulation results on a 100-node network with the optimal value $k=5$. The number of CHs ranged from *0* to *35*. In addition, the percentage of rounds that the number of CHs is equal to *5* is less than 19%.

In this paper, we analyze this cluster number variability problem and spot that there exists a slight inaccuracy in the computation of the node self-selected probability per round, which causes the expected number of cluster per round is not the constant value *k*. As a result, the variance of the number of CHs per round is increased with regard to the pre-determined optimal value *k*. We propose a clustering scheme called I-LEACH (Improved LEACH). I-LEACH improved the cluster head selection algorithms of LEACH in order to correct the inaccuracy problem.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 429 – 437, 2006.

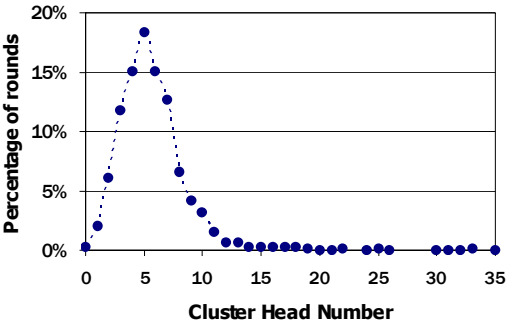


Fig. 1. Cluster Head Number Distribution of LEACH

The remainder of the paper is organized as follows. In section 2, we discuss the related works. Section 3 presents a brief review of the LEACH protocol. In section 4, we analyze the CH selection algorithm of LEACH based on a stochastic process model, and spot the inaccuracy of the cluster head selection algorithm. In Section 5 the improved CH selection algorithm is introduced and simulation results based on Ns-2 are presented to show the effectiveness of I-LEACH. Section 6 concludes this paper.

2 Related Works

In [4], Wang *et al*. also noticed the cluster variability problem of LEACH. They presented a bi-dimensional Markov chain model for analyzing LEACH protocol. The results indicate that the number of clusters generated in LEACH can not concentrate with in a narrow range of the optimal value. They proposed a clustering scheme called StepWise AdapTive Clustering Hierarchy (SWATCH) in [5], which inserted an add-on selection into the cluster head selection of LEACH. SWATCH decreased the variance of cluster number per round. However, since SWATCH splits the selection phase into an initial selection stage and an add-on selection stage, it increased the implemental complexity on the sensor node.

In [6], Monte Carlo simulation was used to analyze the statistical characteristics of the number of CHs selected by LEACH. The simulation results were coincided with that obtained analytically in [4].

In [7], Cao *et al*. pointed out that the clustering algorithm of LEACH may result in faster death of some nodes i.e. shorten system life. A distributed clustering algorithm with an adaptive back-off strategy was proposed to realize load balance among sensor nodes. The primary goal of that algorithm was to prolong the system life.

Although all these authors have noticed the cluster variability problem of LEACH, they considered that the inherent randomness in the CH selection algorithm was the only reason for this variability problem. They did not found there was a slight inaccuracy in the computation of the node self-selected probability per round in LEACH.

3 Brief Review of LEACH

LEACH is one of the most popular hierarchical routing algorithms for sensor networks. It was proposed for an application in which sensor nodes are randomly distributed on a grid-like area and are continuously sensing the environment to send reports to a remote base station (BS). The application assumes that nodes are equally significant and data aggregation is possible. The idea is to form clusters of the sensor nodes based on the received signal strength and use local cluster heads as routers to the BS.

3.1 Operation of LEACH

The operation of LEACH is divided into rounds. Each round consists of a set-up phase and a steady-state phase. In the set-up phase, the CHs are selected and clusters are organized. A TDMA scheme is used for intra-cluster transmissions. In the steady-state phase, the sensor nodes begin sensing and transmitting data to the CHs. After receiving all the data the CH node aggregates it before sending it to the BS. Each cluster communicates using different CDMA codes to reduce interference from nodes belonging to other clusters. The duration of the steady-state phase is longer than the duration of the set-up phase in order to minimize overhead. The timeline of LEACH operation is shown in Fig. 2. LEACH assumes that the nodes are all times synchronized and start the set-up phase at the same time.

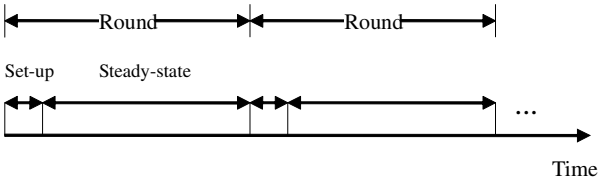


Fig. 2. Illustration of LEACH Timeline

3.2 Cluster Head Selection Algorithms

Cluster head selection is a kernel element of LEACH protocol. At the beginning of a set-up phase, each sensor node, say i, chooses a random number between *0* and *1*. If this random number is less than a threshold value, T_i, the node becomes a CH for the current round. The threshold is given by

$$T_i = \begin{cases} \frac{k}{N - k*(r \bmod (N/k))} & if\ i \in G \\ 0 & otherwise \end{cases}, \tag{1}$$

where k is the expected number of CHs, r is the number of the current round, and G is the set of nodes that have not been selected as a CH in the last (N/k) rounds.

All elected CHs broadcast an advertisement message to the rest of the nodes in the network that they are the new CHs. All the non-CH nodes, after receiving this

advertisement, decide on the cluster to which they want to belong. This decision is based on the signal strength of the advertisement.

4 Inaccuracy in the CH Selection Algorithms

We establish a stochastic process model to inspect the system behavior. Assume that there are N nodes in the sensor network, and the corresponding optimal value of the number of clusters is k. Thus each node elects itself to be a cluster head at the beginning with probability $p=k/N$. The operation of LEACH is periodic. Without losing generality, we assume (N/k) is an integer. A cycle consists of R consecutive round. For simplicity, we only discuss the system behavior in one cycle.

Since CH selection only happens at the beginning of each round, a discrete integer r is adopted to represent a round ($r=1, 2, \ldots, R$). Define $N(r)$ to be the number of CH candidates before the cluster selection at round r, and $H(r)$ to be the number of cluster head after the cluster selection at round r. Then $\{H(r), r=1, 2, \ldots, R\}$ is a stochastic series representing the number of cluster head at round r.

According to LEACH, the node should choose to become a cluster head at round r with probability

$$p(r) = \frac{k}{N - k * r} = \frac{p}{1 - p * r}. \tag{2}$$

Obviously, the distribution of number of CHs at round r follows a binomial distribution $B(N(r),p(r))$:

$$P(H(r) = k) = C_{N(r)}^{k}(p(r))^{k}(1 - p(r))^{N(r)-k}. \tag{3}$$

Therefore, the mean of $H(r)$ is

$$\begin{aligned} \eta(r) = E(H(r)) &= \sum_{i=0}^{N(r)} i * P\{H(r) = i\} \\ &= \sum_{i=0}^{N(r)} i * C_{N(r)}^{i}(p(r)^{i})(1 - p(r))^{N(r)-i} \\ &= N(r) * p(r). \end{aligned} \tag{4}$$

Since the cluster head selection algorithm is full distributed, a single node does not know the exact value of $N(r)$ at current round. The algorithm adopts $(N-k*r)$ as the approximate value of $N(r)$.Thus

$$\eta(r) = N(r) * p(r) = (N - k * r) * \frac{k}{N - k * r} = k. \tag{5}$$

Equation (5) implies the expected number of clusters per round is a constant k. According to [3], the optimal value of k is pre-determined in order to ensure the total energy of the network is minimized.

However, note that $N(r)$ used in equation (5) is an approximate value. Assume the number of CHs at round j is H_j ($H_0=0$), the real value of $N(r)$ is

$$N(r) = N - \sum_{i=0}^{r-1} H_j \neq N - k * r. \tag{6}$$

While the probability with which each node self-select itself as a CH is still $p(r)$. Thus the mean of $H(r)$ is

$$\eta(r) = N(r) * p(r) = (N - \sum_{j=0}^{r-1} H_j) * \frac{k}{N - k * r} \neq k, \tag{7}$$

viz. the expected number of clusters per round is not a constant in fact.

5 I-LEACH

According to the analysis in section 4, we know that because each node calculates $p(r)$ using an approximation value of the number of CH candidates, the expected number of clusters per round is not a constant. This inaccuracy is determined by the inherent characteristic of the distributed self-selection method of LEACH. In this section, we propose an improved clustering scheme (I-LEACH) to guarantee the expected number of clusters per round always equals the optimal target value.

5.1 Improvement Details

The operation of I-LEACH is similar to LEACH, consisting of set-up phase and steady-state phase. However, it differs from LEACH in the following aspects. In the steady-state phase, the cluster member node transfers the information whether it has been a CH node to its cluster head. Then the cluster head sends the aggregated data with this history information to the BS. After receiving all the data from the CHs at round *r-1*, the BS can count the number of nodes which have been a cluster head, then can calculate the probability of the cluster head selection at next round r:

$$p'(r) = \frac{k}{N - \sum_{j=0}^{r-1} H_j}. \tag{8}$$

At the end of round *r-1*, the BS broadcasts a short message that contains the value of $p'(r)$, when it sends out synchronization pulses to the network. Thus at round r each candidate node will become a cluster head with probability $p'(r)$. Using (4), the expected number of CHs per round in I-LEACH is

$$\eta_{imp}(r) = N(r) * p'(r) = (N - \sum_{j=0}^{r-1} H_j) * \frac{k}{N - \sum_{j=0}^{r-1} H_j} = k. \quad (9)$$

That is to say, I-LEACH ensures that the expected number of CHs per round equals the pre-determined optimal value *k*.

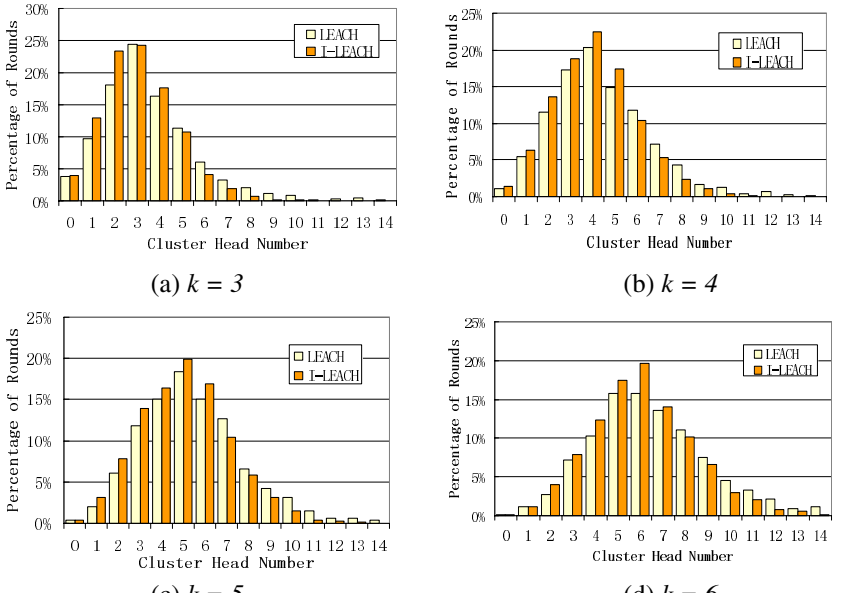
5.2 Simulation Results

We used the network simulator Ns-2 [8] to evaluate I-LEACH and compared it to LEACH. I-LEACH was implemented based on LEACH source code [9] released by the authors of [3].

As a simulation example, a wireless sensor network consisting of 100 nodes was used, where the nodes were randomly distributed between (*x=0, y=0*) and (*x=100, y=100*) with BS at location (*x=50, y=175*).

Both LEACH and I-LEACH are simulated 100 times on the same network with *k=3, 4, 5, 6* (the expected number of cluster per round) separately.

Fig. 3 (a-d) compares the number of CHs with LEACH to I-LEACH. It is worth mentioning that the maximum cluster number produced by I-LEACH was just 14 while the cluster number produced by LEACH ranged from 0 to 35. In Fig. 3, the percentage of cluster head number greater than 14 is negligible. It is clearly that the varying range of the CHs number of I-LEACH is narrower than that of LEACH.



(a) $k = 3$ (b) $k = 4$

(c) $k = 5$ (d) $k = 6$

Fig. 3. CH Number Comparison

Fig. 4 shows that the average cluster head number of I-LEACH is much closer to the target value. Fig. 5 shows that the coefficient of variation of I-LEACH is lower than that of LEACH. Both indicate I-LEACH can guarantee more stable number of CHs than LEACH.

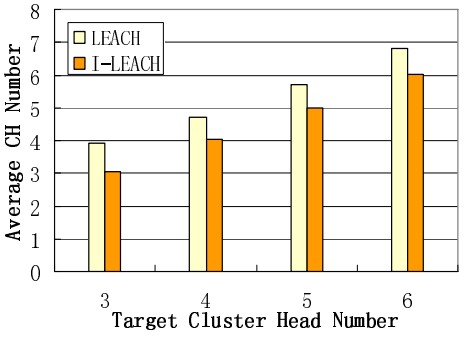


Fig. 4. Average CH Number

Fig. 5. Coefficient of variation of CH Number

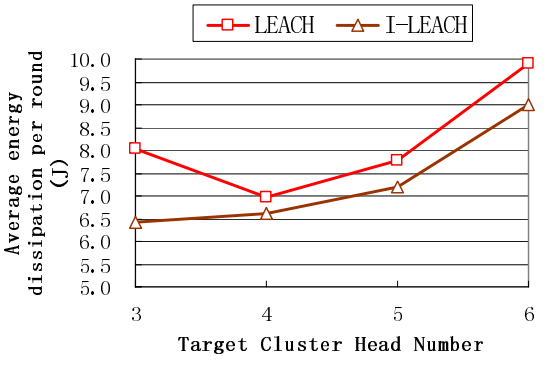


Fig. 6. Average energy dissipation per round for different target CH number

Fig. 6 and Fig. 7 illustrate the average energy dissipation per round and system lifetime (simulation seconds) in various scenarios respectively. These graphs show that the optimum number of clusters is around 4-5 for the 100-node network, and I-LEACH is more energy efficient than LEACH in each case.

Meanwhile, since the main operation of I-LEACH is executed by BS, the network energy dissipation introduced by the improvement algorithm is very small (less than *1%*).

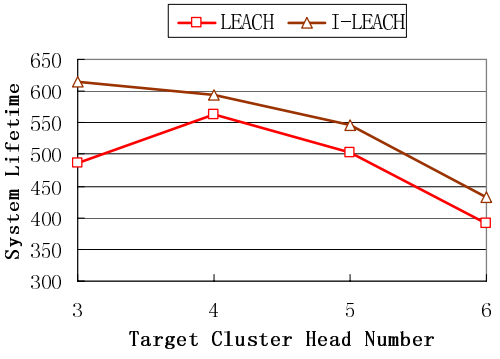


Fig. 7. System lifetime (simulation seconds) for different target CH number

6 Conclusion

In this paper, we analyzed the cluster number variability problem in LEACH. A slight inaccuracy was spotted in the computation of the node self-selected probability per round. To solve such inaccuracy, we proposed a clustering scheme I-LEACH. The experimental results show that our scheme produces a well distributed number of cluster head concentrating in a narrow range around the target value while introducing very small energy dissipation. Since it has been shown in [3] that there exists an optimal value for individual wireless sensor networks, which leads to the minimum energy dissipation, I-LEACH will provide higher energy efficiency than LEACH.

References

1. Jamal N. Al-Karaki, Ahmed E. Kamal: Routing Techniques in Wireless Sensor Networks: A Survey. IEEE Wireless Communications, Vol. 11, No. 6, (2004) 6-28
2. W. Heinzelman, A. Chandrakasan and H. Balakrishnan: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. Proc. 33rd Hawaii Int'l. Conf. System Sciences, Maui, HI (2000)
3. W. B. Heinzelman, A. P. Chandrakasan, H. Balakrishnan: An Application-Specific Protocol Architecture for Wireless Microsensor Networks. IEEE Tran. On Wireless Communications, Vol. 1, No. 4, (2002) 660-670
4. Quanhong Wang, Hossam Hassanein and Glen Takahara: Stochastic Modeling of Distributed, Dynamic, Randomized Clustering Protocols for Wireless Sensor Networks. In: Proceedings of the 2004 International Conference on Parallel Processing Workshops, (2004) 1-8

5. Quanhong Wang, Kenan Xu, Hossam Hassanein and Glen Takahara: Swatch: A Stepwise Adaptive Clustering Hierarchy in Wireless Sensor Networks. NETWORKING 2005: 4th International IFIP-TC6 Networking Conference, Waterloo, Canada (2005)
6. Ying Wang, Mudi Xiong: Monte Carlo Simulation of LEACH Protocol for Wireless Sensor Networks. In: Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, (2005)
7. Yongtao Cao, Chen He: A Distributed Clustering Algorithm with an Adaptive Backoff Strategy for Wireless Sensor Networks. IEICE Trans. Commun., Vol. E89-B, No. 2, (2006) 609-613
8. Ns-2: Network simulator-2. [Online]. Available: http://www.isi.edu/nsnam/ns/
9. MIT uAMPS LEACH ns Extensions. [Online]. Available: http://www.ece.rochester.edu/research/wcng/code/index.htm

A Multipath Routing Algorithm for Wireless Sensor Networks

Jinglun Shi[1,2]

[1] Department of Electronic Engineering, Jinan University, PS 510632, Guangzhou, China
jlshi@scut.edu.cn
[2] School of Computer Engineering, Seoul National University, PS 151744, Seoul, Korea
jlshi@mmlab.snu.ac.kr

Abstract. Wireless Sensor Network (WSN) is expected to have a significant impact on the efficiency of military and civil applications such as target field imaging, intrusion detection, weather monitoring. Sensors with low cost, low power and multifunction are expected to last until their energy drains, so energy-conserving forms of communication and computation are essential to the sensor node lifetime. Given the unreliable nature of the wireless channel and the high failure rate of the individual sensors, a fault tolerant routing protocol with energy-efficiency is expected to overcome these problems. In this paper, we propose an energy efficient multipath routing algorithm (EMRA) for wireless sensor networks. EMRA can efficiently find a disjoint multipath and provide a protection for routing failure. From the observed results, EMRA performs well in terms of average dissipated energy and delay to set up backup path.

1 Introduction

A WSN consists of a large number of densely deployed sensor nodes with limited resource: battery power, computation, communication range and memory. It can be widely used in the areas of medical care, military, and disaster recovery/relief. Routing [1] technique in WSNs is a challenge for the inherent characteristics that distinguish these networks from other wireless networks like mobile ad hoc networks. First due to the large number of sensor node, it is not possible to build a global addressing scheme for the deployment of a large number of sensor nodes as the overhead is high. Furthermore, sensor nodes are tightly constrained in terms of energy, processing, and storage capacities. Thus, it requires saving and managing resources carefully. Third, sensor networks are application-specific (i.e., design requirements of a sensor network change with application). For example, the challenging problem of low-latency precision tactical surveillance is different from that of a periodic weather monitoring task. Finally, data collected in WSNs is typically based on common phenomena, so there are a lot of redundancies. Such redundancies need to be exploited by the routing protocols to improve energy and bandwidth utilization.

Earlier works have explored the design of mechanisms for single-path routing in WSN [4-6]. But with single path, fault tolerance can not be provided because the continuity of end-to-end communication can not be maintained without routing protection and restoration techniques. In [3, 4], to route around failed nodes, periodic,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 438 – 446, 2006.

low-rate, flooding of events is needed to enable local re-routing. But as we know, energy efficiency is an important performance. Such flooding can adversely impact the lifetime of network. It is desirable to find alternative techniques to provide greater resilience in the presence of failure.

Motivated by reliability and energy efficiency requirements, this paper proposed an energy efficient multipath routing algorithm for wireless sensor networks to increase resilience to node failure. In the design, an array of gradients is set up from the sink node to the source node. According to the array of gradients and our schemes, a disjoint multipath can be constructed between the sink node and the source node quickly. Our analysis and simulation show that the disjoint multipath routing algorithm can perform well in the routing overhead, delay to set up paths and average dissipated energy.

The remainder of this paper is organized as follows. A brief review of multipath routing is presented in section 2. In section 3, our energy-efficient multipath routing algorithm is proposed. The performance evaluation is presented in section 4. In section 5, our paper is concluded, and future related topics are discussed.

2 Multipath Routing

Multipath routing means that multiple paths between source and destination are established. Multipath routing has been widely studied in wired and wireless networks [8, 10, and 11]. According to [10], in Ad hoc networks multipath routing is better suited than single path in stability and load balance. Many analyses have shown that the multiple path routing algorithms can increase network throughput and decrease message delay. But in wireless sensor networks, as we concerned, energy efficiency and reliability are important for the limited energy and weakly connection. With multipath, if the working route is broken, the source node doesn't need to cost more time to find a new path for routing, just choose the other path, which can provide a more reliable connection, at the same time save energy.

Based on Directed Diffusion, a disjoint multipath and a novel braided multipath to enable energy efficient recovery from failure of the path between source and sink (DM) are proposed in [2]. It set up the disjoint multipath by alternate path reinforcement. The sink node sends alternate path reinforcement to its next most preferred neighbor. And the neighbor will propagate the reinforcement to its neighbor in the direction of the source. By this mechanism, a disjoint multipath can be constructed. A braided multipath is constructed by relaxing the requirement for node disjointedness and related braided multipath scheme. In the algorithm, the disjoint multipath is constructed during the reinforcement happened.

In our paper, we only consider the disjoint multipath cases, because as state in [13], the braided multipah can easily be transformed into disjoint multipath. Our algorithm is also based on Directed Diffusion. By the array of gradients, we construct the disjoint multipath during the exploratory data message from source to sink. From our analysis and performance evaluation, it can set up the multipath more quickly and be more energy-efficient.

3 Energy-Efficient Multipath Routing Algorithm

Wireless sensor networks typically consist of a large number of nodes, work at a very low data rate, it needs not to assign unique ID for each node. Additionally, for the data- centric – routing to and from a specific node is not required. Similar to DD [3], and DM [2], EMRA is designed to use only the localized information (gradients) to find disjoint paths to protect the working path.

3.1 Array of Gradients

In EMRA, we find the disjoint multipath by the gradients. The disjoint multipath is used to provide stability routing, and the array of gradients is used to control the overhead and energy consumption for exploratory data message. Following are the details.

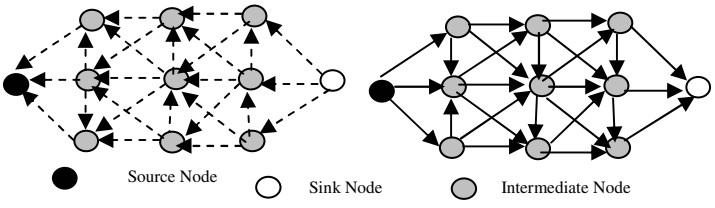


Fig. 1. An example of interests flooding

Firstly, the sink node floods the interest, and interest is diffused in the networks. The intermediate nodes will set up an array of gradients. Here an array of gradients is not for all neighbors from which the interest is received, just for the K neighbors according to the delay time. K is defined as the maximum number of gradients maintained at each node, which means the number of gradient for each node should not more than K. In the mediate node, m gradients are saved, where $m \leq K$. In the array, the gradients are ordered by the increase of delay time. Following are the case study for K=*2*.

If the K=2, then the array of gradients for each intermediate node is set up as Fig. 2.

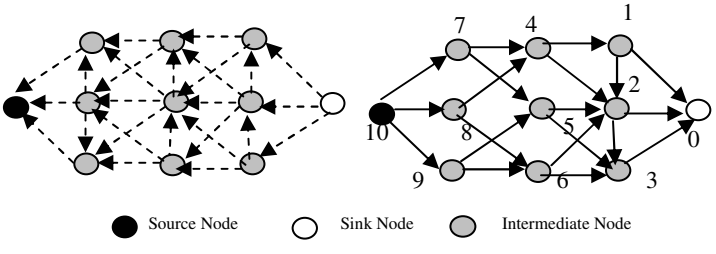


Fig. 2. An Example of gradients as K=2

From the Table 1, we can see the detail value for each intermediate node as K=2. From it, we can get that the array of gradients for each node are ordered with different level. At the same time, we can get that as the K get higher, more gradients is needed to set up.

Table 1. Gradients for each intermediate

Node number	Array of gradients	Array of Levels
Node 1	[0,2]	[1,2]
Node 2	[0,3]	[1,2]
Node 3	[0]	[1]
Node 4	[1,2,5]	[1,2,3]
Node 5	[3,2,1]	[1,2,3]
Node 6	[3,2]	[1,2]
Node 7	[4,5,8]	[1,2,3]
Node 8	[4,6,5]	[1,2,3]
Node 9	[6,5,8]	[1,2,3]

3.2 Setting Up Multipath

If the source receives the interest, it sends the exploratory data message to each neighbor. When the intermediate nodes get the exploratory data message, they will obey four rules to forward it:

i. Each node will forward the exploratory data message to one of its neighbor according to its array of gradients; only one neighbor with lowest delay will be chosen.
ii. Each node will reject the exploratory data message if the same data message has been forwarded already.
iii. If the next neighbor with the lowest delay refuses the exploratory data message by replying with a reject message, the second next neighbor in the array of gradients will be chosen.
iv. If all neighbors in the array of gradients refuse the exploratory data message, the node will give up forwarding the data message and delete the gradients.

The K is crucial for the message forwarding to the sink node. The higher K is, more exploratory messages need to forward, and the number of multipath may be higher. At the same time, more energy will be consumed. The multipath will be guaranteed disjoint by the rules i and iii, both of them can assure that the multipath will have no the same intermediate node. The cases for the K=2 and K=3 are shown in Fig.3 and Fig.4.

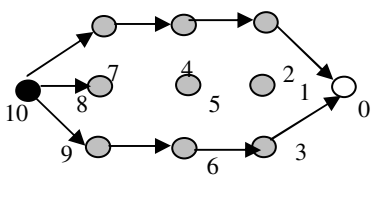


Fig. 3. An Example of exploratory messages as K=2

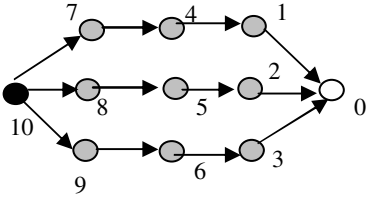


Fig. 4. An example of exploratory messages as K=3

3.3 Reinforcement

If the sink node gets multiple exploratory messages, it means that a disjoint multipath is found, otherwise, only one path is found. If a multipath is found, it will reinforce multiple neighbors with different level, where the reinforcement of level is set 10^{M-i}, where i represents the time order of getting the exploratory message, M represents the number of multipath. Then the reinforcement message is forwarding reversely hop by hop to the source node. When the source node gets the reinforce messages with the different levels, it can get multiple paths to route the related information. In our paper, only one backup path is reinforcement, which means $M \le 2$, if M=2, a backup path is found. And the multipath is disjoint. Similarly, a negative reinforcement message is used to remove a link from a path. Fig.5 and Fig.6 show the results of positive reinforcement with different level.

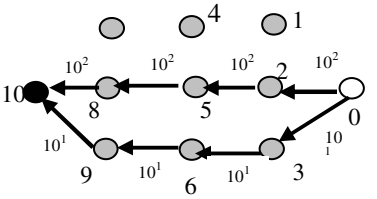


Fig. 5. The disjoint multipath as K=3, M=2

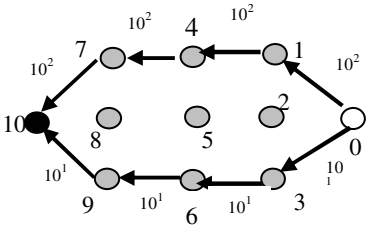


Fig. 6. The disjoint multipath as K=2, M=2

4 Performance Evaluation

Our test consists of two parts. Firstly, two inner performances are tested: one is the number of multipath with the value of K and the network size, the other is the ratio of backup path and primary path's length. Secondly, some performances such energy efficiency and delay to set up a backup path between EMRA, DD, and DM are evaluated.

To study the impact of the value of K on EMRA, we set different values of K in different network size. From the number of multipath found with EMRA, we can get the related performance of K. In Fig.7, we can get that as 50nodes/ 200×200 and 100 nodes/ 200×200, there are little difference in the number of the multipath found with EMRA. As the density gets higher, the difference is still very small. From these, we can conclude that the value of K has little infection on the number of multipath found by EMRA. At the same time, higher K needs more exploratory messages. So in our following experiments, K is set to 2.

To show the quality of the backup path, parameter Ratio of Backup Path and Primary Path's Length is used. In our test, only one backup path is setup. If the ratio is high, which means the quality of the backup path is lower, else means the quality is higher. From Fig. 8, we can see that, at lower densities, the backup paths are relatively longer since fewer alternate paths exist in the topologies, and a long backup path is chosen. It tells that in density network, the backup path found by EMRA is high quality.

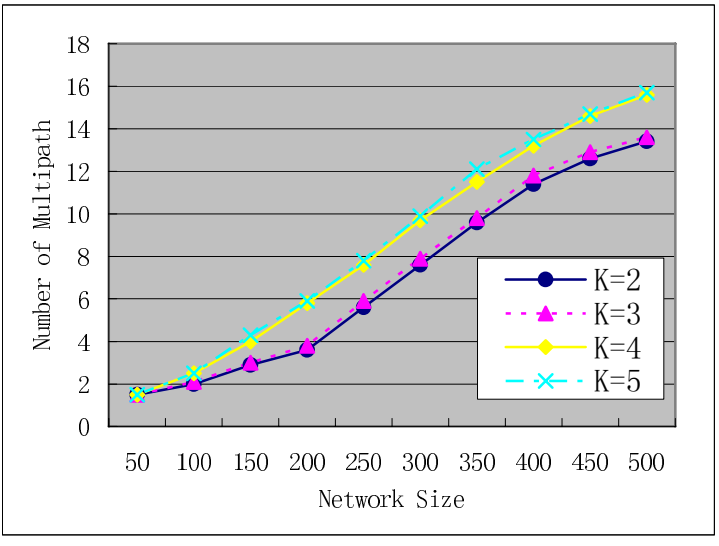


Fig. 7. Number of multipth in different value of K and network size

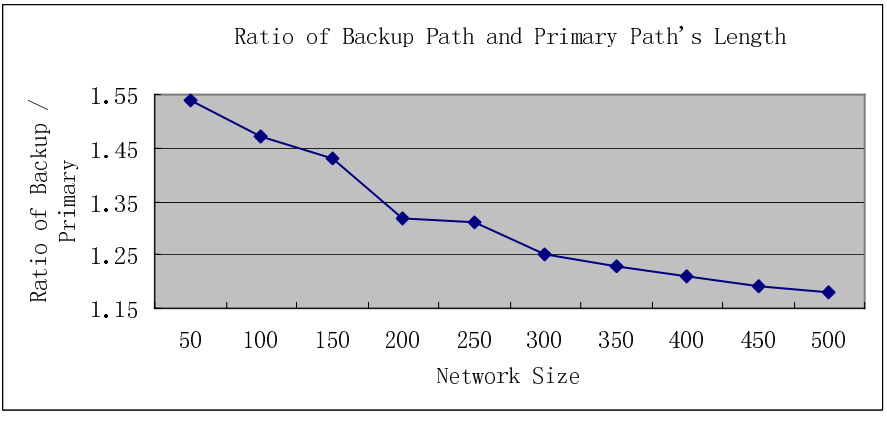


Fig. 8. Ratio of backup path and primary path's length

Two metrics are chosen to analyze the performance of EMRA and to compare it to other algorithms: Average Dissipated Energy measures the ratio of total dissipated energy per node in the network to the number of distinct events seen by sinks. This metric is used to show the average work done by a node in delivering useful tracking information to the sink. It also indicates the overall lifetime of sensor nodes. Delay Time to Setup Path(s) measures the average delay used for setting up path(s). The delay to setup a multipath is an important performance for a multipath routing algorithm. In our test, only one path is set up in DD, and the number of Multipath in EMRA and DM is 2, which means only one backup path was set up. The delay time

to setup path includes three parts: time used for propagation the interests, time used for the exploratory data messages, and time used for reinforcement.

Fig.9shows the average dissipated energy per packet as a function of network size. In the figure, the average dissipated energy per event decreases as the network density increases. DM costs more energy than other algorithms. As the network's density is small such as 50nodes and 100 nodes, the values of DD and EMRA are close. However as the network density becomes higher, EMRA becomes more energy-efficient than DD.

Fig.10 shows the delay to setup path(s). From it, we can conclude that DD has the minimum delay among DD, EMRA, and DM for the reason that it just needs to find one path. DM costs the maximum delay because in DM, it try to find the backup path

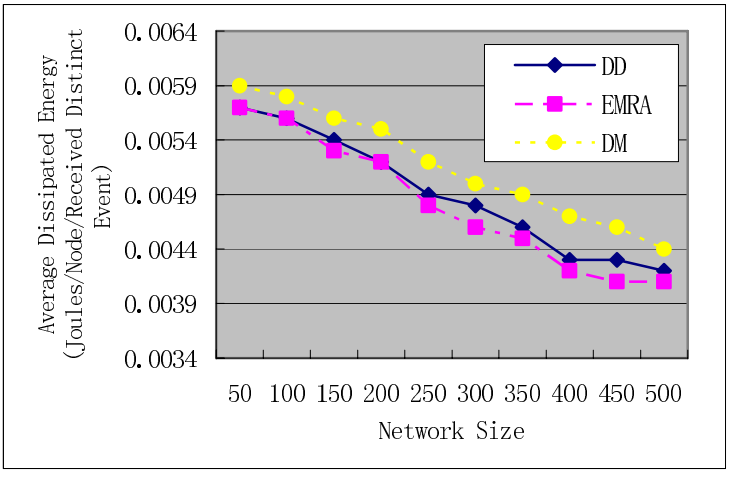


Fig. 9. Average Dissipated Energy with network size

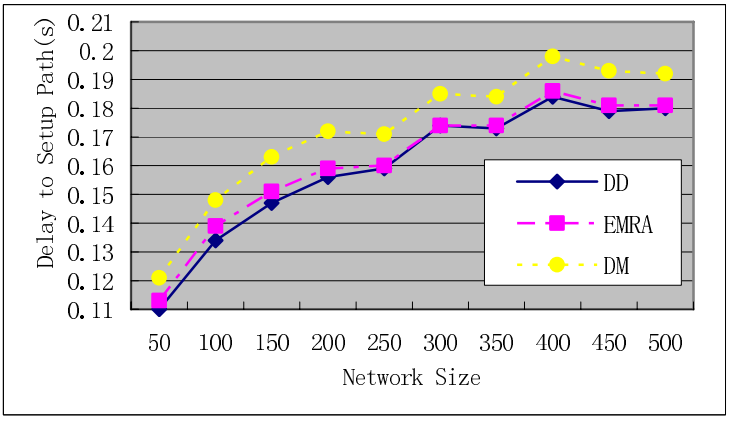


Fig. 10. Delay to set up path(s)

after the sink node gets the exploratory message, and each node tries their neighbors one by one for the backup paths. In the figure, we can see that delay to setup multi-path in EMRA is close with the delay in DD, especially the network's density is high, which also proves that EMRA is very efficient.

5 Conclusions and Future Work

In this paper, we propose an energy-efficient multiple paths routing algorithm (EMRA). Compared with DM that finds the multiple paths in the reinforcement phase, EMRA can find disjoint multiple paths when the source initially disseminates the exploratory data messages. Compared with DD, the number of exploratory data messages forwarded is decreased efficiently in EMRA. From the analysis and the simulation results, EMRA performs well in terms of average dissipated energy and delay to set up multiple paths.

Acknowledgements

This work was supported by grant no.R01-2004-000-10372-0 from the BRP of Korea Science & Engineering Foundation.

References

1. Jamal N. Al-Karaki, Ahmed E.Kamal: Routing techniques in wireless sensor networks: a survey. IEEE Wireless Communications (2004).
2. D.Ganesan, R.Govindan, S.Shenker, and D.Estrin: Highly-resilient, energy-efficient multi-path routing in wireless sensor networks. ACM Mobile Computing and Communications Review (2001), vol.5, no.4.
3. Fabio Silva, John Heidemann, Ramesh Govindan, and Deborah Estrin: Directed diffusion. USC/ISI Technical Report ISI-TR-2004-586 (2004).
4. Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin., John Heidemann, and Fabio Silva: Directed diffusion for wireless sensor networks. ACM/IEEE Transactions on Networking (2003), 11(1):2-16.
5. I. Akyildiz et al.: A survey on sensor networks. IEEE Commun. Mag. (2002), Vol. 40, no.8, pp.102-14.
6. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson: Wireless sensor networks for habitat monitoring. In First ACM Workshop on Wireless Sensor Networks and Applications (2002), Atlanta, GA, USA.
7. W.Heinzelman, A.Chandrakasan and H.Balakrishnan: Energy-efficient communication protocol for wireless microsensor networks. Proc. 33rd Hawaii Int'l. Conf. Sys. Sci. (2000).
8. Lianfang Zhang, Zenghua Zhao, Yantai Shu,Lei Wang,and Oliver W. W. Yang: Load Balancing of Multipath Source Routing in Ad Hoc Networks. IEEE (2002).
9. Jinglun Shi, Zhang Ling, Shoubing Dong, Zhou Jie: A Stability-based Multipath Algorithm For Ad Hoc Networks. IEEE PIMRC (2003), Beijing, China.
10. M. Pearlman, Z. Haas, P. Sholander and S. S. Tabrizi: On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad-Hoc Networks. MobiHoc (2000), Boston, USA.

11. Stefan Dulman, Tim Nieberg, Jian Wu, Paul Havinga: Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks. IEEE (2003).
12. Jinglun Shi, Weiping Liu: A Service-oriented Model for Wireless Sensor Networks with Internet. CIT(2005), pp. 1045-1049.

Improved Dynamic Power Management in Wireless Sensor Networks

Chuan Lin[1,2], Yanxiang He[1], Naixue Xiong[3,1], and Laurence T. Yang[4]

[1] School of Computer,The State Key Lab of Software Engineering, Wuhan University, 430072, PR China
{chlin, yxhe}@whu.edu.cn
[2] School of Mathematics and Statistics, Wuhan University, 430072, PR China
[3] School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), Japan
naixue@jaist.ac.jp
[4] Department of Computer Science,St. Francis Xavier University Antigonish, NS, B2G 2W5, Canada
lyang@stfx.ca

Abstract. Wireless sensor networks play a key role in monitoring remote or inhospitable physical environments. One of the most important constraints is the energy efficiency problem. Power conservation and power management must be taken into account at all levels of the sensor networks system hierarchy. Especially, DPM (Dynamic Power Management) technology, which shuts down the devices when not needed and wake them up when necessary, has been widely used in sensor networks. In this paper, we modify the sleep state policy developed by Simunic and Chdrakasan in [1] and deduce a new threshold satisfies the sleep-state transition policy. Nodes in deeper sleep states consume lower energy while asleep, but require longer delays and higher latency costs to awaken. Implementing dynamic power management with considering the battery status and probability of event generation will reduce the energy consumption and prolong the whole lifetime of the sensor networks. The sensor network consumed less energy in our simulation than that in [1].

1 Introduction

A sensor network is comprised of a large number of miniscule devices equipped with one or more sensors, some processing circuits, and a wireless transceiver. Unlike conventional networks, the main goals are prolonging the life of the network and preventing connectivity degradation through aggressive energy management as the batteries cannot usually be replaced due to the operations in hostile or unattended environments.

To achieve satisfactory network lifetime the problem of energy efficiency needs to be tackled on all levels of the entire network. Many researchers are devoted to reducing power consumption in various aspects of hardware design, data processing, network protocols and operating system [2,3,4,5]. Once the system has been designed, additional energy savings can be achieved by using dynamic power

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 447–456, 2006.

management (DPM), which shuts down the sensor node if no events occur [6]. The basic idea is to shut down devices when not needed and wake them up when necessary. This shutdown yields good savings. But while this power saving method seemingly provides significant energy gains, it is important to remember that sensor nodes communicate using short data packets. The shorter the packets, the more dominance of startup energy [7]. So we have to carefully use Dynamic Power Management (DPM) to get the maximum life of sensor node.

In fact, for example, if we blindly turn the radio off during each idling slot, over a period of time we might end up expending more energy than if the radio had been left on. As a result, there should need other smarter scheme to turn the nodes on/off. In another words, operation in a power-saving mode is energy-efficient only if the time spent in that mode is greater than a certain threshold. There can be a number of such useful modes of operation for a wireless sensor node, depending on the number of the states of the microprocessor, memory, A/D converter, and transceiver.

A variety of DPM techniques have been proposed to reduce the power consumption in sensor nodes and in general battery-powered embedded systems [1,8,9,10,11]. Much work has been done exploiting sleep state and active power management [1,12], Dynamic Voltage Scaling (DVS) [1,10,13] and Dynamic Voltage and Frequency Scaling [10], sentry-based power management [14], an application-driven approach [15], software and operating system power management and battery state awareness power management.

In [1], the authors propose an OS-directed power management technique to improve the energy efficiency of sensor nodes. The node would update the probability of even generation. It is an efficient algorithm for a single node, but not an effective policy for the whole system [16]. We have modified the system model and algorithm proposed in [1] by considering with more factors such as the battery status and the energy waste of wakening up. All these factors decide the sleep state and sleep period of a single node. In addition, the threshold time, which the node should stay in sleep state, derived from [1], is revised.

The organization of this paper will be described as follows. In section 2, the system mode is briefly discussed and the sleep time threshold corresponding to the sleep state is derived. Next in section 3 we propose several power management policies considering the task criticality and the battery state. In section 4, we present the simulation results and compare them to the results in [1]. Finally, section 5 presents our conclusion and some future work.

2 Power-Aware Sensor Node Model

This model describes the power consumption in different levels of node-sleep states. Every component in a node can be in different states. The processor can be in active, idle, or sleep mode, so can the radio, memory and A/D converter. Each node sleep state corresponds to a particular combination of component power modes. In general, if there are N components labeled $(1, 2, ...N)$, each with k_i sleep states, the total number of node sleep states is $\prod k_i$. Every component

power mode has a latency overhead associated with transition to that mode. Therefore each sleep mode is characterized by power consumption and latency overhead. The deeper sleep state of the node, the less power it consumes, and the more the latency it spends. However, from a practical point of view not all sleep states are useful [9].

Let us assume that all sensor nodes will have components such as processor, memory, sensing with A/D converter, radio. So a sensor node will have the following sleep states, as listed in Table 1.

Table 1. Sensor node sleep states (Tx=Trasmit, Rx=Receive)

States	Processor	Memory	Sensor	Radio
S_0	Active	Active	On	Tx/Rx
S_1	Idle	Sleep	On	Rx
S_2	Sleep	Sleep	On	Rx
S_3	Sleep	Sleep	On	Off
S_4	Sleep	Sleep	Off	Off

Table1 describes the component power modes corresponding to five different useful sleep states for the sensor node. These sleep states are chosen based on actual working conditions of the sensor node, for example, it does not make sense to have memory active and everything else completely off. Each sleep state is characterized by latency and power consumption. Nodes in deeper sleep states consume lower energy while asleep, but incur a longer delay and require a higher energy cost to awaken. The design problem is to formulate a policy for transiting between states based on observed events and the battery status, so as to maximize lifetime of the sensor.

This model is called Power-Aware Sensor mode. It's similar to the system power model in the Advanced Configuration and Power Interface (ACPI) standard, which is an open industry specification co-developed by Hewlett-Packard, Intel, Microsoft, Phoenix, and Toshiba that defines a flexible and extensible interface for power management in PCs and related hardware [17].

3 Sleep-State Transition Policy

Let us assume that an event is detected by node k at some time. The node finishes processing the event at t_1 and the next event occurred at $t_2 = t_1 + t_i$. At time t_1 node k decides to transit to sleep state s_k from the active state s_0 , as shown in figure1. We also assume that the process of transiting to sleep states is gradual, not direct, for example, the node is first transited to sleep state s_1 and then state s_2 and so on. Each states s_k has power consumption P_k, and the time transiting to it from the active states and back are given by $\tau_{0,k}$ and $\tau_{k,0}$. By our definition of node sleep states there are several conditions for any $i < j$,

$$P_i > P_j, \tau_{0,j} > \tau_{0,i}, \tau_{j,0} > \tau_{i,0}. \quad (1)$$

Every transition from state s_i to state s_j has a cost in terms of power consumption, denoted by $P_{i,j}$, and of delay overhead, denoted by $\tau_{i,j} = \tau_{0,j} - \tau_{0,i}$. From a practical point of view, the cost associated with transitions from state i to state $j (i < j)$ is usually much lower than that (the cost) associated with the reverse transition, and for the sake of simplicity is neglected [11].

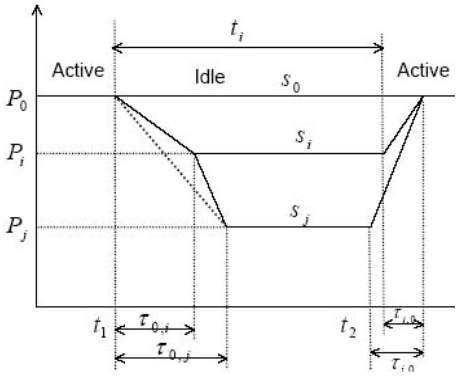


Fig. 1. State transition latency and power

We now derive a set of sleep time thresholds $T_{th,k}$, in which the node sleep states s_k should stay. Transiting to sleep state s_i from state s_0 and awakening up to active state will result in a net energy loss if idle time $t_i < T_{th,i}$ because of the transition energy overhead. This assumes that no productive work can be done in the transition period, which is invariably true. But in [1], the energy saving equation does not consider that during the interval $\tau_{i,0}$, the node can't be working until it enters the active state s_0 completely. Moreover, a DPM policy must take into account the extra energy consumption needed for awakening the node back to active state and should be able to foresee how long it will be remain idle. In our assumption, the active state is directly transited to sleep state s_k and the cost is neglected for reduced computation complexity. Therefore, the energy saving from a state transition to a sleep state and back is given by

$$E_{save,k} = \frac{1}{2}(P_0 - P_k)(t_i - \tau_{0,k} + t_i + \tau_{k,0}) = (P_0 - P_k)(t_i - \frac{\tau_{0,k} - \tau_{k,0}}{2}). \quad (2)$$

Actually the saving energy consumption is the trapezoid area in the figure. Such a transition is only useful when $E_{save,k} \geq \triangle E_{k,0}$, where $\triangle E_{k,0}$ is defined as the additional energy consumption due to awakening the sensor node back to state s_0. It is clear that the node should be in sleep state only when its idle period can be long enough so that the saved energy compensates for the expended transition energy. This implies that the saving energy must be not less than the energy wasted by awakening the node. This leads to the threshold value,

$$T_{th,k} = \frac{1}{2}(\tau_{0,k} - \tau_{k,0}) + \frac{\Delta E_{k,0}}{(P_0 - P_k)}, \quad (3)$$

where $\tau_{0,k} = \tau_{0,1} + \tau_{1,2} + ... + \tau_{k-1,k}$. This equation implies that the longer the delay overhead of transition $s_0 \rightarrow s_k$ or the shorter overhead transition of awakening back to state s_0 the higher the energy-gain threshold. In [1], the author did not consider that awakening a sensor node also need lots of energy and extra time. It is clear that $\triangle E_{i,0} < P_0\tau_{i,0}$, therefore, our threshold will be smaller than that in [1] and there is a greater probability the node will be in the sleep state. Hence it will increase energy consumption saving and prolong the whole lifetime of the sensor networks. The simulation shown in Fig. 5 will prove this.

In the deepest sleep state s_4 the sensor node cannot detect an event or receive a message from the other nodes. When system is in state s_4, there is a chance that some events will get lost. Therefore, whether or not transit to the deepest sleep state s_4 and how to determine the deepest sleeping period T becomes an important issue. It should be noted that in clustering protocols, the cluster head can't be allowed to enter in the deepest sleep state s_4 while the other normal nodes can do so. We can obtain the deepest sleeping period T according to the battery status and the parameter μ defined in [16],

$$T = \mu e^{V_s/V_p}, \tag{4}$$

where V_s denotes the standard working voltage and V_p represents the present voltage of the battery [16]. Therefore, we can define the deepest sleep state period T using any μ as a time counter for state s_4.

To determine the sleep states to which the node will transit, we can refer to the event generation model presented in [1] and the hybrid automata theory [18]. The node in the shallower sleep state would determine its sleep state $s_k(k = 1, 2, 3, 4)$, according to event generation probability λ_k using the formula

$$P_k(T_{th}, 0) = e^{-\lambda_k T_{th}}, \tag{5}$$

where $P_k(T_{th,0})$ denotes the probability of no events occurring in a node sensing area C_k over threshold interval T_{th}. Let $P_{th,k}(T_{th})$ be the probability that at least one event occurs in time t at node k,

$$P_{th,k}(T_{th}) = 1 - P_k(T_{th}, 0) = 1 - e^{-\lambda_k T_{th}}. \tag{6}$$

The probability of at least one event occurring is an exponential distribution characterized by a spatially weighted event arrival λ_k, which indicates the mean rate of event generation (time elapsed divided by the total number of events registered by node k), and the value of λ_k may change with time. This is an important parameter used to determine which sleep state will the node enter. If $P_k(T_{th}(i), 0)$ is bigger than a fixed value P, which is equal to 0.5 in our scheme, the node will enter sleep state s_i.

Because the hybrid automata is the formal representation of a hybrid system, as a finite-state machine, where the states are represented as a finite set of control states [18], we can use the hybrid automata to represent our sleep state transition policy as shown in Fig. 2.

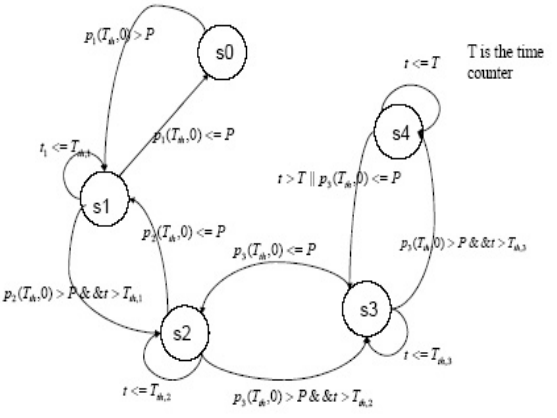


Fig. 2. Graphic representation of a hybrid automation for sleep states transition policy

4 Awaken Up Policy

In our method we can awaken the idle sensor node in several ways, including event driven, message driven policies and a time counter.

An event-driven policy works under the shallower sleep state. When a critical event such as violent changes in temperature or a signal generated by a moving object occurs, the sensor produces an interrupt and awakens the CPU. CPU processes the signal with data fusion algorithm or transmitting to other nodes, then goes to sleep again.

A message-driven policy works under $k = 1$ or 2, because in these sleep states the receiver is still on. When a node j sends a waken-up message to its adjacent nodes, node i is in sleep state s_{k_1} and node k is in state s_{k_2} for transmitting data. Upon receiving this message, node i will check to see if its sleeping time has been more than $T_{th}(k_1)$, if that is true, the node will wake up to state s_0 and send an acknowledgement message to node j, otherwise, it will wait until $t > T_{th}(k_1)$. Does the same node k. Node j will send data to the first one to respond. This method can avoid the huge energy consumption caused by packet transmission failure. We propose a message-driven algorithm that is shown in Fig. 3. The time counter technique can only be used to awaken the node in the deepest sleep state s_4.

5 Simulation and Analysis

We suppose a 100 nodes system distributed uniformly and randomly over a $50m$ by $50m$ area. We also assume that sensor nodes are capable of transmitting directly to the sink node, while do not consider the multihop operation.

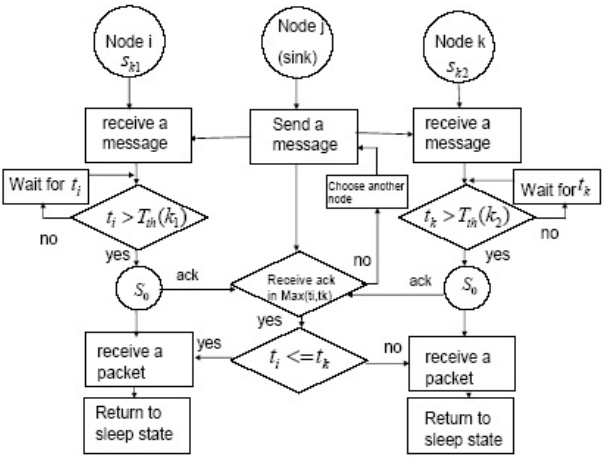


Fig. 3. Diagram of the message-driven policy

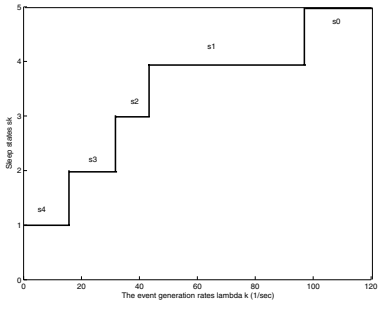


Fig. 4. Sleep states based on the event generation rates

Fig. 4 shows a node's sleep states based on the event arrival rates $lambda_k$. If $\lambda_k < 17.3$, transition to state s_4 is always possible expect for cluster head, it will enter state s_3 if $17.3 \leq \lambda_k \leq 34.7$, s_2 for $34.7 \leq \lambda_k \leq 46.2$, s_1 for $46.2 \leq \lambda_k \leq 99$ and come back to active state s_0 if $\lambda_k \geq 99$.

Fig. 5 shows the normalized energy consumption according to the event arrival rate λ_k. It can be seen from that, with a low event arrival rate, the energy consumption with different sleep states is much less than that with higher event rate. As the event arrival rate increasing, the more nodes will awaken up to the active state that needs more energy. In addition, our modified threshold policy consumes less energy than the original one in [1].

Next we will demonstrate that node energy consumption tracks event probability λ_k. Fig. 6 shows the event frequency with a Gaussian spatial distribution centered around $(25, 25)$, and shows the normalized energy consumption of overall spatial nodes.

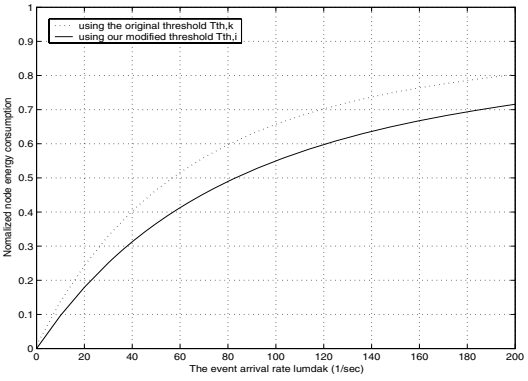


Fig. 5. Normalized energy consumption corresponding to the event arrival rate (modified threshold versus the original in [1])

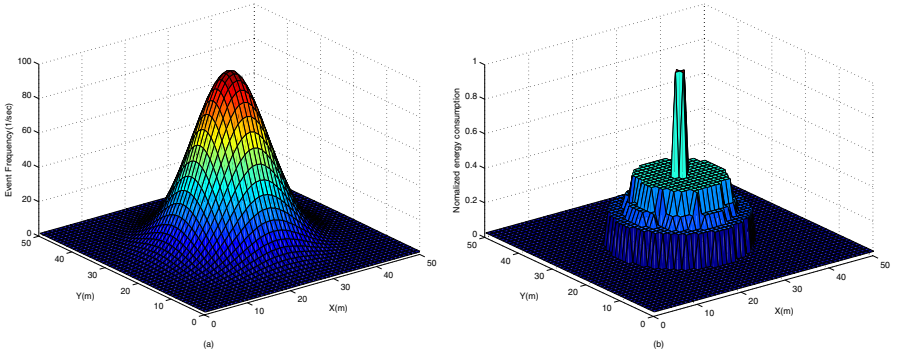


Fig. 6. Simulation of DPM in a sensor network: (a) is the spatial distribution of event arrival rates (Gaussian) and (b) shows the configuration of spatial power consumption in the sensor node

In the scenario without power management, there is uniform energy consumption at all nodes.

6 Conclusion and Future Work

In this paper we modified the dynamic power management sleep state policy of [1], and derived the sleep threshold with respect to the extra energy cost for returning the sleep node to an active state and the battery status.

However, several aspects needed to be done in the future. First, one drawback to our scheme is that we do not analyze the latency, it is very important parameter in a sensor network. Second, we need to address the question of how to avoid event missing effectively when the nodes are in sleep states. Finally, we intend to consider the multihop operation in our DPM sensor network.

References

1. A.Sinha, A.Chandrakasan: Dynamic Power Management in Wireless Sensor Networks. IEEE Design and Test of Computers, Vol.18, Issue 2, pp.62-74, March-April, 2001
2. B. H. Calhoun, D. C. Daly, N. Verma, D. Finchelstein, D. D. Wentzloff, A. Wang, S.-H. Cho, and A. P. Chandrakasan: Design Considerations for Ultra-low Energy Wireless Microsensor Nodes. IEEE Transactions on Computers, June 2005
3. K. Sohrabi, J. Gao, V. Ailawadhi, and G.J. Pottie: Protocols for Self-Organization of a Wireless Sensor Network. IEEE Personal Communications, Vol.7, Issue 5, pp. 16 - 27,October, 2000
4. V. Raghunathan, C. Schurgers, Sung Park, and M.B. Srivastava: Energy-Aware Wireless Microsensor Networks. IEEE Signal Processing Magazine, Vol.19, Issue 2, pp. 40 - 50, March, 2002
5. J. Hill, R. Szewczyk, A. Woo, S. Hollar, D.E. Culler, and K. Pister: System Architecture Directions for Networked Sensors. Architectural Support for programming Languages and Operating Systems, pp.93-104, 2000. Available at http://www.tinyos.net/papers/tos.pdf.
6. L. Benini and G.D. Micheli, Dynamic Power Management: Design Techniques and CAD Tools, Kluwer Academic, NY, 1997
7. I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci: A Survey on Sensor Networks. IEEE Communications Magazine, Vol.40, Issue 8, pp. 102 - 114, August, 2002
8. E.Y.Chung, L.Benini, and G.D.Micheli: Dynamic Power Management Using adaptive Learning Tree. International Conference on Computer-Aided Design (ICCAD), pp.274 - 279, 7-11 November, 1999
9. A.L.A.P. Zuquim, L.F.M. Vieira, M.A. Vieira, A.B. Vieira, H.S. Carvalho, J.A. Nacif, C.N. Jr. Coelho, D.C. Jr. da Silva, A.O. Fernandes, and A.A.F. Loureiro: Efficient Power Management in Real-time Embedded Systems. IEEE International Conference on Emerging Technologies and Factory Automation-RTFA'03, vol.1, pp.496 - 505, 16-19 September, 2003
10. IBM and MontaVista Software: Dynamic Power Management for Embedded System. Ver.1.1, 19-November, 2002. Available at: http://www.research.ibm.com/arl/projects/papers/DPM-V1.1.pdf.
11. C.F Chiasserini, R.R. Rao: Improving Energy Saving in Wireless Systems by Using Dynamic Power Management. IEEE Transactions on wireless Communications, Vol.2, Issue 5, pp.1090-1100, September, 2003
12. B. Brock, K. Rajamani: Dynamic Power Management for Embedded System. IEEE International System-On-Chip (SOC) Conference, pp. 416 - 419, 17-20 September, 2003
13. B.Calhoun, A. P. Chandrakasan: Standby Power Reduction Using Dynamic Voltage Scaling and Canary Flip-Flop Structures. IEEE Journal of Solid-State Circuits, vol. 39, no. 9, September, 2004
14. J.Hui, Z. Ren, B.H. Krogh: Sentry-based Power Management in Wireless Sensor Networks. Second International Workshop on Information Processing in Sensor Networks, pp. 458-472, April, 2003
15. Rodrigo M. Passos, Claudionor J. N. Coelho Jr., Antonio A. F. Loureiro, Raquel A. F. Mini: Dynamic Power Management in Wireless Sensor Networks: An Application-Driven Approach. Second Annual Conference on Wireless On-demand Network Systems and Services (WONS'05), pp. 109-118, January, 2005

16. Ren C. Luo, Liang Chao Tu, Ogst Chen: An Efficient Dynamic Power Management Policy on Sensor Network. Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05), vol. 2, pp. 341 - 344, 28-30 March, 2005
17. Hewlett-Packard, Intel, Microsoft, Phoenix, and Toshiba, Advanced Configuration and Power Interface (ACPI): an Open Industry Specification - Revision 3.0a. Available at: http://www.acpi.info/.
18. T.A.Henzinger: The Theory of Hybrid Automata. Eleventh Annual IEEE Symposium on Logic in Computer Science (LICS), pp. 278-292, July, 1996

A Fast Traffic Planning Algorithm in Lifetime Optimization of Sensor Networks

Yantao Pan, Wei Peng, Xicheng Lu, Shen Ma, and Peidong Zhu

School of Computer, National University of Defense Technology, Changsha, P.R. China
pytmail@126.com

Abstract. The lifetime optimization is a key challenge of sensor networks. Since data transmission is the main energy consumer, it is important to make use of energy efficient communication protocols to prolong the network lifetime. We formalize the lifetime optimization problem to a max flow problem in a directed graph with capacity powers on arcs and vertices. Then we propose a fast algorithm to solve this problem. The method gives the value of maximum lifetime exactly. The time complexity is $O(|V|^2 \cdot |A|)$

1 Introduction

The development of sensor networks provides us a chance to change the way by which we interact with the physical environments. It is a key challenge to maximize the lifetime of a sensor network. The lifetime is usually defined as the time from the deployment of a network to its partition when there exit nodes that cannot send their data to the sink. Literatures [1-2] investigate the upper bound and the expectation of a sensor network's maximum lifetime. Literatures [3-6] propose heuristic algorithms to maximize the lifetime approximately. However, none of those approaches provides the maximum lifetime exactly. In this paper, we exploit this problem by max flow model and propose a fast algorithm to achieve an optimal traffic planning under which the network lifetime is maximized.

The rest of this paper is organized as follows. In section 2, we propose the statement and formulation of the problem. In section 3, we propose the algorithm. In section 4, some concluding remarks are made.

2 Problem Statement and Formulation

Consider a group of wireless static sensor nodes V randomly distributed in a region. Each node $v \in V$ has a limited battery energy supply which is mainly used for data communications. Assume that each node has fixed communication power and transmission radius. Let $p(v)$ be the total quantity of data that vertex v can send. It is limited by initial energy supply and communication power. Let $w(v)$ be the data-generating rate at a source node v. A sensor network can be defined as a directed graph with vertex and arc powers.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 457 – 467, 2006.

Definition-1. A sensor transportation network (It will be shown as SensorNet for short in the rest of this paper.) is defined as the form of $N^s = (V, A, p, X, Y)$, where

1. $G(V, A)$ is a connected simple directed graph, and
2. $p(v): V \mapsto \overline{\mathbb{R}^-}$ is residuary transmission capacity of vertex v, and
3. X, Y are source and sink sets.

Additionally, let $I = V \setminus (X \bigcup Y)$ be intermediate set of N^s. Let $\alpha(H) = \{(u,v) \in A \mid v \in H, u \in V \setminus H\}$ and $\beta(H) = \{(v,u) \in A \mid v \in H, u \in V \setminus H\}$ be the arc sets entering and leaving vertex set H. For $f(a): A \mapsto \overline{\mathbb{R}^-}$, let $f^{\alpha}(H) \triangleq \sum_{a \in \alpha(H)} f(a)$ and $f^{\beta}(H) \triangleq \sum_{a \in \beta(H)} f(a)$. Then, f is a flow of N^s if $f^{\beta}(v) \le p(v), \forall v \in V$ and $f^{\beta}(v) = f^{\alpha}(v), \forall v \in I$. Let $val(f) \triangleq f^{\beta}(X) - f^{\alpha}(X)$ be the value of f.

Definition-2. Let N^s be a SensorNet. Flow f is defined as a traffic planning of N^s if there is a T so as to $f^{\beta}(x) - f^{\alpha}(x) = T \cdot w(x), \forall x \in X$ and $p(v) = f^{\beta}(v), \exists v \in V$. T is called the lifetime of N^s under the traffic planning f. If there is no f' and its corresponding T' so as to $T' > T$, f is an optimal traffic planning and T is the maximum lifetime.

Definition-3. Let N^s be a SensorNet and assume $T > 0$. $N_T^s = (V', A', c, p', s, t)$ is defined as the transformed network of N^s about T, if following conditions are satisfied. (1) $V' = V \bigcup \{s, t\}$. (2) $A' = A \bigcup A_s \bigcup A_t$, $A_s = \{(s,x) \mid x \in X\}$, $A_t = \{(y,t) \mid y \in Y\}$. (3) $c(u,v) = \begin{cases} T \cdot w(v), u = s, \\ \infty, otherwise. \end{cases}$. (4) $p'(v) = \begin{cases} p(v), v \in V \setminus Y, \\ \infty, otherwise. \end{cases}$.

Lemma-1. Let N^s be a SensorNet and let T^* be its maximum lifetime. Suppose N_T^s is the transformed network of N^s about T, and $f_{\max}^{s,T}$ is a maximum flow of N_T^s. Then $T \le T^* \Leftrightarrow val(f_{\max}^{s,T}) = \sum_{x \in X} T \cdot w(x)$.

Proof. $\Rightarrow$

Since $T \le T^*$, there is a flow f of N^s such that $f^{\beta}(x) - f^{\alpha}(x) = T^* \cdot w(x), \forall x \in X$. Let $f'(u,v) = \begin{cases} T \cdot w(v), (u,v) \in A_s, \\ f^{\alpha}(u) - f^{\beta}(u), (u,v) \in A_t, \\ f(u,v), otherwise. \end{cases}$. It is easy to verify that f' is a flow of N_T^s and $val(f') = \sum_{x \in X} T \cdot w(x)$.

Note that $(\{s\}, V'\backslash\{s\})$ is an arc cut of N_T^s whose capacity is not more than $\sum_{x\in X} T\cdot w(x)$. So we get $val(f^{s,T}) \le \sum_{x\in X} T\cdot w(x) = val(f')$ for any flow $f^{s,T}$ of N_T^s. That is to say, f' is a maximum flow of N_T^s and $val(f_{\max}^{s,T}) = val(f') = \sum_{x\in X} T\cdot w(x)$.

$\Leftarrow$ If the statement were not true, we got $T > T^*$. Let $f(a) = f_{\max}^{s,T}(a), \forall a \in A$. It is easy to verify that f is a flow of N^s and $val(f) = val(f_{\max}^{s,T}) = \sum_{x\in X} T\cdot w(x) > \sum_{x\in X} T^*\cdot w(x)$. Then the maximum lifetime of N^s is not less than T according to Definition-2. This contradicts the fact that T^* is the maximum lifetime of N^s. (This completes the proof.)

From Lemma-1, we come to the following theorem.

Theorem-1. Let N^s be a SensorNet and let T^* be its maximum lifetime. Suppose N_T^s is the transformed network of N^s about T and $f_{\max}^{s,T}$ is a maximum flow of N_T^s. Then T^* is the maximum T that satisfies $val(f_{\max}^{s,T}) = \sum_{x\in X} T\cdot w(x)$.

Suppose $\{T_i\}, i = 0,1,2,...$ is a sequence, where $T_0 = 0$ and T_1 is an upper bound of the maximum lifetime. The rest elements are generated by BS algorithm.

```
BS (T_0,T_1)
1  T_low ← T_0 ; T_up ← T_1 ; i ← 0
2  While T_up − T_low > ξ
3    do
4        T_{i+2} ← (T_up − T_low)/2
5        If θ(T_{i+2}) = 0
6            T_up ← T_{i+2}
7        else
8            T_low ← T_{i+2}
9        i = i + 1
10 return (T_up − T_low)/2
```

Fig. 1. BS algorithm

In BS algorithm, ξ is the given precision and $\theta(T) = \begin{cases} 1, val(f_{\max}^{s,T}) = \sum_{x\in X} T\cdot w(x), \\ 0, val(f_{\max}^{s,T}) < \sum_{x\in X} T\cdot w(x). \end{cases}$.

Corollary-1. $\lim_{i\to+\infty}\{T_i\} = T^*$.

According to Corollary-1, we can take a binary search between the lower and upper bounds of the maximum lifetime and finally achieve a value exactly enough if only we could find the maximum flow problem of N_T^s.

N_T^s is a transportation network with vertex and arc powers. The generic form of it is $N=(V,A,c,p,s,t)$. We call it Network for short in the rest of this paper.

Definition-4. $N(f)=(V,A^0,c^0,p^0,s,t)$ is defined as the residual network of N about f, if

1. $p^0(v)=p(v)-f^\beta(v)$ for $\forall v\in V$, and
2. $A^0=A_+^0\bigcup A_-^0$. A_+^0 and A_-^0 are the sets of forward and backward arcs. For each $(u,v)\in A$,
   1) if $f(u,v)=0$, $(u,v)\in A_+^0$ and $c^0(u,v)=c(u,v)$;
   2) if $f(u,v)=c(u,v)$, $(v,u)\in A_-^0$ and $c^0(v,u)=c(u,v)$;
   3) if $0<f(u,v)<c(u,v)$, $(u,v)\in A_+^0$, $c^0(u,v)=c(u,v)-f(u,v)$, $(v,u)\in A_-^0$, $c^0(v,u)=f(u,v)$.

Definition-5. An $s-t$ path P of $N(f)$ is defined as an $s-t$ admissible path, if

1. $c^0(P)>0$, where $c^0(P)\triangleq\min\{c^0(a)\mid a\in P\}$, and
2. for $\forall v\in P\setminus\{t\}$,
   1) if $v=s$ then $p^0(v)>0$,else
   2) for $(u,v),(v,w)\in P$, one of following 3 conditions is satisfied: (1) $(v,w)\in A_-^0$, (2) $p^0(v)>0$, (3) $(u,v)\in A_-^0$.

Let $d^0(v,w)=\begin{cases}c^0(v,w), & (v,w)\in A_-^0\vee(u,v)\in A_-^0,\\ \min\left\{p^0(v),c^0(v,w)\right\}, & (v,w)\in A_+^0\wedge(u,v)\in A_+^0\wedge p^0(v)>0.\end{cases}$ and let $d^0(s,v)=\min\left\{p^0(s),c^0(s,v)\right\}$. The capacity of path P is defined as $d^0(P)\triangleq\min\limits_{a\in P}\left\{d^0(a)\right\}$.

Definition-6. A cut of a Network is defined as a set of arcs and vertices $K=\left\{k\middle|k\in V\bigcup A\right\}$ if the following 2 conditions are satisfied.

1. If all the elements of K are removed from N, there is no $s-t$ path.
2. Any proper subset of K can't satisfy condition 1.

The capacity of cut K is $cap(K)\triangleq\sum\limits_{k\in K\bigcap V}p(k)+\sum\limits_{k\in K\bigcap A}c(k)$. K is a minimum cut if there is no cut K' such that $cap(K)<cap(K')$.

Lemma-2. Let f be a flow of N and let K be a cut of N. Then $val(f) \le cap(K)$.

Proof. Let $K = V_k \bigcup A_k$, where V_k is a cut-vertex set and A_k is a cut-arc set. Then we have $val(f) = \sum_{a \in \beta(V_k)} f(a) - \sum_{a \in \alpha(V_k)} f(a) + \sum_{a \in A_k} f(a) \le \sum_{u \in V_k} p(u) + \sum_{a \in A_k} c(a) = cap(K)$.

Lemma-3. Let f be a flow of N and let K is a cut of N. If $val(f) = cap(K)$, f is a maximum flow and K is a minimum cut.

Proof. Suppose f^* is a maximum flow and K^* is a minimum cut. According to Lemma-2, $val(f) \le val(f^*) \le cap\ \ K^* \le cap\ \ K$. Since $val(f) = cap\ \ K$, f is a maximum flow and K is a minimum cut.

Theorem-2. Let $N(f)$ be the residual network of N about f and let P be an $s-t$ admissible path of $N(f)$. Suppose δ is a number that satisfies $0 \le \delta \le d^0(P)$. Assume $\hat{f}_\delta(u,v) = \begin{cases} f(u,v) + \delta, (u,v) \in P \bigcap A_+^0, \\ f(u,v) - \delta, (u,v) \in P \bigcap A_-^0, \\ f(u,v), otherwise. \end{cases}$. Then $\hat{f}_\delta$ is a flow of N, which is called the adjusted flow of f about P and δ. $val(\hat{f}_\delta) = val(f) + \delta$. Furthermore, if $\delta = d^0(P)$, then $\hat{f}_\delta$ is denoted by $\hat{f}$ and is called the adjusted flow of f about P.

Theorem-3. f is a maximum flow of N if and only if there is no $s-t$ admissible path in $N(f)$.

Proof.

$\Rightarrow$ If the conclusion were not true, we get an admissible path of $N(f)$ denoted by P. According to Theorem-2, there is a $\delta \le d^0(P)$ and its $\hat{f}_\delta$ satisfies $val(\hat{f}_\delta) > val(f)$. This contradicts the given condition that f is a maximum flow.

$\Leftarrow$ Let $H = \{v \mid \text{there is } s\text{-}v \text{ admissible path in } N(f)\}$. Then it is easy to know $s \in H$, $t \in \bar{H} = N(f) \setminus H$. Let $A_k = \left\{(u,v) \in A^0 \middle| u \in H, v \in \bar{H})\right\}$, $V_k = \left\{u \middle| (u,v) \in A_k\right\}$. For $\forall (u,v) \in A_k$, we have $(u,v) \in A_+^0$ because there is an admissible $s-u-v$ path and $v \in H$ if $(u,v) \in A_-^0$. In addition, we have $p(u) = 0, \forall u \in V_k$ because there is an admissible $s-u-v$ path and $v \in H$ for $\forall (u,v) \in A_k$ if $p(u) > 0$. Let $(H, \bar{H}) = \left\{(u,v) \in A \middle| u \in H, v \in \bar{H}\right\}$ and $\overline{A}_k = (H, \bar{H}) \setminus A_k$. Let $K = V_k \bigcup \overline{A}_k$, then K is a cut. It is easy to verify that $f(a) = c(a), \forall a \in \overline{A}_k$. If there are $u \in V_k$ and $v \in H$ so as to $f(u,v) > 0$, there is a backward arc $(v,u) \in A_-^0$ in $N(f)$ and we get an admissible $s-v-u-w$ path for $\forall (u,w) \in A_k$. This lead to $w \in H$ and contradicts the

fact that $(u,w)\in A_k$, so $f(u,v)=0$. Therefore, we have $\sum_{u\in V_k, v\in \bar{H}} f(u,v)=\sum_{u\in V_k, v\in V} f(u,v)=\sum_{a\in\beta(V_k)} f(a)$. If $\exists(v,u)\in(\bar{H},H), f(v,u)>0$, there is an arc $(u,v)\in A_-^0$ so as to $v\in H$. This contradicts the fact that $v\in\bar{H}$, so $f^{\alpha}(H)=0$.

$val(f)=f^{\beta}(H)-f^{\alpha}(H)=f^{\beta}(H)=\sum_{a\in A_k} f(a)+\sum_{a\in\bar{A}_k} f(a)=\sum_{u\in V_k, v\in\bar{H}} f(u,v)+\sum_{a\in\bar{A}_k} f(a)$

$=\sum_{a\in\beta(V_k)} f(a)+\sum_{a\in\bar{A}_k} f(a)=\sum_{u\in V_k} p(u)+\sum_{a\in\bar{A}_k} c(a)=cap(K)$. According to Lemma-3, f is a maximum flow. (This completes the proof.)

3 Fast Algorithm on Optimal Traffic Planning

In this section, we propose a fast algorithm based on MPM method introduced by Malhotra, Pramodh-Kumar and Maheshwari [7]. Key idea of MPM algorithm is to identify the vertex of minimum potential. Then, push flow from it and pull flow to it.

For a Network N and its flow f, we construct its layered network $LN(f)$ firstly. Different from generic layered network, the vertices with $p(u)=0$ must be considered carefully. According to Definition-5, an augment flow could be passed through a vertex u no matter $p(u)$ is zero or not, if only the incoming arc of u where the flow comes from is a backward arc. In addition, if the incoming arc is a forward arc, one of the following conditions must be satisfied to ensure that a flow could be passed through u. (1) $p(u)>0$. (2)There is a backward arc from u to some vertex which does not belong to any constructed layer before.

Definition-7. Suppose $N(f)$ is a residual network. Let $V_0=\{s\}$. $LN(f)=(V^L,A^L,p^0,c^0)$ is defined as a layered network if $V^L=\bigcup_i V_i$ and $A^L=A^0\setminus\{(u,v)\,|\,\exists V_i,\ \text{s.t. } u,v\in V_i\}$, where $V_i, i=1,2,...$ is given by formula (1).

$$V_{i+1}=\left\{v\ \middle|\ \begin{array}{l} v\notin\bigcup_{j=0}^{i}V_j,\ \exists u\in V_i,\ \text{s.t. } (u,v)\in A_-^0\ \vee \\ \left((u,v)\in A_+^0\wedge\left(p^0(v)>0\vee\exists w\notin\bigcup_{j=0}^{i}V_j,\ \text{s.t. } (v,w)\in A_-^0\right)\right)\end{array}\right\} \tag{1}$$

A layered network could be constructed by algorithm LC.

Lemma-4. If $t\notin LN(f)$, there is no admissible $s-t$ path in $N(f)$.

Proof. If the statement were not true, we got an admissible $s-t$ path P. Since $t\notin LN(f)$, there must be two adjacent vertices in $P=\langle s,...,u,v,...,t\rangle$ such that $u\in LN(f)$ and $v\notin LN(f)$. However, according to Definition-5, $v\in LN(f)$ if P is admissible. This leads to a contradiction. We have thus proved the lemma.

LC ($N(f)$)

1 $V_i \leftarrow \varnothing, for \forall i ; V_0 \leftarrow \{s\}$

2 $A^L \leftarrow \varnothing ; V^L \leftarrow V_0 ; i \leftarrow 0$

3 **while** $t \notin \bigcup_{j=0}^{i} V_j$ and $V_i \neq \varnothing$

4 **do**

5 $V_{temp} \leftarrow \varnothing$

6 **for** each $v \in V \setminus \left(\left(\bigcup_{j=0}^{i} V_j \right) \bigcup V_{temp} \right)$

7 **for** each $u \in V_i$

8 **if** $(u,v) \in A^0$

9 **if** $(u,v) \in A_-^0$

10 $V_{i+1} \leftarrow V_{i+1} \bigcup \{v\}$; $A^L \leftarrow A^L \bigcup \{(u,v)\}$

11 **else if** $p(v) > 0$

12 $V_{i+1} \leftarrow V_{i+1} \bigcup \{v\}$; $A^L \leftarrow A^L \bigcup \{(u,v)\}$

13 **else if** $\exists w \in V \setminus \bigcup_{j=0}^{i+1} V_j$ such that $(v,w) \in A_-^0$

14 $V_{temp} \leftarrow V_{temp} \bigcup \{w\}$

15 $V_{i+1} \leftarrow V_{i+1} \bigcup \{v\}$; $A^L \leftarrow A^L \bigcup \{(u,v)\}$

16 $i \leftarrow i+1$

17 $V^L \leftarrow V^L \bigcup V_i$

18 **return** $LN(f) = (V^L, A^L, p^0, c^0)$

Fig. 2. LC algorithm

Now we propose the algorithm to find a blocking flow in the layered network. The in-potential of a vertex u is the sum of its incoming arcs' capacities. The out-potential of u is the sum of its total outgoing backward arcs' capacities and the smaller of the residual vertex capacity and the total outgoing forward arcs' capacities. The potential of u is the smallest of the in- and out-potential.

Definition-8. The potential of a vertex u is defined as formula (2).

$$\eta(u) = \min\left\{ \sum_{(v,u)\in A^L} c^0(v,u), \sum_{(u,v)\in A^L \cap A_-^0} c^0(u,v) + \min\left\{ \sum_{(u,v)\in A^L \cap A_+^0} c^0(u,v), p^0(u) \right\} \right\}. \quad (2)$$

Given a layered network $LN(f)$, we could find out a vertex u with smallest potential and denote it by η^*. Then, we can push a η^* unit flow from s to t. This contains

two stages. The first called Push-Out is to saturate each outward arc of vertex u by the total flow η^* in some order. For each vertex v in the next higher layer of u, let $\eta(v)$ be the flow into v. We saturate the outward arcs of v to pass $\eta(v)$ unit flow through v. When all vertices in that layer have been deal with, repeat this for the next layer. We will never find a vertex with insufficient potential, because we start this with a vertex of minimum potential. When all layers behind u have been deal with, we take the second stage called Pull-In to saturate each incoming arc of u in some order and follow the flow to the next lower layer before u. For each vertex v in the next lower layer of u, let $\eta(v)$ be the flow out of v. Then saturate incoming arcs by $\eta(v)$ unit flow. When all vertices in that layer have been deal with, we repeat this in the next layer before. With these two stages, a η^* unit flow is augmented in $LN(f)$. We update $N(f)$, $LN(f)$ and then go to a new repeat until $t \notin LN(f)$. In addition, we may prune off some vertices (including its arcs), which are not connected to t in $LN(f)$.

Note the following differences from the generic MPM algorithm. In the Push-Out and the Pull-In stages, the residual capacity of a vertex is subtracted by a unit when the vertex sends a unit flow on forward arcs and it remains when the vertex sends a flow on backward arcs. However, the residual capacity of a vertex increases when it receives a flow from backward arcs and remains when it receives a flow from forward arcs. Consequently, we do not delete the vertex u from a new calculated $LN(f)$ even if $\eta(u) = 0$, because it may be useful to construct $s-t$ admissible paths. However, since such zero potential vertexes do not set limits to the capacity of an admissible path, it is excluded from the calculation of the minimum potential η^*.

Theorem-4. The FAT algorithm is finite and terminates with the maximum s-t flow.

Proof. The algorithm is finite because at least one arc or vertex will be saturated after a round of Push-Out and Pull-In operations. The algorithm ends with a flow f where $t \notin LN(f)$. According to Lemma-4 and Theorem-3, f is a max flow.

Since at least one arc or vertex will be saturated after a round of Push-Out and Pull-In, these two operations are executed at most $O(|V|+|A|)$ times before we halt with a blocking flow. The time complexity of LC function is $O(|A|)$. The function Prune could be realized by width-first search in $LN(f)$ whose time complexity is $O(|V|)$. The operation of finding a vertex with minimum potential could be realized as follows. Firstly, we calculate the in- and out- potential of every vertex in N. Each time the flow f and p^0 are changed, these potential are changed correspondingly. The cost of maintaining these values is linear in the number of vertices. In one hand, for each minimum potential vertex, we visit at most $|V|-1$ other vertices. In other hand, we use at most $|V|$ minimum potential vertices altogether. So the cost of Push-Out and Pull-In operations is $O(|V|^2)$. Then, we come to the following theorem.

```
FAT (N)
1  f ← 0
2  LN(f) ← LC(N(f))
3  while t ∈ LN(f)
4        do
5           LN(f) ← Prune(LN(f))
6           η* ← min{η(u) | u ∈ V, η(u) > 0};  u ← v, η(v) = η*
7           f ← PushOut(η*, u, LN(f))
8           f ← PullIn(η*, u, LN(f))
9           LN(f) ← LC(N(f))
10 return f
```

```
Prune(LN(f))
1 for each u ∈ LN(f)
2    if there is no admissible path from u to t
3       delete u and its corresponding arcs from LN(f)
4 return LN(f)
```

```
PushOut(η*, u, LN(f))
1  Δ ← η*
2  for each v ∈ {v | (u,v) ∈ A^L}
3    do
4       λ ← min{Δ, c^0(u,v)}
5       f(u,v) ← f(u,v) + λ
6       if (u,v) ∈ A^0_+
7          p^0(u) ← p^0(u) − λ
8       else
9          p^0(v) ← p^0(v) + λ
10      c^0(u,v) ← c^0(u,v) − λ
11      if v ≠ t
12
   f ← PushOut(λ, v, LN(f))
13      Δ ← Δ − λ
14      if Δ = 0
15        break
16 return f
```

```
PullIn(η*, u, LN(f))
1  Δ ← η*
2  for each v ∈ {v | (v,u) ∈ A^L}
3    do
4       λ ← min{Δ, c^0(v,u)}
5       f(v,u) ← f(v,u) + λ
6       if (v,u) ∈ A^0_+
7          p^0(v) ← p^0(v) − λ
8       else
9          p^0(u) ← p^0(u) + λ
10      c^0(v,u) ← c^0(v,u) − λ
11      if v ≠ s
12         f ← PullIn(λ, v, LN(f))
13      Δ ← Δ − λ
14      if Δ = 0
15        break
16 return f
```

Fig. 3. FAT algorithm

*Theorem-5.*The time complexity of FAT is $O(|V|^2 \cdot |A|)$.

Let's consider an eight vertices Network plotted in Fig. 1 for example. The source vertex is marked by the form of a square and the sink vertex is marked by the form of a triangle. The graphics in the left column are $N(f)$. The values of p and c are marked on vertices and arcs. The graphics in the middle column are $LN(f)$. The vertices that have no admissible path to sink are pruned with "\\" marked on arcs. The graphics in the right column are the blocking flows in $LN(f)$.

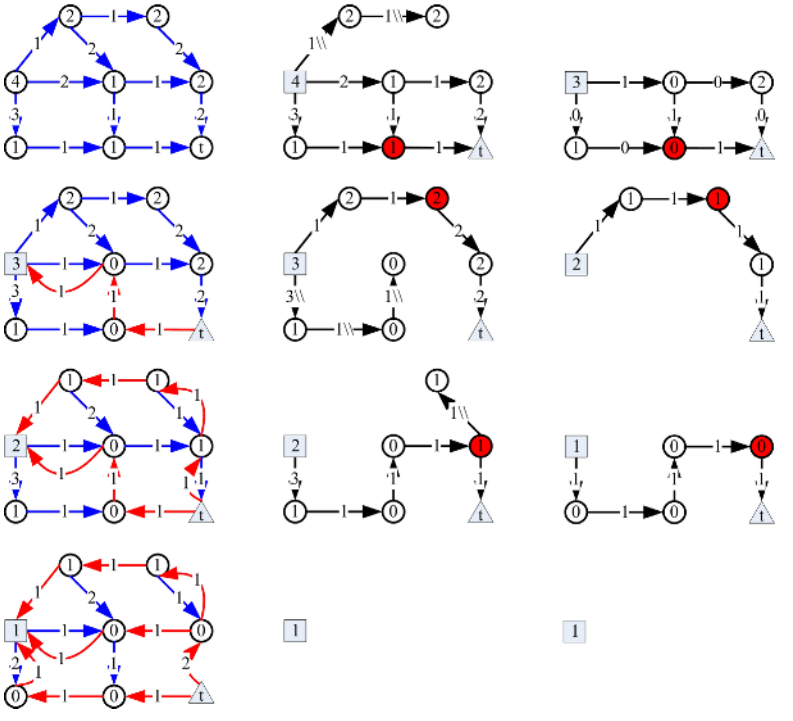


Fig. 4. An example of FAT algorithm

4 Conclusion

In this paper, we formulate the lifetime maximization problem as a max flow problem on a transportation network with powers on vertices and arcs. With BS algorithm, we can achieve optimal traffic planning of a sensor network in any precision if only we can find the max flow on such a transportation network. After analyzing the characters of the flow and cut in such network, we define the admissible path on residual network and prove theorem-3 as a sufficient and necessary condition to check if a flow is maximized. Based on these theoretical analyses, we propose a fast algorithm to solve such a problem. The time complexity of our approach is $O(|V|^2 \cdot |A|)$.

References

1. M. Bhardwaj, A. Chandrakasan, T. Garnett.: Upper Bounds on the Lifetime of Sensor Networks. IEEE International Conference on Communications, Helsinki, Finland, June 2001
2. M. Bhardwaj, A. Chandrakasan.: Bounding the Lifetime of Sensor Networks Via Optimal Role Assignments. IEEE INFOCOM'2002
3. J.-H. Chang, L. Tassiulas.: Energy conserving routing in wireless ad-hoc networks. IEEE INFOCOM'2000
4. K. Dasgupta, K. Kalpakis, P. Namjoshi.: Efficient Algorithms for Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks. Computer Networks, vol. 42, 2003
5. Sankar, Z. Liu.: Maximum Lifetime Routing in Wireless Ad-hoc Networks. INFOCOM'2004
6. R. Madan, S. Lall.: Distributed Algorithms for Maximum Lifetime Routing in Wireless Sensor Networks. Global Telecommunications Conference, IEEE, volume 2, Nov 2004
7. V. M. Malhotra, M. Pramodh Kumar, S. N. Maheshwari.: An O(|V|3) algorithm for finding maximum flows in networks. Information Processing Letters, 7:277--278, 1978

An Adaptive Coverage Algorithm for Large-Scale Mobile Sensor Networks

Peng Guo, Guangxi Zhu, and Liang Fang

Department of Electronics and Information
Huazhong University of Science and Technology, P.R.C.
guopeng@hust.edu.cn, gxzhu@hust.edu.cn,
fangliang@smail.hust.edu.cn

Abstract. Coverage has been an active research area in mobile sensor networks. For a randomly placed large-scale sensor network, sensor nodes would most probably be distributed asymmetrically, and it requires the coverage algorithm to do with the diffusion and contraction of the network. Most of the existed algorithms are on the assumption that sensor nodes are initially densely distributed or the states of the network coverage are known to all the nodes, which does not meet all application scenarios. This paper proposes a new adaptive coverage algorithm based on the combination of boundary contraction and random repulsion. It works well on the scenarios of the asymmetrical initial distribution, the isotropic sensor nodes, and that only the coverage states in communication range being known by nodes. Simulation results show that the algorithm realizes both the diffusion and contraction of the sensor network, and that the deployed nodes tend to be uniformly distributed.

Keywords: mobile sensor networks, adaptive coverage, boundary contraction, random disturbance.

1 Introduction

Coverage is one of the key factors for the performance of sensor networks. The measures of it include validity, redundancy and connectivity. For large-scale sensor networks, it is impossible that the nodes are placed one by one; instead, nodes are randomly dropped. However, this random placement can not guarantee enough uniformity of the network. Besides, when the coverage achieves the optimum, it's possible that some nodes become invalid because of energy exhaust or other troubles, and it would break the equilibrium of the entire network.

Therefore, it's desirable that the sensors have locomotion capability, and with some suitable adaptive coverage methods, the nodes in the network can self-deploy, starting from the current network coverage state to the optimal state. With the development of Micro-Electro-Mechanical system technology recently, tightly packed and low cost mobile sensors become available [1], and the research of adaptive coverage algorithm of mobile sensor networks has become popular.

In this paper, the coverage optimization of sensor networks with initial asymmetrical distribution is discussed. And the assumption is made that each sensor can only

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 468 – 477, 2006.

get the knowledge on the position of its neighboring nodes, whereas a global map of the environment is either unavailable or of little use because the environment is not static.

Under the above constraint, when the initial asymmetrically distributed nodes are deployed, it can not only depend on the attractive and repulsive forces between sensor nodes, as it would lead to some independent and isolated clusters in the final network coverage. The essential reason for it is that each node in the network has no sense of the center of network. When network contracts, nodes inside may make the wrong judgment on the network center by the denseness characters of the neighborhood, and that would result in multiple density centers in the network.

This paper proposes a novel adaptive coverage algorithm based on potential field methods, which can realize both the diffusion and contraction of the sensor network. The simulation results show that mobile sensor networks with initial asymmetrical distribution or with the failure of some sensor nodes can achieve the final effective and uniform coverage via the self-deployment of the nodes, The simulation also demonstrates a lower computational complexity.

The organization of the paper is as follows. In section 2, some of the related work on the coverage of mobile sensor networks is reviewed. Section 3 is about the problem formulation. In section 4, our coverage algorithm is described, and the simulation results are presented in section 5, along with comparisons with three former algorithms. Finally in 6, the conclusions and the future work are given.

2 Related Work

The coverage algorithm of mobile sensor network is parallel to that of the robot system. The concept of coverage as a paradigm for evaluating many-robot systems was introduced by Gage [2]. He defined three kinds of coverage: blanket coverage, barrier coverage and sweep coverage. According to this taxonomy, the deployment problem described in this paper is a blanket coverage problem.

Potential field techniques are a series of typical coverage algorithms, which were first described by Khatib [3] and have since been widely used in the mobile robotics community for tasks such as local navigation and obstacle avoidance [4][5]. It has been pointed out that "potential field methods" are able to achieve good coverage without global maps [6]. And an idea of "social potentials" has been used where the potentials for a robot are constructed with respect to the other robots in [7].

A lot of attention has focused on the problem of dispersing large quantities of mobile robots into an unknown environment. Batalin and Sukhatme's approach is based on using local dispersion of the robots to achieve good global coverage [8]. Pearce, et al. have developed a dispersion behavior for a group of miniature robots inspired by insect colony coordination behaviors [9]. In [10], Sameera, Poduri, et al. proposed a constrained coverage algorithm for mobile sensor networks which is designed to maximize the collective k sensor coverage while simultaneously constraining the degree of the network nodes. As a whole, all these algorithms do with dispersion of the initial densely distributed nodes, without considering the contraction of the initial asymmetrically distributed nodes.

Recently, some people began to study the coverage problem of initial asymmetrical distribution in mobile sensor network. However, these studies are on the assumption that each node has the knowledge of global coverage map [11] or network topology [12] [13], or some nodes are endowed with special properties in advance [14].

It is very complex for other algorithms, such as voronoi and delaunay[15], to build the voronoi and delaunay map for each node without prior knowledge of the global coverage. These algorithms are not cater to dynamic mobile sensor networks, especially in situations that some nodes are disabled or been destroyed. Although they could redistribute the network, they couldn't contract the network. Coverage density would be lower and lower, and a hole unable close up appear in the end.

3 Problem Formulation

Problem

In a boundless two-dimensional region, where the priori knowledge of this field is not available, given N random distributed nodes with isotropic radio communication of range R_C and detection of range R_S ($R_S=\eta R_C$), and only the nodes positions within R_C range is known, how should the adaptive coverage algorithm be designed so that the resulting configuration unifies and maximizes the sensor coverage of the network with the constraint that distance between two nodes is less than $r=R_S$? Moreover, when parts of the mobile nodes fail, how can the algorithm still work?

Definition

Several terms are defined at first:

1) Neighboring Node: if the Euclidean distance between node i and j is less than or equal to R_C, then node j is considered node j's neighboring node.
2) Boundary Node: all nodes on the convex boundary.
3) Isolated Node: node without any neighboring node.
4) Inner Node: node that is inside the boundary, and has more than one neighboring node.

Referring to [10], we make the following assumptions:

1) The nodes are capable of omni-directional motion.
2) Each node has the same right or rank.
3) Each node can sense the exact relative range and its neighbors.
4) No node has the knowledge of the global topology or map of the network.
5) The quality of sensing (communication) is constant within R_S (R_C) and is zero outside the sensing (communication) range, i.e. it follows a binary model.
6) The locomotion algorithm of each node is adaptive and only the localization of its neighbors can be used during deployment.

We use the following two metrics to evaluate the performance of the deployment algorithm.

1) Uniform degree: the standard deviation of the neighbor number of all nodes.
2) Coverage power: power consumed by nodes movement, for coverage mission, are presented by the total path length of the movements.

4 Proposed Algorithm

In large-scale mobile sensor networks, each node adjusts its own location according to the positions of neighboring nodes only, and the Potential field-based techniques have been used extensively in the research. In these methods, sensors will move from a high potential state to a low potential state similar to-the way in which a charged particle would move in an electrostatic field. Therefor, some virtual forces are always constructed. In our algorithm we define three kinds of forces: Border Force F_{border}; Inner Repulsion Force F_{inner}; Random Disturbance Force $F_{disturb}$.

F_{border} is the virtual force exerted on boundary nodes to execute the contraction between nodes and thus eliminate the sparsely covered area. Traditional virtual force is defined as the attraction between nodes. However, as all nodes have the same right, and they have no knowledge on the network center and the sparse region, they can make judgments by the local coverage only. Such kind of "center confusion" and the lack of global cooperation would result in that not all nodes converge on one target, and multiple density centers would appear in the network.

F_{inner} is defined as repulsion force between inner nodes to maximize the coverage under a certain density. When distance between two nodes is close to zero, F_{inner} tends to be infinite. Meanwhile, to guarantee enough redundancy of the coverage, it is required that when distance between two nodes is equal to or smaller than threshold r, F_{inner} be greater than F_{border}.

$F_{disturb}$ is the virtual force exerted on inner nodes and is orthogonal to F_{inner}, and it is given that $F_{disturb} << F_{inner}$. $F_{disturb}$ helps nodes with fine adjustments in order to avoid the situation of critical balance and large-scale coverage leaks during self-deployment.

Mathematically, the forces can be expressed as follows. Consider a network of n nodes 1, 2, 3… n at positions x_1, x_2… x_n, respectively. Let Δx_{ij} represent the Euclidean distance between nodes i and j, i.e.

$$\Delta x_{ij} = \left\| x_i - x_j \right\| \tag{1}$$

F_{border}, F_{inner} and $F_{disturb}$ are defined as follows.

$$F_{border}(i) = F \cdot \frac{\dfrac{x_{i+1} - x_i}{\left\| x_{i+1} - x_i \right\|} + \dfrac{x_{i-1} - x_i}{\left\| x_{i-1} - x_i \right\|}}{\left\| \dfrac{x_{i+1} - x_i}{\left\| x_{i+1} - x_i \right\|} + \dfrac{x_{i-1} - x_i}{\left\| x_{i-1} - x_i \right\|} \right\|} \tag{2}$$

where x_i, x_{i+1}, x_{i-1} are boundary nodes.

$$F_{inner}(i,j) = \begin{cases} \dfrac{F \cdot r \cdot (x_i - x_j)}{\Delta x_{ij}^{\,2}} & \Delta x_{ij} \le r \\ 0 & \Delta x_{ij} > r \end{cases} \tag{3}$$

$$F_{disturb}(i) = k \cdot \theta \cdot F \cdot \frac{\sum F_{inner}(i,j)}{\left\| \sum F_{inner}(i,j) \right\|} \angle 90^{\circ} \tag{4}$$

Where k is the force constant, θ is a random variable changed with time, and it belongs to {1, -1}.

Based on the above definitions, the resultant force of node i is

$$F(\mathrm{i}) = \begin{cases} F_{disturb}(\mathrm{i}) + \sum F_{inner}(\mathrm{i},\mathrm{j}) & i \text{ is an } \textit{inner node} \\ F_{border}(\mathrm{i}) + \sum F_{inner}(\mathrm{i},\mathrm{j}) & i \text{ is a } \textit{boundary node} \end{cases} \tag{5}$$

The equation of motion for node i is formulated as:

$$\ddot{x}(t) = \frac{F(\mathrm{i}) - f}{m} \tag{6}$$

Where f is motional friction which satisfies F >> f > Fdisturb and m is the virtual mass of the node assumed to be 1.With motional friction f, dithering caused by Fdisturb can be restrained when network is close to equilibrium.

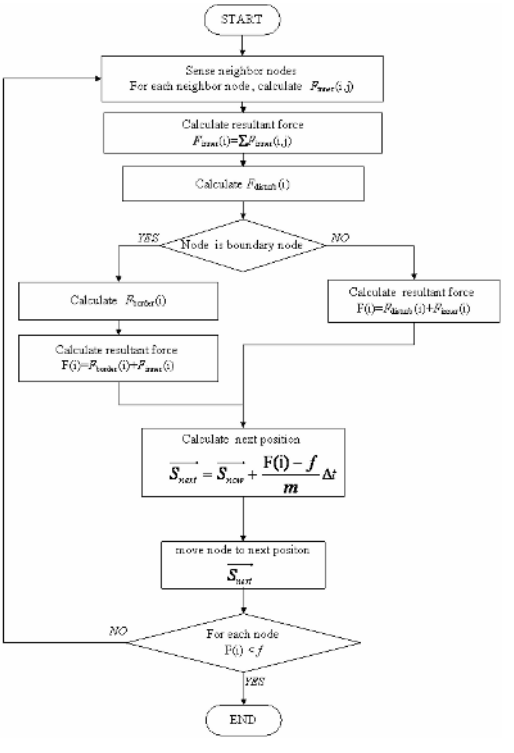


Fig. 1. The coverage algorithm

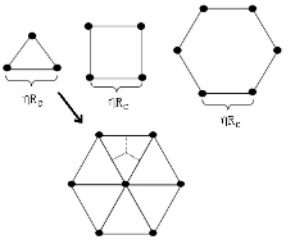


Fig. 2. The possible isogons

Discussions

Given the certain region to calculate the number of nodes to be deployed, or given the number of nodes to calculate the deployed coverage area, in either case, the goal is to achieve the required coverage quality with as few node motions as possible. And the condition of the initial deployment is given as follows:

As the sensor network is expected to be uniformly covered, the ideal coverage result is given that the network consists of isogons, and all the nodes are on the vertexes of the isogons. And as it is known, the internal angle α of an isogon is calculated as:

$\alpha = \frac{n-2}{n} \times 180^\circ$, where n is the number of the edges.

Considering that these isogons should cover the whole region seamlessly, the internal angle α of the isogons is formulated as:

$\alpha \times m = 360^\circ$,

that is, $\frac{n-2}{n} \times 180^\circ \times m = 360^\circ$, where m is a positive integer.

Resulting from these constrains, the possible isogons can only be regular triangles, squares or regular hexagons, which are shown in figure 2.

In this paper, the regular triangles are chosen for the expected coverage, for the distance between each vertex in this kind of isogons is ηRC, which well meets the demand of our problem. We calculate the deployment density according to regular triangles and give the calculation of the initial area that we should choose. In this way, the distance of the node motion is little and it would reduce the consumption of the sensors's energy. We can give the formula as follows:

$$\lambda = \left(\frac{\sqrt{3}}{2} \eta^2 R_C^2 \right)^{-1} . \tag{7}$$

5 Experiments and Results

We have randomly deployed 300 nodes in a unimpeded square region. Limbic length is 1600 according to (7). We also assume F=100, η=0.5, R_S=ηR_C=100, k=0.1.

Based of the above enactment, we have got a set of results using the above virtual force formulas and motion equations, Traditional algorithms (e.g. potential field, voronoi, delaunay) are also simulated in the same condition, all results are shown as following:

Fig 3 (a) indicates the initial random distribution of the network, where we can see the nodes are asymmetrically distributed, that some areas are over-dense and others with large leaks. Fig 3 (b) is the final distribution of the proposed algorithm, (c) represents potential field, (d) voronoi and (e) delaunay. We can see that the final distribution of the proposed algorithm are very uniform, and distances between neighbors are almost around r = 0.5RC, except for few nodes on the boundary.

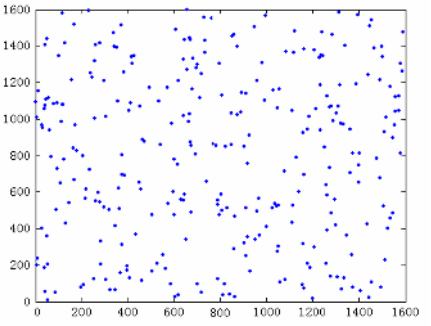


Fig. 3. (a) the initial random distribution

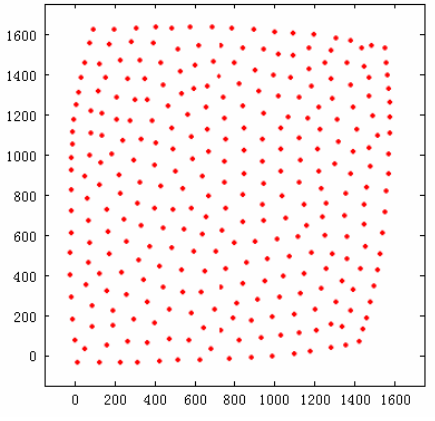


Fig. 3. (b)

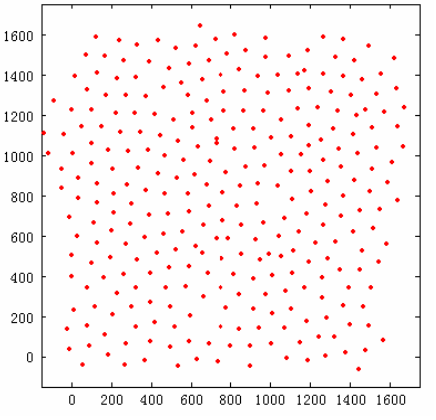


Fig. 3. (c)

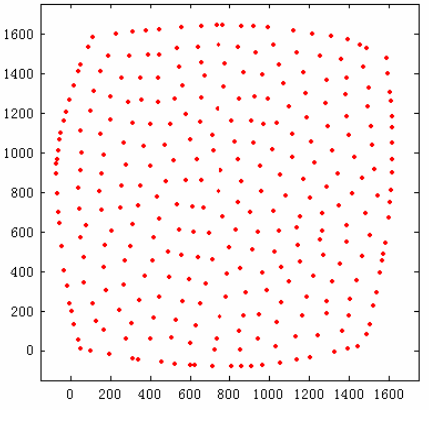


Fig. 3. (d)

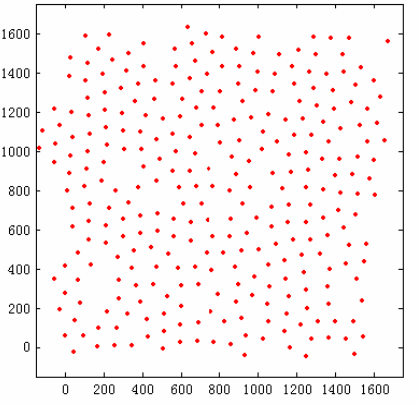


Fig. 3. (e)

Fig. 3. (b)(c)(d)(e) the final distribution of the proposed algorithm, potential field, voronoi and delaunay algorithm

Fig 4 indicates the alteration of node's uniform degree as time goes by. When t=40 second, the curve tends to be steady, which proves that the number of each node's neighbors is almost constant. As we can see, the proposed algorithm and voronoi have the most uniformity redistribution results.

Total power consumption of four algorithms are given in Fig 5. The proposed algorithm is the second lowest one, which is a little higher than delaunay. But, both delaunay and voronoi need to build complicated map repeatedly in each node, which consume lots of power and demand huge computational capability.

Coverage algorithm of mobile sensor networks should be adaptive to many unforeseeable accidents of the network, a total loss of nodes in a small block for example, and redistribute automatic. Fig 6(a) (b) give the simulation results of the proposed algorithm and potential field in this situation. Fig 7 (a) (b) are the redistribution results. The proposed algorithm contracts finally, because it is based on border contraction. While by potential field the nodes stopped only after a small movement, and a hole appear because of the accident.

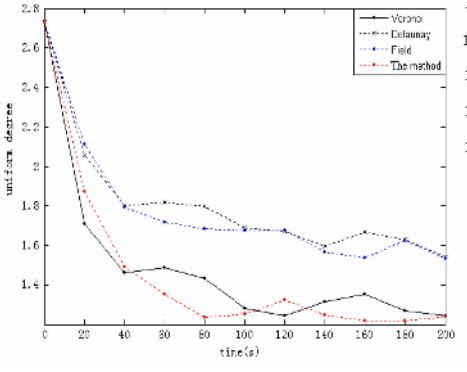


Fig. 4. The uniform degree

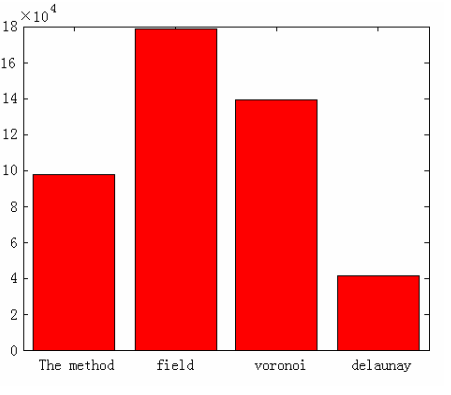


Fig. 5. total power consumption

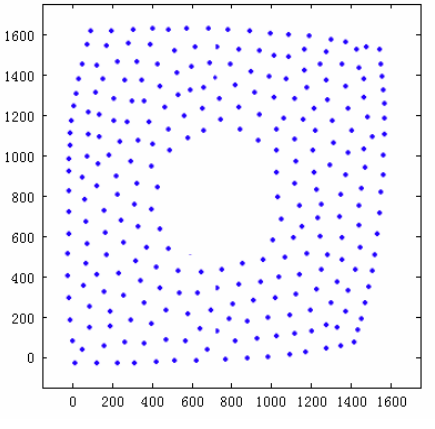


Fig. 6. (a)

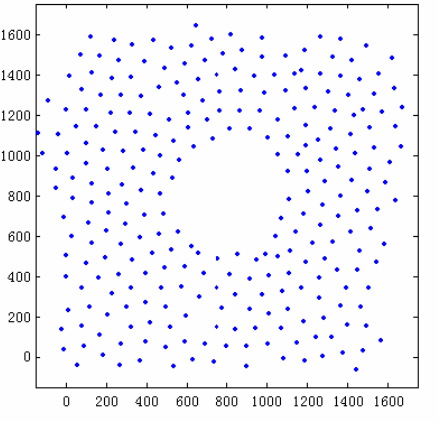


Fig. 6. (b)

Fig. 6. (a)(b) nodes loss for the proposed algorithm and potential field algorithm

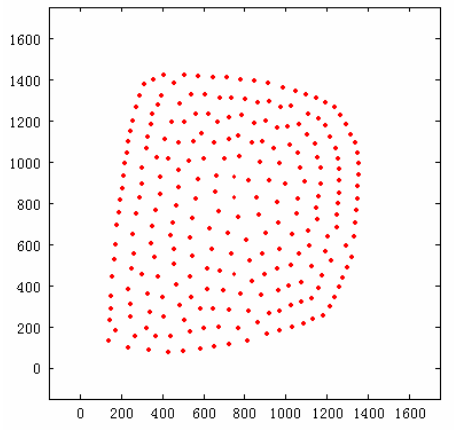


Fig. 7. (a)

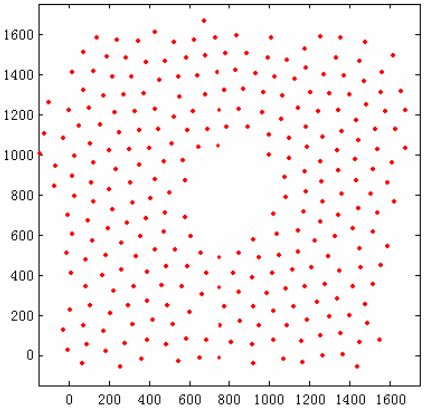


Fig. 7. (b)

Fig. 7. (a)(b) redistribution for nodes loss of the proposed algorithm and potential field

6 Conclusions and Future Work

This paper has presented a coverage algorithm for large-scale mobile sensor networks with asymmetrical initial distribution. With the assumption that each node has no sense of the topology or its relative position in the network, we propose a new method based on the combination of boundary contraction, inner repulsion and random disturbance. Simulation results indicate that the sensor network achieves the coverage uniformity and has low coverage power consumption.

The algorithm also applies to the redeployment scenarios of some optimizing-covered sensor networks resulting from abnormal events of nodes, e.g. nodes failure or removal. However, we think the algorithm efficiency on this situation is low, as most of the network nodes would be involved in the redeployment process even when the abnormal nodes are few. The expected result is that only parts of the network nodes respond to the disturbance. Therefore, the future work is the efficient redeployment algorithm for the optimizing-covered sensor networks after few abnormal events, which is still on the assumption that each node only has the knowledge of the positions of its neighboring nodes.

References

1. Frost Gorder, P.: Sizing up smart dust. IEEE Computational Science and Engineering, Volume 5, Issue 6, Nov.-Dec. 2003 Page(s):6 – 9
2. D. W. Gage.: Command control for many-robot systems. The Nineteenth Annul AUVS Technical Symposium in AUVS-92, Huntsville, Alabama, USA. June 1992. pp. 22-24
3. O. Khatib.: Real-time obstacle avoidance for manipulators and mobile robots. International Journal of Robotics Research, 5(1):90–98, 1986

4. Parlaktuna, O., Bakla, B., Ozkan, M., Yazici, A.: Mobile robot navigation using fuzzy logic methods. Proceedings of the IEEE 13th Conference on Signal Processing and Communications Applications, 16-18 May 2005 Page(s):432 – 435
5. Blanc, G., Mezouar, Y., Martinet, P.: Indoor Navigation of a Wheeled Mobile Robot along Visual Routes. Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 18-22 April 2005 Page(s):3354 – 3359
6. Andrew Howard, Maja J Matari´c, and Gaurav S Sukhatme.: Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem. In Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems (DARS02) Fukuoka, Japan, June 25-27, 2002
7. J. H. Reif and H. Wang.: Social potential fields: A distributed behavioral control for autonomous robots. Robotics and Autonomous Systems. vol. 27, pp, 171-194, 1999
8. M. Batalin and G. S. Sukhatme.: Spreading out: A local approach to multi-robot coverage. In 6th International Conference on Distributed Autonomous Robotic Systems (DSRS02), Fukuoka, Japan. 2002, pp. 373-382
9. J. L. Pearce, P. E. Rybski, S. A. Stoeter, and N. Papanikolopoulous.: Dispersion behaviors for a team of multiple miniature robots. In IEEE International Conference on Robotics and Automation, Taipei, Taiwan, September 2003, pp. 1158-1163
10. Sameera Poduri and Gaurav S. Sukhatme.: Constrained Coverage for Mobile Sensor Networks. Proceedings of the 2004 IEEE International Conference on Robotics and Automation, Volume 1, 2004 Page(s):165 - 171 Vol.1.
11. Nojeong Heo, Varshney, P.K.: Energy-efficient deployment of Intelligent Mobile sensor networks. IEEE Transactions on Systems, Man and Cybernetics, Part A, Volume 35, Issue 1, Jan. 2005 Page(s):78 - 92
12. Zhou, S. Wu, M.-Y. Shu, W.: Finding optimal placements for mobile sensors: wireless sensor network topology adjustment. Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, 2004. Volume 2, 31 May-2 June 2004 Page(s):529 - 532 Vol.2.
13. Wang, G., Cao, G., Porta, T.L., Zhang, W.: Sensor relocation in mobile sensor networks. Proceedings of the IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Volume 4, 13-17 March 2005 Page(s):2302 - 2312 vol. 4.
14. Bin Zhang, Sukhatme, G.S.: Controlling Sensor Density Using Mobility. The Second IEEE Workshop on Embedded Networked Sensors.30-31 May 2005 Page(s):141 – 150
15. Guiling Wang, Guohong Cao, La Porta, T.: Movement-assisted sensor deployment. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies. Volume 4,7-11 March 2004 Page(s):2469-2479 vol.4.

Adaptive Sink Mobility Management Scheme for Wireless Sensor Networks

Kwang-il Hwang and Doo-seop Eom

Department of Electronics and Computer Engineering, Korea University
5-1ga, Anam-dong, Sungbuk-gu, Seoul, Korea
Tel.: +82-2-3290-3802, Fax: +82-2-3290-3895
brightday@final.korea.ac.kr

Abstract. In wireless sensor networks, it is important to efficiently disseminate information from each source to a sink node. In particular, in mobile sink applications, due to the sink mobility, a stationary dissemination path may no longer be effective. The path will have to be continuously reconfigured according to the current location of the sink. In this paper, an Adaptive Reversal Tree (ART) protocol, based on the Adaptive Reversal algorithm and dynamic Root change mechanism, is proposed. Data dissemination from each source to a mobile sink can be easily achieved along the ART without additional control overhead, because the ART proactively performs adaptive sink mobility management. In addition, the ART can maintain a robust tree structure by quickly recovering the partitioned tree with minimum packet transmission. Finally, the simulation results demonstrate that the ART is a considerably energy-efficient and robust protocol.

1 Introduction

In recent years, attractive characteristics of sensor networks have led to incremental utilization in many military and civil applications such as battlefield surveillance, environmental control, and security management. In such applications, it is important to efficiently disseminate information from each source to a sink node. However, if a sink moves to a different location, the dissemination path may no longer be effective. The path will have to be continuously reconfigured according to sink movement. The continuous reconfigurations can bring about tremendous traffic and energy wastage in the network. In particular, energy is considered as one of the most expensive resources in sensor networks.

In order to efficiently manage the dissemination path in sensor networks with a mobile sink, several schemes are proposed [1,2,3,4,5]. Directed Diffusion [2], SAFE [3], and DST [5] concentrate on managing the dissemination path initiated from a sink, whereas source initiated dissemination protocols, such as TTDD [1] and SEAD [4], use a method that mobile sinks access on the dissemination path constructed on the basis of each source. Although these methods aim to solve the problem using different techniques, high maintenance overhead is still required to continuously update the dissemination path.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 478–487, 2006.

In this paper, to address the challenges on the mobile sink problems, an Adaptive Reversal Tree (ART) protocol, which is based the Adaptive Reversal algorithm and dynamic Root change mechanism, is proposed. The ART is able to maintain the dynamic tree by adaptively reversing links with minimum communication overhead in the network, in spite of being in environments with a large number of mobile sinks.

The remainder of this paper is organized as follows. Section 2 overviews the basic algorithms for the ART. Section 3 describes detailed operation of the ART. In section 4, performance evaluation through simulation is presented. Section 5 concludes this paper.

2 Algorithm Overview

In this section the basic idea of the ART protocol, which is motivated by other literature [5,6] discussing network dynamics, is presented. First, the *adaptive reversal* algorithm, which is able to perform fast, efficient link reversal with minimum communication overhead and energy in sensor networks, is introduced. In addition to the basic algorithm, the *Dynamic root change*, to pursue the current location of a dynamic sink, is presented.

2.1 Adaptive Reversal Algorithm

Link reversal algorithms provide a simple mechanism for routing in mobile ad hoc networks with dynamic topology changes. Gafni and Bertsekas [6] proposed *full and partial reversal* algorithms. In the link reversal algorithms, a directed acyclic graph (DAG) directed at the destination, is continuously maintained, according to topology changes. However, such classical reversal algorithms are hardly considered a routing protocol for sensor networks, due to various peculiar characteristics of sensor networks. First, while all nodes in a MANET can become source or sink, all flows of data in sensor networks are eventually concentrated on the sink node. In addition, the majority of sensor nodes are stationary and topology changes result from the movement of the sink node, rather than sensor nodes.

In particular, in traditional link reversal routings (LRR), the invocation of reversal is initiated from nodes detecting an absence of destination. However, in sensor networks, such an update reversal triggered from the node detecting absence of the sink due to sink movement can be extended to the entire network, because each node in the network can hardly identify the current location of the sink. Eventually, due to continuous sink movement, link reversal routing can result in continuous flooding in the network.

Therefore, in order to address such problems, an *Adaptive reversal* (AR) algorithm is proposed for sensor networks with a mobile sink. The AR algorithm has several distinguished characteristics. First, while the traditional LRR maintains a DAG for each destination at each node, the AR is based on a spanning tree directed at a temporary root in which multi paths are not permitted. Second, update reversal invocation is triggered from a node newly appointed by the sink node, not the node detecting absence of sink due to sink movement.

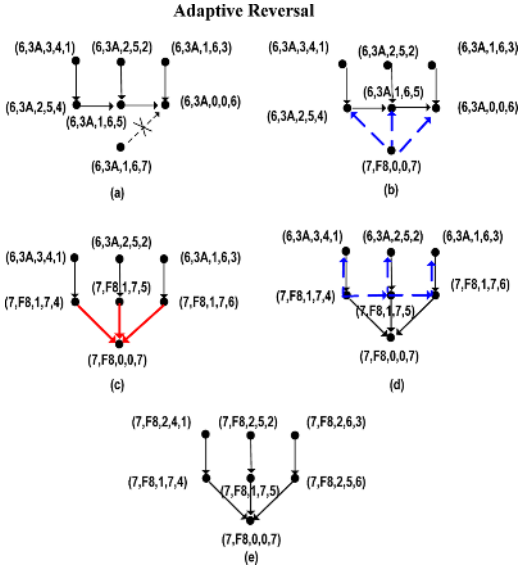


Fig. 1. Adaptive link reversal algorithm

The AR is implemented in a straightforward manner by a cache maintained by each node, and an update reversal message triggered from a node newly appointed by the sink in the network, which is appointed by the sink node. The cache is called the Uplink_info cache. The Uplink_info cache of each node i is the set of quintuple(a_i, b_i, c_i, d_i, i), ordered lexicographically. Essentially, the field a_i is temporary root node id, b_i is a sequence number received from latest update message, c_i represents the distance from the temporary root (i.e. hop distance), d_i designates its current parent node, and i is its unique id in the network. Let N denote the set of nodes in the network. The initial set of Uplink_info cache $\{(a_i^0, b_i^0, c_i^0, d_i^0, i) \mid i \in N\}$ satisfies $a_i^0 = b_i^0 = c_i^0 = d_i^0 = 0$. A temporary root node constructs a spanning tree by flooding. It is assumed that the tree is directed to the temporary root and each node considers only one parent node to reach the root. Accordingly, after constructing spanning tree starting from temporary root node k where $a_k = k \, and \, c_k = d_k = 0$, the set of Uplink_info cache for each node i is changed as follows: $\{(a_i^k, b_i^k, c_i^k, d_i^j, i) \mid i, j, k \in N, \; c_j^k \langle \, c_i^k, \, and \; d_i^j = j\}$.

If a node l is appointed as a new root, it propagates the update reversal message with the set of quadruple$\{(a_l, b_l, c_l, l) \mid l \in N\}$, where a_l, b_l, c_l and l have the same meaning as a_i, b_i, c_i *and* i in the Uplink_info cache, respectively. The link reversal process of the AR algorithm is implemented as follows:

Suppose node i received the update packet from node j. Each node i compares each field in its Uplink_info cache with the update packet from j, including the set of (a_j, b_j, c_j, j). Let the set $(\overline{a}_i, \overline{b}_i, \overline{c}_i, \overline{d}_i, i)$ denote the updated cache after reversal process.

First, the field d_i is compared. If d_i is equal to j, (i.e. j is already my parent), then the cache in node i is updated with $\overline{a}_i = a_j$, $\overline{b}_i = b_j$, *and* $\overline{c}_i = c_j + 1$. It is important to note that, in this case, the node does not propagate the update message to other nodes any more.

On the other hand, if j is different from d_i, according to other conditions, the reversal process of each node goes into the one case of the following:

Case 1: $a_i \neq a_j$ *and* $b_i \neq b_j$ (this means the node should forward the update reversal) Then, the cache is updated with $\overline{a}_i = a_j$, $\overline{b}_i = b_j$, $\overline{c}_i = c_j + 1$, *and* $\overline{d}_i = j$ and the node propagate its update message of $(\overline{a}_i, \overline{b}_i, \overline{c}_i, i)$.

Case 2: $a_i == a_j$ *and* $b_i == b_j$

Then, if $c_i \leq c_j$, the node does nothing, otherwise, $\overline{c}_i = c_j$, *and* $\overline{d}_i = j$. The latter case means that the link of node i for this temporary root has already been reversed, but a more efficient route to the root is discovered.

Fig 1 describes the reversal process of the AR algorithm. It is remarkable that the update reversal packet in the AR is only forwarded to the partial area around the new root node, not to the entire network. As shown in fig. 1, if the other nodes are connected backwards to nodes 1, 2, and 3, the update reversal packet will never be propagated. This reduced amount of update packets can lead to considerable energy conservation in the networks. In section 4, it is shown that maintaining the dynamic spanning tree is available with the number of nodes less than approximately 14% of all nodes, in spite of a highly mobile sink environment.

2.2 Dynamic Root Change

The presented AR algorithm enables fast, efficient link reversal directed to the new root node. The performance largely depends on where the next update will occur since the last update. In the worst case, which in the uniformly deployed structure, the next update occurs at an edge across its carter corner since the last update occurred at an edge in the spanning tree with a rectangular shape, all nodes in the spanning tree will have to operate to reverse their link. However, this degradation of performance can be avoided by changing the root node according to the location of the sink more frequently.

The DST [6] presents a *periodic update request* scheme by periodically broadcasting Update messages. In this scheme, a root node can be changed continuously according to the location of the sink. The presented AR takes the *dynamic root change* scheme similar to that in the DST. However, the root change process employed in the AR is different from that of the DST, which uses a tight periodic function, depending on the node density and maximum sink speed, in that the ART can generate an update asynchronously as well as a longer, flexible update interval. Fig. 2 illustrates the dynamic root change process combined with an adaptive reversal algorithm.

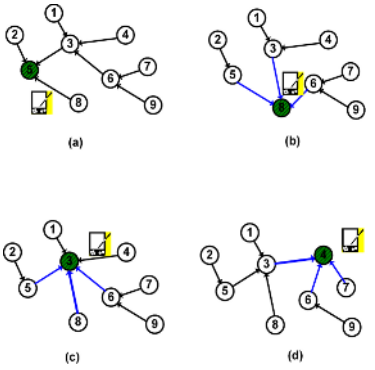


Fig. 2. Dynamic Root change process combined with adaptive reversal

3 Adaptive Reversal Tree (ART)

In this section, the operation of *Adaptive Reversal Tree* (ART), which is based on the Adaptive Reversal algorithm and the dynamic Root change mechanism, is presented. Since the ART proactively performs adaptive sink mobility management, data dissemination from each source to a mobile sink can be easily achieved along the ART, without additional control overhead. In addition, the ART can maintain a robust tree structure by quickly recovering the partitioned tree with minimum packet transmission.

3.1 Basic Model

The application model presented is somewhat different from general sensor network applications which take a stationary sink node into account as a whole. Our application model, in particular, focuses on the mobile sink application where a sink node enters the sensor field directly and performs her of his role based on source data dynamically refreshed from sensor field. Such application model is very useful for a battle field or rescuer activity.

Application model that we suppose also makes the following basic assumptions:

- Homogeneous sensor nodes are densely deployed.
- Sensor nodes communicate with each other through short-range radios. Long distance data delivery is accomplished by forwarding data across multiple hops.
- Each sensor node is aware of its own location (for example through receiving GPS signals or through other localization techniques).
- Sensor nodes remain stationary at their initial location.

3.2 Adaptive Sink Mobility Management of the ART

At the initial stage, a sink node enters the sensor field, broadcasting the Sink_Update message. The update message is used to notice the current sink's location to the network and thereby a temporary root node is changed to a new node that is currently near the sink.

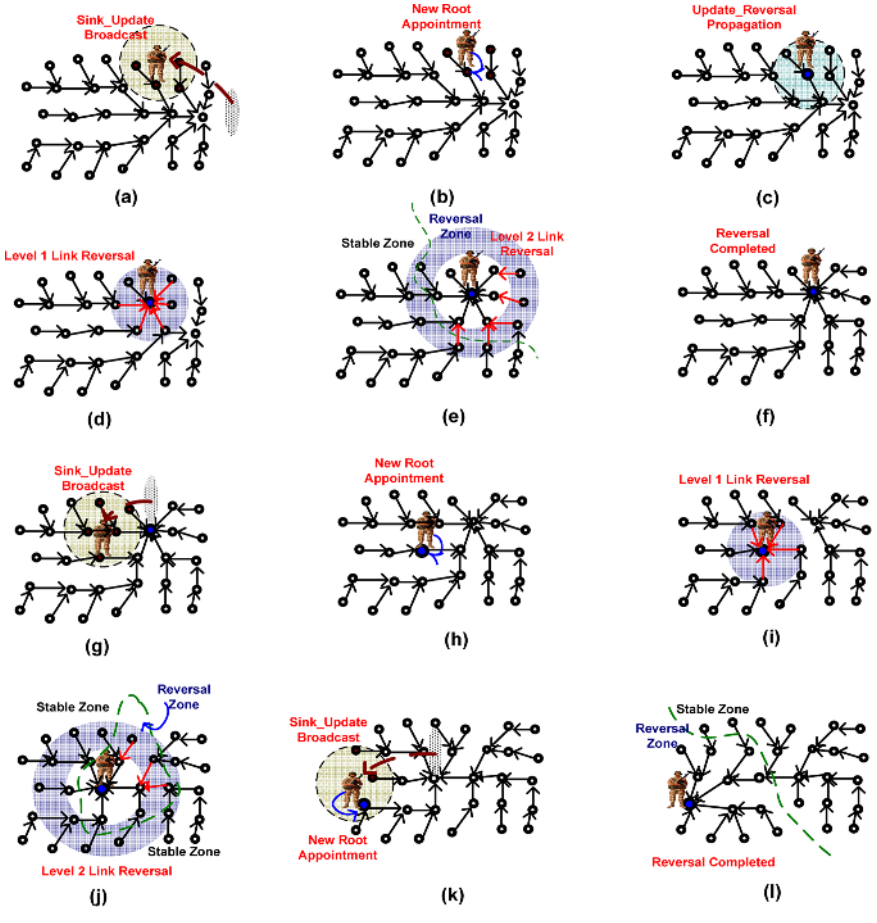


Fig. 3. An example of adaptive sink mobility management via the ART

In general, the Sink_Update message is periodically broadcasted by the update rates, which represents the time interval for a user to obtain information sensed from the sensor field. However, it is also allowed that the update request of the sink is broadcasted asynchronously (Non-periodically) only when the sink wants. All nodes that received the Sink_Update message transmit the Root_Request message to the sink node. The sink appoints one of them as a new root node and replies with the Root_Reply to the node. However, if one of these nodes is

the previous root, the sink does not change the root, because this means the sink has not moved far away from the root since the last update. If the tree construction field in the Root_Reply message is set, (i.e. the sink requests a new dissemination tree), the root node initiates construction of an initial spanning tree by flooding the entire network. The tree is directed to the current root node. Fig. 3 illustrates an example when managing the sink mobility of the ART. As the sink moves to a new location, a new root node is appointed and the adaptive reversal process is triggered from the node. In the ART, the stable zone and the reversal zone is defined as the area where update activity does not occur, and as the area where reversal of each node is updated, respectively. The update reversal process for each update request proceeds until update propagation meets the stable zone. It is important to note that in spite of sink movement, update reversal is only required in a partial area (reversal zone), not the entire network. The size of stable zone and update rate interval represent a trade off. The stable zone presents the set of nodes which are already correctly directed to the new root node. On the other hand, the reversal zone stretches out from an area, where the last update was invoked, to that adjacent to the current root node. Accordingly, in the case of a long update interval, a sink may move further away from the old root, so that it leads the extension of the reversal zone. In section 4, the working node rates are investigated, revealing the ratio of reversal zone over the entire field, with respect to the variation of sink speed.

3.3 Proactive Data Dissemination Via the ART

The ART is intentionally designed to cope well with the mobile sink, so that data dissemination cost from each source to the mobile sink can be accomplished with the $O(n\sqrt{N})$ along the tree, where n is the number of source nodes and N is the number of all deployed sensor node. In addition, In the case of many sources, the cost will be reduced greater by use of in-network processing.

3.4 Error-Resilience to Failure

A link on the tree can be broken by various reasons, such as node failures, obstacles, and so on. Such link breakages result in partitioned trees. However, it is important that the tree should always maintain robust connectivity from all sensor nodes to sinks. Therefore, in order to keep the robust tree, the presented ART exploits simple but fast, robust error-recovery. As shown in fig. 4, due to node failure, the partitioned tree occurs. As soon as the root of a partitioned tree comes to realize the breakage of its uplink, it broadcasts a Join_Request message to its neighbors. Only nodes, where their parent is not the sender, reply with the Join_Reply to the root of the partitioned tree. The node completes joining the active tree by selecting one of them as its parent. The selection condition is the node having the smallest height to current root node of active tree as shown in Fig. 4.

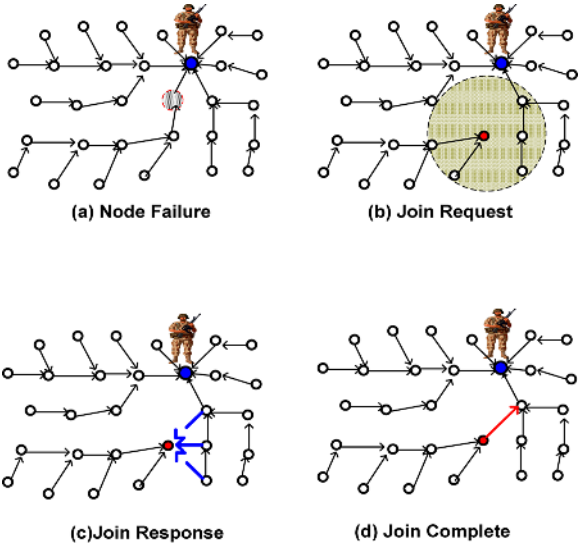


Fig. 4. Error recovery process of the ART for a partitioned tree due to failures

4 Performance Evaluations

In this section, the performance of the presented ART is evaluated in terms of communication overhead and dissipated energy with respect to the sink mobility. The presented ART is implemented as an independent routing agent module in ns-2.27. In the basic simulation setting, the same energy model is used, which is two-ray ground model and omni-directional antenna, as adopted in Directed Diffusion, and TTDD implementation in ns. A 802.11 DCF is used as the underlying MAC protocol. A sensor node's transmitting, receiving, and idling power consumption rate is set to 0.66W, 0.395W and0.035W, respectively. The network in the simulation consists of 400 sensor nodes randomly distributed in a 1000m x 1000m. Each simulation run lasts for 500 seconds. Each query packet is 36 bytes and each data packet is 64 bytes in length, in order to facilitate comparisons with other protocols.

4.1 Evaluations of Communication Overhead

In this subsection, the communication overhead (CO) in the network with respect to sink mobility is evaluated. Communication overhead rate is defined as the ratio of the number of working nodes, participating in the update reversal, over the total number of nodes. First, the CO at each update rate from 5 seconds to 20 seconds with respect to sink movement (to each specific moving distance) is observed. In the second experiment, the Average CO at different sink speeds, from 5m/s to 30m/s, is examined respectively. As presented in fig.5, the simulation results illustrate that the presented ART enables to maintain the

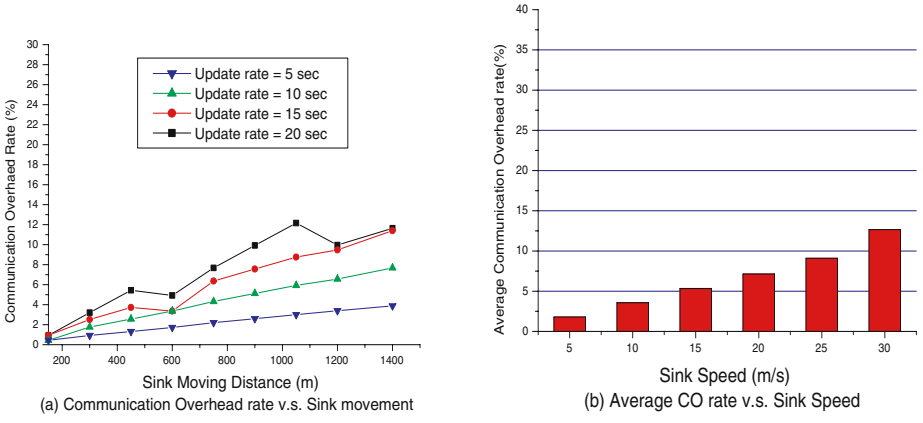


Fig. 5. Evaluations of communication overhead

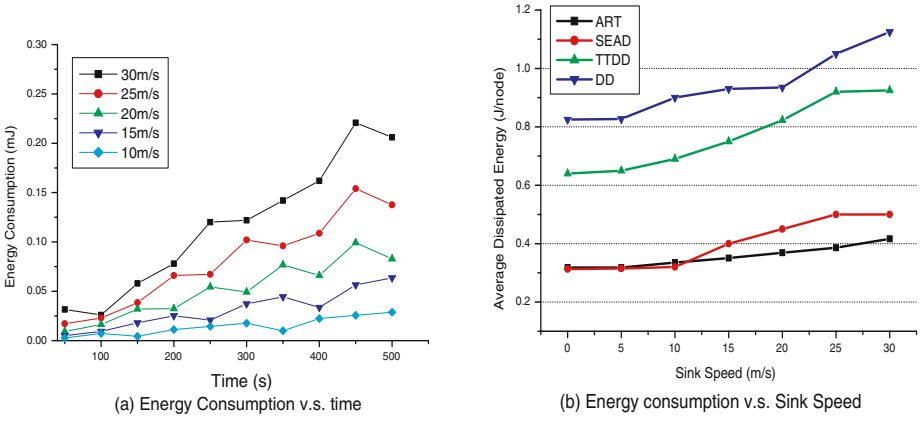


Fig. 6. Evaluations of energy efficiency

dissemination path only with the number of working nodes less than approximately 14% of all nodes in the network, even in highly mobile sink environments.

4.2 Energy Consumption

In this subsection, energy consumption is evaluated. The total dissipated energy at each sink speed (from 10m/s to 30m/s) is observed over time and compared to other data dissemination protocols, directed diffusion, TTDD, and SEAD. It is remarkable that the presented ART outperforms other protocols considerably in terms of total dissipated energy. This indicates that the ART is able to maintain the dynamic tree with minimum communication overhead in the network.

5 Conclusion

In this paper, in order to cope well with mobile sink environment, an *Adaptive Reversal Tree* (ART) protocol, which is based the *Adaptive Reversal algorithm* and the *dynamic Root change mechanism*, was proposed. The ART is able to maintain the dynamic tree by adaptively reversing links with minimum communication overheads in the network in spite of highly mobile sink environment. Since the ART proactively perform adaptive sink mobility management, data dissemination from each source to a mobile sink can be easily achieved along the ART without additional control overheads. Finally, the simulation results showed that the presented ART is a considerably energy-efficient, robust protocol.

References

1. Haiyun Luo, Fan Ye, Jerry Cheng, Songwu Lu, Lixia Zhang: TTDD: Two-tier Data Dissemination in Large-scale Wireless Sensor Networks. In: ACM/Kluwer Mobile Networks and Applications, Special Issue on ACM MOBICOM,2002.
2. C. Intanagonwiwat, R. Govindan, and D. Estrin: Directed diffusion for Wireless Sensor Networking. In:IEEE/ACM Transaction on Networking, Vol. 11, 2003.
3. Sooyeon Kim, Sang H. Son, John A. Stankovic, Shuoqi Li, Yanghee Choi : SAFE: A Data Dissemination Protocol for Periodic Updates in Sensor Networks. In: Proceedings of the 23 rd International Conference on Distributed Computing Systems Workshops (ICDCSW'03),2003.
4. Hyung Seok Kim, Tarek F. Abdelzaher, Wook Hyun Kwon : Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks. In: Proceeding of Embedded Networked Sensor Systems (SenSys03), Los Angeles, California, USA, 2003.
5. Kwang-il Hwang, Jeongsik In, and Doo-seop Eom : Dynamic Shared Tree for Minimum Energy Data Aggregation of Multiple Mobile Sinks in Wireless Sensor Networks. In: EWSN2006, Lecture Notes in Computer Sciences 3868, Zurich, Switzerland, 2006.
6. C. Busch, S. Surapaneni, and S. Tirthapura : Analysis of Link Reversal Routing Protocols for Mobile Ad Hoc Networks. In: SPAA 2003, pp. 210-219, San Diego, California, June 2003.

A Congestion Control Technique for the Near-Sink Nodes in Wireless Sensor Networks

SungHyun Moon, SungMin Lee, and HoJung Cha

Department of Computer Science, Yonsei University
Seodaemun-gu, Shinchon-dong 134, Seoul 120-749, Korea
{shmoon, sulee, hjcha}@mobed.yonsei.ac.kr

Abstract. Without congestion control techniques specifically designed for sensor network applications, the system may not function properly due to data transmission failure. Moreover, most many-to-one communication schemes adopted by sensor network applications cause a practical congestion problem called the funneling effect. With this problem, the hottest area is confined to the connection between the sink and its neighbors. In order to solve the bottleneck problem, we propose a congestion-control technique which uses adaptive time-slot scheduling and a service-differentiated technique. The simulation results show that the proposed method outperforms the conventional technique.

1 Introduction

The unique characteristics of wireless sensor networks have motivated many interesting research issues [1]. One active area of research pertains to network congestion that is closely related to collision, packet-drop and retransmission [2]. In the wireless environment with the random and dense topology, the congestion problem must be addressed in order to allow for the systems' proper operations. The wireless environment generates more unstable connection among nodes than that of wired networks. Due to the unstable connection, packets are transmitted for several times and may cause network congestion. Moreover, the random and dense topology, which is the usual environment for sensor networks, aggravates network congestion. In particular, applications such as multiple-objects tracking often generate countless data transmissions, and thus may suffer from this problem [3].

Recently, there has been active research aimed at solving the congestion problem. However, compared to other networks, wireless sensor networks have a unique problem called the funneling effect, which usually appears in the many-to-one network paradigm. Many applications in sensor networks have one sink and many sensing nodes and suffer from concentrating downstream data that cannot be aggregated toward the sink. In particular, real-time applications such as multiple objects tracking, in which heavy and constant network traffic is expected, require a congestion handling technique. Early research on the congestion problem in wireless sensor networks neglected to take into account the funneling effect.

As shown in Fig. 1, the end of the funnel should be the most critical area, called the bottleneck. The area may consist of a sink and its neighbors, and the number of neighbors is usually too small to handle enormous amount of data. Therefore, a special congestion-control technique for the connection between the sink and its neighbors is necessary.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 488 – 497, 2006.

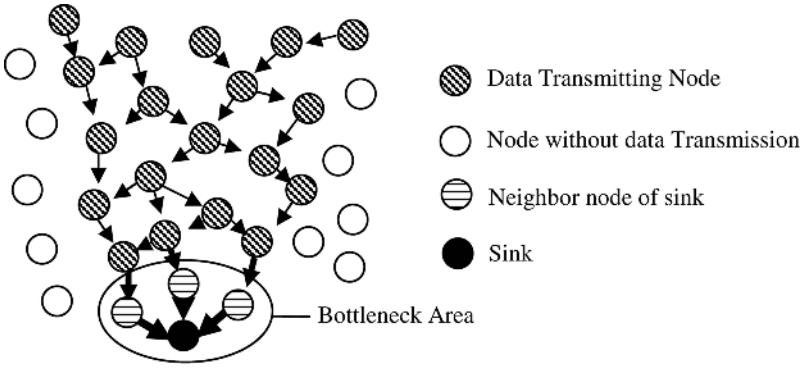


Fig. 1. Illustration of the problem

In this paper, we present a congestion-control technique to handle the bottleneck problem. This technique consists of service differentiation and adaptive time-slot scheduling. With our solution, high-priority data such as those with imminent deadlines can transmit first to the sink, and collision and fairness are handled appropriately. Our adaptive time-slot scheduling is based on TDMA, so it is unlikely to be used for all sensor nodes in the sensor field because of the many technical difficulties involved with such a practice: global time synchronization, overhead of transmitting control messages, etc. Therefore, our solution is restrictively applied to the point where congestion is most acute: the bottleneck area. Congestion in other areas is handled by conventional methods. Applying various congestion-control techniques is possible, so that only the sink's neighbors may trigger the special method, while other areas utilize the normal solutions.

The rest of this paper is organized as follows: Section 2 discusses related work on general congestion control; Section 3 presents the congestion-control technique for the neighborhood of the sink, based on service differentiation and adaptive time-slot scheduling; Section 4 discusses the simulation results; and finally we conclude the paper in Section 5.

2 Related Work

Early research on congestion problems in wireless sensor networks has resulted in efficient techniques for relieving congestion. CODA (Congestion Detection and Avoidance) [4] provides open-loop hop-by-hop backpressure and close-loop multisource regulation. It reduces congestion by controlling the local transmission rate. ESRT (Event to Sink Reliable Transport) [5] uses congestion feedback from sensor nodes, and the sink controls the transmission rate by broadcasting control messages. A hybrid mechanism combining hop-by-hop flow control and rate control [6] has also been presented. Congestion controlled by distributing packets to neighbors [7] has been developed as well. They are designed for sensor networks and reflect their characteristics, but they do not consider the funneling effect.

Recently, some research has started to study the funneling effect on wireless sensor networks. Ee et al [8] present a congestion-control technique that takes the fairness issue into consideration for many-to-one routing. It is similar to other transmitting-rate control techniques, but considers fairness issues to reduce the funneling effect. Siphoning [9] is another method that offers a solution for the problem of the funneling effect. It uses extra hardware called a virtual sink to distribute congestion. Virtual sinks have two kinds of radio transmitters, a short-range radio and a long-range radio. The short-range radio is for communicating with other sensor nodes, and the long-range one is for communicating with other virtual sinks as well as the real sink. COMUT (Congestion Control for Multi-class Traffic) [10] applied a service-differentiation technique to congestion control based on clustering. These approaches all try to solve the funneling effect with unique techniques, but some hardware requirements or the overhead required for clustering management may cause other problems. Most importantly, none of the above solutions addresses the bottleneck problem, which could be the most critical point in the funneling effect.

3 Congestion Control Technique for Near-Sink Nodes

In this section, a congestion control technique for the bottleneck problem is discussed. First, a simple congestion-detection technique, as well as service differentiation, is explained. Second, the solution for the congestion problem, and the adaptive time-slot scheduling method, are discussed. Finally, the overall system design for implementation is described.

3.1 Congestion Detection and Service Differentiation

Congestion detection is the initial step in handling congestion problems. There are many existing methods for carrying this out, such as using channel loading and cumulative traffic, etc. In our situation, however, complicated congestion detection methods may add computation and transmission overhead, because we are considering only a one-hop relationship from a sink to its neighbors. Hence, the amount of data in the neighbors' buffer is used as the standard for congestion. If the amount of data exceeds the fixed threshold, then the system indicates the occurrence of congestion.

When data come into the buffers, they are sequentially stored. This does not cause any problem when there is no congestion. However, when a large amount of incoming data is transmitted, frequent transmissions may cause collision and make the nodes' retransmission and the status of congestion worse. Thus, some packets with imminent deadline expiration may not be sent to the sink because of previously-arrived packets that may have spare time before their deadlines. In order to solve this problem, we present a service-differentiated technique. The basic concept is based on sorting the buffers by priority, priority being defined here as deadline. Sorting data whenever they come into the buffer generates unacceptable overhead, so our system classifies data as urgent or normal. Only urgent data is sorted and placed at the front of the buffer as soon as an occurrence of congestion is detected. After sorting is done, incoming packets are inserted into the appropriate location by comparing their

priorities. If congestion is mitigated, no more packet classification is needed. Thus, the system will operate as it normally would.

3.2 Adaptive Time-Slot Scheduling

A time-slot scheduling technique is one of the effective ways of avoiding collision, which causes serious congestion problems. The time-slot scheduling method creates time-slots for nodes, which will transmit data only at a certain time scheduled by one control node such as a sink. If this method could be used for all sensor nodes, then a significant reduction in the amount of congestion experienced by the whole network would be expected. Unfortunately, however, this is not possible, because the needs for global time synchronization, plus high overhead for transmitting control messages, are hindrances when applied to wireless sensor networks. Our method uses time-slot scheduling only between a sink and its neighbors, while other nodes rely on other existing methods specifically designed for sensor networks.

Before explaining adaptive time-slot scheduling, we discuss the time-slot size, the basic unit of time required to send one message. The time-slot is calculated as shown in Equation 1.

$$time\_slot = \frac{n \times buf\_size \times threshold \times \alpha}{bitrate \times transmission\_time} \tag{1}$$

The above equation is based on the environment of the sink, the neighbor, the hardware specifications, and the given values acquired from experiments. n is the number of neighbors. *threshold* is the relative ratio of buffer usage to the total buffer size and is the deciding factor in determining whether congestion occurs or not. Multiplying n, threshold, and buffer size together gives the amount of data that must be processed to handle the congestion problem. α is the relative ratio of the size of data that has to be processed in one period of time quantum to the total size of data that must be processed to handle the congestion problem. This value determines if the proposed system can support soft real-time. For example, if n=10, *bitrate*=256(kbps), *buf_size*=1320(bytes), and *threshold*=0.6, the total size of data that must be processed to reduce congestion is 7920(bytes). If a time-slot is set at about 1 second, then α=0.32. This means that 10 neighbors need to spend one second to send data sequentially. Thus, if α=0.16, then they need to spend 0.5 seconds, and if a=0.64, two seconds will be spent. The value is decided experimentally. *transmission_time* and *bitrate* are hardware-dependent values which refer respectively to the minimal time required to transfer a data frame and to radio transmission speed. In our test, we set these values at 30(msec) and 256(kbps) respectively, based on MoteIV's Tmote Sky [11].

The adaptive time-slot scheduling method starts to operate as soon as the sink retrieves its neighbors' information. At the beginning, the sink proportionally distributes time-slots among its neighbors while no congestion is detected. Whenever packets are transmitted from neighbors to the sink, their buffer size and weight are included in the packets in order to control the distributing time-slots. Thus, there is no extra transmission cost. If congestion is detected, the sink checks the neighbors' weight, which is calculated at neighbor nodes as shown in Equation 2. This process redistributes time-slots, depending on each neighbor's buffer size and the number of high-priority data.

$$weight_i = \sum_{j=1}^{N} \left(priority_j \times \eta_j \right), \text{ where } N = \text{number of buffer frame}$$
$$\eta_j = \begin{cases} 1.0, & priority_j < 70 \\ 1.5, & priority_j \geq 70 \end{cases} \tag{2}$$

The scale of priority is from 1 to 100 and we decided that any priority over 70 would be set as urgent. Thus, urgent data receives one-and-a-half times the weight given to others. Each priority is multiplied by a different extra weight, and the sum of all frame information in a buffer becomes the final weight for a single neighbor.

After checking each neighbor's final weight, the sink starts to recalculate the proportion of the time-slot known as *time_quantum*. Congestion situations cannot be the same for all neighbors, so adaptive time-slot scheduling is necessary to achieve fairness. Congestion may become worse if the system does not care about fairness. This *time_quantum* is computed by Equation 3.

$$time\_quantum_i = \left\lceil \frac{weight_i}{\sum_{j=1}^{B} weight_j} \times total\_data\_quantum \right\rceil \tag{3}$$

Here, i^{th} node's *time_quantum* is decided by the relative ratio of i^{th} node's weight to the total weight of neighbors, and B is the number of the sink's neighbors. Therefore, if a node has many incoming and high-priority data, the sink increases the node's time quantum. Of course, the time-slot scheduling works dynamically, so it has great flexibility. With this policy, collision and fairness are achieved and the bottleneck problem is mitigated.

3.3 Overall System Structure

In this section, the proposed congestion-handling system is discussed based on the above theoretical ideas. Fig. 2 illustrates the overall process of the mechanism. The system works in both Init and Start state for the time synchronization. The congestion-handling system then operates among Idle, Send, and Control state.

The sink's state changes from Init to Start while broadcasting sync messages to its neighbor nodes. In Start state, the sink waits for the sync_ack messages which contain neighbors' information such as ID, weight, and priority. This information is stored in the neighbor's table. In Idle state, the sink receives data from the neighbors and decides if there is congestion. At the end of the time-slot, the sink sends a control message, and the above steps are repeated. If congestion occurs, *time_quantum* is recalculated and then the updated control message is sent to the neighbors. In the Idle state, neighbor nodes receive data messages from other sensor nodes. When their unique *time_quantum* starts, it triggers them to operate in the Send state. In this state, data from their buffers is transmitted to the sink.

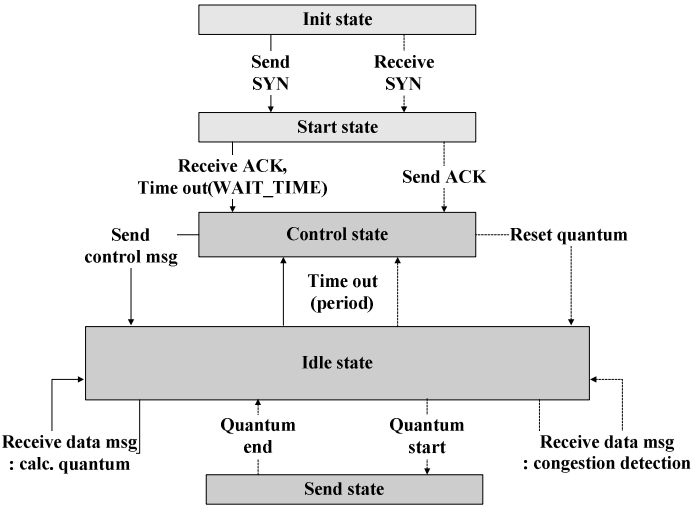


Fig. 2. State diagram for congestion-control system

4 Evaluation

To evaluate the performance of the proposed technique, we implemented it based on TinyOS [12] and used a TOSSIM simulator [13]. Evaluation consists of functionality tests, deadline miss and packet-drop ratio analysis. The last two of analyses were conducted for both the proposed method ("proposed", hereafter) and the normal method without the proposed congestion handling method ("normal", hereafter). Topology for all tests except that evaluating α consists of one sink and three neighbor nodes. For α, we used neighbors 3, 5, 7, 9, 12, and 15. The probability that each neighbor will receive data every 30msec is 0.3. The size of each neighbor' buffer is set as 30 x *packet_size* (44bytes). The threshold for congestion detection is set at 0.6. From the experiment, it is evident that a threshold of less than 0.6 generates overhead for transmitting unnecessary control messages. When the threshold surpasses 0.6, the speed at which data is received is faster than the speed at which data is processed from the buffer. We therefore decided to use 0.6 for the threshold. The bitrate is 256kbps and the size of a single slot is set at 30ms.

First, the system functionality is evaluated. In Fig. 3, we trace randomly-generated deadlines for 10 sec with the maximum value (5sec). The maximum deadline is required that our method is feasible for soft real-time applications. The value of the x-axis is the time required for generating packets. The min-deadline means that a deadline should be bigger than its generating time. As packets approach deadline, their priority has to be increased. From the experiment, therefore, it is evident that randomly-generated deadlines for simulation closely reflect real environments.

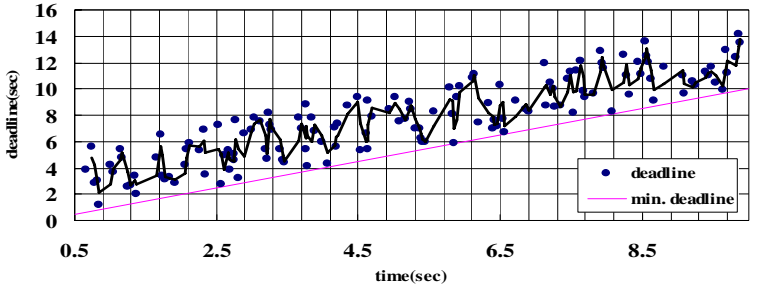
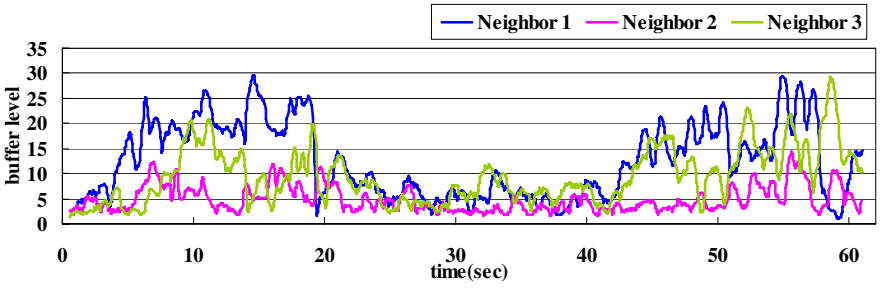
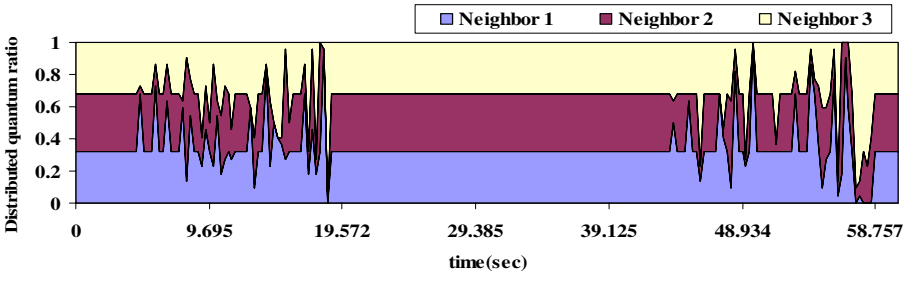


Fig. 3. Deadline trace



(a) Buffer level at neighbors 1, 2, 3

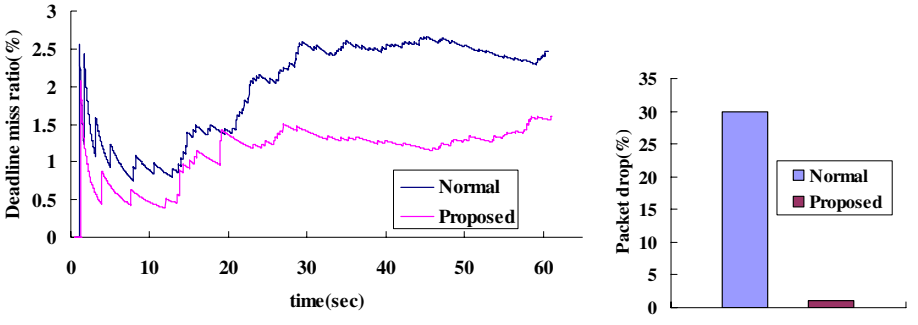


(b) Distributed time-quantum ratio of neighbors 1, 2, 3

Fig. 4. Buffer level and time-quantum ratio of neighbors

Fig. 4 shows the change of buffer level and time-quantum according to congestion level in the proposed congestion control method. Fig. 4(a) shows the variation of the buffer level as the time advances, and Fig. 4(b) shows the ratio of distributed time-quantum. As the buffer level is increased from about 5sec to 20sec, and from 40sec to 50sec, congestion occurs. At this time, our method detects the congestion and the sink divides time-quantum according to the neighbors' congestion level in order to relieve the congestion. When the buffer level of neighbor 1 exceeds the threshold at 5sec, the sink begins to grant more time-quantum to neighbor 1. Afterwards, neighbor 1 and neighbor 3 continuously receive many packets, so the increment of the buffer level is obvious. The sink distributes more time-quantum to 1 and 3 only and reduces the

time-quantum of neighbor 2. As a result, the congestion is solved, and at about 20sec the buffer level of all neighbors remains stable. The above figures also show that only the buffer level of neighbor 3 extremely increases at 57sec hence the amount of the time-quantum of neighbor 3 is assigned more than other neighbors. For other neighbor nodes, the differently assigned amount of time-quantum seems to generate congestion, but continuous changes of the amount of time-quantum according to congestion information eventually alleviate congestion as shown in the section 20sec to 45sec of Fig. 4(b).



(a) Deadline miss rate (b) Packet drop

Fig. 5. Deadline miss rate and packet drop without/with CCNS

Fig.5 shows the comparison of performance between the proposed congestion-control method (proposed) and one without our method (normal). First, deadline miss ratios for both methods are shown in Fig. 5(a). Our congestion-control technique prevents packet drop caused by collision, so the deadline miss ratio is reduced. Of course, complete prevention of packet drop is not possible because of buffer overflow. Our method also cannot avoid packet drop due to buffer overflow. As shown in Fig. 5(b), however, packet drop rarely occurs when the congestion-control mechanism is applied. The adaptive time-slot scheduling technique monitors each neighbor's buffer size, so if one of them has more data in the buffer, it automatically redistributes time-slots. However, the other method, without the time-scheduling technique, suffers from this kind of unbalanced incoming data.

α decides the quantum period of neighbor nodes. If the value increases to a certain point, the period increases as well. This leads to a reduced chance to control congestion; therefore, increasing α may cause severe deadline-miss. On the other hand, if the value decreases to a certain point, more frequent transmission for control messages from the sink to its neighbors may be another cause of congestion. Thus, α is set based on several experiments aimed at finding the optimal value. The evaluation for α is shown in Fig. 6. While changing α from 0.05 to 0.4 and the number of the sink's neighbors from 3 to 15, we observed a change in the miss ratio. When the number of neighbors is fewer than 10, specific α values generate the minimal deadline

miss ratio: when α =0.25 and the number of neighbors=3, the deadline miss ratio=3.15%; when α =0.1 and the number of neighbors=5, the deadline miss ratio=3%; when α =0.1 and the number of neighbors=7, 9 the deadline miss ratio=6.31% and 7.71% respectively. However, if the number of neighbors exceeds 10, the congestion control method starts to malfunction and the miss ratio exceeds 10%, regardless of whether the α value is changed.

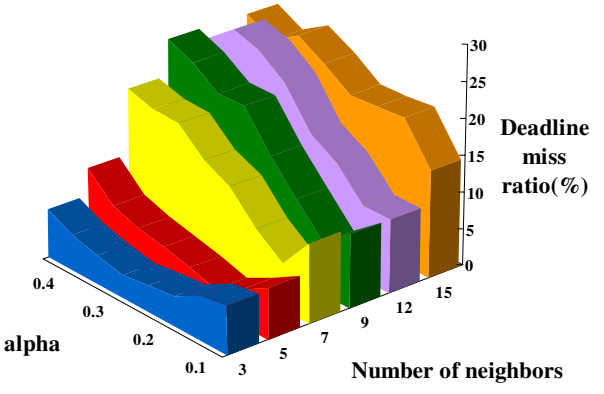


Fig. 6. Deadline miss rate as alpha, number of neighbors

5 Conclusion

There has been a great deal of research done on congestion-control techniques. Most of it, however, has not considered the bottleneck problem generated between a sink and its neighbors when high network traffic is expected. Efforts to avoid and reduce congestion between sources and the sink's neighbors are rendered useless if this problem is not specifically addressed. Our solution reduces collision by scheduling transmitting time-slots for the sink's neighbors and adaptively changing time-slots depending on each buffer size and on data priority. Moreover, fairness is also considered in our technique, in that packets having higher priority occupy earlier transmitting sequences. Applying the proposed solution in combination with some existing congestion-control methods should result in beneficial synergy. For future work, such combining of techniques will be necessary. In order to achieve fusing with other solutions, integrated congestion-detection techniques may be needed.

Acknowledgements

This work was supported by the National Research Laboratory (NRL) program of the Korea Science and Engineering Foundation (2005-01352) and the ITRC programs (MMRC) of IITA, Korea.

References

1. Akylidiz, I. F., Su, W., Sankarasubaramaniam, Y., Cayrici, E.: A Survey on Sensor Networks. IEEE Communication Magazine, vol. 40. (2002) 102-114.
2. Tilak, S., Abu-Ghazaleh, N. B., Heinzelman, W.: Infrastructure Tradeoffs for Sensor Networks. The 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA), Atlanta (2002) 49-58.
3. Jung, W. Shin, S., Pink, Choi, S., Cha, H.: Reducing Congestion in Real-Time Multi-party- tracking Sensor Network Applciations. The 1st International Workshop on RFID and Ubiquitous Sensor Network, Nagasaki (2005) 1191-1200.
4. Wan, C., Eisenman, S. B., Campbell, A. T.: CODA:Congestion Detection and Avoidance in Sensor Networks. In Proc. of ACM SenSys 2003, Los Angeles (2003) 266-279.
5. Sankarasubramaniam, Y., Akan, O. B., Akyildiz., I. F.: ESRT: Event-to-Sink Reliable Transport in Wireless Sensor Networks. In Proc. of MobiHoc 2003, Annapolis, Maryland (2003) 177-189.
6. Hull B., Jamieson, K., Balakrishnan, H.: Mitigating Congestion in Wireless Sensor Networks. In Proc. of ACM SenSys 2004, Baltimore (2004) 134-147.
7. Kang, J., Zhang, Y., Nath, B., Yu, S.: Adaptive Resource Control Scheme to Alleviate Congestion Control in Sensor Networks. In Proc. of the 1st Workshop on Broadcast Advanced Sensor Networks (BASENETS), San Jose (2004).
8. Ee, C. T., Bajcsy, R.: Congestion Control and Fairness for Many-to-one routing in Sensor Networks. In Proc. of ACM SenSys 2004, Baltimore (2004) 148-161.
9. Wan, C., Eisenman, S. B., Campbell, A. T., Crowcroft, J.: Siphon: Overload Traffic Management Using Multi-radio Virtual Sinks in Sensor Networks. In Proc. of ACM SenSys 2005, San Diego (2005) 116-129.
10. Karenos, K., Kalogeraki, V., Krishnamurthy, S. V.: Cluster-Based Efficient Routing in Wireless Sensor Networks. IEEE Workshop on Embedded Networked Sensor (EmNets 05), Sydney (2005) 107-114.
11. MoteIV. Tmote-sky-datasheet. http://www.moteiv.com/products/docs/tmote-sky-datasheet .pdf.
12. Levis, P., Madden, S., Polastre, J., Szewczyk, R., Whitehouse, K., Woo, A., Fay, D., Hill, J., Welsh, M., Brewer, E., Culler, D.: TinyOS: An Operating System for Sensor Networks. Ambient Intelligence, Springer, Berlin (2005) 115-148.
13. Levis, P., Lee, N., Welch, M., Culler, D.: TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications. ACM SenSys (2003) 126-137.

Information-Driven Sensor Selection Algorithm for Kalman Filtering in Sensor Networks*

Yu Liu[1], Yumei Wang[1], Lin Zhang[1], and Chan-hyun Youn[2]

[1] School of Information Engineering, Beijing University of Posts and Telecommunications, Beijing, China
liuyurainy@gmail.com, {ymwang, zhanglin}@bupt.edu.cn
[2] Information and Communications University, Republic of Korea
chyoun@icu.ac.kr

Abstract. In this paper, an information-driven sensor selection algorithm is proposed to select sensors to participate in Kalman filtering for target state estimation in sensor networks. The mutual information between the measurements of sensors and the estimated distribution of the target state is considered as the information utility function to evaluate the information contribution of sensors. And only those sensors with larger mutual information are selected to participate in the Kalman filtering iterations. Then the geographic routing mechanism is utilized to visit these selected sensors sequentially and set up a path to transport the state estimation information to the sink node. Simulation results show that compared with the shortest path tree algorithm, the information-driven sensor selection algorithm involves smaller participated sensors, and shorter total communication distance, while the estimation performance approaches the same bound.

1 Introduction

Wireless sensors have the abilities of sensing, computation and communication, and work under ad hoc mode to compose wireless sensor networks (WSNs). Target state estimation, i.e. collecting the measurements of sensors to estimate the target state and transporting the estimation results to the fixed sink node, is one of the typical applications of WSNs. The routing of these tasks is not only transferring data from one point to another, but should also be optimized for the information aggregation.

Many algorithms have been proposed to address the routing problem in WSNs. Energy-aware routing [1] selected the path to minimize the exhaustion chance of the energy. GPSR [2] is a stateless location-based protocol to route data around sensor holes over a planar network. Geocasting [3] routed data to a geographically defined region. However, none of the aforementioned routing algorithms consider information gathering and aggregation while routing data, which is a major desirable demand of sensor network applications. Directed diffusion [4] used the *publish* and the *subscribe*

* Supported by National Science Foundation of China (NSFC) with contract number (60502036).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 498 – 507, 2006.

messages to set up paths between data source nodes and sink nodes. It routed data based on low-level data attributes rather than sensors addresses, but did not set up optimized routing structures for information gathering and aggregation.

This paper adopts the information utility function to evaluate the contribution of sensors' measurements for target state estimation. Those sensors capable of providing large information contributions are selected to participate in the Kalman filtering iterations. The geographic routing protocol is utilized to pick up the measurements of the selected sensors sequentially and set up a path to the sink node. The sensors on the path combine the received information with their own measurements to execute Kalman filtering, i.e. performing in-network processing to decrease the amount of transmitted data and to refine the estimation result.

The idea of utilizing the information utility to manage sensing resources has been investigated in computer and robotics fields already. [5] introduced an expected utility measure for decentralized sensing system based on local decision, but the communication cost was not explicitly considered. [6] used a simple function to determine the utility of node without explicitly modeling the network spatial configuration. [7] proposed the information-directed routing algorithm to track targets in WSNs, which illumines us to explore the information-driven sensor selection algorithm in this paper for target state estimation in sensor networks.

In section 2, the target state estimation in sensor networks is analyzed, including the sensor model and the Kalman filtering iteration process. The information-driven sensor selection algorithm is introduced in section 3, and the geographic routing method to visit the selected sensors is in section 4. Section 5 is simulation results and analyses, and section 6 is the conclusion.

2 Target State Estimation in Wireless Sensor Networks

Target state estimation can be illustrated as the example of estimating the amplitude of an acoustic source whose location is assumed to be known by all the sensors in the network. In order to decrease the consumption of energy, these sensors are assumed to be managed in clusters [8]. At any time, there is only one active sensor (called leader) in each cluster to monitor the environment. Once observing the emergence of the target, the leader records the measurement, wakes up its neighboring sensors, and initiates a process of information aggregation to estimate the target state. Finally, the estimation result is transmitted to the sink node.

2.1 Sensor Model

Assume that there are N acoustic amplitude sensors [9] deployed in a two-dimensional region. The sensors output the amplitudes of the sound measured by their microphones. Suppose that

$$z_j^{(k)} = x^{(k)} \Big/ \left\| y_j - y \right\|^{\frac{\alpha}{2}} + w_j \tag{1}$$

Where $z_j^{(k)}$ is the measurement of the sensor j at the step k, and $x^{(k)}$ is a random variable representing the amplitude of the target acoustic source. α is the known distance attenuation coefficient of the sound wave (set to 2 in this paper), and $\|\bullet\|$ is

the Euclidean form. w_j is a Gaussian random variable with zero mean and σ_j^2 variance, y is the position of the target, and y_j is the position of the sensor j.

2.2 Estimation Iteration Process

For a linear system with normal distribution, we use the Kalman filtering [10] to estimate the target state, which is a specialization of sequential Bayesian filtering [11]. Assume the following discrete-time linear state model and linear observation model:

$$X^{(k+1)} = FX^{(k)} + V^{(k)} \tag{2}$$

$$Z_j^{(k)} = H_j X^{(k)} + W^{(k)} \tag{3}$$

Here F relates the target state at the previous step k to the state at current step $k+1$, and H_j relates the target state to the measurement. The random variables $V^{(k)}$ and $W^{(k)}$ represent the process noise and the measurement noise respectively. They are assumed to be independent, white, and with normal probability distributions

$$p(V) \sim N(0, \Sigma_v) \tag{4}$$

$$p(W) \sim N(0, \Sigma_w) \tag{5}$$

Here, Σ_v represents the process noise variance and Σ_w represents the measurement noise variance, which may change in each time step in practice, however we assume they are constant.

The probability $p\left(x^{(k)} / \overline{z}^{(k)}\right)$, called belief, represents the distribution of $x^{(k)}$ given the history of measurements up to step k, i.e. $\overline{z}^{(k)} = \left\{z^{(1)}, \cdots, z^{(k)}\right\}$. For a linear Gaussian model with Gaussian prior belief $p\left(x^{(k)} / \overline{z}^{(k)}\right)$, it can be proved that the prediction distribution $p\left(x^{(k+1)} / \overline{z}^{(k)}\right)$ and the posterior belief $p\left(x^{(k+1)} / \overline{z}^{(k+1)}\right)$ after applying the new measurement $z_j^{(k+1)}$ are still Gaussian [11], whose means and variances can be computed through Kalman filtering iterations with the mean and the variance of $p\left(x^{(k)} / \overline{z}^{(k)}\right)$ as well as the measurement $z_j^{(k+1)}$. We denote them as

$$p\left(x^{(k+1)} / \overline{z}^{(k)}\right) \sim N\left(\hat{x}_-^{(k)}, P_-^{(k)}\right) \tag{6}$$

$$p\left(x^{(k+1)} / \overline{z}^{(k+1)}\right) \sim N\left(\hat{x}^{(k+1)}, P^{(k+1)}\right) \tag{7}$$

Where $\hat{x}_-^{(k)}$ and $\hat{x}^{(k+1)}$ are the means, $P_-^{(k)}$ and $P^{(k+1)}$ are the variances of the prediction distribution and the posterior distribution respectively. The detailed processes of Kalman filtering iterations are as follows.

$$\hat{x}_{-}^{(k)} = F\hat{x}^{(k)} \tag{8}$$

$$P_{-}^{(k)} = FP^{(k)}F^{T} + \Sigma_{v} \tag{9}$$

$$K^{(k)} = P_{-}^{(k)}H_{j}^{T}\left(H_{j}P_{-}^{(k)}H_{j}^{T} + \Sigma_{w}\right)^{-1} \tag{10}$$

$$\hat{x}^{(k+1)} = \hat{x}_{-}^{(k)} + K^{(k)}\left(z_{j}^{(k+1)} - H_{j}\hat{x}_{-}^{(k)}\right) \tag{11}$$

$$P^{(k+1)} = \left(I - K^{(k)}H_{j}\right)P_{-}^{(k)} \tag{12}$$

Eq. (8) and (9) update the state and the covariance estimates from the step k to the step $k+1$, which are prediction processes. The Kalman gain $K^{(k)}$ is computed by Eq. (10). Eq. (11) is used to generate a posterior state estimate by incorporating the measurement $z_{j}^{(k+1)}$, and Eq. (12) obtains a posterior estimate error variance.

Along with the iteration process, $\hat{x}^{(k)}$ will approach the actual value of the target state, and $P^{(k)}$ will tend to be zero. Applying the Kalman filtering iterations to the target state estimation in sensor networks with acoustic amplitude sensors and acoustic target, we get $F = 1$, $\Sigma_{v} = 0$, $H_{j} = 1/\left\|y_{i} - y\right\|$ and $\Sigma_{w} = w$.

3 Information-Driven Sensor Selection Algorithm

The target state estimation in sensor networks aims for aggregating measurements of different sensors to improve the estimation accuracy, and to estimate the mean and the variance of the target state through Kalman filtering along with routing. Due to the limitations of battery power and wireless communication bandwidth, in-network processing is utilized to reduce the amount of communication bits, thus to decrease the energy consumption and prolong the lifetime of sensor networks.

Due to the spatial diversities, the measurements of different sensors have different contributions to the target state estimation. We prefer to collect the measurements of those sensors with larger information contributions to effectively aggregate information.

3.1 Information Utility Function

Information utility function is defined as $\psi : P\left(R^{d}\right) \to R$ [12], which acts on the class $P\left(R^{d}\right)$ of all possibility distributions on R^{d} with d being the dimension of target state x. ψ assigns a value to each element $p \in P\left(R^{d}\right)$, which indicates how dispersive or uncertain the distribution p is. Smaller value means a more spread out distribution while larger value means a tighter distribution.

We consider the mutual information [13] to quantify the expected contribution of individual sensors. Mutual information is a fundamental measure in information theory and is commonly used for characterizing the performance of data compression, classification, and estimation algorithms. The mutual information between two random variables U and V with a joint probability mass function $p(u,v)$ and marginal probability mass functions $p(u)$ and $p(v)$ is

$$I(U;V) \triangleq E_{p(u,v)}\left[\log\frac{p(u,v)}{p(u)p(v)}\right] = D\left(p(u,v)\,\|\,p(u)p(v)\right) \tag{13}$$

where $D(\bullet\|\bullet)$ is the Kullback Leibler distance [13] between the two distributions. It indicates how much information V conveys about U.

Under the Kalman filtering method, the information contribution of the sensor j with the measurement $z_j^{(k+1)}$ is

$$I_j = I\left(X^{(k+1)};Z_j^{(k+1)}\,|\,\overline{Z}^{(k)}\right) = E\left(\log\frac{p\left(x^{(k+1)},z_j^{(k+1)}\,|\,\overline{z}^{(k)}\right)}{p\left(x^{(k+1)}\,|\,\overline{z}^{(k)}\right)p\left(z_j^{(k+1)}\,|\,\overline{z}^{(k)}\right)}\right) \tag{14}$$

Intuitively, it indicates how much information $z_j^{(k+1)}$ conveys about the target state $x^{(k+1)}$ given the current belief. It can be interpreted as the Kullback Leibler distance between $p\left(x^{(k+1)}\,|\,\overline{z}^{(k)}\right)$ and $p\left(x^{(k+1)}/\overline{z}^{(k+1)}\right)$, i.e. the belief before and after applying the new measurement $z_j^{(k+1)}$ respectively. Hence, I_j reflects the expected amount of changes in the posterior belief brought upon by the sensor j. Larger change means more information, so we prefer to choose those sensors with larger mutual information to participate in the estimation process. It is worth pointing out that the mutual information I_j is an expected quantity rather than an observation. Hence, it can be computed based on the local knowledge of the location information of neighboring sensors and the current belief without exchanging sensor data in advance.

3.2 Information Utility for Linear Gaussian Model

[14] proposed a computationally feasible approach to calculate the information contribution, and the conclusion is

$$I_j = I\left(Z_j^{(k+1)};X^{(k+1)}\right) - H\left(X^{(k+1)}\right) \tag{15}$$

In case that Kalman filtering is adopted for target state estimation and the prior belief $p\left(x^{(k)}/\overline{z}^{(k)}\right)$ is Gaussian, the observation model in Eq. (3) suggests a normal conditional distribution of $z_j^{(k+1)}$

$$p\left(z_j^{(k+1)}\,|\,x^{(k+1)}\right) \sim N\left(H_j X^{(k+1)},\Sigma_w\right) \tag{16}$$

Given Eq. (7), the distribution of the measurement of the sensor j at the step $k+1$ as

$$p\left(z_j^{(k+1)}/\overline{z}^{(k)}\right)=\int p\left(z_j^{(k+1)}/x^{(k+1)}\right)p\left(x^{(k+1)}/\overline{z}^{(k)}\right)dx^{(k+1)}\sim N\left(H_j\hat{x}_-^{(k)},\Sigma_w+H_jP_-^{(k)}H_j^T\right) \tag{17}$$

The mutual information between the measurement $z_j^{(k+1)}$ of the sensor j and the target state $x^{(k+1)}$ at the step $k+1$ is

$$I\left(Z_j^{(k+1)};X^{(k+1)}\right)=c\log\frac{\left|\Sigma_w+H_jP_-^{(k)}H_j^T\right|}{\left|\Sigma_w\right|} \tag{18}$$

The value of I_j does not depend on the measurement $z_j^{(k+1)}$, but the coefficient H_j. For the acoustic amplitude sensors, $H_j=1/\|y_i-y\|$, and the information contribution I_j is only determined by the distance between the sensor and the target. Those sensors nearer to the target with higher H_j value can provide more information contributions.

4 Geographic Routing Method

Based on the information-driven sensor selection algorithm, if the mutual information is used to evaluate the information contribution of each sensor's measurement for target state estimation, its value is only related to the distance from the sensor to the target. The closer the distance is, the larger the information contribution will be. So we prefer to collect the measurements of sensors closer to the target to participate in the Kalman filtering iterations.

The routing method for target state estimation in sensor networks firstly collects the measurements of the sensors in the selected set for Kalman filtering iterations, and then delivers the aggregated estimation results to the sink node based on the geographic routing protocol. As illustrated in Fig. 1, the target is located at the center of the network, and the sink node is at the bottom left corner. Both of their locations are known by all the sensors. The routing process contains two stages.

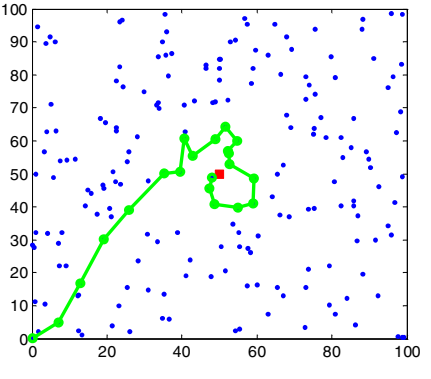


Fig. 1. Information-driven algorithm

Firstly, given the selected sensor set, the iteration result has no relation to the selecting order of the sensor for iteration. In order to decrease the total communication distance and the energy consumption, the leader of the cluster where the target is initiates the Kalman filtering process. It realizes the first Kalman filtering based on its measurement and the initialization of the estimated value as well as the error variance. And it transports the result to the neighboring sensor nearest to the target, which will carry out the second iteration based on the received prior belief and its measurement, and selects the next sensor similarly. The operation repeats until all the neighbor sensors which are in one hop away from the target have been visited. The data gathering path extends around the target to sequentially pick up the measurements of the selected sensors for estimation update, as shown in Fig. 1.

Secondly, the ultimate object of target state estimation is to abstract the estimated mean and variance at the sink node for further processing. So the routing in this stage needs to deliver the results attained from Kalman filtering iterations to the sink node. We adopt the geographic routing protocol to set up a shortest path to the sink node. During routing, the sensors on the path can also do Kalman filtering to refine the estimate, but their contributions are rarely small.

5 Simulation Results

We simulate the three algorithms, i.e. the shortest path algorithm, the shortest path tree algorithm and our information-driven sensor selection algorithm, and compare their performances of the total communication distance and the estimation error variance in the function of the number of participated sensors.

The shortest path algorithm tends to set up the shortest path from the target to the sink, and only those sensors on the path will participate in the estimation, as shown in Fig. 2. The shortest path tree algorithm sets up a multicast tree rooted at the sink node throughout the whole sensor network. The created shortest tree has no relation to the target location. The measurement data are routed to the root along the tree, and the crossing nodes perform in-network processing to aggregate information. In this algorithm, all sensors will participate in the estimation, as illustrated in Fig. 3.

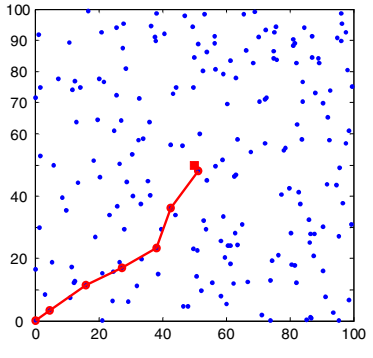


Fig. 2. Shortest path algorithm

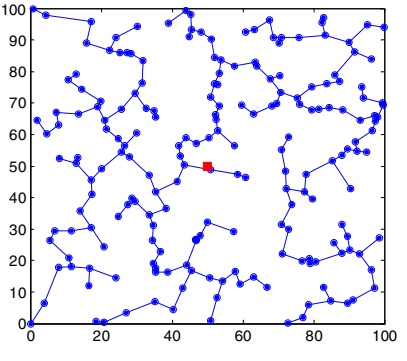


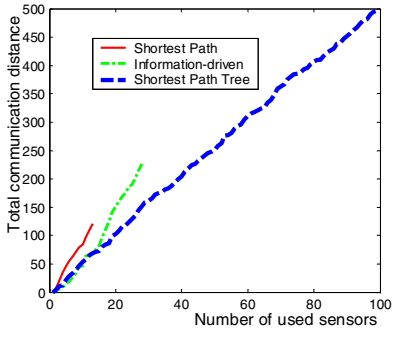
Fig. 3. Shortest path tree algorithm

5.1 Simulation Configuration

There are 200 acoustic amplitude sensors randomly deployed in a two-dimensional area of $100\times100\text{m}^2$. The radio range of each sensor is $15m$. Suppose the target acoustic source is at the center of the network, i.e. $(50,50)$, and the location of the sink node is $(0.1,0.1)$. The amplitude of the acoustic source x is set to 1, and the variance of the measurement noise Σ_w is 0.005. In our information driven sensor selection algorithm, we only select those sensors within $15m$ away to the target to participate in the estimation.

5.2 Comparison of Communication Distance

We first compare the total communication distances of the three algorithms, as shown in Fig. 4. The horizontal axis represents the number of participated sensors and the vertical axis represents the total communication distance. The shortest path algorithm directly builds up the shortest path from the target to the sink, resulting in the smallest number of participated sensors and the shortest communication distance. For the shortest path tree algorithm, no matter where the target is, the measurements are collected along the shortest tree, resulting in the largest number of participated sensors and the longest communication distance. For the information-driven sensor selection and routing algorithm, only one-hop neighboring sensors of the target and those on the path from the target to the sink participate in the estimation, so both the number of participated sensors and the communication distance are moderate.



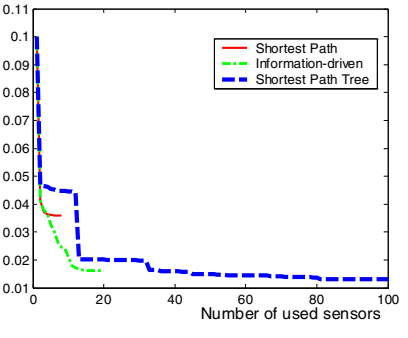


Fig. 4. Comparison of communication distance **Fig. 5.** Comparison of estimation error variance

5.3 Comparison of Estimation Performance

We compare the error variances of Kalman filtering, i.e. the values of $P^{(k)}$ of the posterior beliefs, to evaluate the estimation performance,. As illustrated in Fig. 5, the horizontal axis is the number of participated sensors and the vertical axis is the value of $P^{(k)}$. For the shortest path algorithm, the number of participated sensors with large contribution is small, and along with the extension of the path to the sink, the sensors becomes more and more far away from the target and their information contributions

decrease quickly. So the final estimation error variance is still large. For the shortest path tree algorithm, measurements of all sensors are collected, so its final variance is the bound for all the sensor selection algorithms. In Fig. 5, we only plot the behaviors of the first 100 sensors, and the last 100 sensors bring only a little bit change to the variance. In the information-driven sensor selection and routing algorithm, there are not so many sensors participate in the estimation. Those are chosen for their excellent information contributions based on the information-driven sensor selection algorithm. So the variance of the estimation decreases smoothly, and the final result approaches the bound determined by the shortest path tree algorithm. Compared with the shortest path algorithm, our information driven algorithm gets higher estimation accuracy at the expense of a little more communication cost.

6 Conclusions

Target state estimation is one of the typical applications of sensor networks. We firstly propose an information-driven sensor selection algorithm that utilizes the mutual information between the measurements of sensors and the estimated distribution of the target as the information utility function to evaluate the information contribution of their measurements. And only those sensors with larger information contributions are chosen to participate in the Kalman filtering estimation. Then the geographic routing protocol is adopted to visit the rich information sensors and set up the routing path to the sink. With this algorithm, only those sensors with larger contributions or on the path from the target to the sink are selected to participate in the estimation, which results in fewer participated sensors and shorter communication distance. Simultaneously, the principle of information-driven sensor selection guarantees that the estimation performance approaches the bound determined by the shortest path tree algorithm.

Our information-driven sensor selection algorithm can be widely used in applications such as distributed detection, classification and observation. In the future, we will apply this algorithm to estimate the target state as well as to locate or track the target.

References

1. Shah, R.C., Rabaey, J.M.: Energy aware routing for low energy ad hoc sensor networks. In Proc. IEEE Wireless Commun. Netw. Conf., Orlando, FL, (2001) 350-355
2. Karp, B., Kung, H.T.: Greedy perimeter stateless routing for wireless networks. In Proc. MobiCom, Boston, MA, (2000) 243-254
3. Ko, Y.-B., Vaidya, N.H.: Geocasting in mobile ad hoc networks: Location-based multicast algorithms. In Proc. IEEE Workshop Mobile Comput. Syst. Appl., New Orleans, LA, (1999) 101-110
4. Intanagonwiwat, C. Govindan, R. Estrin, D. Heidemann, J. Silva, F.: Directed diffusion for wireless sensor networking. IEEE/ACM Transactions on Networking. Volume 11, issue 1, (2003) 2-16
5. Manyika, J., Durrant-Whyte, H.: Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach. Ellis Horwood, New York (1994)

6. Byers, J., Nasser, G.: Utility-based decision-making in wireless sensor networks. In Proc. IEEE First Annual Workshop on Mobile and Ad Hoc Networking and Computing, Boston, MA, (2000) 143-144
7. Feng, Zhao, Jaewon, Shin, Reich, J.: Information-driven dynamic sensor collaboration. IEEE Signal Processing Magazine, Volume 19, Issue 2, (2002) 61-72
8. Wei-Peng, Chen, Hou, J.C., Lui, Sha: Dynamic clustering for acoustic target tracking in wireless sensor networks. IEEE Transactions on Mobile Computing. Volume 3, Issue 3, July-Aug. (2004) 258-271
9. Juan. Liu, Reich, J.E., Feng, Zhao: Collaborative in-network processing for target tracking. EURASIP, J. Appl. Signal Process, vol. 2003, Mar. (2003) 378-391
10. Spanos, D.P., Olfati-Saber, R., Murray, R.M.: Approximate distributed kalman filtering in sensor networks with quantifiable performance. Information Processing in Sensor Networks. April (2005) 133-139
11. Y.C., Ho, R.C.K., Lee: A bayesian approach to problems in stochastic estimation and control. IEEE Trans. Automat. Contr.. vol.9, (1964) 333-339
12. M., Chu, H., Haussecker, Feng, Zhao: Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks. Int. J. High-Performance Comput. Appl., vol. 16, no. 3. (2002) 293-313
13. T.M., Cover, J.A., Thomas: Elements of Information Theory. Wiley, New York, (1991)
14. Ertin, Emre, Fisher, John W., C.Potter, Lee: Maximum mutual information principle for dynamic sensor query problems. In Proc. Information Processing in Sensor Networks, Palo Alto, California, USA. (2003) 405-416

TwinsNet: A Cooperative MIMO Mobile Sensor Network

Qingquan Zhang[1], Woong Cho[2], Gerald E. Sobelman[1],
Liuqing Yang[2], and Richard Voyles[3]

[1] Department of Electrical and Computer Engineering,
University of Minnesota Minneapolis, MN 55455 USA
{zhan0511, sobelman}@umn.edu
[2] Department of Electrical and Computer Engineering,
University of Florida Gainesville, FL 32611 USA
woongcho@ufl.edu, lqyang@ece.ufl.edu
[3] Department of Computer Science and Engineering,
University of Minnesota Minneapolis, MN 55455 USA
voyles@cs.umn.edu

Abstract. A distributed sensor network with mobility provides an ideal system platform for surveillance and for search and rescue applications. We consider a system design consisting of a set of autonomous robots communicating with each other and with a base station to provide image and other sensor data. A robot-mounted sensor which detects interesting information coordinates with other mobile robots in its vicinity to stream its data back to the base station in a robust and energy-efficient manner. The system is partitioned into twin sub-networks in such a way that any transmitting sensor will pair itself with another nearby node to cooperatively transmit its data in a multiple-input, multiple-output (MIMO) fashion. At the same time, other robots in the system will cooperatively position themselves so that the overall link quality is maximized and the total transmission energy in minimized. We efficiently simulate the system's behavior using the Transaction Level Modeling (TLM) capability of SystemC. The simulation results demonstrate the utility of our design and provide insights into performance of the system.

1 Introduction

The 1980s and 1990s brought the rise of the industrial robot arm. Robots assembled virtually anything in high-volume, carefully controlled assembly lines. At first, they worked alone as "islands of automation." Gradually, as networking improved, the benefits of teaming were realized to create more capable systems with a trend toward larger numbers of smaller robots [1].

The 2000s have brought the rise of the mobile robot. They have followed a similar trajectory by starting alone, in carefully controlled environments. Yet, a new era has suddenly emerged as mobile robots have begun to appear in uncontrolled environments. The Roomba robot can autonomously vacuum virtually any residential room (with an amenable floor type) in which it is placed while the DARPA Grand Challenge [2] showed robots can navigate difficult mountainous desert terrain for long periods of time. As with robot arms, the natural trend is moving toward robot teams (even "swarms" [3]) of larger number with smaller sized individuals[4].

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 508 – 516, 2006.

Small size presents a significant problem for mobile robot teams. Networking is the key enabler and the only practical method for networking among large numbers of mobile robots is RF wireless. As robots get smaller, their antennas get closer to the ground. This detunes the antennas, reduces line-of-sight, and increases multi-path problems. Furthermore, small robots carry small batteries, limiting the radiated power of onboard RF systems. Finally, as robots move into extremely uncontrolled environments such as subterranean urban search and rescue [5], [6], high performance RF networking becomes a major concern. Teaming to maintain the network and conserve power is just as important, and often synonymous with, teaming to accomplish the task.

For applications in urban search and rescue, high quality video is the sensor mode of choice [7]. Imagery from different viewpoints must be relayed from remote mobile sensors back to the human operator. Therefore, it is desirable to maximize communication bandwidth across a team of robots while minimizing power consumption and allowing the robots to achieve their individual goals. The robots have individual goals for search and exploration, but common goals for maintaining the networking infrastructure. We assume every robot can act as both a remote sensor and a network relay. As robots fan out from the base station into the hostile RF environment, they quickly lose direct contact with the base station and must rely on multi-hop relays to maintain the network. Therefore, issues such as link quality, power minimization and robustness are major factors to be considered when designing wireless sensor networks for such applications.

In recent years, multi-input multi-output (MIMO) systems have been widely adopted to enhance the performance of modern communication systems. In such systems, multiple antennas are employed at the transmitter and/or receiver to combat channel fading with space diversity while enabling significant increases in the capacity of the wireless data link (see e.g. [8]). MIMO techniques are readily applicable to conventional fixed base stations because they have ample processing and power resources and can accommodate multiple antennas with beamforming or transmit diversity capabilities. Unfortunately, multiple antennas are not feasible for mobile robot teams due to their size, power and complexity constraints. An alternative is to let groups of robotic sensors in geographic proximity of one another form cooperative distributed antenna arrays and serve jointly as multiple antennas to enable distributed space diversity and low-power connectivity (see Figure 1(a)).

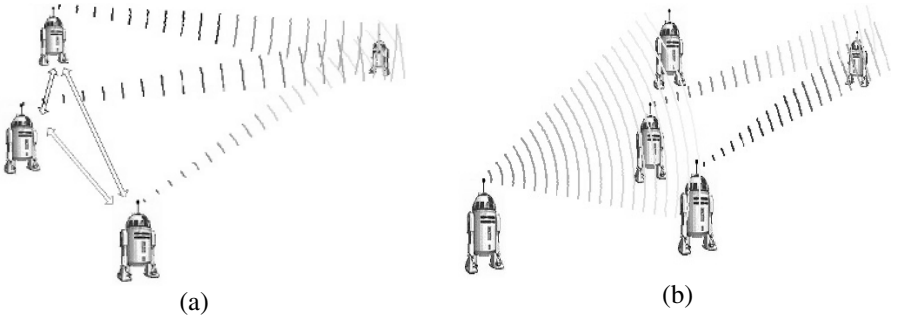


Fig. 1. Examples of cooperative communications

An additional form of cooperation among the small-sized robotic sensors is relayed communications. Specifically, the non-information-bearing nodes can help relay the information from the source nodes to the destination nodes, as depicted in Figure 1(b). This can effectively extend the small coverage range by individual robotic sensors due to their limited power and the intensive multipath channel fading effects.

The remainder of this paper is organized as follows. In Section 2, our cooperative MIMO dual-tree network topology and communications protocol are described. In Section 3, the use of SystemC-based transaction-level modeling techniques for this application are explained. Then, in Section 4, we present simulation results for the behavior of the network as a function of the number sensors and speed of movement of the robots. Finally, our conclusions and directions for future research are given in Section 5.

2 Design of TwinsNet

In this section, we describe the structure and operation of TwinsNet, which creates dual sub-networks to allow for cooperative 2 x 2 MIMO transmission of data packets between robot-mounted sensor nodes and the base station.

2.1 Establishment of Network Topology

Each sensor node is assigned a unique ID within the network. During network initialization, the base station issues messages which propagate throughout the system. Two disjoint sub-networks are created such that all the nodes in one tree have even ID numbers (called the Girl Tree) while all nodes in the other tree have odd ID node numbers (called the Boy Tree). Each node in both trees also has a specific Father and Mother, which are one hop closer to the base station. A pair of even and odd nodes are grouped together as a Sister/Brother. These two nodes correspond either to two sensors on the same robot or two sensors on separate robots which are physically close to one another. In this manner, a virtual 2×2 MIMO structure is overlaid onto the network to increase the throughput capabilities of the system. In addition, robots which do not currently have data to transmit will cooperate with an information bearing sensor by positioning themselves in such a way as to minimize overall energy consumption. (For the purposes of this study, we assume such robots do not have competing positioning goals. This assumption can be relaxed in future work.)

An example network topology for TwinsNet is shown in Figure 2. Nodes 7 and 8 are Brother/Sister, Node 5 is their Father and node 6 is their Mother, as they are closer to the base station. When information is detected by either node 7 or node 8, it will exchange information with its sibling and data will be sent back to the base station via two routing trees (Boy/Girl Trees). Nodes 7 and 8 communicate with nodes 5 and 6 a using a 2 x 2 MIMO transmission technique. A series of such transmissions are used to relay the information back to the base station. After the initialization process, each node maintains a neighbor table and dynamically updates it to reflect changes in link quality due to the robot-mounted sensor movements.

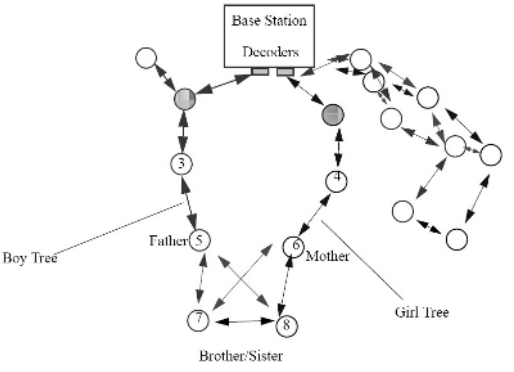


Fig. 2. An example topology of TwinsNet

2.2 Link Quality Estimation

After the tree building process has been completed, the sensor nodes will maintain their Parent and Brother/Sister assignments although they may be moving around within some region. However, a new initialization procedure will be invoked when the link quality for any sensor node drops down below a certain threshold. If this condition is met, then one or more nodes would have to find another candidate link available to them and a portion of the tree building protocol will be invoked.

2.3 MIMO Transmission and Minimum Energy Positioning

The communications between the Sister/Brother nodes to the parent nodes take the form of cooperative antenna array. The Sister/Brother nodes first exchange their information to be transmitted. Depending on the required information rate and error performance, and relying on the availability of channel information, the distributed antenna arrays can deploy different corporation strategies. Examples of such strategies include transmit diversity and beamforming. These are complementary techniques: beamforming is capacity achieving when perfect channel state information (CSI) is available at both the transmitting and the receiving sides; while space-time coding (STC) requires no CSI at the transmitting side. The former is *adaptive* to the acquired CSI while the latter remains *robust* regardless of the unknown CSI.

At the next stage, the transfer of information from the parent nodes to the data fusion center is achieved via non-information-bearing nodes serving as relays. If a low-rate high-performance strategy is adopted in the Sister/Brother-to-Parent link, then the Boy tree and the Girl tree are relaying the same information. Otherwise, they are conveying distinct data streams. A majority of existing works on relay networks focuses on coherent demodulation based on the availability of the channel state information at both the relays and the destination node (see e.g., [9], [10]). Accurate estimation of the CSI, however, can induce considerable communication overhead and transceiver complexity, which increase with the number of relay nodes employed. In addition, CSI estimation may not be feasible when the channel is rapidly

time-varying. To bypass channel estimation, cooperative diversity schemes obviating CSI have been recently introduced. These relay systems rely on non-coherent or differential modulations, including conventional frequency-shift keying (FSK) and differential phase-shift keying (DPSK) [11], [12], [13], , as well as space-time coding (STC-)based ones [14], [15].

To improve the error performance and enhance the energy efficiency of relay networks, optimum resource allocation recently emerged as an important problem attracting increasing research interests (see e.g., [16], [17], [18], [19]). However, all of these works only consider the power allocation issue. To this end, we take a major shift by jointly optimizing the power distribution and the source/relay locations. Interestingly, we have found that location optimization may be more critical than energy optimization [20]. In addition, it can also better exploit the mobility of the robot teams.

3 Transaction-Level Modeling of TwinsNet with SystemC

Recently, a system design methodology based on Transaction-Level Modeling (TLM) has been gaining acceptance in many application areas [21]. This methodology is enabled by using a language such as SystemC [22].

We have implemented an efficient TLM-based simulation model of TwinsNet in order to evaluate its performance. The basic elements in our TLM-based SystemC simulation are the base station and sensor node modules, which are described below.

3.1 Base Station

In order to implement the function of receiving and processing the data arriving from the sensors within the network, a bidirectional port array is created to enable the base station module to gain access to any channel connecting to those sensor nodes. Either of two scanning options can be chosen for the base station to communicate with the sensor nodes. One technique is to respond to its data input ports in an interrupt-driven manner. As the number of channels wishing to communicate at a specific time cannot be predicted in advance, it may be useful for the base station to automatically respond to a channel whenever there is a request from a particular sensor node. The other option is to use polling, i.e. to scan its input ports in a cyclic manner. Both of these options can be implemented by using either the `SC_THREAD` or `SC_ METHOD` process. A key difference between these two types of processes is that an `SC_METHOD` process can't be suspended during its execution while an `SC_THREAD` process may be suspended.

3.2 Sensor Node

The processes inside each sensor node are organized to assure the successful and fast collection and delivery of data throughout the network. A sensor node includes ports which are inherited from the `sc_fifo` interface. It sends its data into the channel making use of the port's inherited functionality.

4 Simulation Results

We have implemented our TLM model on a Sun Solaris based SystemC platform. Specifically, we have used a Sun Blade 150 workstation having a 650 MHz clock frequency and 512 Mbytes of memory. The simulation program has been written using the SystemC 2.1 distribution available form www.systemc.org.

4.1 Link Quality Analysis

To model if the connectivity of mobile sensor nodes is successful for any given transaction, a random process is used for the link quality. If a randomly generated number falls into the range corresponding to a viable transmission, then the message will be transmitted. For example, suppose that the current link quality for a node to connect to its Father is 80%. If the C rand function (0 < rand( ) <1) gives a result of 0.9, then the message is considered to be lost. The link quality is modeled as follows:

$$link \quad quality \quad = \frac{A}{\pi r^2} + B \tag{1}$$

where A and B are adjustable parameters and where r is the distance between the two sensor nodes.

In order to determine the impact of sensor movement and the density of sensors within a physical region, we considered a grid of size 500 meters by 500 meters with a grid spacing of 10 centimeters. Events are randomly generated at the leaf nodes of each of the two trees. This process is controlled by the timer process inside the model. After timing out, a simulated event is triggered. The message related to that event is then transmitted upstream by the network nodes. A packet loss is assumed to occur if a node cannot find any connection around it having a link quality of greater than 70%. The node parameters we considered are:

1. The movement of nodes: A robot's speed is set to be constant in each simulation but the direction is randomly chosen using a Monte Carlo algorithm.
2. The number of nodes: We consider a situation in which all of the mobile nodes are initially placed within a small circle so that all of them can reliably detect other nodes. The effects due to varying the number of nodes are investigated

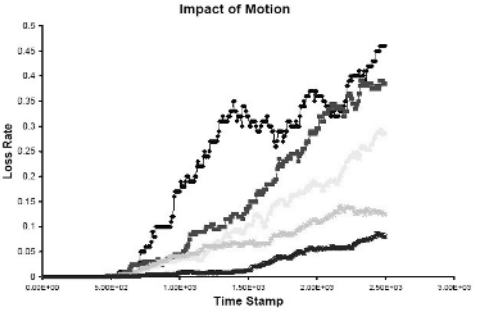


Fig. 3. Loss rate as a function of sensor movement

The impact of node count on loss rate was also studied, with our simulation results shown in Figure 4. As expected, the loss rate decreases for increasing node count. Moreover, the loss rate for a given rate of speed is less for the TwinsNet approach compared to a baseline single transmitter/receiver approach, particularly for the case where the robot-mounted sensors are moving at a high speed.

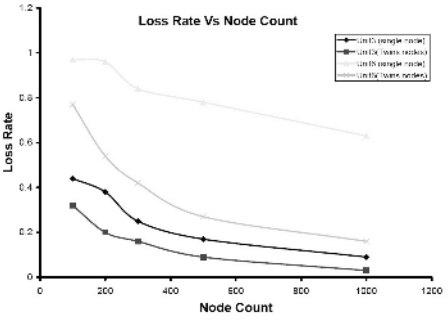


Fig. 4. The loss rate as a function of node count. (Unit3 means moving at a rate of 3 distance units per time unit, and similarly for Unit6.) Results for both single-tree (i.e., non-MIMO) and TwinsNet operational modes are shown for comparison purposes.

5 Conclusions and Future Work

Our simulations show a clear benefit, in terms of link loss rate, to our MIMO-based TwinsNet protocol. The improved link loss rate implies greater availability of critical data under our assumption that continuous data is required by the application, as is the case for search and exploration tasks. The network can be scaled up to a large number of robot-mounted sensors to provide multiple streams of high-quality image data together with coverage over a wide geographical area in a robust and energy-efficient manner. Our TLM-based SystemC modeling technique provides a fast and effective means for evaluating the performance of these large and complex systems.

Although our uniform model for link quality is unrealistic for complex, unstructured environments, the MIMO transmission scheme mitigates asymmetries more so than a point-to-point scheme. Therefore, we believe the trends are accurately represented as the performance of MIMO over non-MIMO systems should only be enhanced by realistic conditions including RF shadows and multi-path interference. In our motion studies, we have thus far only simulated random movement. Planned movement can further improve the performance of the network. In future work, we will continue to investigate the effects of deliberate motion on the protocol effectiveness and the impact of localization accuracy on MIMO transmission as we move toward the implementation of a prototype demonstration.

Furthermore, any active networking scheme involves the fusing or negotiation between the competing network maintenance goals and task goals. In this case, location optimization for the network and location optimization for the search task must be balanced. Comparison of approaches to achieve this balance is another major topic for future study.

References

1. Brown, H., Muir, P., Rizzi, A., Sensi, M.C. and Hollis, R.: A precision manipulator module for assembly in a minifactory environment, Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 2, pp. 1030 – 1035, 2001.
2. Urmson, C., Anhalt, J., Clark, M., Galatali, T., Gonzalez, J.P., Gowdy, J., Gutierrez, A., Harbaugh, S., Johnson-Roberson, M.H., Kato, Koon, P.L., Peterson, K., Smith, B.K., Spiker, S., Tryzelaar, E.and Whittaker, W.L.: High Speed Navigation of Unrehearsed Terrain: Red Team Technology for Grand Challenge 2004, Tech. Report CMU-RI-TR-04-37, Robotics Institute, Carnegie Mellon University, June, 2004.
3. Schultz, Alan C., Parker, Lynne E and Schneider, Frank E. (Eds.): Multi-Robot Systems: From Swarms to Intelligent Automata, Proceedings from the 2003 International Workshop on Multi-Robot Systems, Volume II, Springer, 2003
4. Sibley, G.T., Rahimi, M.H. and Sukhatme, G.S.: Robomote: A Tiny Mobile Robot Platform for Large-Scale Ad-hoc Sensor Networks, Proceedings of the IEEE Conf. on Robotics and Automation, 2002.
5. Murphy, R. R.: Rescue robots at the World Trade Center from Sept. 11-21, 2001, IEEE Robotics and Automation Magazine, June 2004.
6. Voyles, R.M., Larson, A.C., Lapoint, M. and Bae, J.: Core-Bored Search-and-Rescue Applications for an Agile Limbed Robot, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 1, pp. 58-63, 2004.
7. Casper, J.: Human-Robot Interactions during the Robot-Assisted Urban Search and Rescue Response at the World Trade Center, Computer Science and Engineering Report, University of South Florida, Apr, 2002.
8. Alamouti, S. M.: A simple transmit diversity technique for wireless communications, IEEE Journal on Selected Areas in Communications, Vol. 16, No. 8, pp. 1451-1458, Oct. 1998.
9. Laneman, J. N., Tse, D. N. C., and Wornell, G. W.: Cooperative diversity in wireless networks: Efficient protocols and outage behavior, IEEE Trans. on Information Theory, vol. 50, no. 12, pp. 3062-3080, December 2004.
10. Sendonaris, A., Erkip, E.,and Aazhang, B.: User cooperation diversity, part {I: System description, IEEE Trans. on Communications, vol. 51, no. 11, pp. 1927-1938, November 2003.
11. Chen, D., Laneman, J. N.: Cooperative diversity for wireless fading channels without channel state information, Proc. of Asilomar Conf. on Signals, Systems, and Computers, Monterey, CA, November 7-10, 2004, pp. 1307-1312.
12. Tarasak, P., Minn, H.,and Bhargava, V. K.: Differential modulation for two-user cooperative diversity systems, IEEE Journal on Selected Areas in Communications, Vol. 23, No. 9, pp. 1891-1900, September 2005.
13. Zhao, Q.and Li, H.: Performance of a differential modulation scheme with wireless relays in Rayleigh fading channels, Proc. of Asilomar Conf. on Signals, Systems, and Computers, vol. 1, Monterey, CA, November 7-10, 2004, pp. 1198-1202.
14. Cho, W., Yang, L.:Differential modulation schemes for cooperative diversity, Proc. of IEEE International Conference on Networking, Sensing and Control, Ft. Lauderdale, FL, April 23-25, 2006.
15. Cho, W. and Yang, L.: Distributed differential schemes for cooperative wireless networks, Proc. of Intl. Conf. on ASSP, Toulouse, France, May 15-19, 2006.

16. Anghel, P. A., Kaveh, M.,and Luo, Z. Q.: Optimal relayed power allocation in interfernce-free non-regenerative cooperative systems, Proc. of Signal Proc. Workshop on Advances in Wireless Communications, Lisbon, Portugal, July 11-14, 2004, pp. 21-25.
17. Deng, X., Haimovich, A. M.: Power allocation for cooperative relaying in wireless networks, IEEE Communications Letters, Vol. 9, No. 11, pp. 994-996, November 2005.
18. Hasna, M. O., Alouini, M.: Optimal power allocation for relayed transmissions over rayleigh-fading channels, IEEE Trans. on Wireless Communications, Vol. 3, No. 6, pp. 1999-2004, November 2004.
19. Liang, Y.,Veeravalli, V. V.: Gaussian orthogonal relay channels: Optimal resource allocation and capacity, IEEE Trans. on Information Theory, vol. 51, no. 9, pp. 3284-3289, September 2005.
20. Cho, W.and Yang, L.: Joint Energy and Location Optimization for Relay Networks with Differential Modulation, Proc.of Globecom Conf., San Francisco, CA,November 27-December1,2006 (submitted).
21. Cai, L., Gajski, D.: Transaction level modeling: an overview, First IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis, 2003.
22. Habibi, A., Tahar, S.: Design for verification of SystemC transaction level models, Proceedings. Design, Automation and Test in Europe, , Munich, Germany, March, 2005.

Scalable and Low-Cost Acoustic Source Localization for Wireless Sensor Networks

YoungBin You and HoJung Cha

Department of Computer Science, Yonsei University
Seodaemum-gu, Shinchon-dong 134, Seoul 120-749, Korea
{ybyou, hjcha}@cs.yonsei.ac.kr

Abstract. Typical applications of wireless sensor networks require scalability because they are deployed in a large area. Conventional centralized systems for acoustic source localization experience practical difficulties since they focus on high accuracy in a small area rather than scalability in a large region. In this paper, we present a scalable and light-weight acoustic source localization system suitable for use with resource constrained hardware. The distributed mechanism has low complexity and an error-tolerant algorithm for construction of a feasible system in a real environment. The proposed system has been implemented and validated by experiments in a real environment.

1 Introduction

Conventional research in WSN has focused on the centralized system since the acoustic source localization algorithm needs high computational complexity [1-9]. The centralized algorithm is similar to the traditional microphone array (MA) in computational complexity. A powerful base station computes the data from all nodes in the field. However, the nodes in many WSN applications are scattered in a large area and their resources (e.g. computation and network) are constrained. For these reasons, few acoustic source localization systems exist that are implemented in the real world. Recent research investigated constructing the system by a 2-tier architecture consisting of constrained sensor devices and powerful cluster heads [10-12]. The inexpensive sensor devices listen to the signal and pass it to the cluster head through the network, and the cluster head estimates the source location with powerful computational resources. This approach reduces the cost and maintains the computational power of the system. The cluster-based acoustic source localization system has limitations when the system is actually deployed. The sensor nodes as well as the cluster head is deployed uniformly. Topologically, the cluster should be located in the center of the sensor devices. The clustering-based systems have additional overhead for maintaining the clusters.

A centralized approach experiences several problems. First, as multiple sensor devices are deployed, overhead increases in a centralized system because of its low scalability. Second, constrained resources of sensor devices cause implementation difficulties. And also the clustering system requires a specific deployment strategy and maintenance which generates overhead. In this paper, we propose a system that utilizes

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 517 – 526, 2006.

a distributed algorithm to improve scalability. The algorithm is simple to use with resource constrained devices. Due to the constrained device and the uncertainty of acoustic signal propagation, the algorithm is based on a range-free mechanism using *VotingGrid* to provide robustness. The system requires simple sensor devices rather than expensive, stand-alone PCs. The system has been implemented with an off-the-shelf MicaZ [13] platform.

2 Related Work

Research was conducted on centralized systems and 2-tier clustering systems. In research conducted by Li and Hu [1] and Li et al. [2], the system studied, based on an energy attenuation model, utilized centralized computation. The model was too complex to be implemented on tiny motes. The work of Sheng and Hu [3] was based on the ML estimation [14] technique and the system was implemented in a real environment in which the powerful sensors were deployed linearly along a road. Chen et al. [4] implemented the algorithm on the iPAQ because it was too complex to be implemented on the WSN devices. Hawkes and Nehorai [5] proposed using the general closed-form of weighted least-squares, but it did not focus on the implementation in WSN applications. Shirahama et al. [6] demonstrated that the algorithm was less complex than the ML estimation technique, but it is still too complex for implementation on WSN devices. Johnston et al. [7] proposed a less complex mechanism, but the system had centralized architecture and required nontrivial computation to obtain reasonable accuracy. In the research of Lédeczi et al. [8] and Simon et al. [9], the system was deployed in an urban terrain and demonstrated a high degree of accuracy. The system was deployed randomly with uniform devices because it had reduced constraint in deployment. Expensive signal processing devices were required to achieve the accuracy. The scalability was limited because it targets on the specific area based on a centralized system.

Generally, it is hard to implement the acoustic source localization system with inexpensive sensor devices because the algorithm requires non-trivial computations. Hence, 2-tier systems [10-12] based on clustering were proposed to meet the required computational resources, low cost, and scalability issues. Wang et al. [10] described a system having static cluster architecture composed of Mica and stand-alone PCs. The cluster heads were deployed uniformly and all sensor devices recognized the associated cluster head. The static clustering system experienced a problem in that the accuracy decreased when an acoustic source occurred between the clusters. Manual deployment was also necessary to classify each cluster. Chen et al. [11] demonstrated that nodes in the system did not need to recognize their cluster head, reducing the constraints on deployment of the system. The system still experienced fluctuation of accuracy in determining acoustic source location. The system had no localization algorithm, and was not implemented. Huet al. [12] focused on a vast area in Australia. The system was based on 2-tier architecture, which experienced cost and deployment problems especially in the very large target area. The system provided scalability to some degree,

but required expensive devices and manual deployment. Rabbat and Nowak [15] proposed a decentralized algorithm based on the distributed ML estimation technique using token ring architecture. However, the algorithm did not provide robustness when a connection between nodes was broken.

3 Distributed System for Acoustic Source Localization

3.1 System Overview

The proposed system dynamically establishes a *Group* which processes the localization algorithm when an event occurs. As the system has a distributed algorithm, the estimation of source location is performed in the configured nodes rather than in the base station. The proposed system is constructed by using identical sensors to overcome the disadvantages of the 2-tier system. The distributed system is more scalable than the centralized system; it has no constraints on deployment, no cluster management overhead, and no loss of accuracy at boundaries of the clusters.

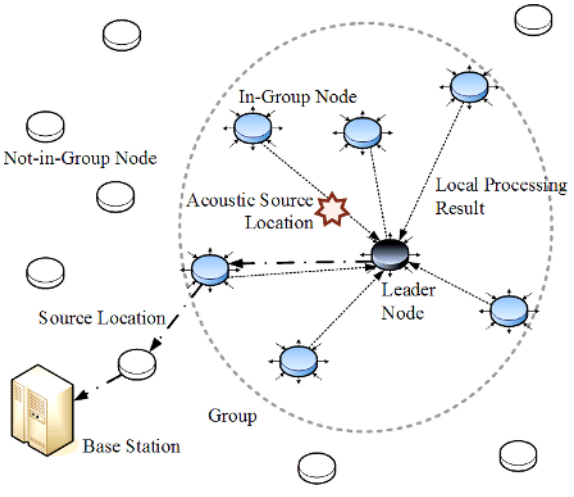


Fig. 1. Overview of the proposed system

Fig. 1 illustrates the structure of the proposed system. The *Group* is constructed with the nodes listening to the acoustic signal. In the *Group*, one leader, nearest to the source, is elected. The members of the *Group* exchange the data of the acoustic signal with each other, and they estimate the source location using data from other members. The leader node confirms the source location by gathering all interim results from the members. The leader processes no additional computations for estimation, but gathers the interim result and confirms the source location by a voting mechanism. The leader has no control over the members of the *Group*. All nodes in the *Group* operate

voluntarily. The confirmed acoustic source location is sent to the base station by a specific routing protocol. The base station can be composed of both stationary and moving devices.

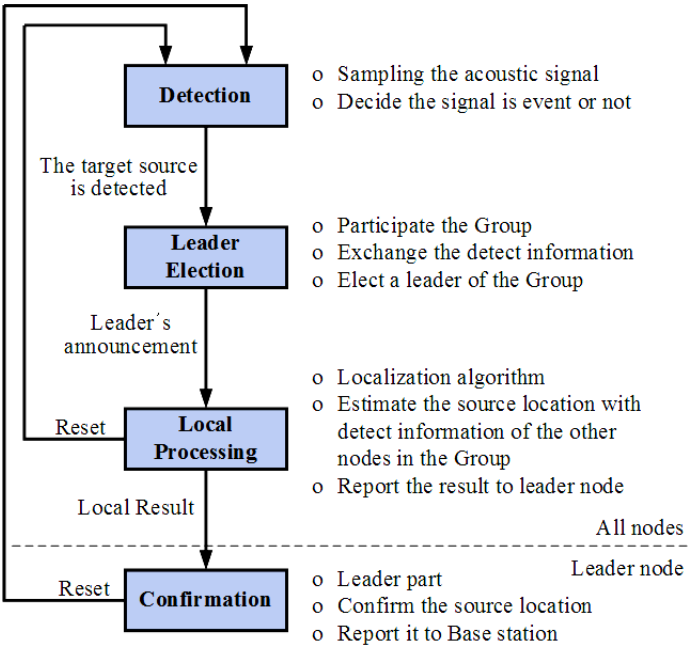


Fig. 2. Mechanism of the proposed system

3.2 System Mechanism

Fig. 2 shows the flow of the working mechanism of the proposed system. The system has four states and the transition between states is carried out by event. All listening nodes follow synchronously the same scenario. In the detection state, a node samples the acoustic signal at a certain rate. If the target acoustic signal is detected, the state is changed to the leader election state. In this state, the node recognizes that it has detected the target acoustic signal, and that it is involved in a certain *Group* originating from the acoustic event. Subsequently, the node exchanges the data of the acoustic signal with other nodes in the same *Group*. The exchange process selects the leader node of the *Group* using the leader election algorithm. The leader is then announced to the *Group*. The other nodes in the *Group* recognize that the leader election state is completed and change to the local processing state. In the local processing state, the nodes estimate the source position. When the estimation process finishes, the nodes have their interim result and report it to the leader node. Finally they revert back to the detection state. When the leader node completes the local processing state, it waits to receive the results of the other nodes. The status is changed into the confirmation state which integrates the results and confirms the acoustic source location. At the end of the confirmation

state, the leader node reports the estimate of the source location to the base station and returns to the detection state. All nodes, including the leader node, are initialized accordingly. The *Group* is dismantled when estimation is finished.

4 Source Localization Algorithm

4.1 Leader Election Algorithm

The goal of the leader election algorithm is to elect a leader node that is the node closest to the acoustic source. After detecting the source, the node acquires the detection time of the signal. Each node which has detected the signal assumes that it is the leader of the *Group*. The time of arrival is exchanged among the nodes in the *Group*, establishing a timer for competition. Each node compares the received arrival times with its own detection time. If the received time is smaller than the detection time of that node, the node relinquishes leadership. As nodes relinquish leader status to nodes having earlier detection times, a single leader node remained. The node located closest to the acoustic source has the earliest time of arrival of the signal, and therefore, is designated the leader. The time or period of competition is based on the maximum time difference between the nodes in a *Group*, and is computed as the maximum Euclidean distance between the nodes in the *Group*, that is, the radius of the *Group*. The radius can be varied by changing the characteristics of the acoustic signal, distinction in the mechanism of the acoustic source of interest, and the shape of the target area. Therefore, the time required for the period of competition is experimentally obtained.

4.2 Localization Algorithm

The range-based algorithm is a general technique which has high accuracy and complexity that requires powerful devices to sample and process the acoustic signal. Because WSN devices lack a high degree of computational power, the range-free mechanism is selected in our work.

Basically, each node estimates the source location based on the detection times of the members in the *Group* using the neighbor information table. The neighbor information table maintains the ID, position, and detection time of all nodes in the *Group*. In each iteration, the node selects a target node in its neighbor information table. After the target node is chosen, the node compares the detection times. A *partition line* is defined which is perpendicular to the midpoint of a line between two nodes. The acoustic source must be located in one of the two areas which are divided by the *partition line*. The detection time of the acoustic signal determines the area which includes the source location. The node marks the expected area of containing the acoustic source and selects next target node iteratively. An estimate of the source location is acquired by determining the area receiving the most votes which results from integrating all estimates of each node.

As the algorithm considers only the result of comparisons, the proposed system is able to operate with inaccurate detection times. Therefore, the system is implemented easily on WSN devices having low capability of sampling. Since each node is able to generate an own estimate, the system is tolerant to failure of a single node. The

integration continues processing without the data of failed node with no effect on data from other nodes. This algorithm is simple, but estimation based on voting reduces the effect of errors.

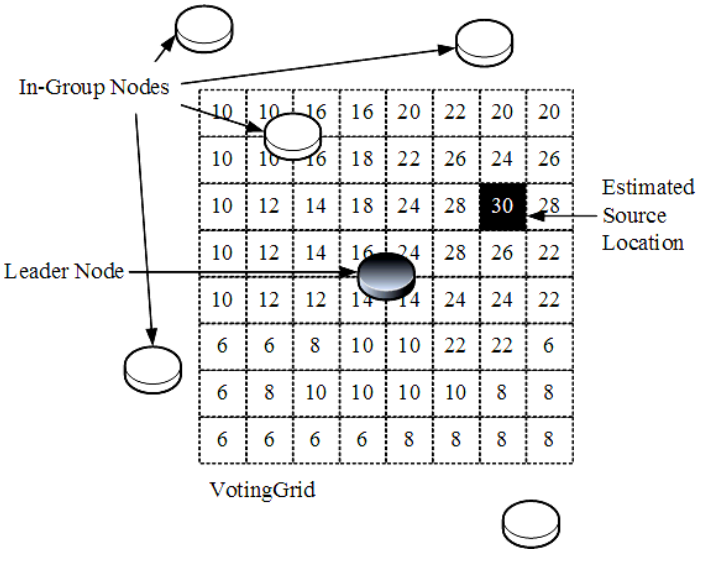


Fig. 3. Example of the 8X8 *VotingGrid*

4.3 VotingGrid

The *VotingGrid*, a virtual and geographical array for estimation, is an important component of the proposed mechanism. Fig. 3 shows the construction of the *VotingGrid* and an example of estimating the source position. The *VotingGrid* is centered on the leader node and the number of each cell indicates the ballot box for the source position. The area of the *VotingGrid* is determined either statically or dynamically. The *VotingGrid* includes all points in the field which are closest to the leader node. The grid size is variable for the refinement of source localization. When the leader announces its election, the area and the size of the *VotingGrid* are included in the message. Therefore, all nodes are informed of the identical *VotingGrid* of the target field. After the leader is announced, the node votes for each cell of the *VotingGrid* iteratively using the localization algorithm described in Section 4.2.

5 Experiments

A number of MicaZ [13] and MDA310CA sensor boards [16] are used for the experiments. Because the proposed algorithm determines the voting by the time of arrival, the system is required to maintain global time. The Flooding Time Synchronization Protocol (FTSP) [17] is used for time synchronization for all nodes in

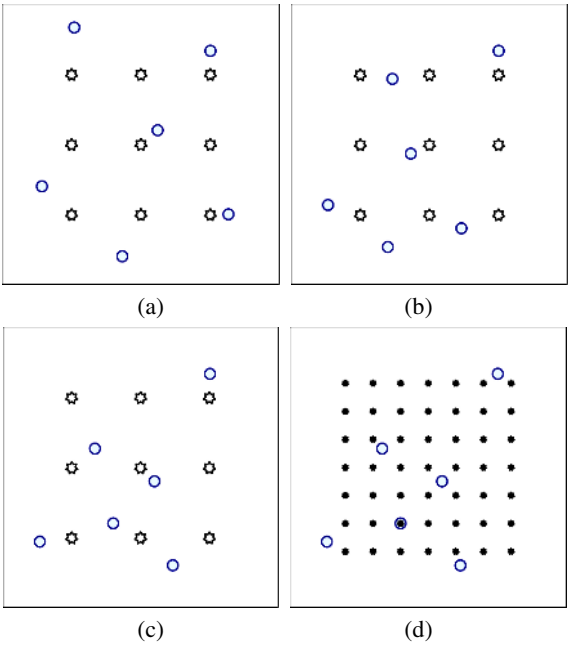


Fig. 4. Experimental configuration

the field. The current implementation of the system decides the existence of the event by the amplitude of the acoustic signal and the event detection method can be substituted by another known one. For sampling, a timer is implemented having microsecond resolution. The timer achieves high sampling rates by accessing to MCU (Atmega128L in MicaZ) directly rather than using millisecond timer provided by TinyOS. In the experiment, the sampling rate is 2.5KHz which is a lower sampling rate than used in traditional MAs. The size of the *VotingGrid* is 8x8. The experiments have been conducted in the lobby of our building in the presence of natural noise.

Fig. 4 illustrates three topologies with 6 nodes. The field size is 6m by 6m, and all nodes shown as circles detect the acoustic event. The nine source positions shown as stars are deployed evenly at 1.5 meters. Twenty experiments are conducted at each point of source position and the error is measured by the distance between the real source position and the worst case estimation in each experiment. Fig. 4a shows the topology having the nodes distributed evenly. Fig. 4b shows the topology in which the nodes are deployed slightly into the southeastern direction. Fig. 4c shows the topology having biased distribution. Fig. 4d has the same node topology as in Fig. 4c, but has forty-nine source positions in order to analyze the accuracy of whole field in detail.

Fig. 5 shows the results of the experiments. The XY-plane represents the geographical coordinates of the field. The Z-axis stands for the maximum error of the estimation. In Fig. 5a, the error decreases as the source moves to the center of the field. Accurate detection in the center of the nodes is a typical result of the acoustic source localization system. In Fig. 5b, the overall tendency is similar to that observed in Fig. 5a. The topology is different between Fig. 5a and Fig. 5b, and the center of the nodes is

moved to the southeast of the field. In 5c, the error is smallest in the center of the field, not in the center of the nodes. Fig. 5b and 5c show the characteristics of the proposed algorithm. The proposed algorithm generates the *partition line* which divides the area of the field. The accuracy is improved when the acoustic source is located in the smaller piece of the field. The partition line is determined by the topology of the nodes. In random topology, the possibility exists for achieving a high degree of accuracy in the center of the field. The accuracy in the low density region of the nodes is low, but the position of the best result is not in the highest density of the nodes. The system achieves .good accuracy in the middle of the *Group* because all nodes in the experiment detect the acoustic source. Considering that the *Group* is constructed by the leader node which is the closest to source in the system, the high degree of accuracy determines in the middle of the *Group* is reasonable.

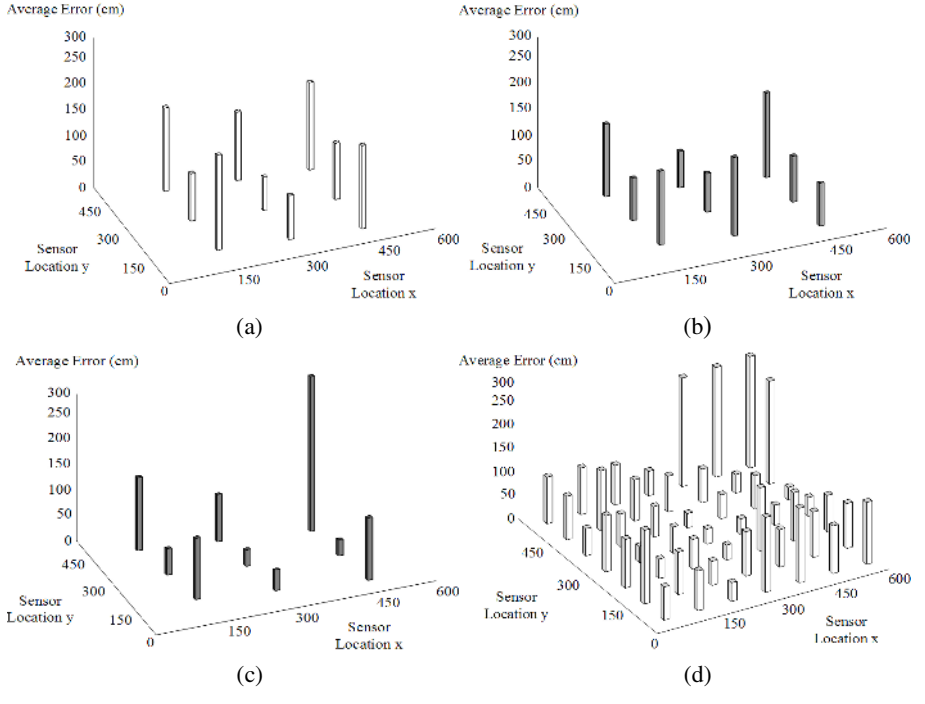


Fig. 5. Experimental results

Table 1 shows the average error of the entire field. The accuracy is similar as the topology varies. The average accuracy does not look outstanding. However, the basis for measuring error used in this study is different from that used in the range-based system. Because this research is based on range-free estimation, a new measuring mechanism is used to analyze the proposed system.

Fig. 6 shows the error measurement mechanism of the proposed system. The cells receiving the most votes are colored in grey. If the real source is assumed to be located at the position of the star, the worst case detection of possible estimate is measured.

Table 1. Average error for all points of the field

Fig.	5a	5b	5c	5d
Average error of the entire field	1.27 m	1.13m	1.03m	1.14m

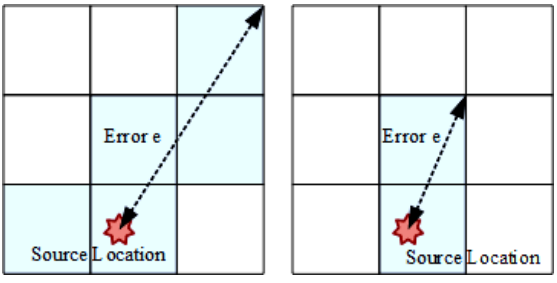


Fig. 6. Error Measurement

This measurement mechanism was used to analyze the characteristics of our system. If the source location was pinpointed by averaging the coordinates of the cells receiving the most votes, an error less than 0.6 meters is achieved which improves accuracy regarding the number of nodes per field area.

6 Conclusions

Since existing research on localization focused not on accuracy but on scalability, previous studies could not easily be applied to the real environment. In this paper, we propose a distributed acoustic source localization system for WSNs. Good scalability is supported by the localized *Group* concept. Therefore, the mechanism can be applied to vast areas where it is difficult to use the centralized system due to its low scalability. The mechanism also supports the application of a light-weight algorithm based on the range-free mechanism suitable for use in resource-constrained devices. For low sampling capability of WSN devices, the algorithm performs well with erroneous data estimated using the voting algorithm, *VotingGrid*. The system is composed of single tier architecture to relax the constraints of deployment and overcome the problem inherent in systems having 2-tier architecture. Using event-driven clustering technique, the system has no degradation of performance. The system is implemented with MicaZ and MDA310CA which contained no extra signal processing hardware. The system demonstrated an average accuracy of 60cm, which is better than existing systems based on similar hardware, in regard to the number of nodes per area.

Further experiments in large networks covering vast areas are planned. Because the size and area of the *VotingGrid* is related to the maximum accuracy of the system, future work will focus on policies for determining the *VotingGrid* more efficiently and accurately. The system will include the data compression mechanism to reduce the cost for communicating with the *VotingGrid*.

Acknowledgements

This work was supported by the National Research Laboratory (NRL) program of the Korean Science and Engineering Foundation (2005-01352) and the ITRC Program (MMRC) of IITA, Korea.

References

1. Li D., Hu Y. H.: Energy-Based Collaborative Source Localization Using Acoustic Microsensor Array. Journal of EURASIP on Applied Signal Processing. vol. 4. (2003) 321-337.
2. Li D., Wong K. D., Hu Y. H., Sayeed A. M.: Detection, Classification, and Tracking of Targets. IEEE Signal Processing Magazine. vol. 19 (2002) 17–29
3. Sheng X., Hu Y. H.: Maximum Likelihood Wireless Sensor Network Source Localization Using Acoustic Signal Energy Measurements. IEEE Transaction on Signal Processing. vol. 53, no. 1 (2005)
4. Chen, et al.: Coherent Acoustic Array Processing and Localization on Wireless Sensor Networks. Proceedings of IEEE. vol. 91, no. 8 (2003)
5. Hawkes M., Nehorai A.: Wideband Source Localization using a Distributed Acoustic Vector-Sensor Array. IEEE Transaction Signal Processing. vol. 51. (2003) 1479-1491
6. Shirahama J., Ohtsuki T., Kaneko T.: Low Complexity Source Localization Algorithms in Sensor Networks. Proceedings of the 2nd ACM International Workshop on Performance Evaluation of Wireless Ad hoc, Sensor, and Ubiquitous Networks. (2005)
7. Johnston B., Yin X., Valenzuela A., Frantz P.: A Fast Algorithm and Testbed Evaluation for Sound Source Localization Using Sensor Networks. Proceedings of IEEE Vehicular Technology Conference. (2005)
8. Lédeczi A., Volgyesi P., Maroti M., et al.: Multiple Simultaneous Acoustic Source Localization in Urban Terrain. Proceedings of the 4th Workshop on Information Processing in Sensor Networks (2005)
9. Simon, G., et al.: Sensor Network-based Countersniper System. Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems (2004)
10. Wang Q., Chen W. P., Zheng R., Lee K., Sha L.: Acoustic Target Tracking Using Tiny Wireless Sensor Devices. Proceedings of the 2nd Workshop on Information Processing in Sensor Networks (2003)
11. Chen W. P., Hou C. J., Sha L.: Dynamic clustering for acoustic target tracking in wireless sensor networks. Proceedings of IEEE International Conference on Network Protocols (2003)
12. Hu W., Tran V. N., Bulusu N., Chou C. T., Jha S., Taylor A.: The Design and Evaluation of a Hybrid Sensor Network For Cane-toad Monitoring. Proceedings of the 4th Workshop on Information Processing in Sensor Networks (2005)
13. http://www.xbow.com/Products/productsdetails.aspx?sid=101
14. Fisher R. A.: On the mathematical foundations of theoretical statistics. Philos. Trans. Roy. Soc. London Ser. A, (1922)
15. Rabbat M. G., Nowak R. D.: Decentralized Source Localization and Tracking. Proceedings of the 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing (2004) 921-924.
16. http://www.xbow.com/Products/productsdetails.aspx?sid=75
17. Maróti M., Kusy B.,Simon G., Ledeczi A.: The Flooding Time Synchronization Protocol. Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems (2004)

REDRP: Reactive Energy Decisive Routing Protocol for Wireless Sensor Networks

Ying-Hong Wang, Yi-Chien Lin, Ping-Fang Fu, and Chih-Hsiao Tsai

Department of Computer Science and Information Engineering TamKang University
Tamsui, Taipei, Taiwan, R.O.C.
inhon@mail.tku.edu.tw, 693191305@s93.tku.edu.tw,
694190074@s94.tku.edu.tw, 890190092@s90.tku.edu.tw

Abstract. The wireless sensor network (WSNs) was extensively deployed and researched for many applications in recently. By taking the advantage of smaller dimension, lower cost and simple structure of the sensor node, the more restrictions come together with sensors. Therefore, the limited power of sensor nodes is the most direct and difficult problem we meet. The limitation on the energy of sensor node makes the bottlenecks for designing the suitable routing protocols. In order to solve the problem of limited energy, the loading of nodes have to be distributed as possible as it can. If the energy consumption can be shared averagely by most nodes, then the lifetime of sensor networks will be enlarged. Thus we propose the routing protocol called Reactive Energy Decision Routing Protocol (REDRP) for sensor networking by considering several representative routing protocols in different structures. This protocol will create the routes in reactive routing method to transmit the data node gathered and the REDRP use the residual energy of nodes as the routing decision for energy-aware. As the results of simulation shows that the more fairness usage of sensor nodes, the total energy consumption of entire network will be distributed fairly by our protocol and the lifetime will also be increased.

1 Introduction

In recently years, the researches of the field on the wireless sensor network had been developed and carried out with time. As the result of the progress in microelectronic and wireless communication, the sensor node will have smaller volume with sufficient ability compared with the traditional mobile node of the Ad-hoc wireless networks in specifically applications. The sensor nodes which adopt the wireless module and the smaller hardware components have not to be limited to the fixed topologies and the constant space. Since the smaller sensor node has these characteristics, it makes the sensor network become more suitable to be applied in many dimensions, such as the monitoring on battlefield in real-time or the detection of the ecological environment which is difficult to be reached by human being and so on. These applications will be simply accomplished by the sensor network.

With the mentioned advantages of the sensor node before, the sensor network also have more restrictions in the other side. The simpler computing ability, fewer capacity of memory and the limited power supply of the sensor nodes are the further

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 527 – 535, 2006.

restrictions come with the reduced size. Generally, the construction of the larger scope in the sensor network may demand hundreds and even thousands of sensor nodes. Because the sensor nodes also have the limitations of confined sensing range and the communication distance, the information which is gathered by sensor node have to be processed by some particular device instead of itself such like fixed Sink or Base Station. So the data has to be forwarded to the sink one by one of sensor nodes to handle the transmissions. To build the routing path of data may be achieved by many nodes to satisfy the long distance communication. Therefore, to use the suitable routing mechanism will directly influence the efficiency and the lifetime of entire sensor network.

In terms of the problems in the sensor network, the immediate difficulty we will face is the energy consumption of the sensor network. So we propose a new routing protocol with energy-aware to called Reactive Energy Decisive Routing Protocol (REDRP) applied in sensor network. The routing protocol is expected to improve the lifetime and distribute the usage of sensor nodes fairly. We classify this paper into several sections as follows. In section 2, we will discuss some existing routing protocols with their properties, and then the detailed routing procedures of REDRP will be described in section 3. Section 4 shows the results of our simulation, the conclusion and the future works will be presented in section 5 at last of this article.

2 Related Works

With more and more routing protocols proposed for sensor network, many different network structures were produced. Mainly of these routing protocols can be classified into three categories of the direct-communication, flat and clustering routing, and the characteristics of proactive, reactive and hybrid are also the routing types depending on how the source finds a route to the destination [1]. In this section, we introduce some general routing protocol in sensor networks as follows and also discuss these routing protocols with their properties.

The Directed Diffusion [2] is the well-known routing protocol of sensor network, without complicated routing procedures, the protocol transmits information of sensor nodes by diffusion directly. This scheme guaranteed the high delivery rate of data and kept low latency of transmission, but the drawback was also obvious to waste the plenty power dissipation by transmitting and receiving packets redundantly.

The author Heinzelman proposed the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol for sensor network [3], this protocol is also the very common used in sensor network. The LEACH has the hierarchical structure and clustering routing method, LEACH partitions the sensor nodes into many different clusters, each cluster have a head node to collect the data of the other nodes in the same cluster and sending the collected data to the other cluster head or sink, but the nodes which is not the head between different clusters could not communicate with each other. The head node will be chosen randomly in the cluster, it causes the more energy consumption of the head nodes. By taking advantages on the data aggregation and chosen the head nodes randomly, LEACH would disperse the energy dissipation of nodes to enhance the lifetime in sensor network, but the longer communications between the head nodes or sink may increase the energy depletion by comparison.

The Power-Efficient Gathering in Sensor Information Systems (PEGASIS) is the familiar reactive routing protocol for sensor network [4]. Similar to the LEACH, the PEGASIS have a leader node to transmit the data to the sink. But the sensor nodes would not report data in periods by PEGASIS. When there's any detected event, the route will be created as a chain by Greedy algorithm, there's one node will be selected as the leader node to gather the data which is transmitted by other nodes along the chain. This reactive model can reduce the energy depletion by switching off the leader node and the sensor nodes in turns and it can also decrease the frequency of the communications of sensor nodes. But the PEGASIS needs the location-aware device to create the chain for data transmissions, and that will also cost the more overheads of the node.

The Rumor routing algorithm is proposed and this hybrid routing protocol use the agent packet with longer TTL to maintain the routes in the network [5]. The agent packet could be randomly generated by sensor nodes with the record of routing paths on the sensor node, the other sensor nodes will also update their routing tables by receiving the agent packet for routing on better paths. With the generation and the traveling of the agent packets, to sustain the routing paths becomes easier, but the records of agent packets would become larger because of going through the more nodes, that will cause the much energy expense.

Because of the location-based routing protocols such like TTDD [6] required the information of position of every node, so the energy would be also consumed by the additional device, so we abandon the idea of location-based protocols. By estimating the protocols above, we proposed the Reactive Energy Decisive Routing Protocol (REDRP) with energy-aware to apply the sensor network. Within the reactive routing model, the more efficiency and simpler routing procedures will be achieved in our protocol and it is also more suitable and practical in applications of the sensor network.

3 REDRP: Reactive Energy Decisive Routing Protocol

Most of the well-known energy-aware routing protocols take the communication distance with energy information for the routing decision, and these mechanisms with the pre-position hardware such likes GPS will also cause the additional cost and energy overhead. Without adopting the pre-position hardware, a large number of nodes can save the significant cost and the energy. Because the time of receiving message is most direct affection of the communication distance and the link quality between nodes, so we consider the mechanism of timestamp with the energy information as one of the factors.

Before addressing the detail of our REDRP in this section, there are some assumptions and requirements have to be described. Every sensor node is fixed in the network and all of the nodes will have some appending capability of energy monitoring in real-time for the decisions of finding routes. We also demand that every node has a unique node id, and every node will record the remaining energy with the node's id of the neighboring nodes. The node will note the path id, the previous node id and the node id of next hop; the recorded information is used to transmit the data or adjust the routing paths. One or more fixed sink without energy restriction can be

applied in the sensor network. Considering these assumptions above, the routing procedures of REDRP are divided into the following steps: Initiation step, Routes Discovery step, Data Transmission step and Routes Adjustment step.

3.1 Initiation Step

In the initiation, the sink will broadcast a timer packet to entire network. When the nodes received this packet, the current time will be recorded as the Dist value. The Dist value can simply presents the reference position to the sink with the other nodes, the smaller Dist value can be noted that the node is near the sink more. After setting the Dist value, the sensor nodes will turn into the sensing mode. In sensing mode, each node can use the minimal energy consumption to sensing the event or receiving the packets. The sensor node will change this status only if some information is sensed or particular packets are received for establishing the routing path or the other manipulations. When the operations finished, the sensor node will be turned into sensing mode immediately. As well as the concept of the other reactive protocols, the routing path will be discovered on demand and that will reduce a great amount of the energy depletion by using the simplest function of sensing in sensor network.

3.2 Routes Discovery Step

Referring to the other proactive routing protocols, our REDRP create the routes reactively by sensing data. While the sensor node without transmitting data detects the occurrence of some events, the node will send a special packet to its neighboring nodes for finding a routing path. The special packet called RP_Request packet is used for requesting a route to deliver the data to sink. The RP_Request packet contains the fields of information as Fig. 1.

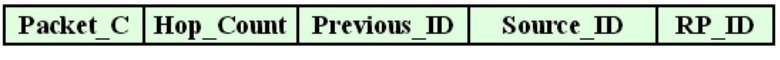


Fig. 1. The definition of RP_Request packet

In the RP_Request packet, the Packet_C field can be used to identify the packet is the RP_Request packet. The Hop_Count value will be increased by one for once transmission from sender to receiver. If any node receives a RP_Request packet, the Hop_Count value of packet greater than total number of nodes, that mean the RP_Request packet has traveled through the whole network and the packet has became invalid. Thus, the packet will not be send anymore. The the Hop_Count field are used to detect the extra energy consumption from the infinite loops. The Previous_ID, Source_ID and RP_ID present the previous node, source node and the routes id of the node on this routing path, the neighboring nodes can get the routing information by the fields. The RP_ID will be set in empty value as startup.

By receiving the RP_Request packet, the receiver will check the Hop_count first to avoid the infinite looping. The nodes which received the RP_Request packet without data transmission will send the special packet back to the sender, which is called RP_Reply packet containing the fields of Packet_C, Node_ID, Dist and R_Energy. The field of Packet_C is also used to identify the packet is the RP_Reply packet. The

Node_ID field indicated which node sends the RP_Reply packet back, and the R_Energy and Dist fields noted the residual energy and the Dist value of the node.

Every received RP_Reply packet will be recorded in the table with a timestamp T_i given by the node. The table is built of the R_Energy_i, T_i and the $Dist_i$ values of node i. We assumed a formula of the node i:

$$P_i = R\_Energy_i / T_i^2 \tag{1}$$

The sender will decide the next hopping node according to the P_i and $Dist_i$, the node will be chosen as the next node by the smaller $Dist_i$ and the higher P_i. The smaller $Dist_i$ can make sure the direction of finding the routing path will go to the sink, and the higher P_i means the next hopping node has higher residual energy or the shorter distance. The node which is chosen as the next hop will resend the RP_Request packet to find the next node of the routing path.

By repeating the sending and receiving of the RP_Request and RP_Reply packets, the routing path to sink can be built. The sink will reply the Confirm packet with the RP_ID in reverse to verify the created route finally. Moreover, the nodes on the routes also have the energy information of neighboring nodes and RP_ID of the route.

3.3 Data Transmission Step

The routes will be established after receiving the confirm packet and every node is also going to be assigned a routes ID by RP_ID of the confirm packet. The unique RP_ID is generated by sink node to identify the different routing paths. After the

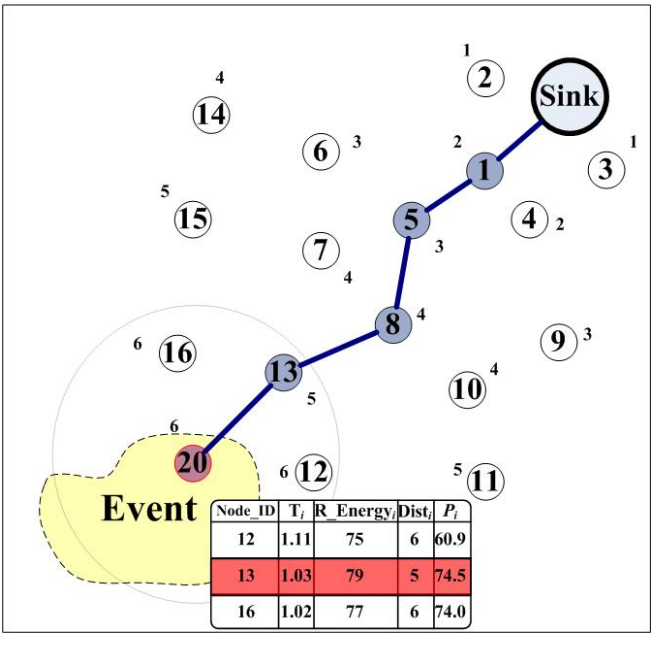


Fig. 2. The network presentation after the event occurring

creation of the routes in Routes Discovery step, the gathering data will be transmitted to the sink along the routes. For example as Fig. 2, the node 1, 5, 8, 13, 20 were assigned to the nodes of the route 1, the information of neighboring nodes is also recorded to adjust the routes if there's any event detected by the transmitting nodes. However, to balance the usage of the routes, the nodes on transmission can not to be assigned the other routes until the finishing of transmission. After the data transmission finished, the sensor nodes of the route would turn in sensing mode and these nodes can be assigned to other routes after.

3.4 Routes Adjustment Step

Since the energy would be depleted by the sensing or the data transmission, and the nodes on the route would also be the source node when sensing events. These situations may break the routes that were created before, so the routes adjustment is necessary to be startup for maintaining availability of the routing paths.

While the nodes on the using route sensed the data, the gathering information and received data will be saved temporarily until the routes adjustment finished. The node will send a RP_Adj packet with the Node_ID to the previous node on the old route. When the previous node receives the packet, it will remove the node from table first and the previous node will take the second advisable node of the table, and it will also notify the new hopping node to the old. At the same time, the old next node will send the temporal receiving data to the new node, and the new next node will discover a new routing path to the sink by receiving the data. After pass the received data to the

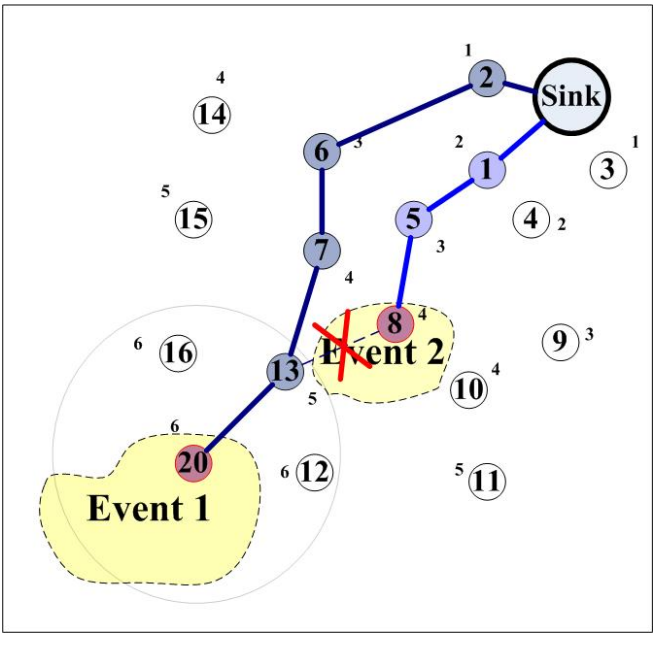


Fig. 3. The network structure after routes adjustment

new hopping node, the node will take the old routing path to transmit the gathering data to the sink. The old routing path and the new discovered routing path will be the better two with higher residual energy, shorter distance or more efficient connection. The fairness is also satisfied by choosing the routing path in the routes adjustment step.

In general, the routes adjustment will be split into routes discovery and data transmission, it may also spread the direction of routing path to make the energy consumption more fairly and the procedures will also be simplified without heavy loading of finding two new routing paths as Fig. 3.

4 Simulation Results

Before discussing the performance of our model, we would like to describe the environment of our simulation at first and then we analyze the simulation results compared with other protocols.

4.1 Environment Setup

To evaluate the performance of our model, we simulate the REDRP to compare with the other protocols in average residual energy measurement and the number of nodes alive for lifetime. We achieve the simulation of network environment by C++ language and compare our REDRP with related protocols of LEACH [3], PEGASIS [4] and the Rumer Routing Algorithm [5]. We consider a network of 1000 fixed sensor nodes allocated on 150m × 150m range. Each node has the restricted ability on sensing and transmission range of 5m. There is one fixed sink node with infinite power supply in the network. There are 500 events generated randomly and the transmitted data packet is 50 bytes.

4.2 Results Analyze

We frame the power consumption model of sensor network by two main formulas according to the wireless radio model and adopt the main operation parameters in wireless sensor network to calculate the performance [7, 8]. The power dissipation of k bits at a transmitter and receiver can be modeled as the following formulas:

$$p_t(n_i, n_k) = (\alpha_{t1} + \alpha_{t2} d^{n}_{ni,nk}) \times k \tag{2}$$

$$p_r = \alpha_r \times k \tag{3}$$

$p_t(n_i, n_k)$ denotes the power dissipation of the sender n_i to transmit to the receiver n_k and theα_{t1} andα_{t2} are the energy consumption coefficients. $d^{n}_{ni,nk}$ is the energy dissipation rate by distance of n_i and n_k, the energy depletion of receiver n_k is formulated as p_r and theα_r indicates the energy consumption for receiving per bits. We use the both formulas to compute the situation of power exhaustion in our simulation.

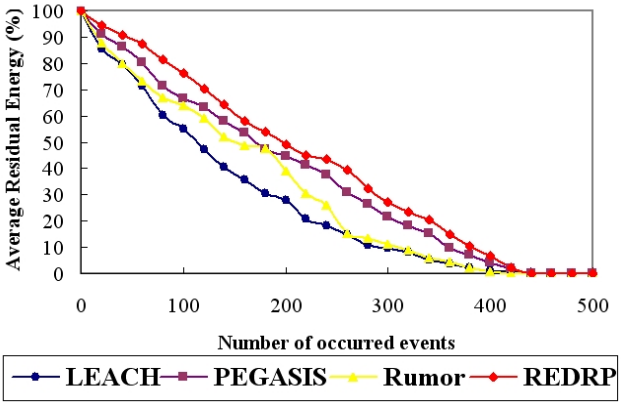


Fig. 4. The average residual energy of REDRP compared with the other protocols

As Fig. 4 shows, we estimated the REDRP in our simulation and the result presented the much higher performance compared with the others; LEACH with the proactive routing has to change the cluster head node periodically for maintaining the clustered network structure, and the longer communication of the cluster head nodes also made more additional dissipation of energy. The Rumer Routing Algorithm would also deplete the extra energy cost on traveling of more and more agent packets. Besides, the longer TTL of the agent packets also cost a great amount of energy, and the randomly generated agent packets also caused the unstable energy consumption of entire network. To consider the performance of PEGASIS, both REDRP and PEGASIS used the reactive routing model, but the REDRP has better efficiency without location-aware device, and the more energy would be expensed for create the chain of all the nodes on PEGASIS.

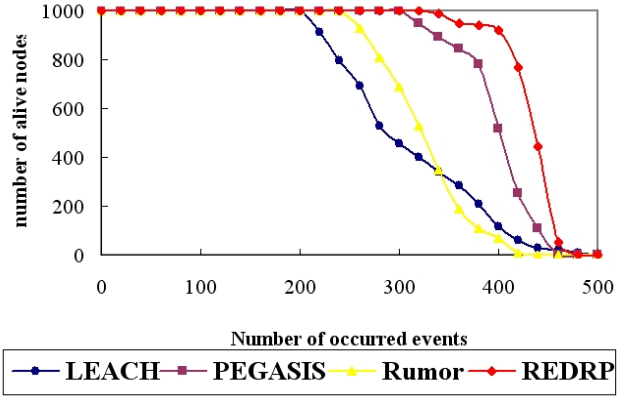


Fig. 5. The number of nodes alive in REDRP compared with the other protocols

The Fig. 5 presents the number of nodes alive that could exhibit the lifetime of the network and the more nodes alive denotes the longer lifetime. The result of simulation shows that the REDRP has more nodes alive than others in our simulation. The simple and reactive manner with energy-aware of our protocol disperse the nodes' utility efficiently and fairly.

5 Conclusions and Future Work

Since the most notable problem in sensor network is the energy consumption and we consider that the depletion of energy should be solved by sharing the operations with sensor nodes fairly. By this notion, we proposed a novel routing protocol: Reactive Energy-aware Decisive Routing Protocol (REDRP) for sensor network in this paper. Our simulation result shows the better accomplishment of REDRP by comparing with other protocols. We respect this simple and reactive routing protocol with energy-aware mechanism to improve the average utility rate of sensor nodes efficiently and increase the lifetime of the entire sensor network. With more deeply consultation, the data aggregation may also be considered to enhance the performance of data transmission in the protocol. The delay of data transmissions and the synchronization problem of packets will be the concerned in the future.

References

1. Qiangfeng, Jiang., Manivannan, D.: Routing protocols for sensor networks. In: 1st IEEE Consumer Communications and Networking Conference, 2004. (2004) 93–98
2. Intanagonwiwat, C., Govindan, R., Estrin,D.: Directed Diffusion: A scalable and robust communication paradigm for sensor networks. In: Proceedings of the 6th Annual ACM/IEEE International Conference Mobile Computing and Networking, 2000. (2000) 56–67
3. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33rd Annual Hawaii International Conference System Sciences, 2000. (2000) 3005–3014
4. Lindsey, S., Raghavendra, C.S.: PEGASIS: Power-Efficient Gathering in Sensor Information Systems. In: Proceedings of the Aerospace Conference. (2002) Vol.3, 1125–1130
5. David, Braginsky., Deborah, Estrin.: Rumor Routing Algorithm For Sensor Networks. In: Proceedings of the 1st ACM international Conference workshop on Wireless sensor networks and applications on Mobile Computing and Networking. (2002) 22–31
6. Fan, Ye., Haiyun, Luo., Jerry, Cheng., Songwu, Lu., Lixia, Zhang.: A Two-tier Data Dissemination Model for Large-scale Wireless Sensor Networks. In: Proceedings of the 8th ACM Annual International Conference on Mobile Computing and Networking. (2002) 148–159
7. Hyun-sook, Kim., Ki-jun, Han.: A Power Efficient Routing Protocol Based on Balanced Tree in Wireless Sensor Networks. In: Proceedings of the 1st International Conference on Distributed Frameworks for Multimedia Applications. (2005) 138–143
8. Bhardwaj, M., Chandrakasan, A.P.: Bounding the lifetime of sensor networks via optimal role assignments. In: Proceedings of INFOCOM 2002. (2002) 1587–1596

Systolic Query Processing for Aggregation in Sensor Networks*

Suraj Pandey, Ho Seok Kim, Sang Hun Eo, and Hae Young Bae

Department of Computer Science and Information Engineering, Inha University,
Yonghyun-dong, Nam-gu, Incheon, 402-751, Korea
{suraj, hskim, eosanghun}@dblab.inha.ac.kr, hybae@inha.ac.kr
http://dblab.inha.ac.kr

Abstract. Pipelining the messaging between sensor nodes increases the overall throughput of the querying system, however at the cost of extra communication. But for long running queries, the messages communicated in pipelined architecture are even less than the normal count of messages in any query processing methodology in sensor networks, as also pointed out in previous work. In this paper we device a novel methodology to process aggregation queries in sensor networks by using the systolic architecture. We explicitly define and stipulate the use of systolic message communication as aggregation query processing technique to yield increased response time with the saving of energy by reduced message communication when considering long running queries. We show through simulation the two-fold gain using the proposed technique as compared to methods without pipelining.

1 Introduction

The onset of technological development of low-cost sensors and networking devices have widened the horizon of stream applications and has further laid stones for cut-through research in sensor networks. A sensor network consists of many spatially distributed sensors, which are used to monitor or detect phenomena at different locations, such as temperature changes or pollutant level. Sensor nodes, such as the Berkeley MICA Mote [1] which already support temperature sensors, a magnetometer, an accelerometer, a microphone, and also several actuators, are getting smaller, cheaper, and able to perform more complex operations, including having mini embedded operating systems. These sensors; mobile and static, have the ability to gather massive amount of spatial as well as temporally dense data over vast geographical areas.

Real-time sensor data have a tendency of being highly compute and memory intensive if handled centrally. Speedups of many orders of magnitude over previous centralized system were found through improvements in new paradigms,

* This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 536–545, 2006.

e.g. in-network query processing and aggregation. A major gain also comes from parallel architectures, interconnected sensor nodes with algorithms using parallelism and pipelining at different levels. Parallel processing among the sensor nodes has to be addressed with more elaborate approach. H.T. Kung claims [7] a perfectly linear speedup for his systolic array architecture. Most recently, it is to be noted that the sensor network topology favors clusters of sensor nodes, with connections programmable as pipelined structures; this paradigm dominates other architectures in performance and economy, even in comparison to message passing, when it comes to long running queries. Clustering of the sensor nodes relaxes the problem of unreliability of sensor communication thus overlooking the inherent approximations and probabilistic errors.

Scanning sensor sequence data for aggregation and/or peculiarity is a common and often repeated task in sensor networks. The need for speeding up this treatment comes from the exponential growth in monitoring applications. Comparison algorithms whose complexities are quadratic with respect to the length of the data sequences detect similarities between query sequence and a subject sequence. For stipulated quality results in a short time is to use parallel processing. Systolic arrays have been proven as a good candidate structure. Systolic communication supports efficient, fine-grained parallelism. In this model, the source cell program sends data items to the destination cell as it generates them, and the destination cell can start processing them as soon as the first word of input has arrived. Systolic algorithms rely on the ability to transfer long streams of intermediate data between processes at high throughput and with low latency. More importantly, the communication cost must be consistently small, because cost variations can greatly increase delays in the overall computation. This implies dedicated communication paths are desirable, which may be neighboring or non-neighboring paths depending on communication topologies of the algorithm. Instruction systolic arrays (ISAs) have been developed in order to combine the speed and simplicity of systolic arrays with flexible programmability. Sensor networks and the systolic array have much in common. Each processing element (PE) of a systolic array can be thought of as a sensor node indulged into the parallel processing of the streaming data. Each sensor node belonging to a particular cluster formation, acts like an array element and processes the streaming data locally. The processing is done in parallel and the operations are pipelined.

In this paper, we lay out the foundation stone by considering throughput improvement using systolic concept. Beginning with parallel structures and then the clustering of the sensor nodes, we describe the query processing especially for aggregation operation within the sensor networks using our approach.

The remainder of this paper is organized as follows. In Section 2, we briefly review related work. In Section 3, we propose our approach to systolic query processing together with the ISA basics. Section 4 presents the performance evaluation based on the simulated environment. Finally, we conclude in Section 5 providing insights into future work.

2 Related Work

Novel research challenges in networking, databases and algorithms [6,5,8] have been propounding into the problems of decentralized routing, network maintenance and data aggregation in sensor networks. In terms of providing database queries over sensor networks, TinyDB[1][9] at UC Berkeley and Cougar [10] at Cornell University are the two major efforts. They provide algorithms for many interesting aggregates such as MAX, MIN, AVERAGE, SUM, COUNT. Zhao et.al. [11] also have suggested algorithms for constructing summaries like MAX, AVG. The focus of their work is however more on network monitoring and maintenance rather than database query. Przydatek et.al [12] have discussed secure ways to aggregate data but only with one aggregating node. Considine et. al. [13] have discussed how to compute COUNT, SUM, AVERAGE in a robust fashion in the presence of failures such as lost and duplicate packets.

In the previous work [9,14,15], grouping computes aggregates over partitions of sensor readings. The basic technique for grouping is to push down a set of predicates that specify the group membership, ask sensors to choose the group they belong to, and as answers flow back, update the aggregate values in the appropriate groups. In this paper the sensor nodes are grouped according to their regions into cluster formations, as our previous work on spatial query processing [16] addressed by proposing a routing tree. Ratnasamy et.all described a novel Geographic Hash Table (GHT) [17] system which hashes keys into geographical coordinates.

Madden et.al., in [5] proposed TAG, an aggregation service as a part of TinyDB [9] which is a query processing system for a network of Berkeley motes. It presents the in-network processing of the aggregation queries on the data generated in the sensor network. They propose the pipelined aggregation. In their pipelined approach, time is divided into intervals of duration i. The value of i can be quite small, about the time it takes for a single sensor to produce and transmit a sensor reading, versus the value of t in the simple multi-round solution proposed in TAG [5]. Their most significant drawback is that a number of additional messages are transmitted to extract the first aggregate over all sensors. However, they have clearly noted that, after the initial message overhead, each additional aggregate arrives at a lower cost even than the normal method, and at a rate of one update per time interval. Optimizations to reduce this overhead were also pointed out; reporting the aggregate values only when they change significantly to affect the overall value. Thus pipelining has been the primary motivating factor to propose the systolic method in processing the query in sensor networks with stress on aggregation queries.

3 Systolic Querying Processing

Systolic arrays are oneto three-dimensional arrays of simple, mostly identical processing elements, with nearest neighbor connection. They both compute and

[1] http://telegraph.cs.berkeley.edu/tinydb/

pass data rhythmically through the system (the word systole is used in physiology, describing the rhythmical pulses of blood through the heart). As with the issues connected with synchronization of a large array of processing units, the asynchronous data-driven wave-array processor is another option, which has the same structure as a systolic array, but without a global clock. Correct timing versus correct sequencing is at the heart of choosing the appropriate array architecture.

In [18] the ISA is suggested as a new architecture for parallel computation. In contrast to the conventional systolic arrays a sequence of instructions is pumped through a mesh-connected array of processing elements being capable of executing a few simple instructions and having a small local memory. We choose to adopt the ISA-concept as it leads to wider flexibility than the general systolic array concept for sensor networks.

The structure of instruction systolic array is being depicted in figure 1. The PEs are equipped with simple processing capabilities and storage space. Instructions are pumped through the columns of the array of PEs from top to bottom.[2] This matrix is called the top program (TP). In addition selector information (0 or 1) is pumped through the rows of the array from left to right. This matrix over 0,1 is called the left program (LP). A 0 causes every PE in its row to stay passive and a 1 causes every PE in its row to execute the instruction what has been provided to it.

The processing nodes of the systolic array have the exact characteristics with the sensor nodes. With even more functionalities like enhanced processing power and increased memory space, the sensor nodes can be substituted with the PE as generally associated while describing the systolic arrays. One major constraint which determines the feasibility issue in using systolic concept in sensor networks is the communication between these PE (sensor nodes). The need for managing the communication resource is more pronounced in the systolic communication model. This is because the production and consumption of a data stream are tied directly to the computation rates of the PE. As the computation on a PE can stall and even deadlock while waiting for data, the lifetime of a communication path can be arbitrarily long.

In an ISA, communication between PE is done in a simple and flexible way. Each PE can read information from its four[3] direct neighbors. Within each clock phase reading access is always performed before writing phase. To avoid read/write conflicts execution of instructions is done in two non overlapping phases. If a PE needs information from one of its four direct neighbors, it reads the neighbors communication register during the first phase of the execution of an instruction. Thus at most five (or eight) processors can read from a communication register simultaneously. During the second phase of the execution of an instruction every PE writes into its own communication register or an internal register. Aggregation over an entire row can be performed by the addition of the

[2] Conventional for ISA, but modified for Sensor Networks.

[3] The count of sensor nodes communicating depends on the topology of the network.

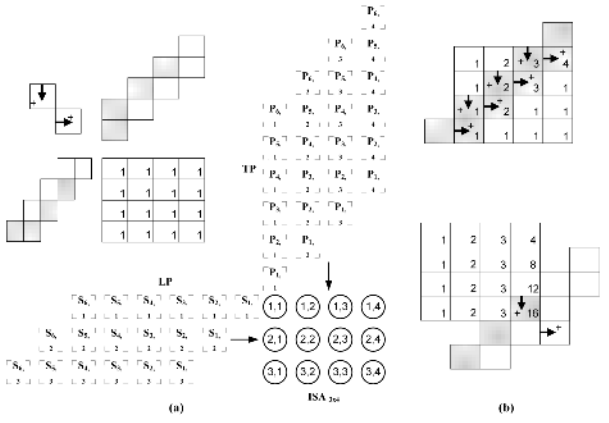


Fig. 1. Flow of data in an ISA. (a) TP and LP being fed to the PEs. (b) An example of data being aggregated within the cells in a pipelined fashion. Shaded areas signify the data and mask column and row wise respectively.

individual values together while forwarding. Both broadcast and aggregation can be achieved simultaneously by following this rule.

Aggregate functions on a processor array are associative and commutative functions to which every processor provides an argument value. As they are commutative and associative, aggregate functions can be evaluated in many different ways (orders). The ISA supports top-down column operations and left-right row operations, due to the systolic flow of the operations. Thus, an aggregate function can be implemented on an ISA by executing it firstly in columns, placing the corresponding results within the last processing element within each column, and secondly applying the aggregation to these results in the last row, executing in the order from left to right. The aggregate functions like SUM, MAX can be easily computed over the data values.

However, simply following this process among sensor nodes in a sensor network is energy demanding. Through a careful analysis and judgment it is evident that the common elements of an ISA can be compromised to be residing in a single PE such as a sensor node. As we can take the advantage of each sensor node in that it can store additional information, the communication registers concept borrowed from ISA can be subsided by the replacement of a single sensor node (or few additional local nodes within a cluster) that is responsible for providing the storage function.

As in TAG, following the stream semantics, each record consists of one {group id, aggregate value} pair per group. Each group is time stamped and the readings used to compute an aggregate record all belong to the same time interval, or epoch. The duration of each epoch is the argument of the EPOCH DURATION clause, which specifies the amount of time (in seconds) devices wait before acquiring and transmitting each successive sample [5]. We describe the aggregation operation in a greater depth in figure 2. The basic process is the same as

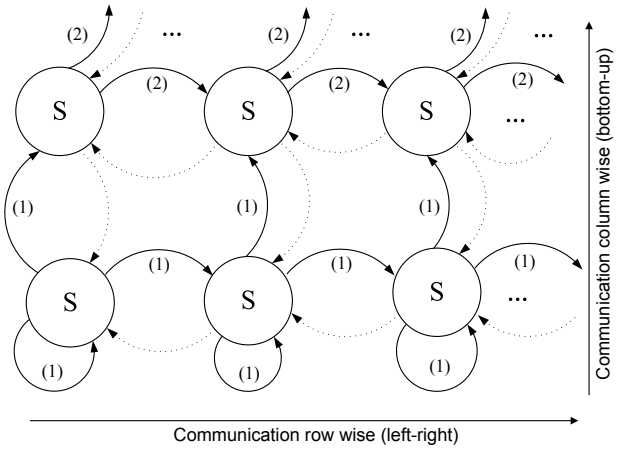


Fig. 2. Data flow network between sensor nodes. Numbers indicate the time sequence when the messages are transferred. Solid lines and dotted lines are for normal and asynchronous messages respectively.

given by the ISA structure. However, instead of forwarding the message one at a time, each sensor node layer forwards the message simultaneously. The sensor nodes within the MBR[4] forming a cluster, as maintained by our previous work, handle the message as the ISA does, described in the above paragraph. The TP and the LP when mapped in the context of sensor networks, is no more than the semantic synchronization between nodes by the help of time stamping. The nodes when dealing with the calculation of aggregate value dwell between aggregation, simply forwarding or suppressing, depending on the attributes of the message received from other node/s. TP is analogous to sensor data and LP is equivalent to the attributes (e.g. time stamping), as the processing is carried out by the sensor node itself. As depicted in figure 2, each node has two outgoing message arrows, which denote the forward message direction. There is the ambiguity when we explicitly denote the direction as left-right and top-bottom within a cluster. Each cluster has its orientation with respect to the cluster head, or the sensor nodes that is/are responsible to forward the aggregation to their parent cluster which leaves the naming of the direction as incorrect but we denote it in the figure for the convenience of the reader. The two solid directional lines have exactly the same associative meaning with this ambiguity. Each cluster is free to choose the direction of clustering, interchangeably between row and columns. However, only a single arrow is to be considered when the message is being forwarded within each cluster. Also shown are the dotted lines pointing to the nodes lower in the chain within a cluster, .i.e. farther away from the cluster head which is responsible to convey the aggregate to its parent node. We associate two semantics with this type of communication:

[4] Minimum Bounding Rectangle.

- The solid lines indicate the direction of the normal flow of messages from one node to another.
- The dotted lines are used for asynchronous messaging.

TAG clearly associates the approach with the risk involved in losing the ability to incorporate nodes that failed to hear the initial request to aggregate when we limit the communication by not forwarding any message that doesnt change the aggregate value. It is here that these asynchronous messaging is used, so that the network is prevented from drastic collapse or forwarding feebly constructed values over to the parent nodes. The asynchronous message is such that it is introduced in the network from few nodes at random within the cluster to neighboring nodes that havent reported their values. The message would keep the network alive for future querying. The nodes receiving the acknowledgements would simply report back to their parents within any time interval before the parents consider them dead. The looping arrow in each sensor node within the boundary is shown for generality when we consider the pipelined structure, showing that its the leaf node.

4 Performance Evaluation

We assume stationary sensor nodes which have a common maximum radio range r and are equipped with omni-directional antennas. The buffers at the sensors are assumed to be of infinite capacity, reasonably valid (hence no losses in the network) and are modeled as FIFO queues. The information sensed by the sources is organized into data units of fixed size, and sent in fixed time slots. The sequential model could also have been used to overcome the problem of synchronization, but we chose to use the time slot for our experiments. For aggregators, we assume that such a node knows the number of its linking nodes, generally specified as children. We assume that the aggregator node, for each cluster, aggregates all the data it receives into one packet which it then forwards to its parent along the aggregation tree toward the sink. Routing is performed by following the aggregation tree whose leaves are normal nodes, as suggested by our previous work on spatial querying in sensor networks [16]. Aggregators constitute the internal nodes of the tree, which are localized in a cluster, and the sink is its root. By following the MBR routing, every node knows prior which node it has to route to, i.e. each node knows (and is within communication range of) its parent along the aggregation tree.

We evaluate the system based on simulated results. The chosen simulator is the widely used NS-2 [2,3]. It provides a flexible structure where modules can be added. The radio model used is the 802.11 MAC layer as it is widely used protocol for a number of other academic works based on the NS simulator compared to the S-MAC. In the simulation used to generate this graph, sensors were placed on a fully packed grid of the specified diameter; sensor values were randomly selected from the uniform distribution over the range [0,1000].

MAX, and COUNT, and AVERAGE were compared by using our approach to that of TAG. The architecture is as comparable to the TAGs approach to

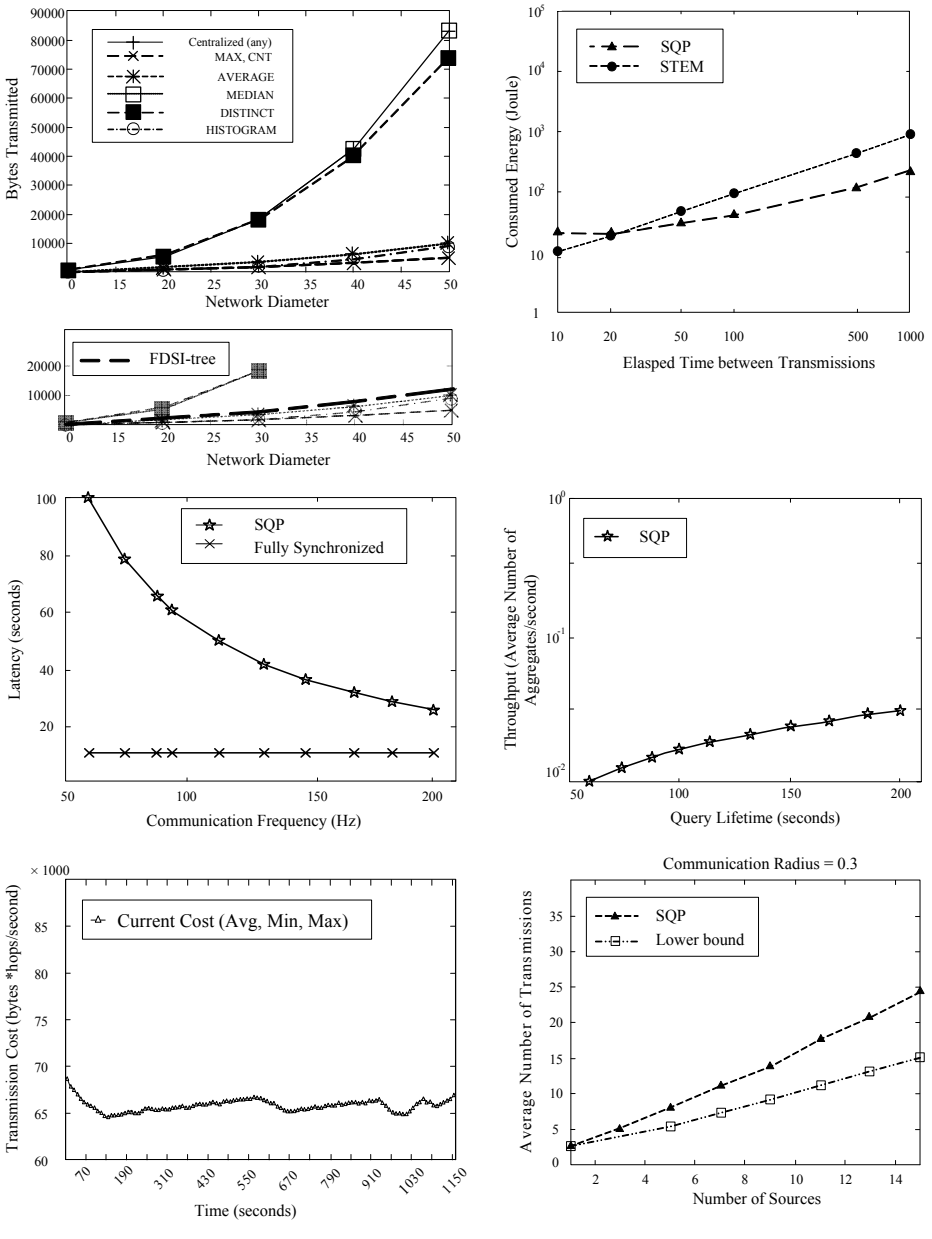


Fig. 3. Performance graphs. Average messages transmitted under distributed processing are as comparable to that of TAG's. Energy consumption slighly increases initially but levels as the time increases. Exponential decrease in latency for increased communication approaches synchronized state. Throughput increases for the long running queries. Reduction of transmission cost versus execution time is dominantly small for prolonged execution of an aggregation query. The number of transmissions to yeild acceptable results is comparably closer to the lower bound for the number of sources.

the substantial reduction in the number of messages: better than an order of magnitude over the centralized approach in the case of simple aggregates like MAX and COUNT, as can be seen from figure 3. They allow extraction of data using a declarative query interface. The routing tree of TAG can be described as vertical routing as every node sends data closer to the sink. For TAG to have complex correlated sub-queries additional routing, permitting horizontal routing is required. Horizontal routing allows several sensor nodes scattered across the network to communicate within the network. Implementing this in TAG will require a substantial change to its routing mechanism. This justifies the use of our routing tree FDSI-tree and the systolic communication paradigm for portability.

Figure 3 lays the performance results based on our simulation. At first, the clustering technique we advocate is based on our previous work, FDSI-tree. The average messages for the aggregate functions are far better than the centralized version, as can also be seen for the TAGs approach. We next analyze the energy consumption, transmission delays, the throughput and latency. We can see that the energy consumption using the proposed SQP technique increases very slowly with the increase of transmission delay. On the other hand, the energy consumption by STEM [4] increases rather rapidly. When the epoch period is very small, the energy consumed during the wakeup procedure contributes most to the total energy consumption. As initially, more messages are exchanged, termed as systolic operation, SQP consumes slightly more energy. With the lapse of the period, energy spent in the operation is reduced as only small number of messages is needed to update states within each cluster. It is readily understood that as the communication frequency increases, the latency decreases exponentially. The comparison basis was presumed to be against a fully synchronized hypothetical system, where every communication is predictable and occurs under predefined or predicted hopping, which is highly unlikely in the case of sensor networks. But the latency curve clearly puts the dilemma between communication frequency and latency. The point to be noted is the communication frequency is comparably small in the case of our approach. The average number of aggregates per second over query lifetime provides us the increased throughput. In general, the query lifetime is dependent on the application. But for monitoring scenarios, the figures that indicate the lifetime in seconds can be considered as appropriately calibrated.

5 Conclusion

In this paper, we proposed a novel methodology in processing aggregate queries using systolic processing in sensor networks. The pipelined messaging technique leads to a high throughput and considerable energy saving for long running queries. Feasibility issues are well justified by the experimental results on message cost and system response time, which clearly brings out a novel approach to sensor data management. With the advent of smart sensors in maneuvering inter-node messaging, we point out to their potential use to improve the energy utilization in a sensor network as our future work.

References

1. Jason L. Hill, David E. Culler. Mica: A Wireless Platform for Deeply Embedded Networks. IEEE Micro, vol.22, no.6, pp. 12-24, Nov/Dec, 2002.
2. S. Bajaj, et.all. Improving simulation for network research. Tech. Report 99-702b, University of Southern California, March 1999, revised September 1999.
3. D. Estrin, M. Handley, J. Heidemann, S. McCanne, Y. Xu, and H. Yu. Network visualization with the Nam, VINT network animator. IEEE Computer 2000, no. 11, 6368.
4. C. Schurgers. Optimizing Sensor Networks in the Energy-Latency-Density Design Space. IEEE Trans. on Mobile Computing, Vol.1, No.1, Jan.-Mar. 2002, pp. 70-80.
5. S. Madden, M.J. Franklin, J. Hellerstein and W. Hong. TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks. In Proc of 5th Annual Symposium on Operating Systems Design and Implementation (OSDI), 2002.
6. Chalermek Intanagonwiwat, Ramesh Govindan, and Deborah Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In Proceedings of the 6th annual ACM/IEEE international conference on mobile computing and networking, Boston, MA, USA, 2000, pp. 56–67.
7. H.T. Kung, and C.E. Leiserson, Systolic arrays for VLSI. In Sparse Matrix Proceedings, 1978, SIAM, Philadelphia, 1979.
8. B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In Proc. of MobiCom, July 2001.
9. Madden, S.R.,Franklin M.J. and Hellerstein, J.M. TinyDB: An Acquisitional Query Processing System for Sensor Networks. ACM Transactions on Database Systems, March 2005, 121-173.
10. Yong.Yao and Johannes.Gehrke. The Cougar Approach to In-Network Query Processing in Sensor Networks. In SIGMOD, 2002.
11. J. Zhao, R. Govindan and D. Estrin. Computing Aggregates for Monitoring Wireless Sensor Networks. The First IEEE Intl. Workshop on Sensor Network Protocols and Applications (SNPA), 2003.
12. B. Przydatek, D. Song, and A. Perrig. Secure Information Aggregation in Sensor Networks. In Proc. of the First ACM Conf. on Embedded Networked Systems (SenSys), 2003.
13. J. Considine, F. Li, G. Kollios and J. Byers. Approximate Aggregation Techniques for Sensor Databases. In Proc. Of the 20th Intl. Conf. on Data Engineering, 2004.
14. Heidemann, J., Silva, F.,Intanagonwiwat, C., Govindan, R., Estrin, D., and Ganesan, D. Building Efficient Wireless Sensor Networks with Low-level Naming. In SOAP (2001).
15. Greenstein, B., Estrin, D., Govindan, R., Ratnasamy, S., and Shenker, S. DIFS: A Distributed Index for Features in Sensor Networks. In Proc. 1st IEEE International Workshop on Sensor Network Protocols and Applications, Anchorage, AK (2003).
16. Sang Hun Eo, Suraj Pandey, Soon-Young Park, Hae-Young Bae. FDSI-Tree: A Fully Distributed Spatial Index Tree for Efficient & Power-Aware Range Queries in Sensor Networks. SOFSEM 2006: 254-261.
17. S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker. GHT: A Geographic Hash Table for Data-Centric Storage in SensorNets. In Proc. of 1st ACM WSNA, September 2002.
18. Lyall A. et al. Implementation of Inexact String Matching on the ICL DAP. In Feilmeier et al eds., Parallel Computing 85, North Holland, 1986.

Adapted Listening in Wireless Sensor Network MAC Protocol

Zhen Fu[1], Yuan Yang[1], Tae-Seok Lee[1,2], and Myong-Soon Park[1,*]

[1] Internet Computing Lab, Dept. of Computer Science and Engineering, Korea University, Korea
{fuzhen, yy, myongsp}@ilab.korea.ac.kr
[2] Korea Institute of Science and Technology Information, Korea
tsyi@kisti.re.kr

Abstract. Research activity in the area of medium-access control protocols of wireless sensor networks (WSN) has grown dramatically in the past few years. A number of MAC protocols are proposed to achieve energy efficiency upon the limitation of WSN which are usually deployed in a special environment, assigned with long-term work, and supported by limited battery. It is found that there is a trade-off between energy efficiency and transmission delay, thus duty cycle of fixed sleep/listening MAC protocols has to be adjusted carefully to achieve the best performance. Taking the challenge to design an adapted listening mechanism, A-MAC is designed to adaptively and dynamically adjust the duty listening time based on traffic load. With A-MAC, it can achieve more power efficiency in low traffic load and much less transmission latency comparing to existing MAC protocols. We simulate A-MAC, and the simulation result shows that A-MAC can significantly prolong the lifetime of network when traffic is low and reduce packet delivery latency.

1 Introduction

A wireless sensor networks is a collection of sensor nodes that is spread around an area in which a certain phenomenon of interest is expected to take place [1]. For example, in a general wireless sensor network scenario, when an abnormal event, such as a fire, is detected, all the sensors which detected the object will send the collected data to the base station by multi-hop connection.

We study the problem of media access control (MAC) protocol in the novel regime of sensor networks, where unique application behavior and tight constraints in computation power, storage, energy resources, and multi-hop transition require have shaped this design space to be very different from traditional wireless networks.

Although many high level architectural and programming aspects of this area are still being resolved, the underlying media access control and transmission control protocols are critical enabling technology for many sensor network applications.

Application behavior in wireless sensor networks lead to very different traffic characteristics from that found in conventional wireless computer networks. Sensor

* Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 546 – 555, 2006.

networks are different from traditional networks in several ways. At first, usually powered by batteries, sensor nodes are often difficult to replace batteries or get charged. Second, sensor networks are often deployed in random fashion rather than pre-planning. Third, as large numbers of node are often deployed in sensor networks, the node density may vary from time to time and place to place. Finally, traffic in sensor a network is often triggered by special events and be burst sometimes. Thus traditional MAC protocols are not suitable for wireless sensor networks and need modifications.

As reducing the energy consumption is the primary concern about wireless sensor networks and at the same time, to reduce the latency in multi-hop data transmission is also very important. Sensor nodes are expected to be switched to the sleep mode in order to reduce energy consumption. However, fixed Sleep/Listening schedule strategy is usually adopted in the existing proposals which cause unnecessary idle listening problem and conspicuous transmission latency due to the diversity of the traffic-load in the network. Thus we proposed A-MAC, a MAC protocol designed to dynamically adjust the duty listening time based on traffic load which reduces idle listening time when traffic load is low and also reduces transmission delay when traffic load is high. To satisfy the application's requirement, A-MAC adjusts the data transfer rate properly. Meanwhile, special strategy is adopted when data transmission bursts occur in the network which requires much less transmission delay compared to common situations.

The rest of this paper is organized as follow. Section 2 describes three requirements to design a MAC protocol for sensor networks. Related works are briefly reviewed in Section 3. Section 4 proposed A-MAC, the adapted listening MAC protocol in wireless sensor networks. Meanwhile, Section 5 illustrates the performance evaluation of A-MAC. Conclusions are finally made in Section 6.

2 Three Wireless Sensor Network Requirements

In this section, we will discuss some important characteristics of sensor networks and how design of MAC protocols can be made to meet the challenges. According to the challenges reflected by the characteristics of sensor networks, a well-designed MAC protocol should achieve following three requirements:

Requirement 1, energy efficiency

One of the most important requirements of wireless sensor networks MAC protocols is Energy efficiency. As stated above, it is very difficult to change or recharge batteries for large numbers of sensor nodes which is usually battery powered. Actually, in general, it is designed for sensor networks to build nodes that are cheap enough to be discarded and efficient enough to operate only on limited power sources. In all cases, prolonging the lifetime of sensor nodes is an essential problem. On many hardware platforms, the radio is a major energy consumer. According to the research of UC Berkeley, radio communication including both receiving data and transmitting data consumes much energy. As the MAC layer controls radio activities directly, the energy saving from the radio control of the MAC protocols significantly affects the sensor node lifetime.

Requirement 2, reduce transmission latency
Latency means the time interval from when a sender node has a packet to send until the packet is successfully received. In sensor networks, the importance of latency is determined by the application. Regarding multitudinous applications are usually on environment surveillance, especially for the special events monitoring such as fire alarm, earthquake and oil leaking, the data transmission is restricted by strict time limitation, collected data should be transferred in time. Overtime data is meaningless for such kinds of applications. Therefore, it is also very important to reduce the data transmission latency.

For most of wireless sensor networks, in some applications such as environment surveillance, sensor nodes are usually remaining inactive until some events are detected. In a period with no sensing event, there is normally very little data need to be transmitted in the network. Thus energy savings are generally more important than message latency in this situation. However, once after detection, low-latency operation becomes more important.

Requirement 3, Scalability and adaptability
Scalability and adaptability are also very important requirement of MAC protocol that handles chances in network node density, topology and size. A well designed MAC protocol should accommodate node changes gracefully. Scalability and adaptively are very important because sensor networks are often deployed in an ad hoc manner and operated in uncertain environments.

In summary, based on the characteristics of a MAC protocol discussed above, the most important requirements for design a MAC protocol for sensor networks are energy efficiency, transmission latency, scalability and adaptability.

3 Related works

Various MAC protocols were proposed for wireless sensor networks. Current MAC protocols for sensor networks can be divided into TDMA and contention based protocols.

The proposed TDMA protocols are based on performing TDMA scheduling in communication clusters [2]. TDMA divides the channel into N time slots, as shown in Figure 1. In each slot, only one node is allowed to transmit. The N slots comprise a frame, which repeats cyclically. The overhead of forming these clusters, and inter-cluster Communication /Interference may eliminate the efficiency of TDMA. Therefore, when adopted into large scale sensor networks, TDMA may not be able to provide sufficient performance for network's requirements in scalability and adaptability.

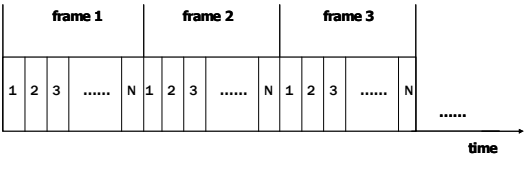


Fig. 1. TDMA time slots assignment

The standardized IEEE 802.11 [4] is an example of the contention based protocol. It is widely used in wireless ad hoc networks because of its simplicity and robustness to the hidden terminal problem. However, recent work has shown that the energy consumption using this MAC is very high when nodes are in idle mode. This is mainly due to the idle listening.

In order to remit the power consumption of nodes' idle listening, the protocol in [5] [9] called S-MAC is proposed, as showed in figure 2, S-MAC makes nodes periodically listen and sleep, neighbor nodes form virtual clusters and follow same sleep schedule. Nodes that reside in two separate virtual clusters are called "border nodes". In order to relay data, border nodes are set to listen when any of the two clusters are in the listening status. The drawback of S-MAC algorithm is that the border nodes have to follow two schedules, which results in more energy consumption via idle listening and over-hearing. Moreover, when the traffic load is higher than the node's data transmission capability provided by the fixed duty listening time, the S-MAC will inevitably cause transmission latency. Especially for multi-hop routing sensor networks, since all immediate nodes have to wait until the next node is wake up for listening. Therefore, S-MAC can achieve the requirement 1 in a certain extent but cannot achieve the requirement 2 in relatively high traffic-load state.

Fig. 2. S-MAC Periodic listening and sleep schedule

To solve the many problems which are led by the border nodes in S-MAC, Tae-Seok Lee introduced a unifying scheduling mechanism called H-SYNC (Heartbeat-SYNC) [6] to synchronize all the sensor nodes' Sleep/Listening time rate with the base station. This mechanism is able to eliminate the "border nodes" in S-MAC. In another word, the mechanism overcomes border node problems by adopting a policy for enabling the whole network to be unified by a single schedule and provides better performance in energy efficiency and remission of transmission latency than S-MAC. However, similar with S-MAC, H-SYNC's fixed Sleep/Listening schedule still lacks capability to deal with the variety of the data traffic load.

4 Proposed Traffic-Load Aware MAC Protocol

For the three requirements described in the section 2, our proposed MAC protocol A-MAC also follows contention based mechanism to achieve the third requirement. Furthermore, we adopt dynamic adjustment of Sleep/Listening rate of the sensor nodes and a synchronization mechanism to achieve the first requirement and the second requirement. This section describes A-MAC in detail.

4.1 Maintain Synchronization

In terms of overcoming the transmission latency problems in virtual clusters in S-MAC, we introduce a new synchronization mechanism. We let a sensor node synchronize to its up level nodes that are called parent nodes.

For example, as the base station is assigned as the level 0 node, the closet nodes to the base station are assigned as level 1 nodes. After the construction of the level structure of the network, nodes will synchronize with their corresponding parent nodes by periodically broadcasting SYNC packets which are very short and contain sender's address and the current scheduling. After the synchronization, time rates of the senders' neighbor nodes will be synchronized with the senders.

In order to receive SYNC packets, Burst-Info packets and data packets for a sensor node, we divide its listening interval into three parts. As showed in figure 3, the first part is for receiving SYNC packets, the second is for receiving Burst-Info packets, and the third is for data transmission, and each part is further divided into many time slots for senders to perform carrier sense. For example, if a sender wants to send a SYNC packet, it starts carrier sense when the receiver begins listening. It randomly selects a time slot to finish its carrier sense. If it has not detected any transmission by the end of the time slot, it wins the medium and starts sending its SYNC packet at that time. The same procedure is followed when sending data packets.

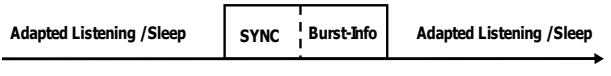


Fig. 3. A-MAC time interval assignment

If multiple neighbors want to talk to a node which is in the duty listening state, they need to contend for the medium. The contention mechanism is the same as that in IEEE 802.11, i.e., using RTS (Request To Send) and CTS (Clear To Send) packets. The node that sends out the RTS packet wins the medium at first, and the receiver will reply with a CTS packet. Once the sender receives the CTS packet, it will start the data transmission process with the receiver.

4.2 Duty Listening Time Adjustment

This section introduces the adjustment of the schedule of sensor nodes' listening and sleep rate.

Because the conflict between the bandwidth required in practice and bandwidth provided by the sensor nodes affects the un-proportion. MAC needs to control its duty listening time length to ensure that the data could obtain channel access and could reach to base station. Our duty listening time control at first measures the data packet length and then uses a gradual increase and fast decrease approach to adjust the duty listening time. In our protocol, we adopt [8] to detect the data traffic of the network and the data packet length. Firstly, the node uses the maximum duty listening time, if there is any duty listening time left when required data transmission finished, it will reduce the duty listening time in the next cycle rapidly. In the following cycle, if there still exists any left duty listening time, it will reduce the listening time again. The

process will be repeated until the bandwidth meets required bandwidth. Otherwise, if all current duty listening time has been utilized but still cannot satisfy required data transition, sensor node will extend the duty listening time.

Furthermore, in order to handle burst data transmission, for example, when a sensor network remaining inactive for long periods suddenly needs to transmit a great deal of data, we add a Burst-Info interval before data transmission, which is showed in figure 3. In this time interval, if a sensor node has too many data packets to transmit suddenly, in order to increase the network's data transmission capability, the node will send a Burst-Info packet to the receiver nodes to request for more listening time to receive the data packet. The nodes that receive the Burst-Info packets immediately increase its duty listening time accordingly. A more detailed description has been given in the figure 4.

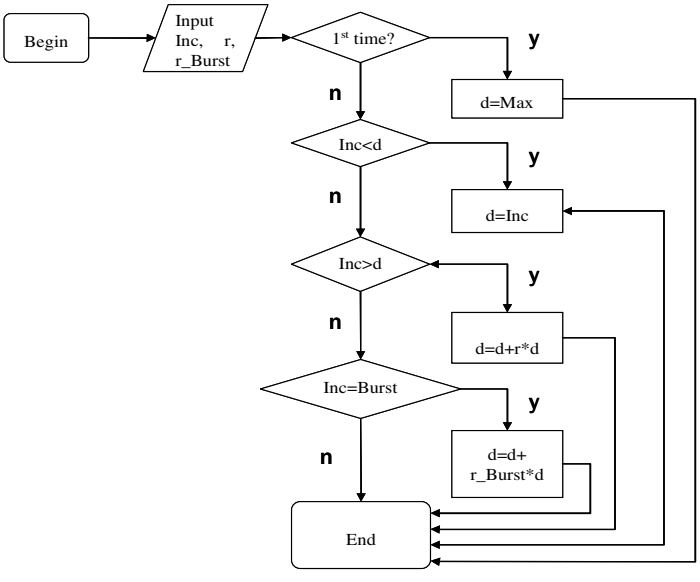


Fig. 4. Algorithm to decide one node's duty listening time

Here r and r_Burst are increase rate of the duty listening time which is according to the application scenario. For example, in our simulation, we assume the r as 5% and r_Burst as 15%. According to our simulation result, that can satisfy most of the application requirements.

The advantage of dynamic duty listening time adjustment mechanism is: it is fit for the diversity of traffic-load in wireless sensor networks. It can not only save the energy consumption by avoid unnecessary idle listening in general low data transmission periods, but also be able to increase its data transmission capability to meet the requirements to reduce the transmission latency in some possible special situations.

5 Simulation

We implemented our prototype in the TinyOS [7] with TOSSIM network simulator. The energy costs of the Tx:Rx:Slp radio modes is 1.78:1:0.06. SMAC and H-SYNC have the basic duty cycle of 50%. This means in a 100ms frame, a sleep period of 50ms, 50ms for listening. A-MAC follows the same frame length but duty listening length is adapted with the traffic load, and the increase rate of the duty listening time r and r_Burst is assigned as 5% and 15% respectively.

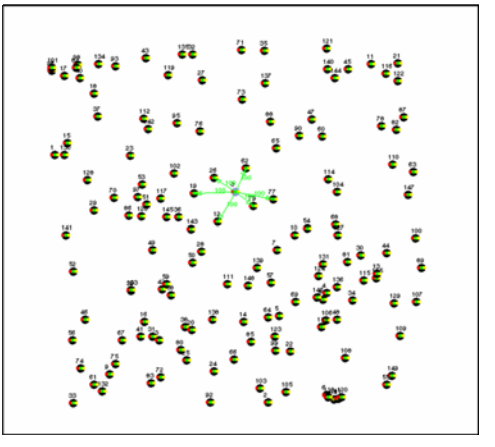


Fig. 5. Network topology

5.1 Energy Consumption Comparison

We compare the energy consumption of different MAC protocol with various traffic loads. In our experiments we have used a network with a 150 nodes randomly deployed, here node 0 is base station, as showed in figure 5. We have chosen a radio range so that all the nodes in the network can communicate with base station by one hop or multiple hops, here blue links around node 0 mean all the possible data links of the node. All the source nodes generate data packets and transmit to base station, which is located in the center of networks.

In each test, the source node sends a fixed amount of data, 20 messages of 100-bytes each which are forwarded to base station. Figure 6 shows how energy consumption on all nodes in the network changes as traffic load varies from heavy (on the left) to light (on the right).

From figure 6, we can find out that, at light load, operating at a low duty cycle can save significant amounts of energy compared to not sleeping. It also shows the importance of adaptive listening time extension when traffic becomes heavy. when data generation interval is large than 0.6 second, which means comparatively less data traffic, A-MAC' s energy consumption is conspicuously less than both S-MAC and H-SYNC. When data generation interval is around 0.5 second, energy consumption of the three protocols is close. Because at this time, to achieve the requirement of network bandwidth, A-MAC adjusts its duty listening time close to other two

protocols. However, when data generation interval is less than 0.4 second, without adaptive listening, S-MAC and H-SYNC consumes less energy than A-MAC. The result shows that, although S-MAC and H-SYNC could both reduce the energy consumption by adopting sleeping/listening schedule, the idle listening energy consumption is still higher than A-MAC when the data transmitted doesn't exceed the transmission capability of node. In contrast, the A-MAC performs better by creating traffic aware listening/sleeping time rate of sensor nodes when traffic load is low.

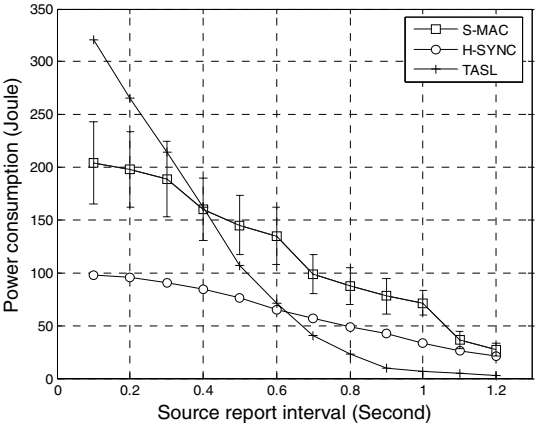


Fig. 6. Energy consumption

When the interval of data generation of source nodes reduces, data transmitted from source nodes will increase and even exceed the maximal duty listening time S-MAC and H-SYNC can bear. In this situation, energy consumption of A-MAC will increase accordingly and finally exceed the other two protocols' energy consumption. It is because of that, the duty listening time of S-MAC and H-SYNC is invariable, even if the actual data transmission requirement exceeds node's capacity. In contrast, in order to reduce the data transmission latency, A-MAC will increase node's duty listening time. Although it brings more energy consumption, it also brings benefits that transmission latency will be conspicuously reduced.

5.2 Latency Analysis

A disadvantage of indirect sleep MAC is that the latency of sending a message can be increased. In this simulation, latency is measured by the time a message takes to travel over one average hop when traffic load of the network varies.

To measure the latency, we use the network with the same network topology which contains 150 random deployed nodes and all the sensor nodes transmit data to base station periodically. Figure 7 shows the measured mean message latency under different traffic loads. When traffic load is low, duty listening time of S-MAC and H-SYNC provides sufficient data relay capability, which can cover required bandwidth for data transfer. At this time, the delay of the three methods is similar. The reason for the latency is that each message may need to wait for one sleep cycle on each hop.

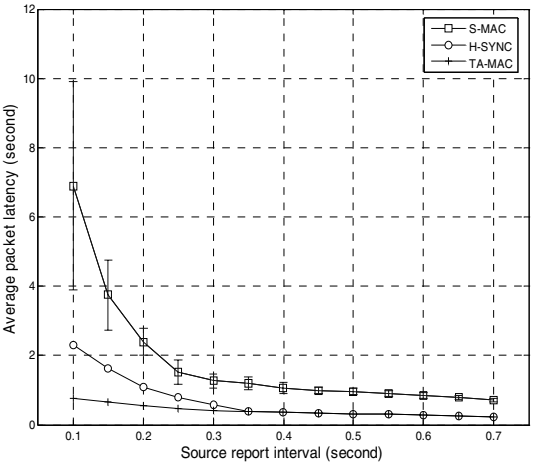


Fig. 7. Average energy consumption

But when the traffic load exceeds that can be covered by S-MAC and H-SYNC fixed duty listening cycle, the latency turns to be higher than A-MAC's. The reason is that messages can reduce the transmission delay caused by inadequate network bandwidth when a good deal of data needs to be transmitted.

6 Conclusion

This paper focuses on designing an adaptive listening mechanism for wireless sensor networks. Our design, A-MAC, can adaptively adjust listening/sleeping schedule of wireless sensor nodes according to the traffic load.

As the traffic load of wireless sensor networks usually changes from time to time, adaptive listening mechanism is very important for sensor network applications. Because when the traffic load is low, such a mechanism can effectively reduce unnecessary energy consumption due to the idle listening; and also reduce transmission latency when traffic load comes to be heavy, e.g. when some special sensing events are detected by sensor nodes.

Our simulation results have shown that A-MAC achieves the goal of adaptive listening mechanism as it provides a well balance between energy consumption and transmission latency. It can effectively reduce idle listening when traffic load is light and reduce transmission latency when timely data delivery becomes the main concern. Considering its simplicity and adaptively, it can help for applications with high requirements of energy efficiency and transmission delay.

Acknowledgement

This work was supported by the Korea Research Foundation Grant and funded by the Korea Government (MOEHRD) (KRF-2005-211-D00274).

References

1. Feng Zhao and Leonidas Guibas, Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann (ISBN 1-55860-914-8), 2004.
2. J. C. Haartsen, “The Bluetooth radio system,” IEEE Pers. Commun. Mag., pp. 28–36, Feb. 2000.
3. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, “Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks”, In Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys), Los Angeles, CA, Nov. 2003.
4. LAN MAN Standards Committee of the IEEE Computer Society, Wireless LAN medium access control (MAC) and physical layer (PHY) specification, IEEE, New York, NY, USA, IEEE Std 802.11-1997 edition, 1997.
5. Ye W, Heidemann J, Estrin D. An energy-efficient MAC protocol for wireless sensor networks. In: Kermani P, ed. Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies. Piscataway: IEEE Press, 2002.
6. Taeseok Lee, Yuan Yang, KiJeong Shin, and MyongSoon Park, "An Energy-Efficient Uni-Scheduling based on S-MAC in Wireless Sensor Network", High Performance Computing and Communications, Italy. September 2005.
7. An open-source OS for the networked sensor regime. http://www.tinyos.net
8. Ivan Stojmenovic, Amiya Nayak, and Jonson Kuruv-ila, Site, Univeristy of Ottawa, “Design Guidelines for Routing Protocols in Ad Hoc and Sensor Networks with a Realistic Physical Layer”.
9. Wei Ye and John Heidemann, “Medium Access Control in Wireless Sensor Networks”, USC/ISI Technical Report, Oct. 2003.

Relay Shift Based Self-deployment for Mobility Limited Sensor Networks

Xiaoling Wu, Yu Niu, Lei Shu, Jinsung Cho*, Youngkoo Lee, and Sungyoung Lee

Department of Computer Engineering, Kyung Hee University, Korea
{xiaoling, niuyu, sl8132, sylee}@oslab.khu.ac.kr,
{chojs, yklee}@khu.ac.kr

Abstract. In this paper, we propose a relay shift based approach to solve uneven sensor distribution problem due to the initial random dropping or the existence of faulty sensors. The distinguishing feature of our work is that the sensors in our model have limited mobility. After determining the optimal cluster head positions by particle swarm optimization (PSO) method, we use proposed Relay Shift Based Algorithm (RSBA) for movement assisted sensor deployment. Dijkstra's algorithm is applied to find a shortest path from a redundant sensor to a virtual node point in an uncovered area, and each sensor moves along this path by relay shift based on the principle that evenly distributed sensors can provide better coverage. Simulation results show that our approach can provide high coverage within a short time and limited movement distance as well as ensuring connectivity and energy efficiency.

1 Introduction

Wireless sensor networks are expected to be widely utilized in the future since they can greatly enhance our capability of monitoring and controlling the physical environment. Due to the inevitable relation with the physical world, the proper deployment of sensors is very important for the successful completion of the sensing tasks.

Sensor deployment has received considerable attention recently. Some of the work [1], [2], [3] assume that the environment is sufficiently known and under control. However, when the environment is unknown or inhospitable such as remote inaccessible areas, disaster fields and toxic urban regions, sensor deployment cannot be performed manually. To scatter sensors by aircraft is one possible solution. However, using this scheme, the actual landing position cannot be controlled due to the existence of wind and obstacles such as trees and buildings. Consequently, the coverage may not be able to satisfy the application requirements. Some researchers suggest simply deploying large amount of static sensors to increase coverage; however it often ends up harming the performance of the network [4]. Moreover, in many cases, such as during in-building toxic-leaks detection [5], chemical sensors must be placed inside a building from the entrance of the building. In such cases, it is necessary to take advantage of mobile sensors, which can move to the appropriate places to provide the required coverage.

* Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 556 – 564, 2006.

To address this issue, a class of work has recently appeared where mobility of sensors is utilized to achieve desired deployment [6], [7], [8], [9], [10]. Typically in such works, the sensors detect lack of desired deployment objectives, then estimate new locations, and move to the resulting locations. For example, in [8], the authors present the virtual force algorithm (VFA) as a new approach for sensor deployment to improve the sensor field coverage after an initial random placement of sensor nodes. The cluster head (CH) executes the VFA algorithm to find new locations for sensors to enhance the overall coverage. They also considered unavoidable uncertainty existing in the precomputed sensor node locations. This uncertainty-aware deployment algorithm provides high coverage with a minimum number of sensor nodes. While the above works are quite novel in their approaches, the mobility of the sensors in their models is assumed unlimited. Specifically, if a sensor node chooses to move to a desired location, it can do so without any limitation in the movement distance.

In fact, the mobility of sensors is limited in most cases. To this extent, a class of Intelligent Mobile Land Mine Units (IMLM) [11] to be deployed in battlefields have been developed by DARPA. The IMLM units are employed to detect breaches, and move with limited mobility to repair them. This mobility system is based on a hopping mechanism that is actuated by a single-cylinder combustion process. For each hop, the fuel is metered into the combustion chamber and ignited to propel the IMLM unit into the air. The hop distance is limited, depending on the amount of fuel and the propeller dynamics. The units contain a righting system to orient itself properly after landing, and a steering system that provides directional control for movement. Some other techniques can also provide such kind of mobility, for instance, sensors supplied by spring actuation etc. This type of model normally trades-off mobility with energy consumption [12, 13]. Moreover, in many applications, the latter goals outweigh the necessity for advanced mobility, making such mobility models quite practical in the future. [13] is one of the very few papers which deal with the mobility limited deployment optimization. The mobility in the sensors they consider is restricted to a flip. However coverage is the only considered objective in their paper and their approach is not feasible in network partition case.

In this paper, we design and evaluate our proposed Relay Shift Based Algorithm (RSBA) for mobility limited sensor self-deployment. In our model, sensors can move only one hop at a time to a new location, i.e., the moving distance is bounded by a certain value (we use transmission range which makes sense in terms of connectivity). A certain number of mobility limited sensors are initially deployed in the sensor network. The sensors nodes are clustered and optimal CH positions are chosen using PSO which is borrowed from our previous work [10]. The initial deployment may not cover all regions in the network. Regions that are not covered by any sensors are coverage holes. In this framework, our problem is to determine an optimal movement plan for the sensors in order to maximize the network coverage and simultaneously minimize the total number of sensor movements. We use Dijkstra's algorithm to find a shortest path from a redundant sensor to the virtual node point in a coverage hole, and design relay shift based sensor deployment protocol based on the principle of moving sensors from densely deployed areas to sparsely deployed areas.

The rest of the paper is organized as follows. Section 2 introduces the energy efficient CH positioning method. In section 3, we present the proposed Relay Shift Based Algorithm (RSBA) for mobile nodes self-deployment. Section 4 evaluates the

performance of the proposed method and compares with related work. Based on the simulation results, we justify our design and discuss future work in Section 5.

2 Energy-Efficient Clustering

2.1 Technical Preliminary: Particle Swarm Optimization

Particle Swarm Optimization (PSO) is an evolutionary computing technique based on the principle of bird flocking. In PSO a set of particles is initialized randomly. Each particle will have a fitness value, which will be evaluated by the fitness function to be optimized in each generation, and knows its best position *pbest* and the best position so far among the entire group of particles *gbest*. The particle will have velocities, which direct the flying of the particle. In each generation the velocity and the position of the particle will be updated. The equation for the velocity and positions are given below as (1) and (2) respectively,

$$v_{id} = \varpi \times v_{id} + c_1 \times rand() \times (p_{id} - x_{id}) + c_2 \times rand() \times (p_{gd} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

where ϖ is the inertia weight, and c_1 and c_2 are acceleration coefficients.

PSO shares many similarities with Genetic Algorithm (GA), however, due to the inexpensive computation in terms of both memory requirements and speed, we choose PSO as the optimization method.

2.2 Determination of Optimal Cluster Head Positions

The model of mobile sensor network is presented as follows. We assume that each node knows its position in the problem space; it is possible by using some localization method [14]. All sensor members in a cluster are homogeneous and cluster heads (CHs) are more powerful than sensor members. Sensing coverage of each node is assumed to have a circular shape without any irregularity. The design variables are 2D coordinates of the sensor nodes, $\{(x_1, y_1), (x_2, y_2), \ldots\ldots\}$.

We intend to minimize energy usage in a cluster based sensor network topology by finding the optimal CH positions. For this purpose, we assume a power consumption model [15] for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics. This is one of the most widely used models in sensor network simulation analysis. Both the free space ($distance^2$ power loss) and the multi-path fading ($distance^4$ power loss) channel models are used here. Assume that the sensor nodes inside a cluster have short distance *dis* to CH but each CH has long distance *Dis* to the base station. Thus for each sensor node inside a cluster, to transmit an *l*-bit message a distance *dis* to CH, the radio expends

$$E_{TS}(l, dis) = lE_{elec} + l\varepsilon_{fs} dis^2 \quad (3)$$

For CH, however, to transmit an l-bit message a distance Dis to base station, the radio expends

$$E_{TH}(l, Dis) = lE_{elec} + l\varepsilon_{mp} Dis^4 \tag{4}$$

In both cases, to receive the message, the radio expends:

$$E_R(l) = lE_{elec} \tag{5}$$

Here we set electronics energy as E_{elec}=50nJ/bit, whereas the amplifier constant, is taken as ε_{fs} =10$pJ/bit/m^2$, ε_{mp} = 0.0013$pJ/bit/m^2$. Since the energy consumption for communication is the most significant, we neglect sensing and computation energy consumption here.

Assume m clusters with n_j sensor members in the j^{th} cluster C_j. We derive the fitness function as in [10]:

$$f = \sum_{j=1}^{m} \sum_{i=1}^{n_j} (0.01 dis_{ij}^2 + \frac{1.3\times10^{-6} Dis_j^4}{n_j}) \tag{6}$$

3 Proposed Deployment Approach: RSBA

Let $G(V, E)$ be the graph defined on V with edges $uv \in$E iff $uv \le R$. Here uv is the Euclidean distance between nodes u and v, R is the communication range.

We have 4 steps for implementing RSBA:

Step 1: Randomly deploy nodes in the network.

Step 2: Detect coverage holes and redundant sensor nodes. We set 2 distance threshold value T_1 and T_2. If the longest distance between 2 nodes A and B along the uncovered area perimeter is larger than T_1, regard it as a coverage hole, and create a virtual node point at the center of AB. If the distance between two neighbors is less than T_2, regard them as redundant nodes. Choose a redundant node nearest to the virtual node point in coverage hole.

Step 3: Use the widely used Dijkstra's algorithm [16] to find a shortest path n_0-n_1-n_2-…-n_{n-1} from a redundant sensor n_0 to the destination n_{n-1} (added virtual node) in a coverage hole. The distance between n_{n-2} to n_{n-1} is bounded by R. Since Dijkstra's algorithm was designed to solve the single-source shortest path problem for a directed graph with nonnegative edge weights, it is feasible here.

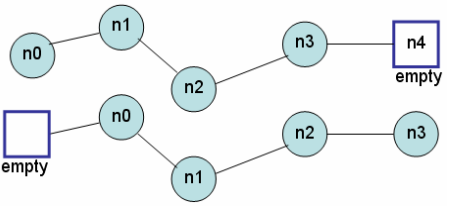


Fig. 1. Illustration of sensor nodes relay shift along the shortest path

```
1. Initialization
   initial_node_locations (netXloc, netYloc);
   sensing_range r;
   communication_range R;
  If     distance (i, j) <= R
         link i and j;
2. Create Virtual Nodes and Redundant Nodes
   calculate the longest length of 2 points A & B along the hole arc;
   detect coverage holes;
   calculate the center point C of edge AB; % C becomes Virtual Nodes
  If     distance(i, j) <= T2
           If   distance (i, C) < distance(j, C)
                define i as source and C as destination;
3. Shortest Path Finding by Dijkstra's algorithm
Function [path, totalCost] = dijkstra (m, netCostMatrix, s, d)
% m: number of nodes in the network, s: source node index, d: destination node index,
% path: node sequence of shortest path, totalCost: distance along shortest path
4. Sensor Nodes Movement
For k=1: length (path)-1
netloc (k) = netloc (k+1);
Update nodes link;
Calculate network coverage
```

Fig. 2. Pseudocode of the proposed RSBA method for sensor nodes reorganization

Step 4: Move sensor node n_{n-2} to the virtual node n_{n-1}, move n_{n-3} to n_{n-2} … finally move the redundant sensor n_0 to n_1, and leave the original location of sensor n_0 empty. The nodes coordinates can be updated by equation (7):

$$NetLoc(n_i) = NetLoc(n_{i+1}), \qquad i = 0, 1, \cdots, n-2 \tag{7}$$

$n_i \in$ nodes on shortest path from source to destination

n_0=source node

n_{n-1} =destination (virtual node)

The process is illustrated in Fig 1 using an example of four sensors and one virtual node along the shortest path. Sensor node n_3 moves to the virtual node point n_4, n_2 moves to n_3 … finally the redundant sensor n_0 moves to n_1, and leave the original location of n_0 empty. The network coverage is defined as the ratio of the union of areas covered by each node and the area of the entire ROI. It can be calculated using Monte-Carlo method by meshing the ROI as has been done in [10].

The pseudocode of the proposed algorithm is illustrated in Fig 2.

4 Performance Evaluations

4.1 Optimal Clustering Results

For PSO based optimal CH determination, a linear decreasing inertia weight value ϖ from 0.95 to 0.4 is used, and acceleration coefficients c_1 and c_2 are set to 2 according to [10]. The coordinates of potential CHs are set as particles in the sensor network. The communication range of each sensor node is 4 units with a fixed remote base station at (5, 20). The minimum number of clusters acceptable in the problem space is 2, but we choose 3 here. The nodes are organized into clusters by the base station. Fitness value is evaluated by the fitness function (6) in each generation. Our purpose is to find the optimal location of CHs. Once the position of the CH is identified, if there is no node in that position then the one nearest to the CH location will become a CH. Here the CHs determined are nodes labeled 27, 23 and 29, as shown in Fig 3.

4.2 Sensor Movement by Relay Shift: Experimental Results

The performance of the proposed movement assisted algorithm RSBA is evaluated by simulation. For the convenience of comparison, we set the initial parameters the same as in [9]: 30 randomly placed nodes in a region of size 10×10 are used for initial deployment; the r and R used in the experiment are 2 and 4 m, respectively. In Fig. 3, the node locations and coverage of the initial random deployment before running the algorithms are shown. Tiny points with red numerical label beside represent the positions of nodes and green circles are used to show the r of the nodes. Communications are possible between nodes that are connected by a dashed line. Sensor information can be collected within the r and communications between nodes are possible within the R. The initial coverage is 0.9273.

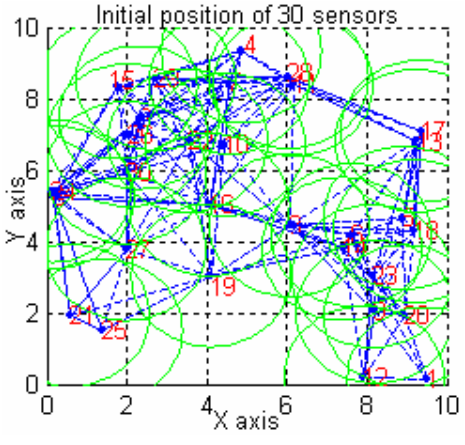


Fig. 3. Initial random deployment with sensing range 2m and communication range 4m

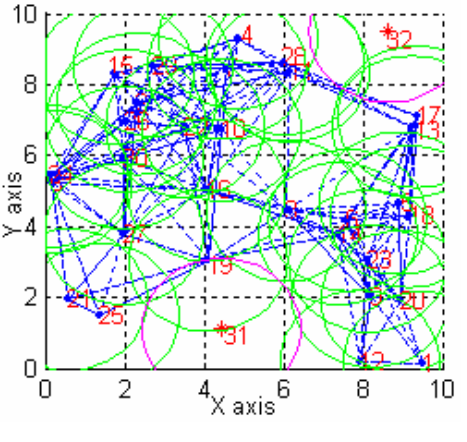


Fig. 4. Determine virtual node point in uncovered area and redundant nodes

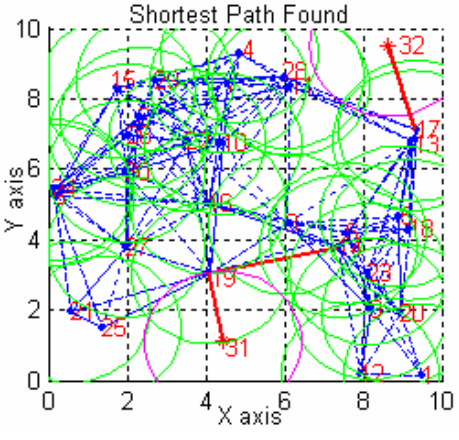


Fig. 5. Find shortest path by Dijkstra's algorithm from redundant node to virtual node point

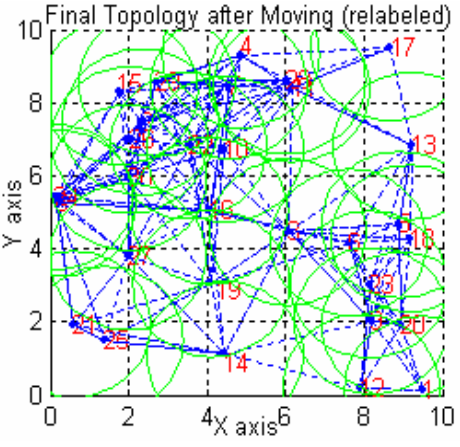


Fig. 6. Final node positions after executing proposed movement-assisted deployment algorithm

Fig. 4 shows the detected virtual node points (labeled as 31 and 32) in coverage hole and the redundant nodes nearest to 31 and 32 are 14 and 17 respectively. Both the coverage holes and the redundant nodes are judged by CHs. This information is then broadcasted by CHs to the whole network. The parameter values needed are: threshold value T_1=1.2 and T_2 = r/4.

Fig. 5 shows the 2 shortest paths found (14→19→31 and 17→32) by Dijkstra's algorithm from redundant nodes to virtual node points. This is also actual path of individual nodes as they move by relay shift, in which sensor node move only one hop at a time which guarantees the connectivity. For the initial distribution of Fig. 3, each node moves a distance of 2.6157 on average and the standard deviation of distance traveled is 0.5714. When the average distance traveled is small, the corresponding energy for locomotion is small. Also, when the standard deviation of distance traveled is small, the variation in energy remaining at each node is not significant and a longer system lifetime with desired coverage can be achieved. Fig 6 shows the final node positions with desired coverage=0.9923 after executing RSBA. Note that the original 30 sensor nodes are finally reorganized and relabeled.

Next, the performances of RSBA are compared with DSSA, IDCA, and VDDA [9] in terms of coverage, standard deviation of distance, movement distance until convergence, and time. Results are presented in Figs. 7–10. Fig. 7 shows the improvement in coverage area from the initial random deployment for RSBA, DSSA, IDCA, and VDDA. All four algorithms exhibit a similar performance. Although the coverage of RSBA (≈99.2%) is slightly lower than other 3 algorithms, this number is often satisfactory for many application requirements. Fig. 8 shows RSBA has lower standard deviation of distance compared with others. It means the variation in energy remaining at each node is small, so that longer lifetime can be achieved. Fig. 9 shows the significant reduction of total distance traveled by RSBA compared with other 3 algorithms. In RSBA, only very few numbers of nodes need to move and each sensor movement is bounded by only one hop. However, almost every node needs to move in the other 3 algorithms. Fig. 10 shows that RSBA leads to faster deployment than

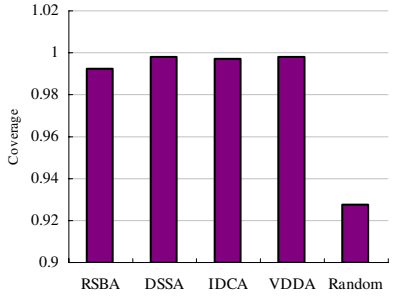


Fig. 7. Coverage comparison

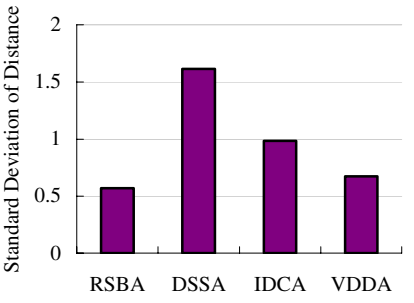


Fig. 8. Standard deviation of distance comparison

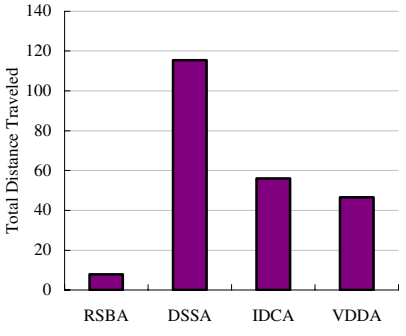


Fig. 9. Total distance traveled comparison

Fig. 10. Termination time comparison

the other 3 algorithms. Termination time is measured in the number of iterations until the algorithms stop.

5 Conclusion and Future Work

In this paper, we designed and evaluated our proposed movement assisted self-deployment approach using sensors with limited mobility. More specifically, sensors can move only one hop at a time to a new location, i.e., the moving distance is bounded by transmission range which guarantees the network connectivity. After initially deploying a certain number of mobility limited sensors in the ROI, the sensors were clustered and the optimal CHs positions were chosen by PSO before movement. We determined an optimal movement plan by proposed RSBA algorithm for the sensors in order to maximize the network coverage and simultaneously minimize the total number of movements. Dijkstra's algorithm was used to find a shortest path from a redundant sensor to the virtual node point in a coverage hole, and mobility limited sensors move by relay shift from densely deployed areas to sparsely deployed areas. Based on simulation, we evaluated and compared our approach RSBA with other related works from various aspects: coverage, standard deviation of distance

traveled, total moving distance, and deployment time, and show that RSBA is very effective in terms of the above standards.

In the future work, we will address varying sensing ranges and investigate such cases. Moreover, the uniformity and scaling problem will be further studied.

Acknowledgement

This work was supported by grant No. R01-2005-000-10267-0 from Korea Science and Engineering Foundation in Ministry of Science and Technology.

References

1. S. Meguerdichian , F. Koushanfar, G. Qu and M. Potkonjak: Exposure in Wireless Ad-Hoc Sensor Networks. Mobicom (2001)
2. S. Dhillon, K. Chakrabarty and S. Iyengar: Sensor placement for grid coverage under imprecise detections. Proc. International Conference on Information Fusion (2002)
3. T. Clouqueur, V. Phipatanasuphorn, P. Ramanathan and K. k. Saluja: Sensor Deployment Strategy for Target Detection. WSNA, (2002)
4. Sameer Tilak , Nael B. AbuGhazaleh, and Wendi Heinzelman: Infrastructure Tradeoffs for Sensor Networks.WSNA (2002)
5. A. Howard, M. J. Mataric and G. S. Sukhatme: An Incremental Self-Deployment Algorithm for Mobile Sensor Networks. Autonomous Robots, Special Issue on Intelligent Embedded Systems, September (2002)
6. J. Wu and S. Wang: Smart: A scan-based movement-assisted deployment method in wireless sensor networks. Proc. IEEE INFOCOM Conference, Miami, March (2005)
7. G. Wang, G. Cao, and T. La Porta: Movement-assisted sensor deployment. Proc. IEEE INFOCOM Conference, Hong Kong (2004)
8. Y. Zou and K. Chakrabarty: Sensor deployment and target localization based on virtual forces. Proc. IEEE INFOCOM Conference, Vol. 2 (2003) 1293-1303
9. Nojeong Heo and Pramod K. Varshney: Energy-Efficient Deployment of Intelligent Mobile Sensor Networks. IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems And Humans, Vol. 35, No. 1 (2005) 78 - 92
10. Xiaoling Wu, Shu Lei, Yang Jie, Xu Hui, Jinsung Cho and Sungyoung Lee: Swarm Based Sensor Deployment Optimization in Ad hoc Sensor Networks. Proc. of ICESS' 05/ LNCS, Xi'an, China, (2005) 533-541
11. http://www.darpa.mil/ato/programs/shm/index.html
12. Sriram Chellappan, Xiaole Bai, Bin Ma and Dong Xuan: Mobility Limited Flip-based Sensor Networks Deployment. Dept of Computer Science and Eng, Ohio-State Univ. Technique report (2005)
13. Sriram Chellappan, Xiaole Bai, Bin Ma, and Dong Xuan: Sensor Networks Deployment Using Flip-based Sensors. Proc. of IEEE International Conference MASS'05 (2005)
14. Radu Stoleru, Tian He, John A. Stankovic, David Luebke: A High-Accuracy, Low-Cost Localization System for Wireless Sensor Networks. ACM conference Sensys (2005)
15. Wendi B. Heinzelman, Anantha P. Chandrakasan, and Hari Balakrishnan: An Application-Specific Protocol Architecture for Wireless Microsensor Networks. IEEE Transactions on Wireless Communications, Vol. 1, No. 4 (2002) 660 - 670
16. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein: Introduction to Algorithms. 2nd Edition. MIT Press and McGraw-Hill (2001) 595–601

Energy-Efficient Data Dissemination in Wireless Sensor Networks

JiHan Jiang[1], KuoHua Kao[2], and SingLing Lee[3]

[1] Department of Computer Science and Information Engineering
National Formosa University, Yun-Lin, Taiwan
jhjiang@nfu.edu.tw
[2] Department of Communications Engineering National Chung Cheng University,
Chia-Yi, Taiwan
geoge.kao@gmail.com
[3] Department of Computer Science and Information Engineering National Chung Cheng
University, Chia-Yi, Taiwan
singling@cs.ccu.edu.tw

Abstract. In this paper, a novel data dissemination scheme in wireless sensor network is proposed. A global location-based structure called *transfer posts* are adopted to act as the data forwarding stations among sink nodes and source nodes with mobility. Multiple source nodes share the transfer posts. The sink nodes can easily request or collect data through transfer posts in which store the location information of source nodes. Evaluation results show that our approach consumes less energy compared to recently results such as two-tier data dissemination (TTDD) [3] and Railroad [6]. We have resolved the problem of query forwarding and data delivering among mobile sinks and multiple sources with the remarkable performance.

1 Introduction

Wireless sensor network is a multi-hop ad hoc wireless network consisting of large amount of sensor nodes deployed randomly. The sensor nodes are used to monitoring the phenomena in the network, collect interested data, and forward them toward sink nodes that request this data. Each sensor node is stationary and is equipped with a sensing device, a limited built-in battery and a short-range wireless transceiver. Some algorithms have been proposed to implement energy efficient communication protocols in wireless sensor network [2, 4, 7, 8].

Mobility support brings new challenge to wireless sensor network [1, 3, 4, 6]. Luo et al. [3] proposed the TTDD method to build a grid structure according to each source location to handle the requests of multiple mobile sink nodes. However, this method consumes too much energy on building a grid structure for each source node and on executing local query flooding for sink nodes' requests. There is another approach recently, the Railroad [6] utilizes a virtual infrastructure called rail, which is placed in the middle area of the networks to communicate among sensor nodes, sink nodes and source nodes. However it is inefficient in some situation where the query request has to transmit around the whole rail until it reaches a node that has relevant data.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 565 – 575, 2006.

In this paper, a new infrastructure called *transfer posts* are proposed for data dissemination. It is global location-based structure that acts as the data forwarding stations among sink nodes and source nodes with mobility. Transfer posts are shared by multiple source nodes and use to store the location information of source nodes. For analysis convenient, in this paper the transfer posts are organized as a grid structure shared by all source nodes to reduce energy consumption on building infrastructure. Our method is more energy-efficient by using the transfer posts to resolve the problem for query forwarding of sinks and data delivering of sources. Furthermore, it also avoids local query flooding as well.

Our approach can be applicable to a wildlife park, battlefield, etc. For example, tourists or zoologists (sink nodes) with a mobile device (for example a PDA) observe the behavior of lion (source nodes) through wireless sensor network in a wildlife park. Figure 1 shows this example.

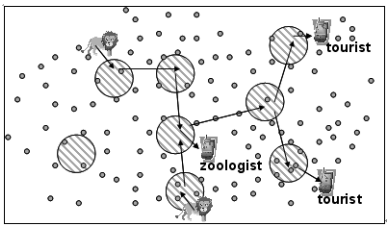


Fig. 1. A wireless sensor network example

The rest of this paper is organized as follows. Related works are shown in Section 2. Section 3 describes the main concept of our scheme including grid construction, query forwarding, data delivery and mobility support. The communication cost of our method and other approaches are compared in Section 4. Section 5 concludes our work.

2 Related Works

In wireless sensor network, the global flooding and frequently location updating of mobile sinks are energy consuming. Thus, recently, extensive researches dedicated to the study of energy efficient protocols for data dissemination [1, 3, 4, 6].

Visvanathan et al. proposed a hierarchical scheme (HDDS) [1] for large-scale wireless sensor networks (WSN). HDDS routes data towards sinks using a hierarchy of randomly selected dissemination nodes. Because dissemination nodes have limited resources, whenever a dissemination node is overloaded, it inserts another level of dissemination nodes to reduce its load. HDDS reduces energy consumption in WSN that dynamically adapt to demands for sensor data.

In [3], TTDD grid structures are created according to the location of the source node to prevent global flooding and frequent location updates. The grid structure supports mobility of the sink nodes. Query and data are transmitted along the grids and flooding is restricted within the local grids only. However grid construction for each source node and local query flooding also consume great energy. Furthermore, TTDD does not support source mobility.

Shin et al. proposed an approach named Railroad [6]. It defines data dissemination structure for large-scale WSN. Railroad adopts a virtual infrastructure called *rail*, which is placed in the middle area of the networks so that every node can easily access it. There is only one rail in the network and it acts as a rendezvous area of the events and the queries. Rail communicates among nodes, sinks and sources, and mobility of source nodes are also supported by the Railroad.

3 Our Method

The main goal in our scheme is to reduce energy consumption and support good mobility for source nodes and sink nodes. Our scheme is based on the following assumptions:

- Sensor nodes are uniformly distributed in the wireless sensor networks.
- All the sensor nodes in the field are homogeneous and have a constrained energy. Sensor nodes remain stationary after being deployed.
- Sensor nodes deliver packets to destination hop by hop by simple greedy forwarding algorithm.
- Each sensor node is aware of its geographic location (for example through receiving GPS signals)
- There are multiple sinks and multiple sources moving around in the sensor field.
- For analysis convenient, in the following the transfer posts are organized as a grid structure.

We found the TTDD consumes a considerable energy on grid construction and local query flooding. By using a global grid structure, our method will reduce the energy consumption of grid construction and will avoid local query flooding. Furthermore, the query forwarding in the Railroad system is inefficient. A query request might have to travel the whole rail until it reaches a station that stored the relevant data. We will solve this problem by using the *immediate transfer post*, which is the transfer post in the same grid with sink node or source node. An example of our scheme is shown in Figure 2.

Once the sensor nodes are deployed, the grid construction process is executed. Then every sensor node is aware of its own grid region and immediate transfer post. Immediate transfer post plays an important role in our method, it directly reduces the energy consumption by avoiding query flooding when sink nodes query. Immediate transfer post also simplifies the communication among source nodes and sink nodes. As soon as a source node generates data, it starts preparing for data dissemination by informing all transfer posts in the sensor field using geocasting with its geographic location. When a sink node needs data, it selects a neighbor as its *mobile agent*. The mobile agent sends a data-query packet to the immediate transfer post, which then propagates the query through other transfer post toward the source. Requested data will be forwarded in the reverse path from the source to the sink. The following subsections will describe our method step by step.

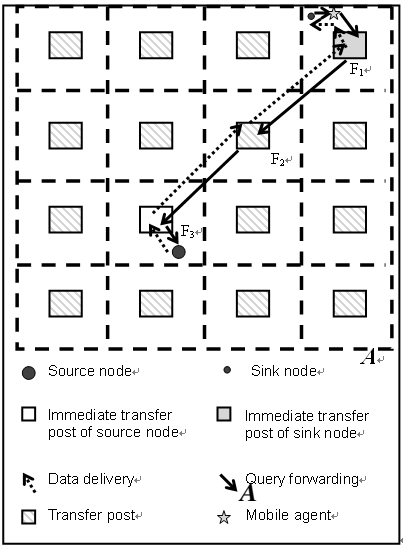


Fig. 2. An example of our scheme in the wireless sensor network

3.1 Grid Construction

Grid construction is executed only once after the sensor nodes are deployed. We assume that a sensor field spans a two-dimensional plane. We divide the sensor field into grids. Each grid is an $\alpha \times \alpha$ square. For a sensor node s at location (x, y), it can be aware of that it is in the grid g_{ij}, where $\{i=x/\alpha, j=y/\alpha\}$. The immediate transfer post of s can be get by the rectangular region of coordinates of the bottom-left corner

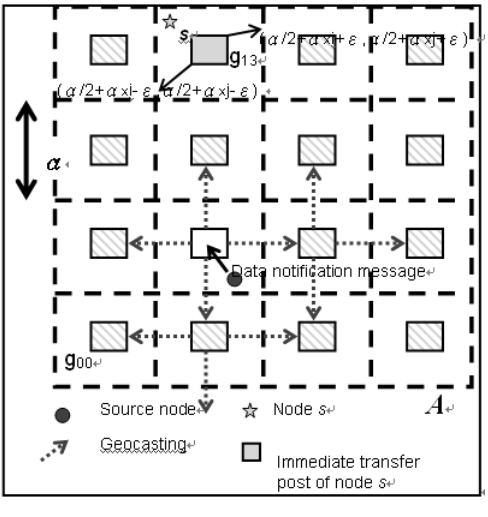


Fig. 3. An example of grid construction and source notification

($\alpha/2+\alpha*i-\varepsilon$, $\alpha/2+\alpha*j-\varepsilon$) and top-right corner ($\alpha/2+\alpha*i+\varepsilon$, $\alpha/2+\alpha*j+\varepsilon$), where ε is a positive real number set by the system. Figure 3 shows the grid construction.

The size of transfer post directly affects the network performance. The number of nodes in transfer post decreases, as the transfer post gets smaller. Since there are few nodes in transfer post, they have to deal with more operations, such as query forwarding and data delivering. They will consume much more energy than other nodes outside the transfer post and die first. It leads to the hot spot problem and loss of connectivity of the network. On the contrary, if the size of transfer post is too big, the source notification process consume more energy while geocasting information of source nodes to these fields. These geocast messages consume a considerable energy such that the network lifetime decreases.

3.2 Source Notification

When a source generates data, it will inform its immediate transfer post with a data-notification message, which includes data type, its location and other information. Immediate transfer post calculates the transfer post of its four neighboring grid and forwards the data-notification message to these fields by geocasting. The neighbor transfer post continues propagating the data-notification message in a similar way. Nodes in the transfer post store the information included in the data-notification message. In our scheme, multiple source nodes share the same global grid structure. Figure 3 shows the source notification.

3.3 Query Forwarding

Once a sink begins to collect data, it will select a neighbor sensor node as its mobile agent at first. Then the sink sends out the data-query message to its mobile agent, thus mobile agent forwards the message to sink's immediate transfer post. Since all the transfer posts are aware of that where the source node is, while the sensor node in the immediate transfer post receiving the data-query message, it can forward the data-query message to the source node easily.

When a transfer post forwards data-query message to the source node, it utilizes a virtual straight line connecting the source node and transfer post itself. Then it selects the next transfer post from its neighbor transfer posts that its distance with the straight line is the smallest. Every transfer post forwards the message to the source node in the similar way till the message reaches the immediate transfer post of source node.

During the query forwarding stage, once the transfer post receives the message from upstream transfer post, it stores the location of upstream transfer post. An example is shown in figure 2, transfer post F_2 stores the location of F_1 and transfer post F_3 stores the location of F_1.

3.4 Data Delivery

Since the transfer post has stored the location of its upstream transfer post, source node can easily deliver the data message according to the location of upstream transfer post toward the sink node after it received the data-query message. An example is shown in figure 2, source node forwards data messages through F_3-F_2-F_1 to the sink node.

3.5 Mobility Support

3.5.1 Sink Mobility

As mentioned earlier, the sink node selects a neighbor as its mobile agent. A mobile sink node only has to report its location to its mobile agent periodically. The mobile sink node's immediate transfer post sends data message to the mobile agent, which in turn relays data message to the sink. As the sink node moves, no new mobile agent is chosen until the sink node moves out of the range of its original grid. Once the sink node leaves the original grid, it will select a neighbor as its new mobile agent and inform its original mobile agent that which grid it enters now and location of its new mobile agent. Then the original mobile agent sends a location update message includes the information of which grid the sink node is in now and location of sink node's new mobile agent to its immediate transfer post. Immediate transfer post utilizes a virtual right-angled triangle separated by middle-line which is connecting the upstream transfer post and immediate transfer post itself into two, each one is 45 degree. Then it checks whether the new immediate transfer post is in the range of the virtual right-angled triangle. If the new immediate transfer post is in the range of the virtual right-angled triangle, the original immediate transfer post will inform the upstream transfer post that sink node has moved to another grid.

Then the upstream transfer post will forward the data message to the new immediate transfer post such that sink node can keep receiving data message. Figure 4 and 5

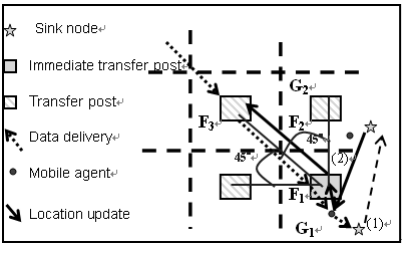


Fig. 4. (1) Sink node moves out of original grid G_1 to grid G_2 and (2) informs its mobile agent and selects a new mobile agent. Mobile agent sends location update to its immediate transfer post F_1 and upstream transfer post F_2.

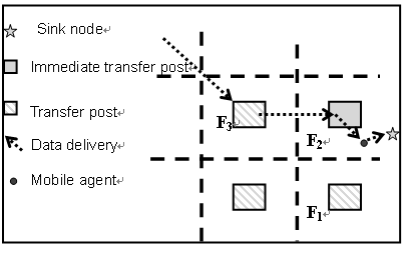


Fig. 5. Upstream transfer post F_3 forwards data messages to transfer post F_2 so that sink node is able to keep receiving data messages

shows the operation of mobility support of the first case. Otherwise, the original immediate transfer post will forward the data message to the new immediate transfer post directly. Figure 6 and 7 shows the operation of mobility support of the second case.

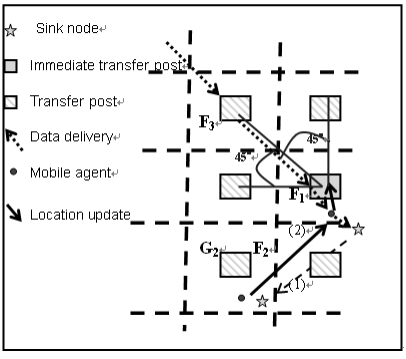


Fig. 6. (1) Sink node moves out of original grid G_1 to grid G_2 and (2) informs its mobile agent and selects a new mobile agent. Mobile agent sends location update to its immediate transfer post F_1.

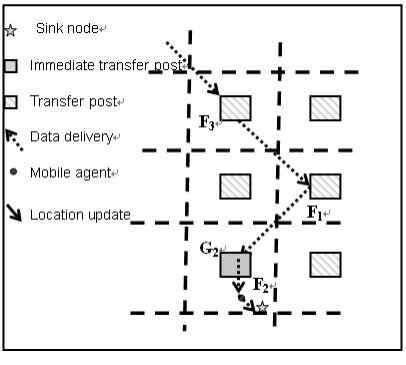


Fig. 7. The original immediate transfer post F_1 forwards data messages to the new immediate transfer post F_2 so that sink node is able to keep receiving data messages

3.5.2 Source Mobility

In our scheme, all the source nodes share the same global grid structure and inform all the grids when they begin to generate data. It would be inefficient that if we re-inform all the grids with the source node's location while source node moving. As a source node moves, it sends a location update message to its immediate transfer post without informing all the grids with its new location. We exploit immediate transfer post to represent the mobile source node. Once the immediate transfer post receives the data-query message, it forwards the data-query message to the mobile source node. As the source node moves, no source notification process is executed until the grid count between the source node and its immediate transfer post exceeds a threshold. The value of the threshold allows trade-off to be made between energy consumption on re-informing all the grids and energy consumption on the location update.

4 Performance Analysis

In this section, we analyze the communication cost of our scheme. The communication cost of a network is the total amount of traffic generated in the network. The total communication cost directly affects network lifetime. We compare our scheme with TTDD and Railroad.

4.1 Model and Notation

We assume that a square field of area A in which N sensor nodes are uniformly distributed. There are about $\sqrt{N}$ nodes on each side of the field. According to the analysis model of Railroad, we assume that there are four types of message: event notification (p_e), query (p_q), data (p_d), and control message (p_c).

There are m sinks and n sources in the sensor field. The total number of events and queries can be written as $n\bar{e}$ and $m\bar{q}$, where $\bar{e}$ is the average number of events and $\bar{q}$ is the average number of queries.

4.2 Communication Cost

We first analyze the worst-case communication cost of our scheme and compare the performance with those of TTDD and Railroad.

In our scheme, every source node informs all the grids in the sensor network with event notification messages by geocasting. The message meets about $\sqrt{2}/2 \cdot \sqrt{N_L}$ nodes until it starts geocasting in the immediate transfer post, where N_L is the number of sensor nodes in a grid in our scheme. N_F is the number of sensor nodes in the transfer post. Thus, the communication cost for all the sources notification can be written as $4n(N/N_L)(\frac{\sqrt{2}}{2}\sqrt{N_L}+N_F)P_e$.

The cost for the query to reach a source is $(c\cdot\sqrt{N})p_q$, where $c\cdot\sqrt{N}$ is the average number of sensor nodes along the straight-line path from the source to the sink $(0<c\le\sqrt{2})$. The communication cost of query forwarding can be written as $m\bar{q}(c\cdot\sqrt{N})p_q$. The cost for the data to reach a sink is $(c\cdot\sqrt{N})p_d$, where $c\cdot\sqrt{N}$ is the average number of sensor nodes along the straight-line path from the source to the sink $(0<c\le\sqrt{2})$. The communication cost of data transmitting can be written as $n\bar{e}(c\cdot\sqrt{N})p_d$. The total communication cost of our scheme is the sum of communication cost of sources notification, communication cost of query forwarding and communication cost of data transmitting $4n(N/N_L)(\frac{\sqrt{2}}{2}\sqrt{N_L}+N_F)P_e+m\bar{q}(c\cdot\sqrt{N})p_q+n\bar{e}(c\cdot\sqrt{N})p_d$. Communication cost of TTDD can be computed with a similar approach, that is $n\frac{4N}{\sqrt{N_L}}p_c+m\bar{q}\left[N_L+c\cdot\sqrt{2N}\right]p_q+n\bar{e}\left[c(\sqrt{2N}+\frac{1}{2}\sqrt{N_L})\right]p_d$. Communication cost of Railroad can be computed with a similar approach, that is $2m\bar{q}\sqrt{2N}p_q+\frac{1}{4}\left[n\bar{e}(p_e+p_q+4p_d)+m\bar{q}p_q\right]c\cdot\sqrt{N}$.

There are two simulations are performed. In our simulations, there are 10,000 sensor nodes uniformly distributed in the wireless sensor networks.

- SimA: there are 5 sink nodes, 10 source nodes and 100 events.
- SimB: there are 10 sink nodes, 5 source nodes and total number of queries 100.

Figure 8(a) shows the total communication cost of our scheme compared with TTDD and Railroad for SimA. We can observe that Railroad consumes less energy while the number of queries is small. Railroad consumes less energy on source notification such that it consumes less energy than our method before the number of queries increase to 35. As the number of queries increase, Railroad consumes more energy than our method because its inefficient query forwarding method. Figure 8(b) shows that our scheme consumes less energy than other approaches while query forwarding and data delivering. Obviously TTDD consumes more energy than other two approaches because of its grid construction and local flooding strategy.

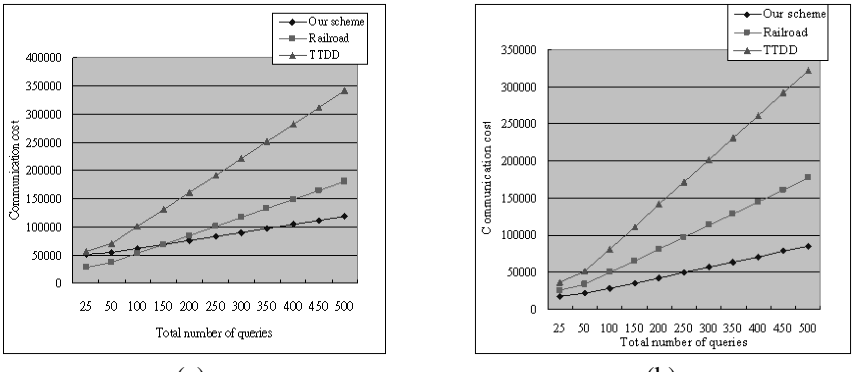


Fig. 8. SimA: the communication cost comparison of our scheme, TTDD and Railroad (a) The total communication cost. (b) The communication cost for query forwarding and data delivery.

Figure 10 shows the communication cost for SimB. Since the there are fewer source nodes than SimA, our method consumes relatively less energy on source notification. Obviously our scheme with fewer number of source nodes outperforms than the Railroad does.

Figure 11 and figure 12 show the effect of amount of sink nodes and source nodes, respectively. In figure 11, we can observe that communication cost increases slightly while the number of sink nodes increase since our method is energy-efficient while query forwarding and data delivering. Figure 12 shows that the communication cost increase conspicuously when the number of source node increase, because we utilize geocasting in the source notification stage.

Our scheme exploits a global grid structure to support multiple mobile sink nodes and multiple mobile source nodes. Although our scheme consumes more energy than Railroad during source notification stage, it is more energy-efficient for query forwarding and data delivering by utilizing the transfer posts. Accordingly, we can conclude that our scheme consumes less energy than TTDD and Railroad and increases network lifetime.

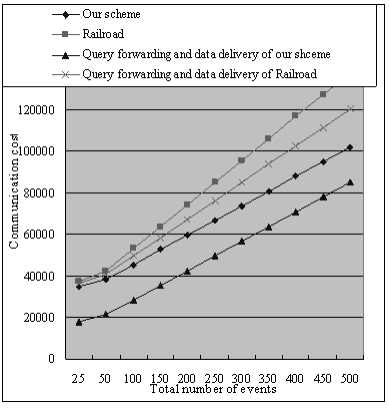


Fig. 9. SimB: the communication cost comparison of our scheme and Railroad. The total communication cost and communication cost for query forwarding and data delivery.

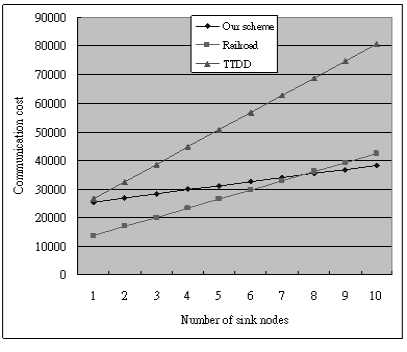


Fig. 10. Total communication cost comparison for number of sink nodes

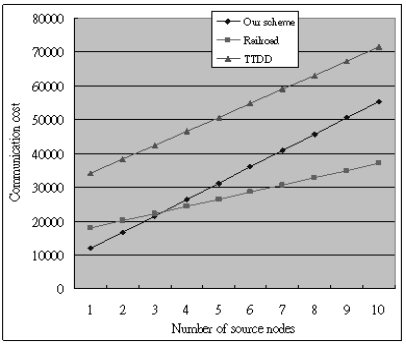


Fig. 11. Total communication cost comparison for number of source nodes

5 Conclusions

In this paper, an energy efficient data dissemination method in wireless sensor network is proposed. Our approach exploits a global location-based structure to support the mobility of sink nodes and source nodes. It provides energy efficient query forwarding and data delivery as well. Evaluation results show that our approach consumes less energy than TTDD and Railroad especially in query forwarding and data delivering.

In the future work, we will solve this problem for wireless sensor networks with other topologies or random deployment. Furthermore, we will consider the adaptability of the transfer posts to meet the practical situation that the wireless sensor networks.

References

1. A. Visvanathan, J. H. Youn, and J. Deogun.: Hierarchical Data Dissemination Scheme for Large-scale Sensor Networks. In Proceedings of the IEEE International Conference on Communications, Seoul, Korea, May 16-20, 2005, Vol. 5, pp. 3030-3036.
2. C. Intanagonwiwat, R. Govindan, and D. Estrin.: Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom 2000), 2000.
3. F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang.: A two-tier Data Dissemination Model for Large-scale Wireless Sensor Networks. In Proceedings of Mobile Computing and Networks (Mobicom 2002), Atlanta, Georgia, USA, 2002, pp. 148 - 159.
4. H. S. Kim, T. F. Abdelzaher, and W. H. Kwon.: Minimum Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks. In Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys 2003), Los Angeles, CA, Nov. 5–7, 2003, pp. 193–204.
5. J. Chen, Y. Guan and U. Pooch.: An Efficient Data Dissemination Method in Wireless Sensor Networks. In Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'04), Dallas, Texas, Nov. 29-Dec. 3, 2004, Vol.5 pp. 3200–3204.
6. J. H. Shin, J. Kim, K. Park and D. Park.: Railroad: Virtual Infrastructure for Data Dissemination in Wireless Sensor Networks. In the Proceedings of the 2nd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN'05), Montreal, Quebec, Canada, Oct. 2005, pp. 168 – 174.
7. T. Shu, M. Krunz and S. Vrudhula.: Power Balanced Coverage-time Optimization for Clustered Wireless Sensor Networks. In the Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'05), Urbana-Champaign, IL, USA, May 2005, pp. 111-120.
8. W. Heinzelman and A. Chandrakasan and H. Balakrishnan.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In the Proceedings of the Hawaii Conf. on System Sciences, Jan. 4-7, 2000.

Proposal of Visualization of Reasoning Processes in Sensor Network Environment

Naoki Matsushita[1], Takashi Yoshino[2],
Takashi Hattori[3], Kaoru Hiramatsu[3], and Takeshi Okadome[3]

[1] Graduate School of Systems Engineering, Wakayama University, Japan
[2] Faculty of Systems Engineering, Wakayama University, Japan
[3] NTT Communication Science Laboratories,
Nippon Telegraph and Telephone Corporation, Japan

Abstract. The sensor network environment will become the standard daily environment in the next generation. The technology of the sensor network can combine with computing and daily space. Combinations of computing and sensor networks happen only inside a computer. In other words, a user cannot understand what happens inside the computer. In particular, our system carries out reasoning. There is no research on the visualization of reasoning processes. The visualization of reasoning processes is a complex process. Therefore, we need to develop visualization of reasoning processes. The main technologies to visualize reasoning are to resolve motion and to use 3D space expression. This paper proposes a method of visualization of reasoning processes in a sensor network environment.

1 Introduction

Beginning in the 1990's, Mark Weiser published the concept of ubiquitous computing, and many researchers have since been trying to actualize the concept [1]. In recent years, our personal belongings can have embedded sensors, because sensors improve miniaturization, save electric power and receive wireless communications [2]. It is a major area of study how to provide various services to build a sensor network using these technologies. There will be numerous miniaturized sensors in the environment around us in the near future [3], which will help with human tasks. Identification (ID) systems using wireless communication is spreading. Some companies have adopted Radio Frequency Identification (RFID) systems for merchandise management and personnel security. Srivastava and his group have built smart environments. Their "Smart Kindergarten" is an example of using a sensor network. The environments have familiar physical objects with embedded sensors, cameras and microphones at kindergartens [4]. They tried to record the interactions of young children and their environments as toys, and use the interaction data for learning.

On the other hand, many researchers have studied the semantic web. Meaning process like building ontology is developed in the field of artificial intelligence.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 576–585, 2006.

Our study aims to deduce the roles and names of objects from behavior. We made the assumption that all artifacts have RFID, wireless communication device and versatility sensor nodes [5]. When a person substitutes the object for something, the computer deduces that the role of the object is "a role that is not original". For instance, a desk is a piece of furniture for writing and drawing. However, if a person sits on the desk, the role of the desk is "chair" in that instant. We want to find the role, because we consider that computers recommend the substitute object for supporting human tasks. However, we cannot know how to change sensor values and how to deduce the role of the object, while computers process. In addition, we consider that applications that use an inference engine will not understand how to deduce, because applications will apply some sensor data and results of reasoning. Therefore, this paper presents one way to visualize the change of sensor values and the process of reasoning.

2 The Need for Visualization

We envision environments in which our personal belongings will have sensor nodes. It is important that users can know the situations of the system, because environments include the home and office.

Reasoning is a task in a computer. The computer sends sensor data to databases. Applications can read these data that are strings or digit sequences from databases. However, it is too difficult for users and application developers to understand the meaning of the data from string or sequence data. Therefore, we suggest two kinds of visualization to assist users and developers.

A. Dynamic Visualization of Reasoning Process: A purpose of this visualization is for users to draw an image of the way of deduction, to comprehend the transition of deduction. Moreover, developers of inference engines can check extraordinary reasoning to use this application for system tuning.

B. Spatial Correspondence Visualization of Sensor data: This application can show the transition of sensor values. The visualization system retrieves sensor data from the sensors directly or from databases. Users or developers can check whether RFIDs are broken or not using this application. Moreover, users can know about the changing of the room from sensors, because the system builds a virtual world from sensor data in the real world. Therefore, users can understand different points from the real world and events that sensors catch.

3 Outline of Inference Engine

The purpose of the inference engine is name deduction for new objects coming into the room, finding new ways that the object do not have their original purpose, and event deduction in the room.

We built a smart room that assumes an office for role reasoning. In this room, there are doors, a table, chairs, books and so on. All have sensor nodes. A sensor

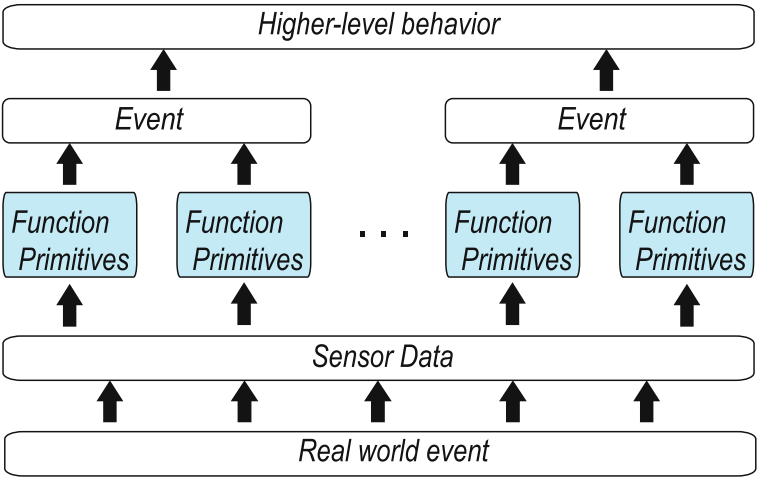


Fig. 1. The process of the system

node consists of a memory, a CPU, a communication device and five sensors. The sensors are an acceleration sensor(X, Y, Z-axis), a temperature sensor, a humidity sensor, an illumination sensor and a Pyroelectricity sensor. Figure 1 shows the processes of the inference engine. The following are the three stages of deduction:

First stage: Sensor data. The situation around an object changes by touching the object or moving other objects. The embedded sensors in the environment and the object can detect the change. The sensor node sends the changing data instantly. "(B) Spatial Correspondence Visualization of Sensor data" visualizes this stage.

Second stage: Function Primitives. We describe object motion using "Function Primitives". Function Primitives is object moving and the relation between an object and another object. Function Primitives has preference from functional definitions in value engineering and Therblig motions in industrial engineering. There are 19 kinds of Function Primitives (Table2). These are distributed into two major compartments "State"(e.g. Move, Contact-Press, Restrict-Blockade and so on) and "State Variation"(e.g. Assemble-Lock in, Disassemble, Contact-Relax and so on). We tried to describe 100 artifacts in WordNet using Function Primitives , and ninety percent in it can be depicted. WordNet is a web dictionary [6].

Third stage: Event. We consider that the inference engine can deduce what events occur using ontology. Ontology defines the relation between an object and other objects. Events can be described combining Function Primitives.

"(A)Dynamic Visualization of Reasoning Process" visualizes Function Primitives stage and Event stage. We consider that express higher-level behavior to combine the events.

			Function Primitves	Meaning	Expression
State	1		Move (oneself)	A subject moves by itself.	He main character moves.
	2		Transport others	A subject transports a target.	The main character carries another character.
	3		Transform (others)	A subject transforms a target.	The main character touches another character, and the color is changed.
	4		Emit (heat, light, etc)	A subject emits something energetic.	The main character emits many spheres.
	5		Absorb (heat, light, etc)	A subject absorbs something energetic.	The main character absorbs many spheres.
	Contact			A subject contacts a target directly.	
		6	Twist	A subject torques a target.	The main character rotates another character.
		7	Press	A subject applies pressure.	The main character pushes another character.
		8	Support	A subject supports the weight of a target.	He main character lifts another character.
	Restrict			A subject restricts something.	
		9	Isolate	A subject shields energy propagation.	The main character catches spheres from ahead.
		10	Guide	A subject leads to move another target.	The main character pulls another character.
		11	Blockade	A subject cuts off a target.	The main character blocks another character.
	12		Stand by	A subject does nothing at all.	The main character comes to rest.
Change in state	Assemble(two artifacts)			A subject joins two targets.	
		13	Lock in	Joining is enduring.	Two characters are united.
		14	Connect	Joining is temporary.	Two characters are touched.
	15		Disassemble	A subject takes apart a target.	One character is resolved to two characters.
	16		Release	A subject releases a target.	Two characters are separated.
	Contact			A subject interacts with a target.	
		17	Apply Force	A subject begins to apply force to a target.	The main character begins to push another character.
		18	Relax	A subject finishes applying force to a target.	The main character stops pushing another character.
		19	Posses	A subject holds a target.	The main character begins to hold another character.

Fig. 2. Function Primitives and their expression

4 Dynamic Visualization of Reasoning Process

First of all, we visualized "Function Primitives", and next we tried to express other reasoning process. This paper explains "Function Primitives". It is difficult for users to viscerally understand reasoning that expresses words or a scheme. We try to express dynamically expression. A purpose of the expression is that users can understand reasoning easily. A problem of the deduced expression is

Character:Each object is expressed as a character.

"?"tail:When the inference engine decides the object, the visualization system changes "?" to the object.

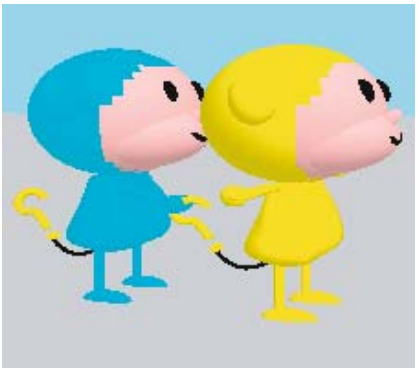


Fig. 3. Characters of the system

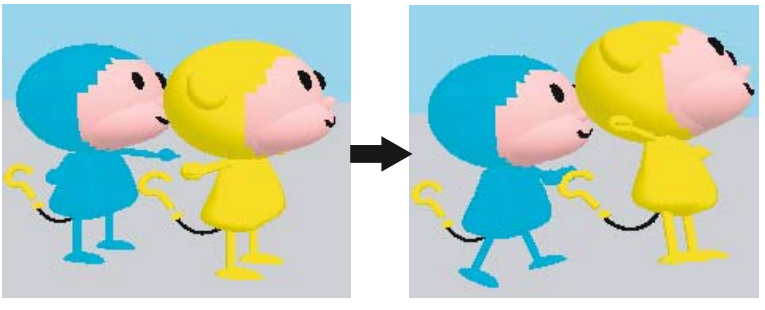


Fig. 4. "Transport" Expression

that nobody knows what object is deduced by the inference engine. It means that a sensor node catches the changing of objects' situation, but the inference engine does not know what objects have a sensor node. We decided upon two policies.

a. Using object abstraction. It is impossible to express something no one knows as it is. Thus, we express abstract entities. We use a character that has a head, arms, legs, and a tail, showing the metaphor as a personification . The reason for using a metaphor is that the user feels a sense of affinity easily[7]. The concept of the character is "thinking", therefore the figure of the tail is "?". This character can express action like a human model (Figure 3).

b. Action of "Function Primitives". We express "Function Primitives" using character animation. When a "Function" depicts the relation of an object to other objects, the visualization system displays the same figures but different colored characters. Several expressions of the "Function Primitives" are explained as follows:

(1) "Transport": The main character lifts another character and moves (Figure 4).
(2) "Lock in": The main character has the arms of two characters with both left and right hand, and the main character has to put one character's arm on another character's shoulder. To unite two objects it is expressed that characters are done not only by unit, but also expressed by color mixture of these surfaces(Figure 5).
(3) "Emit" and "Absorb": These mean that the main character emits (absorbs) energy as heat or light. We draw these functions to emit (absorb) numerous small balls. We think these expressions are useful to understand reasoning by instinct.

We consider that a combination of "Function Primitives" can describe an "Event".

When the inference engine understands an object's name, the visualization system replaces "?" of the tail by an object 3D model. It is necessary to proceed with the visualization of inference and for a user to know what object is deduced.

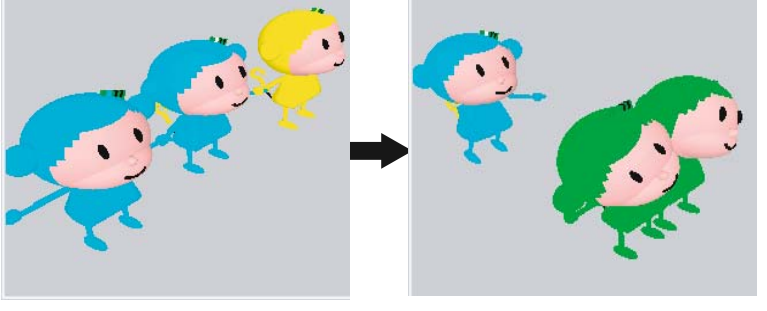


Fig. 5. "Lock in" Expression

5 Spatial Correspondence Visualization of Sensor data

We visualized "First stage: Sensor data" which is a part of the inference engine. It is necessary for a user to know sensor information.

When all artifacts have a sensor node-like bar-code, the number of nodes becomes huge. Therefore we use a virtual 3D space, because a 3D space can be more expressive than a 2D space. Moreover, a 3D space corresponds with the real world. When an object overlaps another object, the user can view it to rotate view point in the virtual world. Users can see the inside walls but cannot see outside. When view location is out of the room, users can see the wall of back but cannot see the wall of forth.

We built a virtual room like a real laboratory and set up several objects (e.g. furniture) in the virtual room (Figure 6). The reasoning computer knows these objects. It means that the computer finishes reasoning or the computer

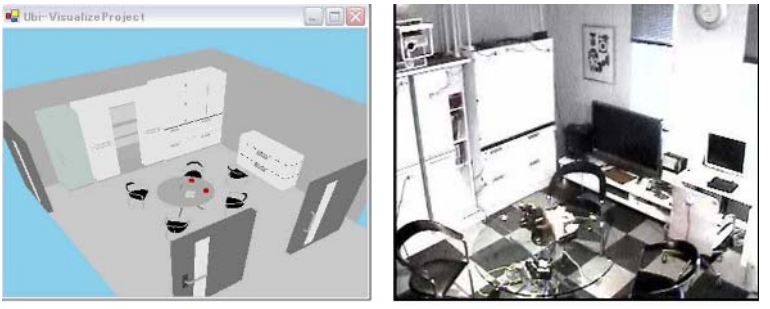


Fig. 6. Virtual sensor room and Real sensor room

gets information about these objects from the Internet. However, there are still unknown objects. The computer has not finished reasoning them yet. We set up the sphere as an abstract object. We reflect 3 kinds of sensor data, acceleration sensor data, temperature sensor data and pyroelectricity sensor data.

Acceleration sensor: The visualization computer calculates the moving distance from the data of acceleration sensors, and expresses movement.

Temperature sensor: The visualization system expresses the value of the temperature to change the color of the surface like an infrared thermograph. When it is hot, the color becomes redder, and when it is cold, the color becomes bluer.

Pyroelectricity sensor: This sensor outputs 2 values, true or false. The sensor has directivity. When the sensor data is true, we express the range using a square pyramid.

Pressure sensor: We lined the floor with the sensors. The size of each sensor is 18 square centimeters. Each sensor returns true or false according to the situation. In the virtual room, we divide the virtual floor into the same number of pressure sensors. When sensors respond, the color of a part of the floor in the virtual room changes.

We guess the position of the person according to pyroelectricity sensors and pressure sensors. When a person walks in the room, pressure sensors respond at the pace of his/her walking and his/her length of stride. Therefore, we guess that when a pressure sensor responds and the pressure sensor exists in a useful range of pyroelectricity sensors, there is a person on the respond sensor.

We explain the situation in which a person comes into the room and then sits on a chair. The color of the furniture in the virtual room is at the red or blue end of the spectrum with the temperature. When he/she opens a door, the virtual door moves. When he/she walks on the floor, the color of part of the virtual floor changes. When he/she walks front of a piece of furniture, the virtual furniture flashes on and off. When the position of the person is decided from sensors, a person shaped fog rises in the virtual room. When he sits on a chair, the virtual chair flashes on and off due to a pyroelectricity sensor. At this moment, the chair

and furniture that look toward him are flashing. The color of the floor at the points of his legs and the legs of the chair change.

6 Speculation Experiment of Spatial Correspondence Visualization

6.1 Experimental Methodology

The purpose of this experiment is research to determine whether ordinary people can speculate from spatial correspondence visualization or not. We reflect sensor data of real events to the virtual room. The event takes about 40 seconds in the real room. The following 10 things occur:

(1) A person enters the smart room. (1sec)
(2) The person walks to a cabinet. (1-3sec)
(3) The person takes a book from the cabinet. (4-5sec)
(4) The person walks to the center table. (6-7sec)
(5) The person sits on chair A. (8-9sec)
(6) The person reads the book. (10-29sec)
(7) The person closes the book and stands up. (30sec)
(8) The person walks to a desk. (31-34sec)
(9) The person sits on chair B. (35sec)
(10) The person drafts the chair and sitting. (36-40sec)

The virtual room describes three kinds of sensors. These are temperature sensors, pyroelectricity sensors and pressure sensors. Moreover, we guess the position of the person according to pyroelectrictiy sensors and pressure sensors, and draw a humanoid 3D model at the position. The book did not have a sensor node. Examinees were university students ranging from 18 to 21 years of age.

The experimental procedure was as follows:

(i) Listen to an explanation about the smart room.(room layout, functions of the sensors node)
(ii) Watch a movie in which the virtual room reflects sensor data
(iii) Guess and Write down what is happening in the smart room.
(iv) Watch a movie in which a camera shot the real room.
(v) Write down different points from the guess of the virtual room and the real camera movie.

6.2 Appraisal Method

We evaluated examinees' guesses about 10 things in the room from description sheets. Each of the 10 things were checked on a 7-point scale of difficulty . (1:The guess is wrong, 2:The guess is more or less wrong, 3:The guess is nearly wrong, 4: Neither can be said, 5:The guess is nearly correct, 6:The guess is more or less correct , 7:The guess is correct) Estimators were 3 graduate school students who were not examinees.

Table 1. Evaluation result

	A	B	C	D	E	F	G	H	I	J
(1) A person enters the smart room	6.7	6.7	6.0	6.7	6.7	6.7	6.7	6.7	6.7	6.5
(2) The person walks to a cabinet	4.0	-	6.7	1.5	-	-	-	-	6.3	6.5
(3) The person takes a book from the cabinet	-	1.0	-	-	1.0	-	1.0	-	6.3	6.7
(4) The person walks to the center table	5.0	5.3	4.7	4.5	5.5	6.0	6.5	6.7	6.0	7.0
(5) The person sits on chair A	2.0	6.3	1.0	-	4.0	-	6.7	-	6.0	6.7
(6) The person reads the book	-	-	-	-	-	2.5	2.0	3.0	4.0	-
(7) The person closes the book and stands up	-	5.3	-	-	-	-	-	-	-	-
(8) The person walks to a desk	-	6.3	-	-	-	6.3	6.7	6.5	5.7	6.7
(9) The person sits on chair B	-	1.0	-	-	7.0	-	6.7	-	1.0	6.7
(10) The person drafts the chair and sitting	-	1.0	-	-	-	-	6.5	-	-	-

It is not necessarily the case that examinees wrote clearly about all things, so estimators had to evaluate context. For instance, if an examinee wrote "The person sits on a chair, after that goes to the desk", the estimators can consider that there is "stand" between "sit" and "go". About the different points in experimental procedure (v), the estimators' judged that the examinees could not guess the thing, therefore estimators checked this point as 1on the 7-point scale.

6.3 Evaluation Results

Table 1 shows the evaluation results. A-J are examinees. Each value is an average amount when two or three estimators can evaluate. "-"is when nobody or one estimator can evaluate.

It is easy to guess the position of the person as "enter the smart room", "walk to the table" and "walk to the desk". We consider that this causes drawing the humanoid model at the position. About the half examinees could guess "sit on a chair". When the humanoid model overlaps a chair, it is easy to interpret it as "sit on a chair". When the person stands in front of the cabinet, most people cannot guess that the person is doing something.

It is difficult to guess what the person is doing according to their position. At a later stage, we need to show objects' behavior in the virtual room.

7 Conclusion

This paper presented visualization of both sensor data and reasoning processes in the study of reasoning an object name and object role using a sensor network.

Dynamic visualization of the reasoning process can express Function Primitives using characters.

Spatial correspondence visualization of sensor data can express moving objects and changing color. The temperature sensor depicts the surface color changes and the pyroelectricity sensor can describe the range. We draw a humanoid model in the virtual room.

It is necessary to show moisture sensors and illuminance sensors from this time. Moreover, concerning the showing of reasoning processes, a problem is how to combine character's actions in combining "Function Primitives" for reasoning an event.

References

1. Mark Weiser: The Computer for the 21st Century. Scientific American, 265(3) (1991) 94–104
2. Hans-W.Gellersen, Albrecht Schmidt, Michael Beigl: Multi-Sensor Context-Awareness in Mobile Devices and Smart Artefacts. Mobile Networks and Applications, 7(5) (2002) 341–351
3. Irfan A. Essa: Ubiquitous Sensing for Smart and Aware Environments: Technologies towards the building of an Aware Home. IEEE Personal Communications (2000) 47–79
4. Mani Srivastava, Richard Muntz, Miodrag Potkonjak: Smart Kindergatren: Sensor-based Wireless Networks for Smart Developmental Problem-solving Environments. Proceedings of the ACM SIGMOBILE Annual International Conference on Mobile Computing and Networking (MOBICOM) 2001 (2001) 132–138
5. Takeshi Okadome, Takashi Hattori, Kaoru Hiramatsu, Yutaka Yanagisawa, Tatsumi Yamada, Tetsuji Satoh: Pervasive Association: Semantic Integration for Ubiquitous Computing Environment(1) -Towards Semantic Sensor Web-. Multimedia, Distributed, Cooperative and Mobile Symposium(DICOMO) 2005 (2005) 165–168
6. Christiane Fellbaum: WordNet: An Electronic Lexical Database. MIT Press(1998)
7. Kusumi Takashi: The Role of Metaphors in User Interface Design: Going From the Desktop to Virtual Space and Returning to language. Special issue of Japanese society for the science of design, 10(1) (2002) 64–73

Energy-Efficient, Traffic-Adaptive, Fast Collision Resolution MAC for WSNs

Younggoo Kwon

Konkuk University, 1 Hwayang-dong, Kwangjin-gu, Seoul, 143-701, Korea
ygkwon@konkuk.ac.kr

Abstract. Development of energy-efficient, traffic-adaptive MAC algorithms that provide both high reliability and easy implementation property is the current major focus in wireless sensor network research. The operational characteristic of the station in packet transmission is completely different from those of deferring stations in steady state. In this paper, the fast collision resolution MAC algorithm is combined with traffic-aware queue status managements to achieve high energy efficiency and high performance at the same time. Through the implementations and various performance studies, the proposed algorithm shows significant performance improvements in wireless sensor networks.

1 Introduction

In many performance analysis papers for the binary exponential backoff based algorithms, the performance analysis starts from the assumption that all stations have the same average contention window range in steady state[1][2]. The same average contention window range for all active stations can be understood that all stations have the same average probability of packet transmission in steady state. The performance of many binary exponential backoff based MAC algorithms can be explained well by using the assumption that all stations have the same average contention window range in steady state. Furthermore, the optimum value which will minimize the wasting overheads during the contention procedure can be derived for a given number of active stations[3]. However, there are still the wasting overheads come from the inherent limitation of the assumption that all stations have the same contention window range for packet transmission in steady state. Wireless sensor networks are operated with limited battery power[7]. By considering the sporadic traffic pattern of WSNs, the actual duty cycle of each station is pretty small, which needs energy for operations[8][11]. Traffic information can be determined by checking the queue status of each sensor station which explicitly specify its traffic characteristics. Depending on the application at hand, the length of the active and sleep period can be dynamically assigned to support the traffic adaptive mechanism. In this paper, we use an efficient collision resolution algorithm which achieves highly efficient transmission procedure. This fast collision resolution MAC is designed with adaptive queue management mechanism which can save power consumptions by controlling the active period of each wireless sensor node. Through the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 586–594, 2006.

implementations and simulations, the proposed algorithm shows high network performance in energy consumptions and end to end latency.

In the next section, we explain related research works. Then, we present the newly proposed algorithm in Section 3. The performance evaluations is given in Section 4. In the final section, we present the conclusions.

2 Related Works

In IEEE 802.11 MAC, if a station has a packet to transmit, it will check the medium status by using the carrier sensing mechanism[5]. If the medium is idle, the transmission may proceed. If the medium is determined to be busy, the station will defer until the medium is determined to be idle for a distributed coordination function inter-frame space (DIFS) and the backoff procedure will be invoked. The station will set its backoff timer to a random backoff time based on the current contention window size. After DIFS idle time, the station performs the backoff procedure with the carrier sensing mechanism by determining whether there is any activity during each backoff slot. If the medium is determined to be idle during a particular backoff slot, then the backoff procedure will continue. If the medium is determined to be busy at any time during a backoff slot with a non-zero backoff timer, then the backoff procedure is suspended. Transmission will begin whenever the backoff timer reaches zero. If the transmission is successfully completed, the contention window (CW) for the source station will be reset to the initial (minimum) value minCW. If the transmission is not successfully completed, the contention window (CW) size will be increased, beginning with the initial value $minCW$, up to the maximum value $maxCW$. The basic channel access mechanism and packet transmission structure is shown in Figure 1.

As we can see from the Figure 1, at the end of the current packet transmission and a DCF InterFrame Space waiting time, each station will choose a random number for the backoff procedure. This random backoff time has a uniform distribution to provide equal fairness for all stations. A major deficiency of the CSMA based MAC protocol is that it is very slow to resolve collisions as the number of stations increases. Based on these observation, the aggregate network throughput can be derived as follows:

$$\rho = \frac{\bar{m}}{\frac{(1-p)}{Mp} t_{slot} + \frac{1-(1-p)^M - Mp(1-p)^{M-1}}{Mp(1-p)^{M-1}} [E[coll] + SIFS + ACK + DIFS] + E[S]} \tag{1}$$

where $\bar{m}$ is the average packet length, M is the number of active stations, $E[coll]$ is the average collision length, and $E[S]$ is the average time to complete a successful packet transmission without any collisions.

If we know the probability of a packet transmission p and the number of active stations M, then, we can calculate the network throughput from (1). If the number of active stations M is fixed and given, then we can obtain the optimal p_{min} value which maximizes the network throughput and the corresponding the

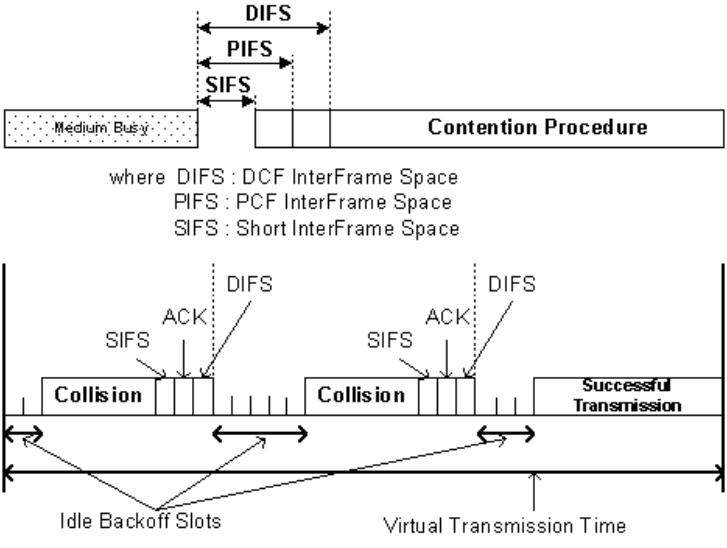


Fig. 1. Basic Channel Access Scheme and Packet Transmission Structure

average size of the contention window. The capacity of the MAC protocol with a proper backoff window tuned to the optimal p value for each M is improved significantly[3]. However, the p_{min} value, and hence the optimal contention window size of transmitting a packet, depends on both the number of active stations and the distribution of packet length.

The IEEE 802.15.4 standard defines beacon enabled mode and superframe structure for power saving purposes[6][7]. It can operate in either beacon enabled mode or beacon disabled mode. In beacon enabled mode, a network coordinator periodically broadcasts beacons so that other nodes in the network hear the beacons to synchronize to the superframe structure suggested by the coordinator. In beacon disabled mode, however, a network coordinator does not broadcast beacons except when other nodes request beacons for scanning or association purpose. In beacon enabled networks, a coordinator broadcasts beacons with superframe structure information recorded in the beacon. When other nodes in the network receive the beacon, they obtain the superframe information and start to synchronize to the coordinator's superframe structure. A superframe structure is defined by the network beacons. A network beacon marks the start of a superframe, while it also marks the end of previous superframe at the same time. A superframe generally consists of two parts - an active and an inactive part. The length of a superframe (beacon interval, BI) and its active part (superframe duration, SD) are determined by beacon order (BO) and superframe order (SO), respectively. BI can be calculated by using the equation

$$BI = aBaseSuperframeDuration \cdot 2^{BO} \tag{2}$$

and the time for superframe duration (SD) can be computed with

$$SD = aBaseSuperframeDuration \cdot 2^{SO} \tag{3}$$

The possible value for the two parameters BO and SO varies from 0 to 14 and must satisfy the following condition.

$$0 \leq SO \leq BO \leq 14 \tag{4}$$

Additionally, if both BO and SO are set to 15, IEEE 802.15.4 MAC will operate in beacon disabled mode. The length of inactive part can be determined by subtracting superframe duration from beacon interval. The active part is divided into 16 equally sized slots and has two periods - a contention access period (CAP) and an optional contention free period (CFP). During the CAP, IEEE 802.15.4 MAC utilizes slotted carrier sense multiple access with collision avoidance (CSMA-CA) mechanism for channel access. Following the CAP, CFP can be assigned for low latency applications or applications requiring specific data bandwidth. CFP may accommodate up to seven guaranteed time slots (GTSs), each of which may occupy one or more slots.

3 Energy-Efficient and Traffic-Adaptive Fast Collision Resolution MAC

We use a simple traffic adaptive, distributed scheme for energy-efficient channel access. It requires the information for queue status of each sensor station to expect the current traffic situations. Unlike previous attempts at achieving adaptive scheduling in sensor networks, this mechanism does not require explicit schedule announcements during scheduled access periods. Alternatively, application-specific traffic information is expected by checking the queue status which reflect the driving application's specific traffic patterns or network situations. This allows the proposed mechanism adapt to changes in traffic behavior and topology easily. We achieves traffic adaptiveness by assigning the length of active period to a station depending on the queue status of each station. The queue status has three different traffic threshold values which decide the length of the active period of the whole cycle. The information of the queue status of each station is transferred to the coordinator and the coordinator determine and broadcast the longest active period as the length of the active period for the next cycle. The net effect is that all stations have the same active period by considering the implementation simplicity. If we dynamically assign different active periods for each station, it may improve the network performance a little, but there is the trade off for the system complexity of network management to deliver the traffic information of each station.

The performance of channel access algorithms in distributed contention-based MAC protocols generally depends on two main factors: the probability of collision p_{coll} and the wasting backoff slots in the contention procedure. To design a good channel access algorithm, we should simultaneously decrease p_{coll} and the average number of the wasting backoff slots for each contention period. Fast collision resolution decrements the backoff time by $aSlotTime$, and when the backoff

time reaches to zero, a packet is transmitted. The outcome of a transmission can be either collision or successful transmission. When a collision occurs, the Contention Window size is increased by an Increasing Factor and a random Backoff Time (BT) for any station involving collisions is chosen. When a successful packet transmission occurs, the Contention Window size is reduced to the minimum contention window size $minCW$ for the successful transmitting station and a random Backoff Time (BT) value is chosen. All other deferred stations will increase their Contention Window size by a factor of IF and a random Backoff Time (BT) is chosen whenever a collision or a successful packet transmission occurs. If a station consecutively succeeds in packet transmissions over a predetermined transmission limit, it will be assigned the maximum contention window size $maxCW$ to invite other stations for packet transmissions. All active stations will monitor the medium. If a station senses the medium idle for a slot, then it will decrement its backoff time (BT) by a slot time, i.e., $BT_{new} = BT_{old} - aSlotTime$. When its backoff timer reaches zero, the station will transmit a packet. If there are $[(minCW + 1) \times 2 - 1]$ consecutive idle slots being detected, its backoff timer should be decreased much faster. The net effect is that after the random backoff timer is chosen from the large contention window range, it will be realigned to reduce the unnecessary wasted idle backoff time. After the random backoff timer is chosen from the large contention window range, it will be realigned to reduce the unnecessary wasted idle backoff time. If a station, which has just performed a successful packet transmission, runs out of packets for transmission or reaches its maximum successive packet transmission limit, then, all stations may have large contention window ranges. The concept of the backoff timer realignment is in Figure 2.

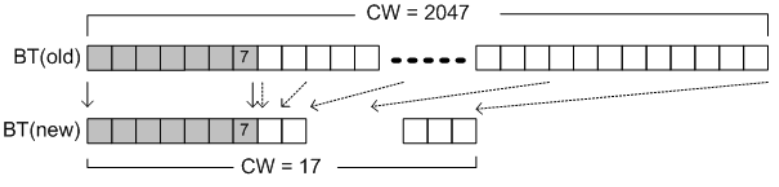


Fig. 2. Backoff Realignment

4 Performance Evaluations

In this section, we present the performance results for the proposed algorithm and the IEEE802.15.4 power-saving MAC algorithm. We use Chipcon 2420DBK as our development platform and testbed. The implementation board are running on IEEE802.15.4 PHY/MAC features with Atmel AVR ATmega128L microcontroller[13]. The radio provides a bandwidth of 250 kb/s. It has three operational modes: receiving, transmitting and sleep, consuming 37.6 mW, 22.0 mW, and 0.852 mW, respectively. There is no power difference between listening and receiving. We implemented three MAC algorithms: 1) IEEE802.15.4 with no sleep

cycle; 2) IEEE802.15.4 with 10% duty cycle; 3) The proposed MAC algorithm. The whole frame length(the beacon interval in IEEE802.15.4) is 1 s. The active period can be changed to reflect different duty cycles. The goal of the experimentation is to reveal the fundamental tradeoffs of energy consumptions and end to end latency for three different MAC algorithms. To facilitate the measurement of multiple messages traveling through a multihop network, we add a message queue at the application layer to buffer the outgoing message on each station. The seven multihop stations are linearly distributed and the CBR generator is used for traffic generation by generating packets of 94 byte length.

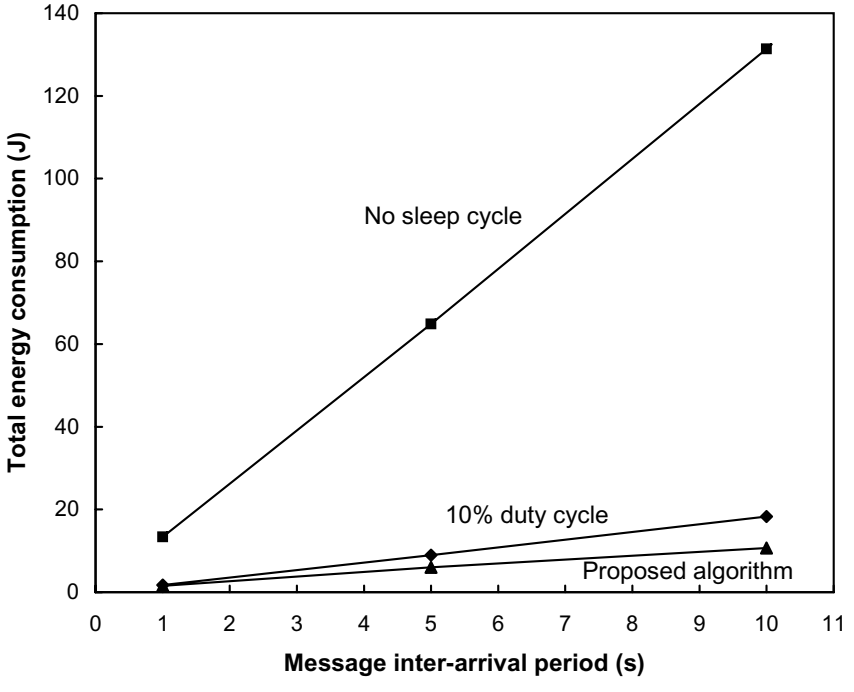


Fig. 3. Energy Consumption

To measure the energy consumption on the radio, we measure the amount of time that the radio on each station has spent in different modes: sleep, idle, receiving or transmitting. The energy consumption in each mode is then calculated by multiplying the time with the required power to operate the radio in that mode. 50 CBR packets from source station flow through the destination station through 7 multihops. Figure 3 shows that the total energy consumption vs. message inter arrival time for the three different MAC algorithms. We can see that the MAC with no sleep cycle consumes the energy much higher than the other two MAC algorithms with sleeping mechanisms. The proposed algorithms consume less energy than the MAC with 10% duty cycle as the traffic load

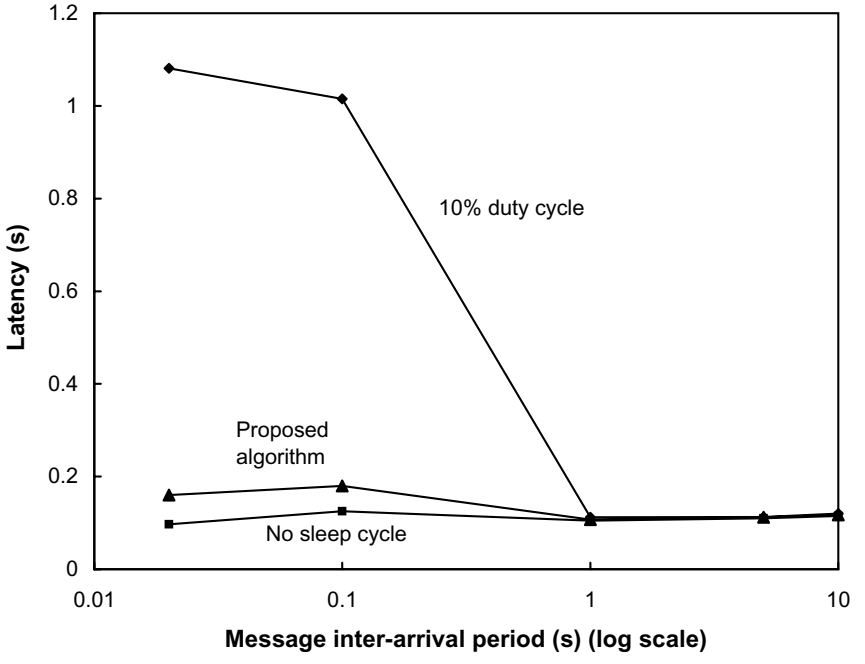


Fig. 4. Latency vs. Message Inter-arrival Period

decreases. The reason is that the proposed algorithm reduces the active period less than 10% duty cycle under light traffic conditions.

Figure 4 shows that the end-to-end latency vs. message inter arrival time for the three different MAC algorithms. If inter-arrival time is larger than 1 s, then, the traffic is too light to show the latency difference among three MAC algorithms. Therefore, we checked the latency performance carefully for the range of the inter-arrival period less than 1 s, and showed the latency results by using log scale graph. As expected, the latency of 10% duty cycle is pretty worse under heavy traffic situations because of unconditional sleeping period which does not consider traffic situations. The proposed algorithm showed that the latency is highly improved compared with 10% duty cycle MAC algorithm. The latency of no sleep cycle MAC is the lowest among three algorithms, however, the energy consumption is expected to be the highest value. From these results, we can see that the fixed sleeping period may cause critical latency problems when the sensor networks are implemented in real world. There is the tradeoff of latency for energy savings.

5 Conclusions

The general assumption that all stations have the same contention window range in steady state results in sub-optimal solutions for performance analysis of

distributed contention-based MAC algorithms because of its inherent limitations. The fast collision resolution algorithm reduces the wasting overheads come from each contention procedure. By considering sporadic traffic patterns in WSNs, the coordinated sleeping algorithm and the traffic-adaptive algorithm provide significantly high performance in energy consumptions and end to end latency. We use simple and general ideas for power saving and traffic management to implement them onto sensor network stations with high reliability. We combined these important attributes in WSNs and present the energy-efficient and traffic-adaptive fast collision resolution MAC algorithm. The proposed MAC algorithm significantly improves the energy efficiency and still provides easy implementation property in WSNs. The performance analysis and implementation studies have demonstrated that the proposed algorithm reduces the energy consumptions and the wasting overheads come from each contention procedure efficiently.

References

1. Bharghavan, V.: MACAW: A Media Access Protocol for Wireless LAN's. SIGCOMM'94, London, England, Aug. (1994) 212-225
2. Bianchi, G.: Performance Analysis of the IEEE802.11 Distributed Coordination Function. IEEE Journal on Selected Areas in Commun. **18** (2000) 535-547
3. Cali, F., Conti, M., Gregori, E.: Dynamin Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit. IEEE/ACM Trans. on Networking **8** (2000) 785-799
4. Chandra, A., Gummalla, V., Limb, J.: Wireless Medium Access Control Protocols. IEEE Communi. Sur. (2000)
5. Crow, B., Widjaja, I., Kim, J., Sakai, P.: IEEE 802.11 Wireless Local Area Networks. IEEE Commun. Mag. **35** (1997) 116-126
6. Dam, T., Langendoen, K.: An adaptive energy efficient MAC protocol for wireless sensor networks. Proc. 1st international conf. an embedded networked sensor systems (2003) 171-180
7. Ye, W., Heidemann, J., Estrin, D.: Medium access control with coordinated adaptive sleeping for Wireless Sensor Networks. IEEE/ACM Trans. Networking **vol.12** (2004) 493-506
8. Singh, S., Raghavendra, C.: Pamas: Power avare multi access protocol with signalling for ad hoc network (1998) 5-26
9. Fullmer, C., Garcia-Luna-Aceves, J.: Floor acquition multiple access (FAMA) for packet-ratio networks. Proc. SIGCOMM, Cambridge, MA. (1995) 262-273
10. Goodman, D., Valenzuela, R., Gayliard, K., Ramamurthi, B.: Packet Reservation Multiple Access for Local Wireless Communications. IEEE Trans. Commun. **37** (1989) 885-890
11. Rajendran, V., Obraczka, K., Garcia, J.: Energy-efficient collision-free medium access control for wireless sensor networks. The First ACM Conference on Embedded. Networked Sensor Systems (SenSys 2003), (2003) 182-191
12. Lu, G., Krishnamachari, B., Raghavendra, S.: Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks. in Proc. EWCN04,. April (2004) 701-706

13. CC2420DBK Demonstration Board Kit Data Sheet.: http: //www.chipcon.com/index.cfmkatid=2subkatid=12dokid=140
14. Timmons, N., Scanlon, W.: Analysis of the performance of IEEE 802.15.4 for medical sensor body area networking. IEEE Sensor and Ad Hoc Communications and Networks Conference (SECON), (2004) 16-24

Bidirectional Data Aggregation Scheme for Wireless Sensor Networks

Sungrae Cho

Department of Computer Sciences
Georgia Southern University
Statesboro, P.O. Box 7997, GA 30460 USA
Tel: 912-486-7375, Fax: 912-486-7672
srcho@georgiasouthern.edu

Abstract. In this paper, bidirectional data aggregation (BDA) scheme is proposed for wireless sensor networks. Traditionally, data aggregation bas been performed in backward direction (from source to sink) where each node in the network combines data from its child nodes. BDA algorithm, however, aggregates sensory data in both directions (sink to sources and sources to sink) when the sink is interested in gathering singular aggregates such as `MAX` and `MIN`. In forward aggregation (sink to sources), each node tags its sensor reading to the ongoing query only if its local reading is not redundant. Node receiving the tagged query suppresses its response if its local reading is redundant. By doing so, we can limit a number of redundant and unnecessary responses from the sensor nodes, saving energy. Performance evaluation shows that BDA algorithm significantly improves energy-efficiency as well as provides an accurate response for a given singular query in the presence of time-varying sensor readings.

1 Introduction

Wireless sensor networks have drawn immense attentions recently from industries and research institutions as an enabling technology for invisible ubiquitous computing arena. Spurred by the rapid convergence of key technologies such as digital circuitry, wireless communications, and micro electro mechanical systems (MEMS), a number of components in a sensor node can be integrated into a single chip with reduction in size, power consumption, and cost [1]. These small sensor nodes could be deployed in home, military, science, and industry applications such as transportation, health care, disaster recovery, warfare, security, industrial and building automation, and even space exploration. Among a large variety of applications, phenomena monitoring is one of the key areas in wireless sensor networks.

In phenomena monitoring, sensor nodes are scattered densely in a sensor field as in Fig. 1. A node called *sink* requests sensory information by sending a *query* throughout the sensor field. This query is received at sensor nodes (or *sources*). We refer the period that the query propagates throughout the sensor field as a *query phase*, and the direction from sink to source nodes as a *forward direction*.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 595–604, 2006.

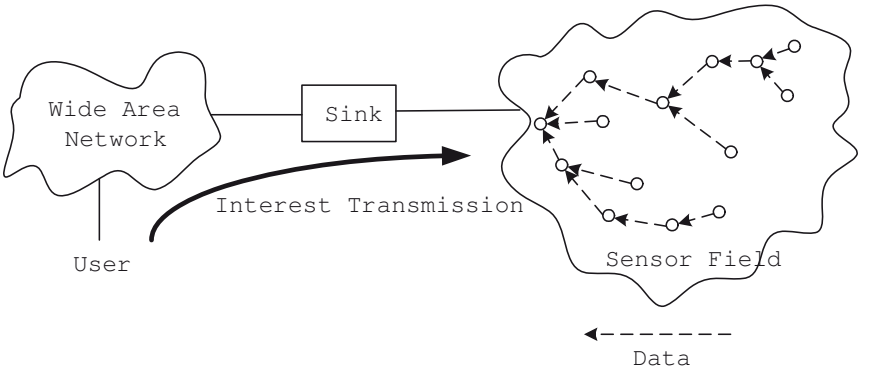


Fig. 1. Phenomenon gathering

When the node finds data matching the query, the data (or response) is routed back to the sink by a multihop infrastructureless networked sensors. We call the time duration that the responses are reached back to the sink as a *response phase*, and the direction from source to sink as a *backward direction*.

The key challenge in such phenomena monitoring is conserving the sensor energy, so as to maximize their lifetime. Since various sensor nodes often detect common phenomena, there is likely to be some redundancy in the sensory data that the sources generate. In-network filtering and processing technique can therefore help to conserve the scarce energy resources. *Data aggregation* or *data fusion* has been identified as an essential paradigm for wireless routing in sensor networks [7]. The idea is to combine the data coming from different sources en-route – eliminating redundancy, minimizing the number of transmissions and thus saving energy.

Intanagonwiwat *et al.* [5] discussed direct diffusion, a set of data-centric technique throughout the network. Their proposed operators are used to provide energy-efficient in-network data aggregation. Madden *et al.* [9] proposed TAG, an aggregation service as a part of TinyDB [11] which is a query processing system for a network of Berkeley motes. The service employs a SQL interface to the sensor data streams. It presents in-network processing of the aggregation queries on the data generated in the sensor network. Zhao *et al.* [14] introduced an architecture for sensor network monitoring. Their architecture benefits from an energy-efficient aggregate for network properties (digest functions). An average query is computed on the digest tree which their digest diffusion scheme constructs. Yuan *et al.* [13] introduced synchronization scheme for data aggregation when an event is detected at each sensor node. They proposed multi-level fusion synchronization (MFS) protocol which synchronizes the transmission time of each intermediate node so that the aggregation is performed effectively.

Most of the previous data aggregation techniques [3,4,5,6,7,8,10,12,13,14] aim at aggregating data during the response phase. However, we can further reduce the energy consumption by exploiting the query phase. We propose in this paper a mechanism to aggregate data during both query and response phases referred

to as bidirectional data aggregation (BDA) algorithm. BDA algorithm is performed when the sink is interested in gathering singular aggregates such as `MAX` and `MIN`. During query phase (sink to sources), each node tags its sensor reading to the ongoing query only if its local reading is not redundant. Node receiving the tagged query suppresses its response if its local reading is redundant. By doing so, we can limit a number of redundant and unnecessary responses from the sensor nodes, saving energy.

The remainder of this paper is organized as follows. The proposed bidirectional data aggregation (BDA) algorithm is given in Section 2. In Section 3, we compare the energy-efficiency performance of BDA algorithm with traditional data aggregation. Finally, contributions and future work are discussed in Section 4.

2 Bidirectional Data Aggregation

Coverage of deployed sensors will overlap to ensure robust sensing task, so one event will likely trigger multiple sensors in the same phenomenon. In this case, it is likely to receive multiple identical copies of a sensory data. Also, some queries inherit redundant responses as follows:

- **Max:** The sink is interested in gathering maximum value from the sensor field. In this case, other values less than or equal to the maximum are redundant.
- **Min:** The sink is interested in gathering minimum value from the sensor field. In this case, other values greater than or equal to the minimum are redundant.
- **Existence:** Some application needs to identify the existence of a target object. For example, in directed diffusion [5], an initial query dissemination is used to determine if there indeed are any sensor nodes that detect the interested object.

We refer the query with the above types as a *singular* query which expects only one response from source nodes. Redundant and unnecessary responses will generate unnecessary transmissions at the underlying layers. For instance, unnecessary response will cause high duty cycle at the medium access control (MAC) layer which in turn generates high contention from multiple nodes. Consequently, sensor nodes suffer from unnecessary energy consumption.

The proposed BDA algorithm proactively suppresses the redundant and unnecessary responses from sensor nodes so as to reduce the energy consumption. Consider a network of tree rooted from a sink node to leaf nodes. While the discussion of algorithms that help to generate and maintain this tree are beyond the scope of this work, our proposed bidirectional algorithm can function in any topology of the tree. We assume that the aggregation tree is formed at the network initialization phase, and is dynamically re-organized as sensors sleep, wake up, or fail. Let define the depth of node i as the number of edges from the sink

to i. Suppose that the depth of the tree D is known *a priori* through a simple probing technique[1] in the network initialization phase.

The proposed BDA algorithm consists of two aggregation schemes: (1) forward aggregation and (2) backward aggregation. The forward aggregation is performed during the query phase while the backward aggregation is done in the response phase. The forward aggregation behaves as the following:

- Unlike the traditional query [9], BDA algorithm exploits a special field called *tag* in the query.
- The sink disseminates a query to its child nodes with a null tag. The sink then waits for DT for responses from its child nodes where T is the maximum allowable round-trip time between two neighboring nodes.
- When the query is received, each node updates the tag of the received query with its local reading *only if it finds its local reading is not redundant*[2]. Whether or not redundant, the node simply forwards the query to its children, stores the non-redundant response at its local memory, and waits for $(D - d)T$ for responses from its child nodes where d is the depth of the node. By permitting this waiting time, each node is able to aggregate all the necessary responses from its subtree. Since it is possible that the updating node does not receive any responses from its subtree, it activates a timer for $(D - d)T$.

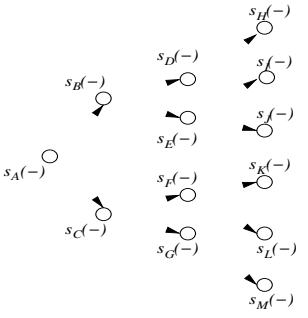
Also, the backward aggregation behaves as follows:

- Once the query is reached at the leaf node, the leaf compares its local sensor reading with the received query tag. If its sensor reading is redundant, the leaf suppresses its response; otherwise, it transmits its response. If the leaf receives responses from other nodes before its response transmission, the leaf compares its local reading with the received response. The leaf suppresses its response if its local response redeems to be redundant.
- When an intermediate node receives responses from its child nodes, the node compares the received responses with its current local reading. If the local reading is not redundant, the node changes the received response with its local reading. Whether or not redundant, the node forwards the received response at $(D - d)T$ where d is the depth of the node. When the node receives a response from other nodes before its forwarding, the node does not forward its response if its local response is redundant.
- When its timer expires, each node transmits its local reading to its parent.

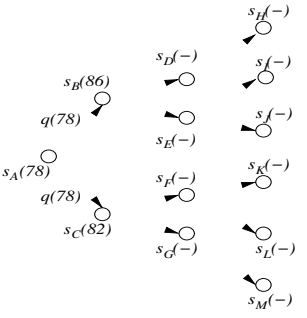
For easy understanding of the proposed BDA algorithm, Fig. 2 shows an example of the BDA algorithm where we assume time $t_i < t_{i+1}$ for i=0, 1, ... Let s_i denote the sensor reading at node i, and by $s_A(78)$ we imply that sensor

[1] This can be easily done by sending a simple query from a sink where intermediate nodes simply increment the hop counter of the query. Then, the maximum hop counter will be the depth of the tree.

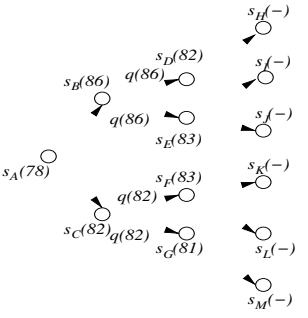
[2] We already described the redundancy condition for singular queries such as `MAX`, `MIN`, and `Existence` in the beginning of this section.



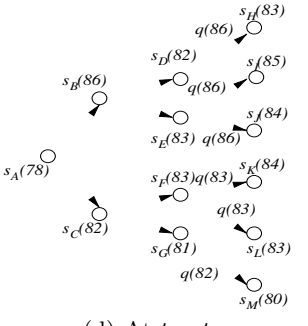
(a) A subtree formed after the network initialization phase



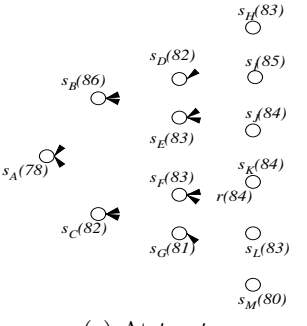
(b) At $t = t_0$



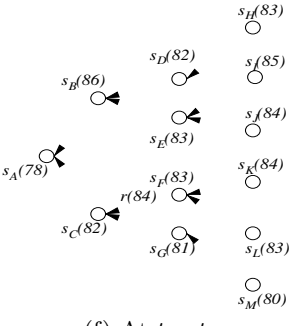
(c) At $t = t_1$



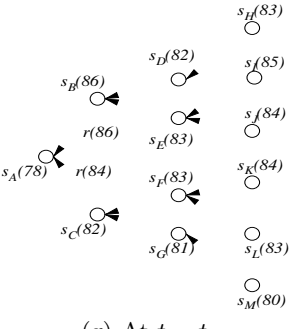
(d) At $t = t_2$



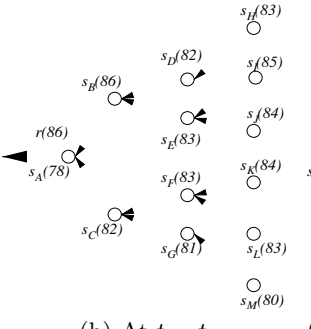
(e) At $t = t_3$



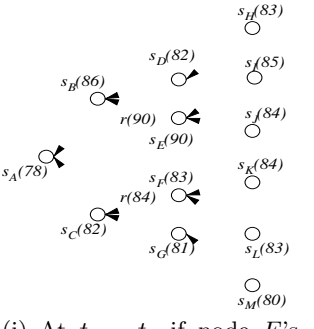
(f) At $t = t_4$



(g) At $t = t_5$



(h) At $t = t_6$



(i) At $t = t_4$ if node E's reading changed to 90

Fig. 2. An example of BDA scheme for gathering the `MAX` aggregate (On-going transmission is highlighted by solid line)

reading at node A is 78. Let $q(i)$ denote the query with tag i, and $r(j)$ denote the response with sensor reading of j. Suppose that the sink is interested in gathering the `MAX` aggregate. Fig. 2(a) shows a particular subtree in the entire network formed after the network initialization phase. Before a query is reached to the tree, all the sensor reading is unknown. At $t = t_0$ (Fig. 2(b)), a query

arrives at node A and node A's sensor reading is known to be 78. Suppose that 78 is the current maximum along the path up to node A. Node A then forwards the query ($q(78)$) by updating the tag with 78.

At $t = t_1$ (see Fig. 2(c)), nodes B and C receive the query ($q(78)$). Since both nodes' reading is greater than 78, nodes B and C update and forward $q(86)$ and $q(82)$, respectively. Since they update the query, they activate their timer for their response transmission. At $t = t_2$ (see Fig. 2(d)), nodes D and E receive $q(86)$ while nodes F and G receive $q(82)$. Nodes D, E, and G forward the received query without updating; however, node F updates $q(82)$ by $q(83)$ and forwards $q(83)$ since node F's local reading is 83. Then, Node F activates its timer for its response transmission of local reading. At $t = t_3$ (see Fig. 2(e)), all leaf nodes receive the query. Since node K's local reading is greater than the received tag, it responses with $r(84)$. At $t = t_4$ (see Fig. 2(f)), node F receives the response $q(84)$ from node K. Since its local reading is less than the query tag 84, it simply forwards the query to node C.

At $t = t_5$ (see Fig. 2(g)), node C receives and forwards $r(84)$. On the other hand at the same time, the timer activated at node B expires, therefore, node B transmits its local response ($q(86)$) to node A. At $t = t_6$ (see Fig. 2(h)), node A aggregates $r(90)$ and $r(82)$, and forwards $r(86)$.

Remark 1. In this example, we assume rather static sensor readings in the sensor field. But, in reality, the sensor reading will dynamically changes as time proceeds. Our BDA algorithm can handle this dynamics without any errors. For instance, let us assume node E's sensor reading is changed to 90 suddenly after $t = t_2$. Since each node keeps the record of the current maximum up to itself when forwarding the query, node E finds that it should send its local response ($r(90)$). For this, node E activates the timer which will expire at $t = t_4$ as shown in Fig. 2(i).

3 Performance Evaluation

To evaluate the energy-efficiency performance of BDA algorithm, we developed a simulator based on event-driven simulation using Java as shown in Fig. 3. The simulator generates a random topology as follows. We assume that the sensors have a fixed radio range and are placed in a square area randomly. Fig. 4 shows a typical network routing tree. This tree is formed based on the proximity metric of each node using breadth first search tree [2]. The root of the tree (sink) is randomly selected in the simulator. When we vary the number of sensors, we vary the size of the area over which they are distributed so as to keep the density of sensors constant. For instance, we use a 1000×1000 area for 1000 sensors. For 4000 sensors, the dimensions are enlarged to 2000×2000.

Based on the tree formed, the sink disseminates a query to its child nodes which forward this query to their children. This process is continued until the query is reached to the deepest nodes. The depth of the tree is computed based on the tree formed, and is used for response waiting time ($(D - d)T$) at each of

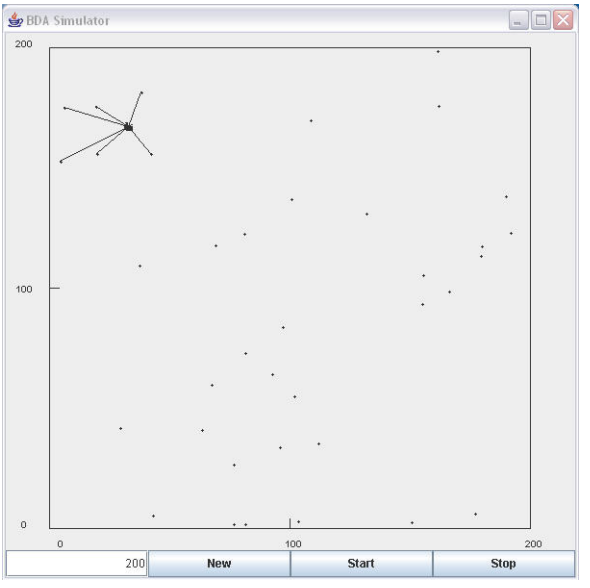


Fig. 3. A screenshot of the simulator when the sink node aggregates its direct children's responses (40 nodes generated in a 200 × 200 area)

the node. When the query is reached to the deepest nodes, the deepest nodes respond with their sensor reading which are aggregated at their parent node, and so on towards the sink.

The sensor reading values are generated uniformly or non-uniformly. When uniform sensor reading is used, we generate the integer numbers in range of [10, 90] where we assume the minimum and maximum sensor readings are 1 and 100, respectively. In other words, the sink expects the sensor readings are from 1 to 100, but actual readings are between 10 and 90. The choice of the range is rather arbitrary, but we observed that expanding the reading range does not affect the performance when we also increase the node density. In real networks, the values of sensors are not uniformly distributed, but rather correlated with their location. For this reason, we consider non-uniform scenario in the simulation. In case of non-uniform sensor reading, we divide the area into four regions called Region I, II, III, and IV as shown in Fig. 4 with the following sensor reading ranges:

- In Region I, uniform sensor readings from 10 to 30,
- In Region II, uniform sensor readings from 30 to 50,
- In Region III, uniform sensor readings from 50 to 70, and
- In Region IV, uniform sensor readings from 70 to 90.

We performed the `MAX` query process as described in Section 2. In the simulation, we consider two protocols: (1) *backward aggregation* (Algorithm 1) [9], and (2) *bidirectional data aggregation* (BDA). Algorithm 1 [9] is a technique that exploits other sensors' reading, but without forward aggregation. In other

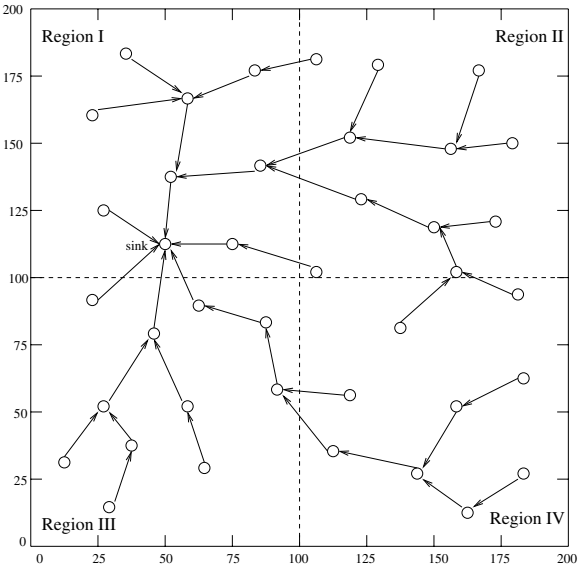


Fig. 4. An exemplary network routing tree for 40 nodes placed in a 200×200 area

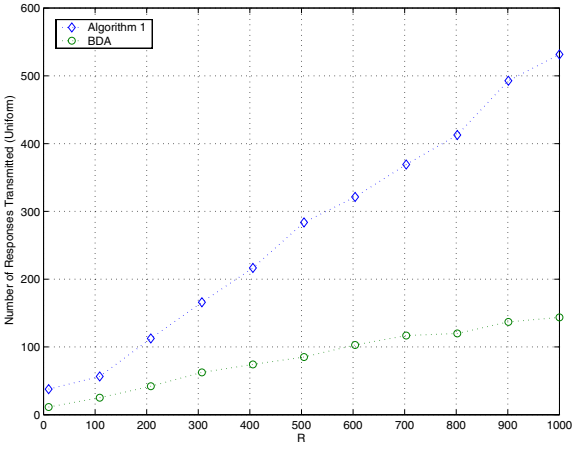


Fig. 5. Energy-efficiency performance (uniform distribution)

words, node that receives other nodes' reading in response phase suppresses its transmission if that node finds its local sensor reading is redundant.

In our simulation, we measure the number of responses transmitted[3]. The number of responses transmitted will be taken into consideration as an

[3] We measured the number of responses transmitted in the network instead of the number of responses received at the sink node. Counting the responses at the sink node only considers energy consumption at neighboring nodes of the sink.

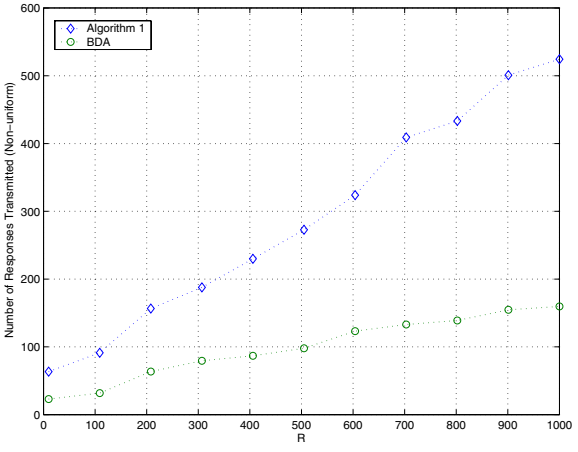


Fig. 6. Energy-efficiency performance (non-uniform distribution)

energy-budget. All performance data we present in this section is averaged over 12 different topologies. Throughout the simulation, our BDA algorithm always provides an accurate maximum value of the sensor readings.

Fig. 5 shows the number of responses transmitted in the network versus the number of nodes under uniform distribution of the sensor readings. As can be seen, the number of responses in both algorithms linearly increases with the number of nodes but with different slope. We observe that BDA algorithm significantly reduces the number of responses. Compared with Algorithm 1, BDA algorithm improves the energy-efficiency (the number of responses) with factor of 0.25.

Fig. 6 shows the energy-efficiency performance under non-uniform distribution of the sensor readings. The results are similar to the uniform case but with a little more fluctuation and with a slight increase of the number of responses. We conclude that the BDA algorithm outperforms conventional backward aggregation, independent of the distribution of the sensor readings.

4 Contributions and Future Work

In this paper, bidirectional data aggregation (BDA) scheme is proposed. Traditionally, data aggregation bas been performed in backward direction. BDA algorithm, however, aggregates sensory data in both directions (sink to sources and sources to sink) when the sink is interested in gathering singular aggregates such as `MAX` and `MIN`. In forward aggregation (sink to sources), each node tags its sensor reading to the ongoing query only if its local reading is not redundant. Node receiving the tagged query suppresses its response if its local reading is redundant. By doing so, we can limit a number of redundant and unnecessary responses from the sensor nodes, saving energy. To our best knowledge, our work is unique and different from previous work.

Performance evaluation shows that BDA algorithm significantly improves energy-efficiency as well as provides accurate response for a given singular query in the presence of time-varying sensor readings.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A Survey on Sensor Networks. IEEE Communications Magazine, August(2002) 102–114
2. Cormen, T.H., Leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. MIT Press, 1990
3. Heinzelman, W.R., Kulik, J., Balakrishnan, H.: Adaptive Protocols for Information Dissemination in Wireless Sensor Networks. in Proc. of ACM MOBICOM, August(1999)
4. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. in Proc. of Hawaii International Conference on System Sciences, January (2000)
5. Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. in Proc. of ACM MOBICOM, August (2000)
6. Krishnamachari, B., Estrin, D., Wicker, S.: Modeling Data-Centric Routing in Wireless Sensor Networks. in Proc. of IEEE INFOCOM, June (2002)
7. Krishnamachari, B., Estrin, D., Wicker, S.: The Impact of Data Aggregation in Wireless Sensor Networks. in Proc. of IEEE ICDCSW, July (2002)
8. Lindsey, S., Raghavendra, C.: PEGASIS: Power-Efficient Gathering in Sensor Information Systems. in Proc. of Aerospace Conference, March (2002)
9. Madden, S., Szewczyk, S., Franklin, M.J., Culler, D.: Supporting Aggregate Queries over Ad Hoc Sensor Networks. in Proc. of Operating Systems Design and Implementation, December (2002)
10. Shrivastava, N., Buragohain, C., Agrawal, D.: Medians and Beyond: New Aggregation Techniques for Sensor Networks. in Proc. of ACM Sensys, November (2004)
11. `http://tlelgraph.cs.berkeley.edu/tinydb`
12. Yao, Y., Gehrke, J.: The Cougar Approach to In-Network Query Processing. in Proc. of ACM SIGMOD, June (2002)
13. Yuan, W., Krishnamurthy, S.V., Tripathi, S.K.: Synchronization of Multiple Levels of Data Fusion in Wireless Sensor Networks. in Proc. of IEEE GLOBECOM, December (2003)
14. Zhao, J., Govindan, R., Estrin, D.: Computing Aggregates for Monitoring Wireless Sensor Networks. in Proc. of IEEE SNPA May (2003)

A Base Station-Coordinated Contention Resolution for IEEE 802.16 PMP Networks

Wenyan Lu[1], Weijia Jia[1,2], Wenfeng Du[1], and Lidong Lin[2]

[1] School of Information Science & Engineering, Central South University, Changsha, 410083, China
[2] Deprtment of Computer Science, City University of Hong Kong, 83 Tat Chee Ave. Hong Kong, China
{wenyanlu, itjia, wenfeng}@cityu.edu.hk

Abstract. IEEE 802.16 PMP mode recommends using truncated Binary Exponential Backoff (BEB) algorithm to resolve the contention when multiple Subscriber Stations (SSs) compete for the connection and resource allocation from the Base Station (BS). The BEB approach may not be effective and transmission opportunities (TOs) may be wasted due to independent contention and backoff of each SS. This paper proposes an efficient coordinated backoff algorithm (COB) through BS coordination so that a global window can be assigned by the BS and the TOs can be effectively consumed by the SSs. Theoretical analysis and simulation results show that COB is adaptive to the dynamic change of active SSs and outperforms the BEB approach.

1 Introduction

The IEEE 802.16 family of standards and its associated industry consortium, WiMax (Worldwide Interoperability for Microwave Access), promising to deliver high data rates over large areas to a large number of users in the near future, are of great concern in recent years[11]. The MAC layer of 802.16 supports a primarily Point-to-Mulitpoint (PMP) architecture, with an optional mesh topology.

A PMP network consists of one Base Station (BS) and multiple Subscriber Stations (SSs). The downlink (from BS to SS) is generally broadcast. But the uplink (from SS to BS) is shared by the SSs. IEEE 802.16 has defined the MAC protocol stack for BS to assign the uplink channel to SSs. But during initial maintenance and bandwidth contention periods, all SSs still need to contend the uplink channel. An effective contention resolution is crucial to the whole system.

At present the mandatory method of contention resolution that shall be supported by 802.16 is based on a truncated Binary Exponential Backoff (BEB) algorithm. BEB has been wildly investigated in IEEE 802.11 networks[2][4][7][10]. However, the MAC layer of IEEE 802.16 is much different from that of IEEE 802.11, which makes the disadvantage of BEB more obvious in IEEE 802.16. It is too cautious that SS enlarges its backoff window whenever there is a collision. Some SSs may be treated unfairly during the contention process. The radical reason for these drawbacks is that BEB only use a small amount of contention information to make a decision for the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 605 – 614, 2006.

whole networks. In fact there are many literatures [2][3][10] trying to improve the performance of BEB with more system information in IEEE 802.11.

In this paper, we present an efficient coordinated backoff algorithm through BS coordination. Based on the observation that in one time frame the backoff window is optimal if it equals to the number of active SSs, we propose an algorithm to calculate this number. In each time frame, all active SSs will contend for the transmission opportunity with the window equal to this number. Theoretical analysis and simulations results show that COB (coordinated backoff) resolution outperforms BEB resolution.

The reminder of this paper is organized as follows. Section 2 briefly describes the BEB based contention resolution of IEEE 802.16. The disadvantages of BEB are also been pointed out. In section 3, COB resolution and the algorithm to calculate the number of active SSs are presented. Section 4 is the performance analysis and simulation results. Finally, section 5 concludes the paper.

2 BEB Based Contention Resolution in IEEE 802.16

In IEEE 802.16, both initial maintenance and Request Contention are contention-based. Since they have the same contention resolution, in this paper we take the Request Contention as example.

2.1 Background of BEB

In order to effectively schedule the data transmission among SSs, IEEE 802.16 defines a complex data structure. The detail is shown in Fig.1. Data are sent and received in time frame cycle. Each time frame is divided into uplink subframe and downlink subframe. The BS controls the assignment on uplink subframe through the UL_MAP which is broadcasted in downlink subframe. The portion of request contention can be defined by a request Information Element (IE) which may consist of multiple transmission opportunities. A transmission opportunity is defined as an allocation provided in a UL_MAP or part thereof intended for a group of SS authorized to transmit BW requests.

An SS which has information to send is called active SS. Before entering its contention resolution process, an active SS first gets the initial backoff window W and the maximum backoff window W_{max} from BS and then randomly select a backoff value within the initial backoff window. This random value indicates the number of transmission opportunities that the SS shall defer before transmitting. Backoff value decreases by one on every transmission opportunity. When this value reach zero, in the next transmission opportunity the SS sends out its BW request.

After a transmission, the SS waits for a Data Grant Burst type IE in the subsequent time frame. Once received, which means the request is succeed and the contention resolution is complete. Otherwise the BW request transmission is unsuccessful. The SS now increases its backoff window by a factor of two as long as the it is less than W_{max}, the SS shall randomly select a backoff value within its new window and repeat the deferring process described above. This retry process continues until the maximum number of retries f has been reached. Then the BW request shall be discarded.

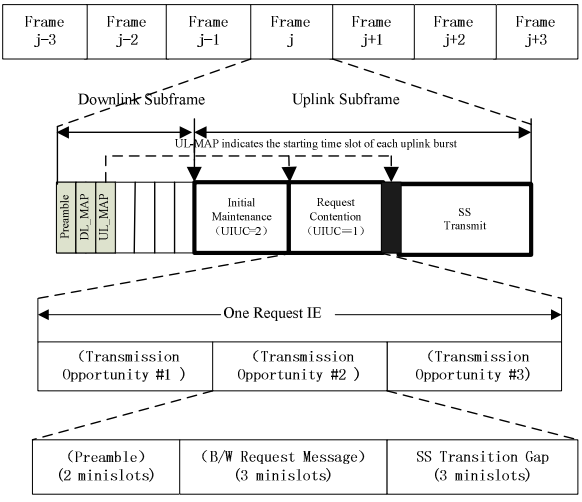


Fig. 1. Data Structure of BW request

2.2 Disadvantages of BEB

The BEB has been wildly used in DCF, but is not perfect in IEEE 802.16. First, there is blindfold for the SS enlarging its backoff window whenever it encounters a collision. Unlike in IEEE 802.11, the SS in IEEE 802.16 doesn't sense whether the channel is busy or not before it transmits a request. This leads to a large collision probability which is caused by the nature feature of random access method. It doesn't always indicate that the contention is severe. When there is a severe collision SS should enlarge its backoff window. But it does not mean that SS should enlarge its backoff window whenever there is a collision.

Another disadvantage is that BEB may be unfair to some SS. In time frame i, when SS_j chose its backoff value $b_{i,j}$ in interval [0, w], if $b_{i,j}$ is bigger than the number of transmission opportunities O_i, SS_j shall defer the opportunity and contends with the backoff values $b_{i,j}$-O_i in time frame i+1. But the SSs that contend in time frame i+1 have no idea of the SS_j. So the transmission opportunities from 1 to $b_{i,j}$ -O_i in frame i+1 are incident to collisions. In another word, the SSs that have $b_{i,j}$>O_i, have more opportunities of collisions .

3 Coordinated Backoff Algorithm

From the insufficiencies of BEB it is easy to know that lacking global contention information causes the SSs take an unperfected policy. If we can use the characteristic of central scheduling of PMP and let the BS control the contention, we may improve the performance. Based on this ideal we present a coordinated backoff algorithm.

3.1 Statement of COB

In Our algorithm the contention is coordinated by BS. In each time frame BS provides a global window which is closed to be an optimal window in one time frame. All SSs will content with this window. The detail of the algorithm is as follows.

1) In time frame i, BS calculates the global backoff windows w_i. Along with O_i, w_i is broadcasted to all SS through UCD message.
2) The active SS_j random select a backoff value $b_{i,j}$ in[0 w_i]. If $b_{i,j} > O_i$, SS_j do not send in this frame. If $b_{i,j} \leq O_i$, SS_j sends out its BW request.
3) BS assigns the bandwidth and records the BW request results.
4) If SS_j has been assigned the bandwidth, the contention process is complete. If SS_j's request encounter collision SS_j will content with the new window w_{i+1} in time frame i+1.

3.2 Global Backoff Window

One of the key steps of COB is to provide a proper global window in each time frame. This window should make the BS correctly receive the BW request as many as possible. We define the success probability p_s as the probability that BS receive a correct BW request in a generic (i.e., randomly chosen) transmission opportunity. It is not hard to know that p_s is the probability when anyone of the active SS sends out a BW request while the other n-1 SSs don't send. Note that when $w<O$, SS can not use those transmission opportunities whose ordinal numbers are bigger than w. These transmission opportunities are wasted. We will always keep $w \geq O$ and then we have

$$p_s = C_n^1 \frac{1}{w}(1-\frac{1}{w})^{n-1} = \frac{n}{w}(\frac{w-1}{w})^{n-1} \tag{1}$$

To maximize the p_s we have

$$\frac{dp_s}{dw} = \frac{-n}{w^2}(\frac{w-1}{w})^{n-1} + \frac{n(n-1)}{w^3}(\frac{w-1}{w})^{n-2} = 0 \tag{2}$$

Form (2) it is very easy to know that the optimal window is

$$w = n \tag{3}$$

For the convenience of further discuss, we also calculate the probability of collision p_c for a random chosen transmission opportunities.

$$p_c = 1-(1-\frac{1}{w})^n - \frac{n}{w}(1-\frac{1}{w})^{n-1} \tag{4}$$

3.3 Algorithm of Calculating the Number of Active SS

The optimal window equals to the number of active SS. while this number is changing all the while. How to estimate the number is crucial to COB method. In i-1th time

frame, denote O_{i-1} as the number of transmission opportunities, C_{i-1} as the number of transmission opportunities that suffer collisions, S_{i-1} as the number of succeeded transmission opportunities, $p_{si-1}=S_{i-1}/O_{i-1}$ as the success rate, $\hat{p}_{si-1}$ as the estimated success rate, $p_{ci-1}=C_{i-1}/O_{i-1}$ as the collision rate and $\hat{p}_{ci-1}$ as the estimated collision rate. Using the technology of exponential smoothness we get the success rate of *i*th time frame:

$$\hat{p}_{si} = \beta \hat{p}_{si-1} + (1-\beta) p_{si} \tag{5}$$

In (5), β is called smooth index which is determined by the variable rate of active SS. If it changes rapidly the value will be large. Suppose we know the number of active SSs n, according to (1), we can calculate $\hat{p}_{si}$ as

$$p_{si} = \frac{n_i}{w_i}(1-\frac{1}{w_i})^{n_i-1} \tag{6}$$

Let $p_{si} = \hat{p}_{si}$, with (5) and (6) we can get an equation on n_i. But it is not a single-valued function, given a $\hat{p}_{si}$ there may be two answers. Which one is true need further analyze. Similar to p_s, we also estimate the collision rate by

$$\hat{p}_{ci} = \beta \hat{p}_{ci-1} + (1-\beta) p_{ci} \tag{7}$$

$$p_{ci} = 1-(1-\frac{1}{w_i})^{n_i} - \frac{n_i}{w_i}(1-\frac{1}{w_i})^{n_i-1} \tag{8}$$

From (7) and (8) we get a single-valued function of n_i. Solving the equations we get an estimated number of active SS denoted as $\hat{n}_{ci}$. If there are two solutions to (5) and (6) we use the one that is closer to $\hat{n}_{ci}$ as the estimated value $\hat{n}_{si}$. Considering success and collision rate are stochastic we use the average value as the result,

$$\hat{n}_i = \frac{\hat{n}_{si} + \hat{n}_{ci}}{2} \tag{9}$$

Taking the number of transmission opportunity O_i into account we set the window as

$$w_i = Max(\hat{n}_i, O_i) \tag{10}$$

This window is the global indicator on the contention. If contention is server, which means there are many active nodes, the window is big so as each active node sends request in each with small probability and eventually achieve better performance.

4 Performance Analyses and Simulation

As we have mentioned above, p_s reflects the use of transmission opportunity. We use it as the main metric of performance. From the viewpoint of SS, whenever it wants to send data, it hopes its BW request can be received by BS as soon as possible. We use delay d to measure this character.

4.1 The Maximal Success Probability

If we know the number of active SSs exactly, substitute w=n into (1) we have

$$p_{s\max} = (1-\frac{1}{n})^{n-1} \tag{11}$$

But unfortunately we have to estimate the number. Suppose there is an error between estimated value and actual value. Denoting $\hat{n} = n \pm \Delta n$, according to COB resolution, all active SSs shall use $1/\hat{n}$ as their backoff windows. So the success probability can be calculated by:

$$p_s = C_n^1 \frac{1}{\hat{n}}(1-\frac{1}{\hat{n}})^{n-1} = \frac{n}{n \pm \Delta n}(1-\frac{1}{n \pm \Delta n})^{n-1} \tag{12}$$

Since (12) express the actual success probability of COB resolution, we use $\triangle p_s$ to denote the error between p_s and p_{smax}:

$$\Delta p_s = (1-\frac{1}{n})^{n-1} - \frac{n}{n \pm \Delta n}(1-\frac{1}{n \pm \Delta n})^{n-1} \tag{13}$$

A numeric example can show that $\triangle p_s$ is not sensitive to $\triangle n$. Supposing n=20, the error of n is 20%, it is easy to find out that when $\hat{n} = 16$, $\triangle p_s$ get the maximal value. The value is 0.0106 which is only about 2.8% of the p_{smax}. So even if there is a big error between the estimated value and the actual value the p_s can remain a high values. It indicates that COB is robust.

4.2 Calculation of Delay

When SS sends out a BW request it needs to wait the next time frame to know if the BW request is succeeded or not. Since transmission opportunity is much shorter than a time frame. We ignore the delay within a time frame. We calculate the delay with the unit of time frame. According to COB resolution, in the first time frame the success probability is $O_1 \cdot p_{s1}/w_1$, so the probability for the delay equals to one time frame is $(1- O_1 \cdot p_{s1}/w_1) \cdot O_1 \cdot p_{s1}/w_1$. The probability that the delay is k time frames can be calculated by

$$p(\text{delay } k \text{ frame}) = \frac{O_k}{w_k} p_{sk} \prod_{i=1}^{k} (1-\frac{O_i}{w_i} p_{si}) \tag{14}$$

The expectation values of delay shall be

$$E(D) = \sum_{k=1}^{\infty} k \frac{O_k}{w_k} p_{sk} \prod_{i=1}^{k} (1 - \frac{O_i}{w_i} p_{si}) \tag{15}$$

Supposing O_i, w_i, p_{si} be constant O, w, p_s we have

$$E(D) = \sum_{k=1}^{\infty} k \frac{O}{w} p_s (1 - \frac{O}{w} p_s)^k = \frac{w - Op_s}{Op_s} \tag{16}$$

By (16), we know that delay is determined by O_i,w,p_s. The bigger of p_s is the less of d is. The more of active SSs is the less of d is. And the bigger of backoff window is the bigger d is.

4.3 Simulation Results

To the best our knowledge, there is no literature to analyze the performance of BEB in IEEE 802.16 by now. So we main compare the performance of BEB and COB by simulation. We develop a program to simulate the process of BW request in IEEE 802.16. The program consist of two kinds of objects BS and SS. BS send out O_i transmission opportunities in each time frame and the active SSs randomly chose a backoff value within its backoff window $w_{i,j}$. The BS reads the status and backoff values of each SS and then give a judgment if the request is succeeded or not. Here we assume the wireless channel is ideal, that is, we ignore the possibility of errors due to noise and also ignore the possibility of "capture" techniques by which a SS sometime capture one transmission in the presence of multiple transmissions. The whole simulation is written with event driving model.

First, we compare the performances in a "steady status". Under this condition, we keep the number of active SS as a constant. Each SS generates a new BW request immediately after a succeed one and SS will not discard its BW request till it has been successfully transmitted. For simplify we keep the number of transmission opportunities unchanged. We record the number of successful BW request and corresponding delay in each time frame. Fig. 2 and Fig. 3 are the corresponding value of BEB and COB from time frame 1500 to 1550. Here n=32, initial backoff window in BEB is 32 and the smooth index β in COB is 0.8. Note that the delays in Fig. 3 are the average delay of all the successful transmission opportunities in one time frame. From Fig 2 we may observe that the success times in COB is bigger than that in TEBE. Fig.3 enable us to conclude that: (1) The delay of COB is less than that of BEB in general; (2) in some fame the delay are very large in BEB which illustrate the unfair problem in section 2.2.

Given an n we simulate the contention process with 100,000 time frames. Let n change from 5 to 100, we draw the curve of p_s and d for BEB and COB respectively in Fig. 4 and Fig. 5. From Fig. 4, we can observe that When O=32 and n<60 COB is better than BEB both on p_s and d. But when n>60, BEB perform better than COB. Taking a careful study we can find out that when n>60 the delay of BW request is big (more than 4 time frames). So the BS should provide more transmission opportunity to reduce the delay. When we enlarge transmission opportunity to 64, COB outperforms BEB.

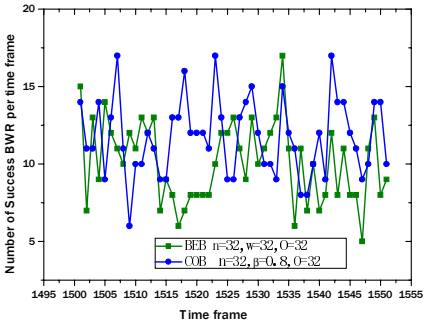


Fig. 2. Variation of success BWR with time frame

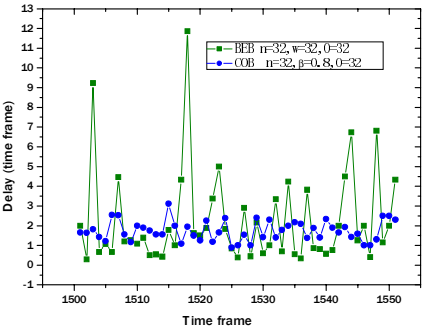


Fig. 3. Variation of delay with time frame

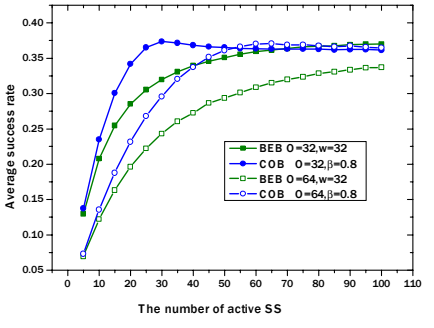


Fig. 4. Variation of p_s with n

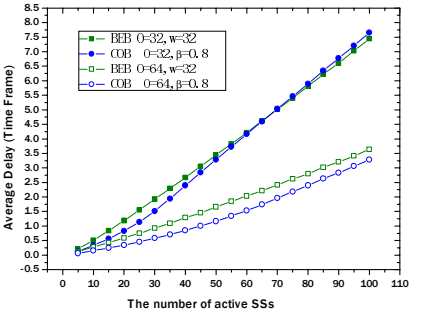


Fig. 5. Variation of d with n

Be used in practice, the number of active SS may change dynamically. It may need a period of times for SS generating a new BW request. At the same time, the number of transmission opportunity that BS provides in a time frame is limited by some other factor such as bandwidth assignment method. The number may vary from frame to frame. Takes these factors into account, we simulate the dynamic feature of IEEE 802.16 in two aspects: on one hand, the arriving of BW request obeys negative exponential distribution; on the other hand, the number of transmission opportunity obeys constant distribution. Furthermore, after trying sending a BW request more than f time fame we discard it. We don't do it in f retransmission just in order to compare them easily. And more, the SS who discards a BW request may generate one after a time t_d. We set N equal to 1000. By adjust the value of λ we control the number of active SS. The exact vales are: w=32, β=0.5, f=25, w_{max}=1024, t_d=1000, λ= 0.005, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.08 (i.e., we control the arrive rate approximate to 5, 10, 20, 30, 40, 50, 60, 80 per time frame). We simulate the O in two different ways: 1) O is randomly chosen from 10 to 80; 2) O is calculated by COB resolution. In this case we limit the O to 50. Fig. 6, 7 and 8 are the curve of average success rate, average delay and losing rate under three conditions respectively.

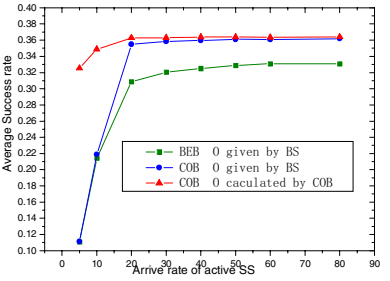


Fig. 6. Variation of p_s with the arrive rate of active SS

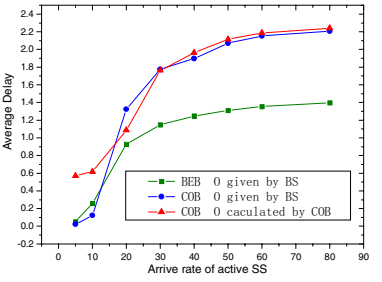


Fig. 7. Variation of d with the arrive rate of active SS

From these Figures, we may observe that: firstly, the average success probability of COB is larger than BEB, and is much better when O is decided by COB. Secondly, the delay of COB is less than that of BEB when the number of active SS is less than 12, but the delay of COB tends to be bigger than BEB when the arrival rate of request continues increasing. However, by analyzing the delay and request losing rate carefully, it's not hard to know that the request losing rate of BEB is much more than that of COB. That is to say, we decrease the delay of BEB by throwing away BW request. When the value of f is large enough, the delay of BEB would increase remarkably, and will bigger than COB. If O lies on COB, the request losing rate may be much less.

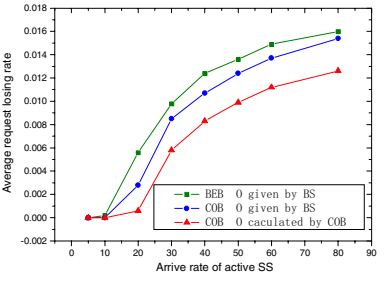


Fig. 8. Variation of p_d with the arrive rate of active SS

5 Conclusions

The foundation of BEB based contention resolution is distributed coordinated function. In BEB, all the SSs judge the degree of contention individually and take action individually. Since each SS only use a small amount of information (the result of its transmission), their contention policy may be incompletion. In this paper we propose an effective coordinated backoff algorithm in which the evaluation of the contention condition is conducted by BS. Comparing with the single feedback of SS, BS is able to know of the competition condition of the whole networks, therefore be able to control the system's competition in a better way. Simulation results show this resolution outperforms BEB.

Acknowledgement

This work is supported by Strategy Grant of City University of Hong Kong under No. 7001709 and 7001777 and partially supported by the National Grand Fundamental Research 973 Program of China under Grant No. 2003CB317003.

References

[1]. IEEE 802.16-2004 IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems [S]
[2]. G. BIANCHI, L. FRATTA, M. OLIVERIC, Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LAN [A] [C] Proc. of the IEEE Int'l Symp on Personal, Indoor and Mobile Radio Communications (PIMRC'96). 1996. 2. 392~396.
[3]. FEDERICO CALÌ, MARCO CONTI, AND ENRICO GREGORI, IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism [J], IEEE Journal on Selected Area in Communications, 2000. 18(9), 1774~1786.
[4]. G. BIANCHI, Performance Analysis of IEEE 802.11 Distributed Coordination Function [J], IEEE JSAC, 2000. 18(3), 535~547.
[5]. CALI F, CONTI M, GREGORI E. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit [J]. IEEE/ACM Trans. on Networking, 2000, 8(6):785~799.
[6]. RAPHAEL ROM MOSHE SIDI Multiple Access Protocols Performance and analysis [M], 1990. Springer-Verlag New York, Inc.
[7]. Qixiang Pang, Soung C.Lievw, Performance evaluation of an adaptive backoff scheme for WLAN [J] Wireless Communications and Mobile Computing. 2004; 4:867~879
[8]. R.L. RIVEST, Network Control by Bayesian Broadcast [J], IEEE Trans. on Information Theory, 1987, IT-33(3). 323~328
[9]. ZHANG Zhao-Feng, WEI Gang A new random access mode for mobile Internet JOURNAL OF CHINA INSTITUTE OF COMMUNICATIONS Vol.24 No.4 April 2003
[10]. PENG Yong, CHENG Shi-DuanA Self-Adaptive Wireless LAN Protocol Journal of Software China Vol.15, No.4 2004
[11]. Arunabha Ghosh, David R, Wolter, Jeffrey G.Andrews and Runhua Chen,Broadband Wireless Access with WiMax/8O2.16: Current Performance Benchmarks and Future Potential IEEE Communications Magazine February 2005, p129-136

A Promise Theory Approach to Collaborative Power Reduction in a Pervasive Computing Environment

Mark Burgess and Frode Eika Sandnes

Faculty of Engineering, Oslo University College, P.O. Box 4, St. Olavs plass, N-0130 Oslo, Norway
mark@iu.hio.no, frodes@iu.hio.no

Abstract. A grid-like environment may be constructed from ad hoc processing devices, including portable battery-powered devices. Battery lifetime is a current limitation here. In this paper we propose policies for minimizing power consumption using voluntary collaboration between the autonomously controlled nodes. We exploit the quadratic relationship between processor clock-speed and power consumption to identify processing devices which can be slowed down to save energy while maintaining an overall computational performance across a collaboration of nodes.

1 Introduction

Power consumption is rapidly becoming the central focus in pervasive computing scenarios where computers are densely packed. Whether one considers a future vision of mobile and embedded devices, in smart homes and workplaces, where power is often supplied by batteries with limited capacity, or more conventional computer systems, power is a fundamental resource to be managed. Pervasive computing environments exist today in Web Hotels and at Internet Service Providers' server rooms and data centres, where power consumption costs and cooling requirements are the main threat to operation. Here the problem is not that power will run out, but that it will raise temperatures to dangerous levels in a confined space. Either way, power consumption is an enemy to efficiency and survivability.

In this paper, we consider strategies for minimizing total power consumption using a paradigm for policy based management that can easily deal with the autonomous nature of devices in a pervasive scenario. This involves voluntary sharing of resources. We use the new framework of promise theory, where the idea is to treat each processor as an autonomous agent in a voluntary cooperative grid[1,2]. In the autonomous agent point of view, each node has complete control over its own decisions in response to events, and no authority is capable of forcing any node to perform services without its explicit consent. This scenario is clearly appropriate for a pervasive scenario, either in a mobile ad hoc network, or in a server room full of business competitors.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 615–624, 2006.

A node or agent relates to others in the environment by making *promises*, or declaring constraints about its behaviour, of various types. To give up its autonomy and allow for external control, a node can simply promise to follow someone else's orders. In this work, we attempt to ask the question: could rationally behaving agents share their abilities to save power using a policy that sacrifices none of their autonomy, and hence minimises the risk of exploitation?

2 Power Saving

Portable, consumer electronic products are increasing in computational power with the advances in microprocessor technology. Such devices include mobile phones, portable digital assistants, MP3 players, digital cameras and notebook computers. Increasingly, the boundaries between such products are blurred as they offer overlapping functionalities. As the personal devices become more powerful, they are used to conduct a wider range of tasks, but mostly their processors are idle. It is worth asking whether one could utilize this latent processing power for a beneficial purpose.

Wireless sensor networks of autonomous embedded devices are in a similar predicament; they might also reap the benefits of migratory processing power: distributed processing can be built into the nodes for immediate analysis of data, and perhaps to save power required to transmit large amounts of data back to a base-station over a potentially noisy or insecure wireless network. Distributed facial face recognition in security sensors, intrusion detection analysis, or rule matching could bring large fluctuations in power demand suddenly to a single node amongst many. What if that load could be balanced across several devices in order to save power?

Batteries are often a limiting factor in portable systems. Either the battery is unable to provide a sufficiently long uptime, or the available power capacity is unable to support the desirable hardware components. Although a component may be functionally perfect for the task at hand, the limited power provided by a particular battery may not match the consumption needs of the component.

A vast body of research has been conducted to overcome the limits imposed by batteries such that the operational time of the device can be prolonged [3,4,5,6]. These techniques are generally labelled power management. The most obvious approach is to switch off subsystems that are not in use. It is common to see power management in action on notebook computers. The disc will spin down and go to rest if the disc has not been accessed for a specified time-interval [7]. A similar spin-up delay is experienced when the disk is first accessed again. The light emitting screens of notebook computers are often switched off just after a few minutes of user inactivity. Inter-device communication also affects power – WLAN is known to consume large amounts of power while Bluetooth is more economic in power usage. Other techniques used in power aware computing systems include disks that spin at different speeds [7], where a low speed consumes less power than high speed, power consuming memory architectures and finally power aware microprocessors.

One technique commonly cited in the literature includes frequency scaled microprocessors. Such processors allow the clock speed of the processors to be scaled either continuously or in discrete steps. A processor running at a lower clock speed consumes less power than one running at a high clock speed, hence computational power can be traded against power consumption. However, there is an interesting quadratic relationship between the clock speed and the power consumption. The power consumption increases as a quadratic function with clock speed. This physical phenomenon is exploited in a number of power aware scheduling algorithms [8,9,10,11,12]. The most common technique is to adjust the slack time between tasks, i.e. when there is slack time between tasks the task before or after the slack is shifted to cover the slack by slowing down the processor [11]. The net effect is that the same computation is performed in the same time-interval, however power has been saved as one or more of the processor has been running at a lower clock speed for some portion of the schedule.

3 Speed-Power Consumption Relationship

The quadratic relationship between the processing speed and the power consumption is the prime driving force behind power saving. Generally, the energy stored in a capacitive CMOS circuit is of the familiar form $E = \frac{1}{2}CV^2$, for applied D.C. voltage V. It is the capacitance C that makes power consumption frequency dependent. For an alternating voltage $V(t)$ with fundamental clock frequency f Hertz, one has a Fourier series $V(t) \sim V_0 \sin(2\pi ft) + \ldots$, and hence from the fundamental capacitive charge law $Q = CV$, one has the current $I = \frac{dQ}{dt} = C\frac{dV}{dt}$, thus $I(t) \sim fCV(t)$. The power dissipated on releasing this current through resistive elements is, by Ohm's law ($V = IR$),

$$\overline{W} = IV = I^2R \sim f^2\, RC^2\overline{V}^2(t). \tag{1}$$

Thus, it depends – on average – approximately on the square of the fundamental clock frequency.

The rationale behind the idea of reducing power consumption is as follows. If a processing problem admits *parallelism* then the task can either be computed using one fast processor or several slower and cheaper processors in parallel with the same computational delay. Since the power consumption is quadratic with respect to the clock speed, the total power consumed by the parallel system may be less than the power consumed by the fast processor. Consequently, the system consuming the least power is preferred.

More formally, if a compound task θ comprised of the N partially ordered tasks $T_1, T_2, .., T_N$ can be computed in sequential time using one processor P running at a clock speed of S_{high} Hz, then the same compound task can be performed using M processors $P_1, P_2, ..P_M$ running at S_{low} Hz. Then, if the sum of the power consumed by the single processor exceeds the power consumed by the parallel processor the parallel processors are used, namely:

$$W(P(S_{\text{high}})) > \sum_{i=1}^{M} W(P_i(S_{\text{low}})), \tag{2}$$

where $W(P)$ denotes the accumulated power consumed by processor P. The idea of using multiple processors in a single embedded system has been standard practice for over a decade in specialized digital signal processing applications [13,14], and multiple DSP processors can be used to construct such a system. There are even multiprocessor chips such as the classic Texas Instruments TMS320C80, which had four shared memory slave processors and one master processor on a single chip. Recent trends include the multiple core microprocessors which are similar in principle. The option of integrating multiple processors into the same low cost consumer electronics device is increasingly viable with current technology.

The quadratic relationship between the clock-speed and the voltage level can be utilized such that the overall battery level can be prolonged. A grid already comprises a large set of distributed processing nodes. The processing nodes are often highly heterogeneous in terms of function, processing power and processing load. A grid problem can be job farmed into reasonably sized chunks. Sometimes these chunks can be customized in terms of size such that processing elements with a history of slower performance is given smaller computation loads than processing nodes with a history of larger capacity. However, the nature of some problems makes it difficult, if not impossible, to choose a specific chunk-size. Furthermore, in other situations it might be difficult to merge results of varying size. Traditionally, in such situations the processors run until they have finished the allotted work and are then given a new chunk. To achieve synchronisation processing nodes are left idle while waiting for the last results. In this paper the idea of slowing down faster processing nodes such that their performance matches more closely those of the slower processing nodes is explored. The general benefit of slowing down nodes is that they will consume less power.

4 Common Currency, Cooperative Payoff

We wish to turn this discussion into one about policy based management in a framework of voluntary cooperation between autonomous systems. This is representative of both independent organizations in a data centre and personal mobile devices in a public space. The central question now is what would induce the agents to cooperate in reducing the total power budget? What does a device or agent stand to gain by collaborating with its neighbours to exploit parallel processing?

- A mobile device must balance the cost of transmitting a parallelizable job to a neighbouring node against the power saving of exploiting the parallelism at lower clock rates.
- A fixed node in a data centre incurs essentially no additional power consumption for farming out tasks. The potential payoff is a saving that can reduce the total electricity bill for the data centre so that every customer can reduce their expenditure on electricity and cooling. The total share of the final bill can therefore be proportionally reduced for all.

The energy saving in a wired network is somewhat greater here, since cables channel all of the power directly along the waveguide to the receiver. A wireless signal, on the other hand, radiates in all directions and most of its power is wasted. The efficiency of wireless communication falls of as $1/r^3$ and WAN protocols step up the transmission power in fixed steps to compensate as the distance grows. The mobile situation is thus potentially more complex, and perhaps harder to imagine in practice. Nonetheless, in exploratory robotic applications, e.g. planetary exploration, where power is extremely scarce, the issues are even more essential[15].

Imagine that the number of jobs is N and the number of processing devices is P. Further, imagine for simplicity that each job has the same time and space complexity and are computationally identical. The processing delay of computing a job on processor i is given by d_i. Since the processing elements are heterogeneous and have different computational loads the computation delays follows some distribution with mean $\overline{d}$ and a maximum delay of $d_{\max}$. If slack retention is used the processing delay d_i of some task is extended from $\overline{d}$ to $d_{\max}$ by slowing down the processor such that the new mean processing delay is $\overline{d}' \simeq d_{\max}$. To achieve this, the processor speed is reduced by a factor of: $s_i = \frac{d_i}{d_{\max}}$, where a factor of 1 means no reduction, i.e. full speed, and a factor of 0 means that the processor has ceased to operate. The mean speed reduction ratio is therefore $\overline{s} = \frac{\overline{d}}{d_{\max}}$.

The power consumed using a specific processor speed radio is given by Ks_i^2 where K is a constant. The mean power saved using slack retention per node is therefore

$$W_{\text{saved}} = W_{\text{before}} - W_{\text{after}} = Ks_{\text{before}}^2 - K\overline{s}^2 = K(1 - \overline{s}^2), \tag{3}$$

since $s_{\text{before}} = 1$ (maximum processor speed). The total power saved is therefore

$$W_{\text{total}} = (M \bmod P)K(1 - \overline{s}^2). \tag{4}$$

This holds for the situations where $N > P$, i.e. more-jobs limited-processors and $P > N$, i.e. limited-jobs multiple-processors, since W_{total} reduces to $NK(1 - \overline{s}^2)$. Clearly, the potential for reducing the power consumption in such a computing environment depends on the spread of the computation delay distribution, i.e. the mean is very different to the maximum, and the relationship between the number of jobs and the number of processing elements.

Figure 1 shows the relationship between the reductions in power consumption in relation to mean speed reduction. By reducing the speed by 10% for the case where $N < P$ then 20% of the power consumption is saved, and by reducing the speed by 30% then 50% of the power consumption is saved. However a speed reduction of 30% represents a large spread in the computation delay distribution. Furthermore, for the cases where $N > P$ then the ratio of jobs that are subjected to slack reduction serves as an upper bound for the theoretical achievable reduction in power consumption. i.e. if only 10% of the jobs are subjected to slack retention than it is obviously not possible to achieve more than 10% reduction in power consumption.

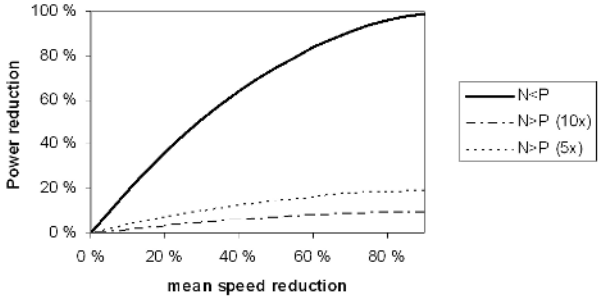


Fig. 1. Reduction in power consumption as a function of mean speed reduction

5 Promise Theory

Policy has two aspects for autonomous systems: voluntary cooperation (with attendant trust issues) and the logic of constraints. What policy rules are needed to encourage the individual agents to cooperate towards a common goal, without having to sacrifice their autonomy?

We use promise theory[1] to discuss this. Promise theory is an attempt to represent policy constraints using an atomic theory of assertions that makes no *a priori* assumptions about agents' willingness to comply with one another's wishes[16,17]. In promise theory, each assertion of policy is made from a source node to a receiver node, in the manner of a service provided. Promises are therefore labelled links in a graph of nodes representing the autonomous devices in the system.

Three main categories of promise are known to exist. These are basic service promises π, promises to cooperate with someone else's promise (denoted $C(\pi)$), and a promise to accept and use a service provided by another agent (denoted $U(\pi)$). The promises are essentially fixed in time (over many computational epochs). Promises do not represent separate events, or messages, but rather they summarize the entire history of interaction over the interval. Their role is to guide the behaviour of the agents in response to individual events or needs that arise in some unknown manner.

In the formulation of promises, one considers both the topology of the relationships between nodes and the nature of the constraints linking the nodes. Agents can form loose coalitions of nodes, or they can be tightly coordinated depending on the number of promises made. The promises from any sender s to any receiver r we need here are of the general form:

1. s promises r to accept jobs/tasks for completion with delay d_s and return the result.
2. s promises r to cooperate with a promise r makes to a third party, i.e. will behave the same as a neighbour in respect to the party.
3. s promises r to reimburse the power outlay r made on s's behalf, in some form of currency. This is payment, made perhaps in the form of a reduced

bill for power and cooling, or perhaps in some fictitious currency used simply to balance the energy accounts.

4. Additional promises could be numerous in regard to other policies concerned with the independent functioning of the devices, e.g. promises to report alarms to a central node, or to relay network traffic in an ad hoc network, etc.

Extensive details of the specific promise model are not possible within the space allowed. We mention only some summarial remarks.

Promise 1 is the basic promise for task sharing. Without further embellishment it enables nodes to perform power sharing. However, it easily allows nodes to be exploited too. A malicious node could easily send requests that are taken on trust in order to force the receiver to use up its power[2].

Promise 2 allows nodes to organize itself into a 'club' in which groups of nodes will behave similarly. This allows regions of autonomous devices to work together voluntarily, without sacrificing their autonomy.

Promise 1 can be made conditional on reimbursement, so that one programs a tit-for-tat relation into the interaction[18,19]. One might also make the promise contingent on a return promise that the sending node will actually use the result, having expended energy on it, but such requirements rapidly start to expend the energy budget on mistrust.

Could trust be exploited? Clearly there are several ways in which trust can be abused, both accidentally and by means of ill-conceived policy. If a sender sends a request at too high a power level, it might fool distant nodes into accepting a task that it too expensive to reply to. This would either waste their power, or they would not transmit a return signal strongly enough and the processing would be wasted. Thus a node could maliciously attack such a collective of mobile devices, forcing them to exhaust their batteries on pointless calculation. An explicit promise from all senders to all other receivers to transmit within an agreed power range would codify policy on this point. Alternatively, a reciprocal promise to fairly weight the tasks could be required by receivers that must be in place before expending significant power.

If a central node in the system has unlimited, cheap power then it might be used as an intermediary for brokering transactions between the individual agents. However, such a situation would not work for robotic probes on the Martian surface: in that case, all of the nodes must be considered as equal peers, none of which should be penalised. Control, in this case, must be distributed and based on implicit trust.

Mobile promises are the same as the fixed infrastructure ones, but they add additional issues. Again transmission power is an issue. Promises to stay in range of one another could have a significant saving effect[15,20].

Three scenarios are shown in fig. 2.

– In figure 2a, the nodes pick a representative node (e.g. a base station) within the nodes (or perhaps externally) and promise that representative to adjust their power according to an appropriate sharing algorithm. This binds the nodes together as a group or role in the graph. By the algebra of ref. [16],

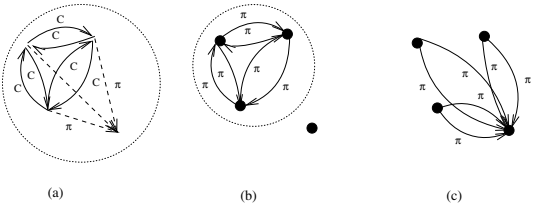


Fig. 2. Three management policy topologies

the collections of agents that make binding bilateral agreements represent a natural *role* within the graph. Typical promises between the agents are cooperative promises to one another to honour the promise to share (which is formally made to the arbiter).

– In fig.2b the agents promise to help one another, independently of an external arbiter. In doing so they place considerable trust in one another. What do they do if they fail to keep their promises to cooperate?
– In fig, 2c the nodes do not communicate directly at all, but always use an external authority to control the scheduling.

6 Promise 1 Body Remarks

Most literature on power aware scheduling assumes that there is one central power source. However, in a mobile setting there are strong reasons to argue for multiple power sources, where each processor has its own power source. In the context of wearable computers, or exploratory robotic agents, these processors can be distributed at different locations.

We denote these architectures according to the spirit of Flynns taxonomy: single instruction single battery (SISB), single instruction multiple battery (SIMB), multiple instruction single battery (MISB) and multiple instruction multiple battery (MIMB).

Many-jobs many-processors (MJMP): The many-jobs, many-processors scenario is typical of the traditional grid. A pervasive grid might expect that the overall job to be computed has a lifetime that exceeds the the battery life of the processing device. It is therefore natural to introduce the notion of a computational epoch, i.e. a subset or chunk of the overall computation that can be computed using the battery capacity available in the pervasive grid, and the overall goal is to maximize the amount of computation per unit of power. This scenario therefore can be reduced to a limited-jobs many processors scenario, where the limited jobs are the jobs that can fit into a computational epoch.

Reducing power consumption by slack reclamation is only applicable once the number of tasks is less than the number of processors; if the number of jobs is much larger than the number of processors the potential power savings are moderate. Given a tight power-budget a small overall reduction in performance can result in large power savings due to the speed-energy quadratic relationship.

The speed of all the processors should be adjusted proportionally to its load and capability, such that the performance scales evenly across the processing devices. This would require another collaborative promise.

Limited-jobs many processors (LJMP): In the limited-jobs many-processors scenario there is a known set of jobs and a large number of processing elements. The objective is to compute the result in an as short time as possible while at the same time maximize the uptime of the system. This is achieved by splitting the problem into as small chunks as possible for distribution. Based on the computational power and history, it is possible to estimate the finishing time from the size of the problem. According to Amdahls law the completion time of a problem is the sum of the parallel parts and the sequential parts, and clearly the overall problem cannot be solved faster than computation delay of the slowest device in use and consequently faster devices will finish the computation too early and perhaps remain idle. Instead, the clock speed of these devices can be slowed down such that the completion-time of computing the given task on the device matches that of the slowest device. The end result is that the result is computed in the same time, but the overall system has consumed less power.

Many-jobs limited-processors (MJLP): If one has more jobs than processors, each processor must execute more than one single job. Again given a variety of time-variant completion times for the various processing devices, jobs waiting to be computed are assigned processing devices that become idle. It is then only once the last task has been assigned that slack reclamation can be used. This thus becomes a many-jobs many-processors phase followed by a limited-jobs many-processors phase.

Limited-jobs limited-processors (LJLP): A limited-jobs limited-processors scenario can be reclassified as either being a limited-jobs many-processors, if there are the same or more processors than jobs, or a many-jobs limited-processors scenario, if there are more jobs than processors.

7 Conclusion

It is known that, in a regime of authoratative control over a group of devices, it it possible to make power savings to reduce the energy cost of parallel processing tasks. A group of autonomous agents can function similarly by voluntary cooperation in a pervasive computer setting without loss of autonomy.

References

1. Mark Burgess. An approach to policy based on autonomy and voluntary cooperation. *Lecture Notes on Computer Science*, 3775:97–108, 2005.
2. M. Burgess. Voluntary cooperation in a pervasive computing environment. *Proceedings of the Nineteenth Systems Administration Conference (LISA XIX) (USENIX Association: Berkeley, CA)*, page 143, 2005.
3. L. Benini, D. Bruni, A. Mach, E. Macii, and M. Poncino. Discharge current steering for battery lifetime optimization. *IEEE Transactions on Computers*, 52(8):985–995, 2001.

4. L. Benini et al. Extending lifetime of portable systems by battery scheduling. In *Proceedings of Design, Automation and Test in Europe 2001.*, pages 197 – 201, 2001.
5. L. Benini et al. Scheduling battery usage in mobile systems. *IEEE Transactions on Very Large Scale Integration (VLSI) System*, 11(6):1136–1143, 2003.
6. L. Bloom, R. Eardley, and E. Geelhoed. Investigating the relationship between battery life and user acceptance of dynamic, energy-aware interfaces on handhelds. *Lecture Notes on Computer Science*, 3160:13–24, 2004.
7. S. Gurumurthy, A. Sivasubramaniam, M. Kamdemir, and H. Franke. Reducing disk power consumption in servers with drpm. *IEEE Computer*, 36(12):59–66, 2003.
8. H. Aydin, R. Melhem, D. Mosse, and P. Mejia-Alvarez. Power-aware scheduling for periodic real-time tasks. *IEEE Transactions on Computers*, 53(5):584–600, 2004.
9. J.-J Han and Q.-H Li. Dynamic power-aware scheduling algorithms for real-time task sets with fault-tolerance in parallel and distributed computing environment. In *Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium*, pages 6–6, 2005.
10. D. Zhu, R. Melhem, and B. Childers. Scheduling with dynamic voltage/speed adjustment using slack reclamation in multi-processor real-time systems. In *Proceedings of 22th IEEE Real-Time Systems Symposium*, pages 84–94, 2001.
11. D. Zhu, R. Melhem, and B. Childers. Scheduling with dynamic voltage/speed adjustment using slack reclamation. *IEEE Transactions on Parallel and Distributed Systems*, 14:686–700, 2003.
12. O. Sinnen, L. Sousa, and F. E. Sandnes. Towards a realistic task scheduling model. *IEEE Transactions on Parallel and Distributed Systems*, 17(3):263–275, 2006.
13. F.E. Sandnes and O. Sinnen. A new scheduling algorithm for cyclic graphs. *International Journal of High Performance Computing and Networking*, 3(1):62–71, 2005.
14. F.E. Sandnes and O. Sinnen. Stochastic dfs for multiprocessor scheduling of cyclic taskgraphs. *Lecture Notes on Computer Science*, 3320:342–350, 2004.
15. J.M. Hecrickx et al. Rigidity and persistence of three and higher dimensional forms. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 39, 2005.
16. M. Burgess and S. Fagernes. Pervasive computing management i: A model of network policy with local autonomy. *IEEE eTransactions on Network and Service Management*, page (submitted).
17. M. Burgess and S. Fagernes. Pervasive computing management ii: Voluntary cooperation. *IEEE eTransactions on Network and Service Management*, page (submitted).
18. R. Axelrod. *The Complexity of Cooperation: Agent-based Models of Competition and Collaboration*. Princeton Studies in Complexity, Princeton, 1997.
19. R. Axelrod. *The Evolution of Co-operation*. Penguin Books, 1990 (1984).
20. J.M. Hecrickx et al. Structural persistence of three dimensional autonomous formations. In *Proceedings of the MARS 2005 Workshop on Multi-Agent Robotic Systems*, page 47, 2005.

CityVoyager: An Outdoor Recommendation System Based on User Location History

Yuichiro Takeuchi and Masanori Sugimoto

School of Frontier Sciences, The University of Tokyo
5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-8561 Japan
{takeuchi, sugi}@itl.k.u-tokyo.ac.jp

Abstract. Recommendation systems, which automatically understand user preferences and make recommendations, are now widely used in online shopping. However, so far there have been few attempts of applying them to real-world shopping. In this paper, we propose a novel real-world recommendation system, which makes recommendations of shops based on users' past location data history. The system uses a newly devised place learning algorithm, which can efficiently find users' frequented places, complete with their proper names (e.g. "The Ueno Royal Museum"). Users' frequented shops are used as input to the item-based collaborative filtering algorithm to make recommendations. In addition, we provide a method for further narrowing down shops based on prediction of user movement and geographical conditions of the city. We have evaluated our system at a popular shopping district inside Tokyo, and the results demonstrate the effectiveness of our overall approach.

1 Introduction

Today, recommendation systems are widely used in online shopping sites. They recommend items that match each customer's preferences and needs, which are estimated by analyzing their past activities at the site, for example which items they bought, or which items they showed interest in. Not only are they a helpful tool for customers, they are also beneficial for site owners in that they have the ability to increase the perceived credibility of the site[7].

But despite the numerous benefits, we believe that a single fact is severely limiting us from appreciating recommendation systems to their full potential: the single fact that recommendation systems are only used for shopping on the Internet, not for shopping in the city, or in other words, *in the real world*.

Since the majority of shopping activities is still done at real-world shops, applying recommendation systems to real-world shopping would be greatly beneficial for many people. The difficulty of this task lies in that in real-world shopping, it is extremely more difficult to acquire customer activity records needed for estimating preferences compared to online shopping.

In this paper, we introduce CityVoyager (Figure 1), a real-world recommendation system designed for mobile devices, which recommends shops to users based on data analyzed from their past location history. Location data can be easily

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 625–636, 2006.

Fig. 1. Using CityVoyager, a recommendation system for real-world shopping

acquired using means such as GPS or Wi-Fi, and it contains rich information about each user's personal preferences.

Our system effectively applies location data to the item-based collaborative filtering algorithm, a proven algorithm used in many online recommendation systems, by transforming location data history into a list that contains the names of each user's frequently visited shops and rating values which indicate how fond the user is of each shop. This list can be directly used as input to the filtering algorithm to make recommendations in the exact same manner as conventional recommendation systems. The transformation of data is done using our newly devised place learning algorithm, which can efficiently find users' frequented places complete with their proper names (e.g. "The Ueno Royal Museum"). No explicit user manipulation is required in the process.

In addition, our system is able to further narrow down shops based on prediction of user movement and geographical conditions of the city, such as the layouts of streets, resulting in more timely recommendations.

We have evaluated the performance of our system at Daikanyama, one of Tokyo's most popular shopping districts. The results demonstrate the effectiveness of our overall approach, although some aspects of the system still need further evaluation to be fully validated.

2 Related Work

2.1 Location Acquisition

Early location-based systems[14][15] have mainly used infrared waves for acquiring location. In more recent systems, Wi-Fi[4][5] is increasingly becoming popular as the method of choice, since in most cases no special equipments are needed, as many modern PCs and mobile devices are equipped with built-in W-Fi capabilities. Also, ultrasound[16] is used in situations where high precision (of up to a few centimeters) is needed.

In outdoor settings, GPS[1] is the most commonly used method for location acquisition. Wi-Fi is another popular choice, but at the current moment there are few areas with a network of access points dense enough for robust location acquisition. More recently, the use of Bluetooth and GSM has been investigated[9]. GSM has the advantage of its wide coverage, and Bluetooth is gaining popularity for its low power consumption.

2.2 Place Learning

At the core of our system is the place learning algorithm, which automatically finds users' frequently visited shops. Place learning has been a popular research topic these years, and several unique algorithms have been proposed. Marmasse and Schmandt's comMotion[11] identifies frequently visited places by keeping track of positions where GPS signals were lost. If signals were lost more than three times within a predefined radius, the system assumes the presence of a frequented place. Ashbrook and Starner[2] uses k-means clustering of visited places to find frequented places, where visited places are defined as places where GPS signals were continuously lost, or places where user movement was slower than one mile per hour.

2.3 Recommendation Systems

Several recommendation systems for real-world shopping have been proposed in the past, such as the personal shopping assistant of Asthana et al.[3]. But existing systems do not use the sophisticated filtering algorithms described below, and thus are unable to adapt themselves to detailed user preferences.

Content-based filtering[6][10] expresses the content of each data in a form that can be objectively evaluated, and filters out data whose content doesn't match the user's preferences. In the most common implementation, the content of each piece of data is expressed as a vector. The preferences of each user is also expressed as a vector, and filtering is done by comparing the similarities of the vectors. Content-based filtering is rarely used outside the recommendation of text data, due to the difficulty of defining effective content representations.

Collaborative filtering[8][13] recommends data that were given high ratings by a number of users who presumably possess similar preferences as the user who requested the recommendation. The biggest advantage of collaborative filtering is that it requires no previous knowledge of the content of the data, and systems for recommending various data have been developed, for example movies, music, etc. This algorithm is widely used for recommendations in online shopping sites.

3 System Overview

The basic function of our system is to estimate users' preferences from the history of their location data, and recommend shops upon request. Whereas online recommendation systems estimate users' preferences from their online activity records, such as items bought or checked in the past, our system estimates preferences based on their location history during shopping in the city. The system is intended to be useful for various kinds of shoppers in various situations. For example, the system can be helpful for shoppers new to an area wanting to find shops that match their tastes, or for shoppers more familiar to the area willing to try something new.

CityVoyager is basically designed to work with GPS receivers, which we believe is a reasonable choice given the current state of technology. However, our

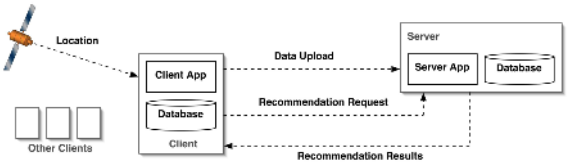


Fig. 2. The CityVoyager system architecture

place learning algorithm can work with any location acquisition technique, as long as visits to shops can be detected. This fact enables CityVoyager to take advantage of introductions of future technologies. For example, we can modify our system to use Wi-Fi as the underlying technology if we detect visits by looking for instances where the user location stays roughly consistent throughout a stretch of time.

Figure 2 illustrates the system architecture. The main components of our system are client devices carried by users, and a server that performs recommendations. Any device which can be equipped with Internet connection and GPS capabilities can be used as the client device. Potential client platforms include PDAs, notebook PCs and mobile phones.

Below, we describe the process in which recommendations are made. The process consists of the place learning phase and the recommendation phase.

3.1 Place Learning Phase

In the place learning phase, raw location data from GPS is reconstructed into a list of each user's frequently visited shops. This process can be further divided into two sub-phases: detecting visits to shops, and finding frequented shops.

Detecting Visits to Shops. We use the unavailability of GPS signals as evidence that the user has gone indoors. GPS signals cannot penetrate through most building walls, so visits can be detected fairly accurately using this method. But since GPS signals frequently become lost in urban areas even when the user is outdoors, we must be aware of the possibilities of false detections. The system judges that the user has visited a shop when GPS signals are continuously unavailable for a period of time longer than a threshold. The system then records the location of the visit, and also records the length of time that signals were lost, as the approximate duration of the visit.

Finding Frequented Shops. Following the above procedure, we can plot the locations of a user's past visits. By analyzing these plots of past visits, we can reveal which shops the user has frequently visited. Unfortunately, this task is more difficult than it seems, since GPS is known to produce errors, of around 10 meters in clear-sky conditions and significantly larger in urban areas. A cluster

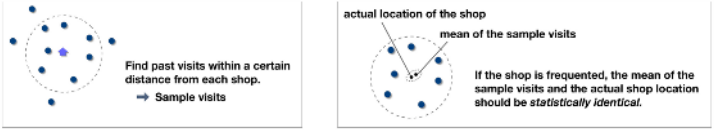


Fig. 3. Our place learning algorithm

of past visits implies the existence of one or more frequented shops in the nearby area, but we could not know exactly *which* shop(s) the user has frequently visited. The best we can do is to estimate the frequented shops, and this is where our new place learning algorithm comes into use.

While previous algorithms look at the entire collection of past visits and try to find clusters of visits, our algorithm focuses on a particular shop and the recorded visits nearby that shop, and then try to judge whether that shop was frequently visited by the user or not. Therefore, our algorithm is able to give the proper names of frequented places (e.g. "The Ueno Royal Museum" or "The Univ. of Tokyo Hongo central cafeteria"), while previous algorithms could only output latitude and longitude values. By knowing the proper names of frequented shops, our system is able to compare one user's frequented shops with those of other users, which forms the basis of our filtering algorithm. In order to achieve the same function using previous algorithms, one would have to take an extra step of mapping latitude and longitude values with their proper names. This extra step will inevitably result in lower accuracy of the overall procedure than that of the place learning algorithm itself.

The actual procedure of our algorithm is as follows. First, for each shop in the database (the system must have a database containing the names and the locations of the shops in the area) the system searches for past visits within a predefined distance from the location of the shop. We call these visits *sample visits*. Then, by applying two-tailed t-test to the sample visits, the system tries to judge if it is plausible that the shop was frequently visited by the user. In other words, if the assumption that the shop was frequented by the user reasonably explains the past visits recorded nearby that shop, the system judges that the shop was frequently visited by the user.

The algorithm is based on the assumption that the observed latitude and longitude values of a visit to a certain shop follow a normal distribution, with the actual shop location as the mean. Given this assumption, the latitude and longitude values of the sample visits around a shop will follow t-distributions. Thus, we can use t-test to evaluate if the means of the t-distributions can be regarded as *statistically identical* to the actual location of the shop. If the shop is a frequently visited shop, the two should be statistically identical (Figure 3).

If the system judges a shop as frequented by the user, the shop is included in the user's frequented shops list, with a rating value calculated from the number of sample visits and the average duration of those visits. This list is stored in the server and is used for recommendation.

3.2 Recommendation Phase

Upon user request, the server recommends shops using the lists obtained in the previous phase. Recommendation is done through two steps described below.

Filtering. First, shops are filtered using the item-based collaborative filtering algorithm[12]. Similarities between shops are calculated as below.

$$Sim(A,B) = \frac{\sum_u R_{u,A} R_{u,B}}{\sqrt{\sum_u {R_{u,A}}^2}\sqrt{\sum_u {R_{u,B}}^2}} \tag{1}$$

Here, $R_{u,A}$ indicates the rating value for shop A by user u, and $R_{u,B}$ indicates the rating value for shop B by user u. The similarity increases when there is an observed tendency that users who frequently visit shop A also frequently visit shop B. Since the algorithm does not take into account the content of the data, correlations between shops of different categories, such as restaurants and clothing stores, can be defined. The system picks out several shops which have high similarities with the user's frequently visited shops. These shops are regarded as having high chances of matching the user's preferences.

Calculating similarities does not need to be done frequently. The similarities reflect users' long-term shopping activities, and thus it can be reasonably assumed that their changes within a short amount of time are subtle.

Adding Weights According to Areas. Our system is for use in the city, which means there will be physical distances between users and recommended shops. In cities like Tokyo where many people shop on foot, the distances can easily become too demanding for users. Therefore, we must make sure that the recommended shops are easily accessible from the users. To meet this requirement, we first divide the city map into *areas*. Areas are defined so that any two points located in the same area are easily accessible from one to the other. Our algorithm for this task, based on cluster analysis, is as follows:

```
1.  Divide the city map using a square grid. The grid should be fairly dense.
2.  Pick the grid elements which are located on places that can be walked through,
    such as on streets, and define them as initial clusters.
3.  For all combinations of two clusters, do{
4.      For all combinations of two grid elements in the cluster, do{
5.          Calculate the Manhattan distance between the two grid elements.
            Keep a record of the calculated distances.
6.      }
7.      Look for the two grid elements that give the largest distance, and define their
        distance as the accessibility of the combination of clusters.
8.  }
9.  Merge the two clusters that combine to make the smallest accessibility.
10. Repeat 3 - 9 until the number of clusters become sufficiently small.
```

The resulting clusters are chosen as the areas. Areas should be redefined anytime there are significant changes in the city landscape, such as openings of new streets, etc. Next, we model the user's movement using a first-order Markov

Fig. 4. CityVoyager screenshots

model, with the areas as nodes. Transition probabilities are calculated from periodically plotted user locations. Higher transition probability indicates more chances of the user advancing to the area. The shops picked out by the filtering algorithm are weighted according to the areas in which they are located, with large weight values being added to shops in the same area as the user, or in areas with large transition probabilities. A few shops with the largest weights are picked out, and presented to the user as the final recommendations.

4 Implementation

As the client device for our initial implementation of CityVoyager, we used a Toshiba Genio e 550G Pocket PC 2002 PDA, equipped with a GPS receiver and a CF-type card for connecting to the Internet using PHS. Since the memory and computational capabilities of our client device is rather weak, transforming location data into a list of frequented shops has to be done in the server, and thus raw location data must be sent through the network, which generates significant privacy risks. Since our place learning algorithm is computationally inexpensive, this should be avoidable in future implementations by using more powerful hardware.

The software consists of the main client application (C++), the client user interface (Macromedia Flash), and the server application (Java Servlet). Client-server communication is done by sending and receiving encrypted XML sentences through TCP/IP sockets, via dial-up Internet connection. A MySQL database is installed in the server, containing information of 213 shops located within a small area inside Daikanyama, Tokyo. Information of shops were acquired through magazines, websites, and by actually walking around the area and checking for missing shops. We believe that we managed to construct a fairly complete list of shops, albeit within a small area. The database also contains each user's list of frequented shops, and transition probabilities for the Markov models.

The user interface consists of several screens (Figure 4), each serving different functions to the user. Users can send recommendation requests to the server by tapping on "request recommendation" from the menu. Recommended shops will be displayed in a list, and will also be shown using star-shaped icons on the map.

5 Evaluation Test

We conducted an evaluation test of CityVoyager at Daikanyama, Tokyo. The test was carried out in two phases. In the first phase, we recorded long-term location data for a number of users and evaluated how well our system could find users' frequently visited shops. In the second phase, we evaluated the effectiveness of our system's recommendation results.

5.1 Place Learning Phase

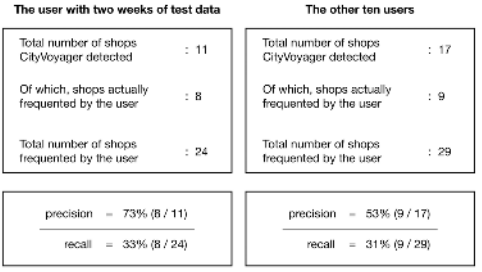
We asked 11 users (ages 18 to 25, 7 male and 4 female) to shop freely inside our test area at Daikanyama, with our client devices in their bags. Due to the limited time available for the test, the length of time we could use for the test varied greatly with the user. For one of the users, we were able to use two weeks of test time, which should be sufficient considering that our system is intended for learning frequented shops, not for learning everyday places like home or office. For example, if we suppose that the average person goes shopping on a pace of once a week, gathering two weeks worth of shopping data would take approximately three and a half months. Our view is that if a place learning system still cannot yield usable results using that amount of time, the system is practically useless. But for each of the other ten users, we were only able to use three to six hours for the test, spanning over a period of one or two days.

In addition to enjoying shopping and letting CityVoyager automatically gather data, we also asked users to write down the names of the shops they visited into a notebook. After the test, we compared the frequented shops detected by CityVoyager with the shops actually frequented by each user.

As the parameters for our place learning algorithm, we set the rejection region of the two-tailed t-test to 10%, and the radius used for picking out each shop's sample visits to 25 meters.

We analyzed the test results using the evaluation framework proposed by Zhou et al[17], which is an extension of the traditional metrics used in evaluating information retrieval systems, modified to suit place learning systems. The framework introduces three metrics: precision, recall, and surprise factor. When applied to our system, these metrics have the following meaning. Precision is the percentage in which a shop detected by CityVoyager was actually frequented by the user, and recall refers to what percentage of the users' actual frequented shops were successfully detected by CityVoyager. Surprise factor refers to the percentage in which CityVoyager detected shops that were actually frequented by a user, but the user failed to report. Since in our evaluation test we kept track of every visit the users made, the surprise factor is always zero.

The left two tables in Figure 5 show the results of the place learning phase test. For the one user with whom we collected two weeks of test time, we defined actual frequented shops as shops which the user has visited 3 or more times during the test period. For this user, there were 24 actual frequented shops in all. But for the other ten users, due to the severely limited test time, we had to lower the bar and define frequented shops as shops which a user has visited 2



	User 1's ratings	User 2's ratings	Average ratings (standardized)
CityVoyager	1 3 7	4 6 3	**4.08**
Shopping without any guides	5 4 3	2 6 3	**3.75**
Conventional location-aware	1 6 3	4 6 6	**4.67**

Fig. 5. Evaluation test results

or more times. The fact that we used different conditions make it impossible to treat the test results of these two groups of users equally, and thus the results must be discussed separately. Discussion will be provided in the next section.

5.2 Recommendation Phase

To evaluate the effectiveness of our system's recommendations, we compared our system with two other methods. One was normal shopping, without the use of any guides. The other was conventional location-aware recommendation, like the services available on mobile phones, in which shops closest to the user's current location are presented. We asked 2 users (age 24, male and age 25, female) to visit three shops for each of the above three methods, and give each shop a rating value in a scale of seven points. A scale of seven is commonly used in surveys, since it is known that reliabilities of subjective ratings do not increase dramatically with scales of more than seven. Both of the 2 users were from the group of users for which we could only spare three to six hours of test time per user in the place learning phase. The table in the far right in Figure 5 shows the results of the test. We will discuss these results in the next section.

6 Discussion

6.1 Place Learning Phase

First, we discuss the results we acquired from the user with whom we collected two weeks worth of shopping data. The recall (33%) may seem quite low, but this is due more to the technical limitations of the GPS than to our place learning algorithm. As we have conducted our evaluation test in a shopping district in Tokyo, famous for being crowded with buildings, signals were frequently lost even when the user was outdoors. This not only led to false detections of visits, but also caused many actual visits to be undetected, since when a user enters and leaves a shop, and enters another before GPS signals become available again, these two visits will be detected as one long visit. The recorded visits show that for 12 of the 24 shops actually frequented by the user, no sample visits were obtained. This means that visits were not recorded anywhere near those 12 shops,

even though the user has actually frequented those shops. If we only look at the 12 shops where sample visits *were* obtained, 8 of those shops were successfully detected, which yields the recall rate of 67%. This implies that the low recall is caused by the limitations of GPS itself, not by a flaw in our algorithm. Fortunately, our place learning algorithm can be applied to other location acquisition techniques with slight modifications, so a higher recall value may be obtained by using alternative technology with better coverage than GPS.

The relatively high precision (73%) came as a surprise to us. Because of the frequent signal loss inside the test area, and because we set the threshold for detecting visits to 5 minutes, shorter than conventional systems, we anticipated many false detections of visits. Actually, there really were many false detections of *visits*, but the number of falsely determined *frequented shops* somehow stayed low. The reason for this may be that false detections of visits do not occur at fixed places, and so they seldom form a cluster of recorded visits necessary for a shop to be judged as frequented.

Next, we discuss the results acquired from the other ten users, for each of whom we could only spare a day or two of test time. Comparing the numbers (precision: 53%, recall: 31%) with the results of the user with whom we collected two weeks of data, we can see that while the precision is significantly lower, the recall is approximately the same. Since we defined "frequented shops" differently between these two groups of users we cannot simply compare the resulting metrics, but we believe it is fair to say that precision increases with the length of time used for collecting data.

Obviously, the performance of the place learning algorithm depends on the geographical conditions of the test area. While we have yet to conduct a thorough evaluation in other shopping districts, we have gathered a small amount of data that may serve as a reference, by tracking the GPS location history of one of the authors of this paper through three days of his life at Kashiwa campus, The University of Tokyo. The parameters used for the place learning algorithm is exactly the same as the test conducted at Daikanyama, and frequented buildings were defined as buildings which the author has visited 3 or more times. As a result, all three of the author's frequented buildings were correctly determined without any false detections, which amounts to perfect precision and recall. This result is obviously due to the sparseness of buildings around our campus, and is by no means intended to serve as a formal evaluation. But it seems reasonable to assume, that if our system was used inside shopping areas that are not as crowded as Daikanyama but more so than the area around Kashiwa campus (which we believe is a category most shopping areas would fall into), the resulting metrics are likely to end up somewhere between the results acquired in these two areas.

6.2 Recommendation Phase

The results of the recommendation phase test show that, the average rating value for shops recommended by our system is higher than that for shopping without external guides, but lower than that for conventional location-aware services.

However, to our disappointment, these results were found to be statistically insignificant, and does not prove if our system is superior or inferior to the other methods. Obviously, the lack of test time (and quite possibly, the number of users) we could use in the place learning phase was largely responsible for this.

Let us finish the discussion by pointing out the advantages of using our system from a nonstatistical point of view. Shopping without external guides may be effective for finding good shops, but only if there is plenty of time. In a city like Tokyo, with a vast number of shops and notoriously complicated streets, finding shops that match personal tastes will require much time and effort. CityVoyager can help in such situations by narrowing down the search space. Conventional location-aware services give shops that are closest to the user's current position, which should be helpful in areas where shops are relatively sparse. But in areas like Daikanyama, there are countless shops within walking distance, and presenting shops closest to the user will not be so much of a help. CityVoyager should be able to make more effective recommendations in such cases, by picking out shops according to personal tastes and preferences, not just geographical conditions.

7 Conclusion and Future Work

In this paper, we proposed CityVoyager, a shop recommendation system for real-world shopping based on user location history. The results of the evaluation test show that our place learning algorithm has the potential to estimate users' frequently visited shops with considerably good accuracy. However, we could not validate the effectiveness of our recommendations, and gathering enough data to acquire statistically significant results remains our most important future work. Some other important aspects of our system must also be evaluated. For example, since shopping behavior differs considerably between users, we need to gather data about how different shopping behaviors affect the accuracy of the system. Also, it would be useful to evaluate how our place learning algorithm performs under extreme conditions, such as frequently visiting two very small shops that are located next to each other, and seeing if the algorithm can detect both of these shops as frequented.

One modification which should be definitely done in future versions is introducing some constraints to the recommendation results regarding some basic information about the user, such as age or sex. In our evaluation test, a shop which only sells fashion items for women was recommended to a male user, and consequently received a rating of 1. Obvious mismatches like this should be excluded from the recommendation results, by defining several basic attributes for each shop and filtering out data with unwanted attributes. Also, there are some types of data not exploited in our current implementation that may be useful, such as walking distance or the time of day. For example, we can recommend restaurants when the recommendation was requested around noon, or recommend cafes if the user has walked long distances.

References

1. Abowd, G.D., Atkeson, C.G., Hong, J., Long, S., Kooper,R., Pinkerton, M., Cyberguide: A mobile context-aware tourguide, ACM Wireless Networks, 1997, pp.421-433.
2. Ashbrook, D. and Starner, T. Learning signifcant locations and predicting user movement with GPS. In Proc. of 6th IEEE Intl. Symp. on Wearable Computers, 2002.
3. Asthana, A., Cravatts, M., and Krzyzanouwski, P. An Indoor Wireless System for Personalized Shopping Assistance. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pp. 69-74, IEEE Computer Society Press.
4. Bahl, P., and Padmanabhan, V. N. RADAR: An In-Building RF Based User Location and Tracking System. In INFOCOM 2000, pages 775–784, 2000.
5. Bahl, P., and Padmanabhan, V. N. A Software System for Locating Mobile Users: Design, Evaluation, and Lessons. MSR Technical Report, 2000.
6. Chen, L., and Sycara, K. WebMate: Personal Agent for Browsing and Searching. Proceedings of the 2nd International Conference on Autonomous Agents and Multi Agent Systems, AGENTS '98, ACM, 1998, pp. 132 - 139.
7. Fogg, B. J., Persuasive Technology : Using Computers to Change What We Think and Do. Morgan Kaufmann Publishers, 2003.
8. Goldberg, D., Nichols, D., Oki, Brian M., Terry, D., Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, v.35 n.12, p.61-70, 1992.
9. LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., and Schilit, B. Place Lab: Device Positioning Using Radio Beacons in the Wild. In 3rd Annual Conference on Pervasive Computing, 2005.
10. Lang, K. NewsWeeder: Learning to Filter Netnews. In Proceedings of 12 th International Conference on Machine Learning, Lake Tahoe, CA, Morgan Kaufmann, pp. 331-339, 1995.
11. Marmasse, N. Schmandt, C. Location-aware Information Delivery with Commotion. In Proceedings of the 2nd international symposium on Handheld and Ubiquitous Computing, Pages: 157 - 171, 2000.
12. Sarwar, B., Karypis, G., Konstan, J., Riedl, J Item-based Collaborative Filtering Recommendation Algorithms. In Proceedings of 10th International World Wide Web Conference, ACM Press, 2001, pp. 285-295.
13. Shardanand, U., Maes, P., Social Information Filtering: Algorithms for Automating "word of mouth". In Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems, (pp. 210-217).
14. Want, R., Hopper, A., Falcao, V., and Gibbons, J. The Active Badge Location System. ACM Transactions on Information Systems (TOIS), v.10 n.1, p.91-102, Jan. 1992.
15. Want, R., Schilit, B., Adams, A., Gold, R., Petersen, K., Goldberg, D., Ellis, J., and Weiser, M. The ParcTab Ubiquitous Computing Experiment. Technical Report CSL-95-1, Xerox Palo Alto Research Center, March 1995.
16. Ward, A., Jones. A., Hopper, A. A New Location Technique for the Active Office. In IEEE Personal Communications, Vol. 4, No. 5, pp 42-47, 1997.
17. Zhou, C., Frankowski, D., Ludford, P., Shekhar, S., Terveen, L. Discovering Personal Gazetteers: An Interactive Clustering Approach. In Proceedings of ACM GIS 2004, 2004, pp. 266-273.

Energy Saving of Mobile Devices Based on Component Migration and Replication in Pervasive Computing*

Songqiao Han, Shensheng Zhang, and Yong Zhang

Department of Computer Science and Engineering
Shanghai Jiaotong University, Shanghai, P.R. China
{hansq, sszhang, zycs926}@sjtu.edu.cn

Abstract. Energy is a vital resource in pervasive computing. Remote execution, a static approach to energy saving of mobile devices, is not applicable to the constantly varying environment in pervasive computing. This paper presents a dynamic software configuration approach to minimizing energy consumption by moving or/and replicating the appropriate components of an application among the machines. After analyzing three types of energy costs of the distributed applications, we set up a math optimization model of energy consumption. Based on the graph theory, the optimization problem of energy cost can be transformed into the Min-cut problem of a cost graph. Then, we propose two novel optimal software allocation algorithms for saving power. The first makes use of component migration to reasonably allocate the components among the machines at runtime, and the second is to replicate some components among machines to further save more energy than component migration. The simulations reveal that the two proposed algorithms can effectively save energy of mobile devices, and obtain better performance than the previous approaches in most of cases.

1 Introduction

Mobile devices coupling with wireless network interface, such as palmtop computers, Personal Digital Assistants (PDAs), mobile phone, and digital cameras are becoming increasingly ubiquitous[1]. However, as the mobile devices become more widely used for more advanced applications, their resource limitations are becoming more apparent. Among the resources, power is vital resource for pervasive computing, because it is the only perishable resource –once consumed, it cannot be replenished by actions performed within the pervasive computing system. Incremental improvements of battery technology are likely, but some of those improvements will be eaten up by the energy demands of increased functionality in mobile devices [2].

It is therefore essential to explore alternative approaches to extending battery life. Besides the approach to improving hardware power efficiency, one effective approach is to perform cyber foraging by offloading work to nearby servers. A typical example of this approach is remote execution that moves most of tasks of an application to the resource-rich server before its execution time. Recently, the idea of using remote

* This work is supported by grants 05SN07114 and 03DZ19320, all from the Shanghai Commission of Science and Technology.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 637 – 647, 2006.

execution to save power has been explored by some significant simulations and experiments[3] that show the effectiveness of remote executions. Besides, some researchers [4,5,6] presented some static algorithms of job scheduling or program partitioning to allocate one or more applications before the applications start to run. To reallocate an application at runtime, Chen et al [7] proposed adaptive execution strategy and adaptive compilation strategy to decide statically or dynamically where to compile a method, and how to execute it, to get better energy conservation. But these algorithms often can neither guarantee to get the optimal solution, nor consider energy conservation potential of component replication.

Motivated by the need of adapting to the ever-changing environment in pervasive computing, we propose two software allocation approaches that can allocate statically or dynamically an application between a mobile device and a server, aimed at saving more battery power of mobile devices. The first is to move the appropriate components among machines in order to minimize the energy consumption. Although component migration would consume some energy, it may reduce energy costs that are induced by the communication between machines or/and the computation on the mobile device. To further save energy, the second approach is to replicate some components between machines without interrupting the logic of the application. In essence, component replication makes the granularity of the replicated components smaller, which provides more potential to assign reasonably these components. For the two approaches, we present the two optimal software partitioning algorithms for component migration and replication respectively. By using the network flow theory, we can transform the problems of optimal software partitions into the bipartition problems of the cost graphs, which can be further transformed into the Min-cut problems of the corresponding flow network that can be easily solved by some well-known graph algorithms.

The rest of this paper is organized as follows: Section 2 set up a power cost model of mobile devices. In Section 3, we present the optimal partitioning algorithm of cost graph with component migration. Section 4 proposes the optimal partitioning algorithm of cost graph with component replication. The simulation results are presented in Section 5. Finally, Section 6 concludes this paper.

2 Energy Cost Math Model

2.1 Energy Cost of Mobile Devices

From the software perspective, we classify the power cost of mobile devices into three types of costs: computation cost, communication cost and migration cost. Computation cost is the cost incurred by the computation of the application. Sending and receiving data between the components on different hosts often consume significant energy, which produces communication cost. But communication cost of the components within a same host is trivial, thus ignored. Migration cost is the cost induced by moving components over wireless network. Although it is still a hard problem to accurately measure these costs, some researches[8, 9] have demonstrated it is feasible to estimate them by energy estimation interfaces or profiles.

Suppose that there are a resource-constraint mobile device, called client, and nearby resource-rich computer, called server, and both of them are connected by wireless

network. To characterize the energy costs of computation, communication and migration, we define the following parameters.

p_i and p_c: mean power consumption rates when mobile device is idle and performs computation, respectively;

p_s and p_r: mean power consumption rates when mobile device sends and receives data, respectively;

b_s and b_r: available sending and receiving bandwidths of the wireless network, respectively;

$t_c(c_i)$ and $t_s(c_i)$: execution times of component c_i on client and server, respectively;

$s(c_i)$: the size of component c_i;

$s(c_i, c_j)$: the size of data transferred from component c_i to c_j;

S_c and S_s: the sets of components locating on the client and server, respectively;

$S_{c->s}$ and $S_{s->c}$: the sets of components that migrate from client to server, and vice versa, respectively.

Then the computation energy costs of component c_i:

$$\begin{cases} e(c_i) = p_c \times t_c(c_i) & c_i \in S_c \\ e(c_i) = p_i \times t_s(c_i) & c_i \in S_s \end{cases} \tag{1}$$

The computation cost of the application:

$$E_c = \sum_{c_i \in S_c} p_c \times t_c(c_i) + \sum_{c_i \in S_s} p_i \times t_s(c_i) \tag{2}$$

Communication energy costs occurred by sending and receiving data from component c_i on the client to component c_j on the server:

$$\begin{cases} e_s(c_i, c_j) = s(c_i, c_j) \times p_s / b_s \\ e_r(c_i, c_j) = s(c_i, c_j) \times p_r / b_r \end{cases} \tag{3}$$

Communication energy cost between the client and server:

$$E_{sr} = \sum_{c_i \in S_c, c_j \in S_s} [\frac{s(c_i, c_j) \times p_s}{b_s} + \frac{s(c_j, c_i) \times p_r}{b_r}] \tag{4}$$

Component migration costs induced by moving component c_i from the client to server, and vice versa:

$$\begin{cases} e_{c \to s}(c_i) = s(c_i) \times p_s / b_s \\ e_{s \to c}(c_i) = s(c_i) \times p_r / b_r \end{cases} \tag{5}$$

Component migration cost between the client and server:

$$E_m = \sum_{c_i \in S_{c \to s}} \frac{s(c_i) \times p_s}{b_s} + \sum_{c_i \in S_{s \to c}} \frac{s(c_i) \times p_r}{b_r} \tag{6}$$

Thus the total energy cost E after the application runs N cycles.

$$E = NE_c + NE_{sr} + E_m \tag{7}$$

2.2 Energy Saving Problem Formulation

The software architecture of an application can be represented by a directed graph $G = (V, E)$ with vertex set V and edge set $E \subseteq V \times V$, where vertex v_i represents component c_i and edge e_{ij} indicates that component c_i depends on component c_j. In order to denote the three types of energy costs, we add the weights to the vertices and the edges of the graph, which is called cost graph. The weight of vertex v_i is represented by a triple set $(C_m(c_i), C_c(c_i), C_s(c_i))$, where $C_m(c_i)$ denotes the migration cost of component c_i that is calculated using Formula 5, $C_c(c_i)$ or $C_s(c_i)$ represent its computation cost on the client or server that is obtained from Formula 1. The weight of edge e_{ij} is notated by $C_c(c_i, c_j)$ whose value is equal to the communication cost induced by component c_i invoking c_j, obtained from Formula 3. Note that if a component can't run on a host, its computation cost on the host will become infinite.

Therefore, the problem of optimal component allocation on the client and server for energy cost minimization can be transformed into the optimal bipartition problem of the cost graph, so that the minimum cost Formula 8 is satisfied. Suppose S_c' and S_s' are the new sets of components on the client and server after partitioning the application, respectively, and S_m is the set of migrating components.

$$Min\left(\sum_{c_i \in S_c'} C_c(c_i) + \sum_{c_j \in S_s'} C_s(c_j) + \sum_{c_i \in S_c', c_j \in S_s'} C_c(c_i, c_j) + \sum_{c_k \in S_m} C_m(c_k)\right) \tag{8}$$

3 Partitioning Algorithm with Component Migration

For the cost graph of an application, we need construct its Min-cut algorithm to satisfy the minimum cost formula (Formula 8). Although the well-know Max-flow Min-cut theorem can find an optimal solution of a flow network bipartition problem, the cost graph is not a flow network, thereby unable to directly use the theorem. So we transform the cost graph into an equivalent flow network that ensures that the capacity of cut edge set is equal to total energy consumption of the mobile device.

Suppose that an application is comprised of three components, called A, B and R. According to its software architecture, we can construct a corresponding cost graph $G = (V, E)$, as shown in Fig.1, by using Formulas (1, 3, 5). Initially, component B locates on the client while other two components, A and R, locate on the server. The cost graph can be transformed into a flow network $G' = (V', E')$, as shown in Fig.2, according to the following steps.

STEP 1. Each node $v \in V$ in graph G is transformed into a node $v' \in V'$ in network G'.

STEP 2. Add the edges and their capacities in network G' adhering to the rules: if there exists one directed edges e_{ij} between node v_i and v_j in graph G, we add two edges e_{ij}' and e_{ji}' between node v_i' and v_j' in network G', and their capacities both are equal to the weight of e_{ij}. If there exist two directed edges e_{ij} and e_{ji} in graph G', we add two edges e_{ij}' and e_{ji}' whose capacities both are equal to the sum of the two weights of e_{ij} and e_{ji}.

STEP 3. Add the node labeled S and T that represent the client and server machine, respectively, where S denotes the unique source node and T denotes the unique sink node in network G'.

STEP 4. For each node other than *S* and *T*, add an edge from *S* to that node and an edge from that node to *T*. The capacities of the newly added edges may be different due to the different initial location of the component corresponding to the node. Let *cap*(*e*) be the capacity of edge *e*, then the capacities of the edges in network *G'* can be obtained using the following expressions.

Component c_i	$cap(e(S, c_i))$	$cap(e(c_i, T))$
$c_i \in S_c$	$C_s(c_i) + C_m(c_i)$	$C_c(c_i)$
$c_i \in S_s$	$C_s(c_i)$	$C_c(c_i) + C_m(c_i)$

STEP 5. Using a Min-cut algorithm, such as Preflow-Push algorithm which has $O(N^3)$ complexity where *N* is the number of components, to cut network *G'*, thereby obtaining two disjoint node sets V_s and V_t and $V' = V_s \bigcup V_t, S \in V_s, T \in V_t$. The components corresponding to the nodes other than *S* in V_s are assigned to the client and the components corresponding to the nodes other than *T* in V_t are assigned to the server. The minimal energy consumption is equal to the weight of a min-cut set or the value of Max-flow in network *G'*.

A cut edge is an edge (*u, v*) where $u \in V_s$ and $v \in V_t$. The cut set is the set of all cut edges in network *G'*. For example, the cut set denoted by the dashed line is {e_1', e_6', e_{rt}, e_{bt}, e_{sa}} in Fig.2, so its weight is equal to $cap(e_1') + cap(e_6') + cap(e_{rt}) + cap(e_{bt}) + cap(e_{sa})$. For more explanations of this algorithm, please refer to [10].

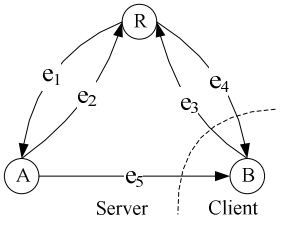


Fig. 1. Cost graph with nodes allocation

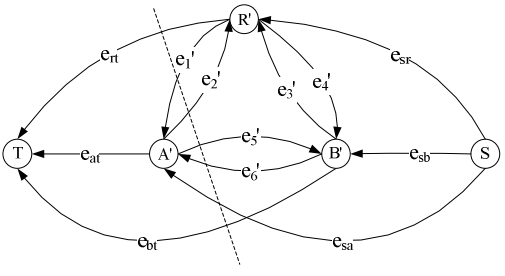


Fig. 2. Flow network transformed from cost graph

4 Partitioning Algorithm with Component Replication

Although component replication can produce some migration cost, a reasonable component replication may reduce execution cost or / and communication cost. From the software logic perspective, a component can be divided into a few subcomponents which have same or different sub functions of the component. Give an application, if there exist *k* components $V_1 = \{v_1, v_2, \ldots, v_k\}$ to invoke a component *R*, then we think that component *R* consists of *k* subcomponents $S_R = \{R_1, R_2, \ldots, R_k\}$, so that each components v_i invokes one corresponding subcomponent R_i rather than component *R*. The weight of edge $e(v_i, R_i)$ is equal to the weight of edge $e(v_i, R)$. However, component

R_i may have several out edges which are decided according to the component invocation sequence diagram. For example, Fig.3 shows the invocation sequence diagram of the cost graph in Fig.1. Suppose both component B and R can be replicated between machines. Firstly, we divide component B into subcomponents B_1 and B_2, and component R into subcomponents R_1 and R_2. Fig.3 shows the whole execution procedure of the application consists of two method invocations: one is $e_5 \rightarrow e_3 \rightarrow e_{12}$, and the other is $e_2 \rightarrow e_{41} \rightarrow e_{11}$. Then the edges are added to connect the corresponding subcomponents in Fig.4, and their weights are obtained by Formula 3.

Because the task set of a component is equal to the union of the task sets of its subcomponents, the sum of the computation costs of all its subcomponents must be equal to the computation cost of the component. For example, $\sum_{R_i \in S_R} C_c(R_i) = C_c(R)$ and $\sum_{R_i \in S_R} C_s(R_i) = C_s(R)$, where the value of $C_c(R_i)$ and $C_s(R_i)$ can be obtained using Formula 1 or the profiling technology.

When a cost graph with component division is partitioned by a min-cut algorithm, it is possible for some subcomponents of a component are assigned to host1, but other subcomponents of this component are assigned to host2, which indicates this component needs replicate itself between the two hosts. Therefore, no matter which subcomponents move to a new host, the sum of their migration costs must be the migration cost of the ancestor component. Then a challenging problem is how to assign the migration cost of the ancestor component to the subcomponents.

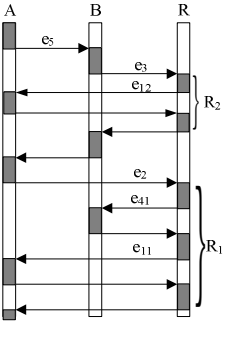


Fig. 3. Component invocation sequence diagram

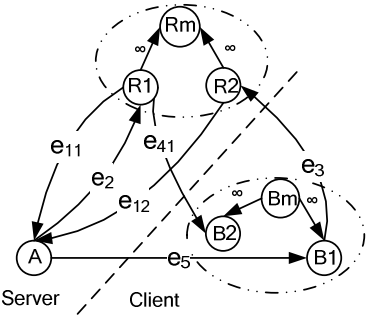


Fig. 4. Cost graph after component division

We assign zero to migration cost of each subcomponent, and turn to another way to represent this type of cost. We introduce a migration subcomponent R_m of component R, to stand for the migration cost of component R. Subcomponent R_m is a logical subcomponent which does not implement any function, so its node weight in cost graph is ($C_m(R_m)$, *0*, *0*), where $C_m(R_m) = C_m(R)$. In addition, for each subcomponent R_i, if component R initially locates on the server, we add edge $e(R_i, R_m)$ with infinite edge weight, but if R initially locates on the client, we add edge $e(R_m, R_i)$ with infinite edge weight. Fig. 4 shows the cost graph G_m after component B and R is divided. Further, the graph G_m can be transformed into a flow network G_f according to transformation steps

Section 3. The following lemma guarantees the validity of the above approach. Due to space limitations, we omit the proofs of the following lemma, theorem and corollary. The interested readers are referred to Reference [10] for details.

Lemma 1. If network G_f is partitioned by the Max-flow Min-cut theorem, so that one or more subcomponents R_i of component R move from one host to another, its migration subcomponent R_m must move together with the moving subcomponents.

If we consider all subcomponents, including logical migration one, in G_f as general components, network G_f is very similar to the network G' in Section 3. We can easily get the following theorem and corollary applied to component replication.

Theorem 1. The weight of a cut set of flow network G_f is equal to the energy cost of the corresponding component assignment with component replication.

Corollary 1. The minimum energy cost of component assignment with component replication is equal to the weight of a min-cut set or the value of max-flow in network G_f.

Therefore, we use the max-flow min-cut algorithm to partition flow network G_f. The minimum energy cost with component replication is equal to the value of maximum flow of G_f, and the components belonging to the part including node S are assigned to the client, but the components belonging to the part including node T are assigned to the server. Moreover, if some subcomponents of a component are assigned to a host and other subcomponents of the component are assigned to another host, this component should be replicated between the two hosts to save more power.

5 Simulations

We apply the two proposed partitioning algorithms to allocate a Mini-Aegis Weapon System (MAWS), a online game developed by referring to Reference[11].

According to software architecture of MAWS, as shown in Fig.5, we can construct its corresponding cost graph, as shown in Fig.6 where the edge weight labeled on an edge is its communication energy cost within an execution cycle, notated by N=1, and under the standard network condition whose bandwidth is notated by b_0.

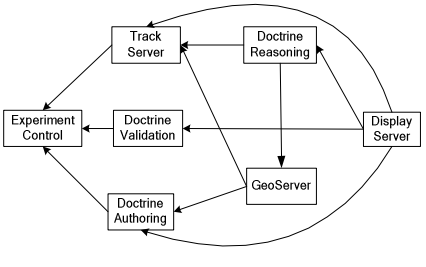


Fig. 5. Software architecture of MAWS

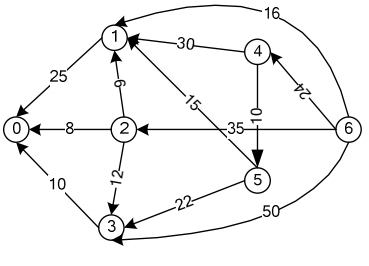


Fig. 6. Cost graph of MAWS

By using the profiling approach, we can obtain the node's weights in the MAWS, as listed in Table 1, where the symbol ∞ indicates an infinite cost.

Table 1. Nodes' weights in the MAWS

Node	*V0*	*V1*	*V2*	*V3*	*V4*	*V5*	*V6*
C_m	∞	360	200	120	500	450	100
C_c	∞	200	120	80	480	240	300
C_s	80	50	30	20	120	60	∞

The simulations consider two major factors of energy cost, execution cycle N and network bandwidth $b = b_0/\alpha$, where α is called as network bandwidth coefficient. Hence, the weight of edge e (c_1, c_2) becomes $\alpha \cdot N \cdot C_c(c_1, c_2)$, and the weight of node c_i becomes ($\alpha \cdot C_m(c_i)$, $N \cdot C_c(c_i)$, $N \cdot C_s(c_i)$). Besides network bandwidth, there are other factors to impact the energy cost, such as the distance and packet error. So we can adjust moderately the value of α to reflect their influences.

Given a partition of an application that runs under a standard network condition, let C_{c0}, C_{m0} and C_{n0} are computation cost, migration cost and network communication cost respectively. If the system runs for N cycles over the wireless network whose bandwidth $b = b_0/\alpha$, the average energy cost per execution cycle, called P, is expressed as follows.

$$P = C_{c0} + (C_{n0} + \frac{C_{m0}}{N})\alpha \tag{9}$$

Without the loss of generality, we assume the whole system initially locates on the server. Our simulations discuss four software partitioning strategies as follows.

(1) Remote execution (RE)

In general, remote execution strategy is to merely download the necessary components, such as UI component *V6* in the MAWS, from the server to the client. Given an execution cycle N, Equation 9 indicates that P increases linearly with the increase of α, as shown in Fig.7

(2) Component migration without migration cost (CMWOMC)

In Reference[6], the authors presented their bipartition algorithm of a program which considered the communication and computation costs, but neglected migration cost. After using this algorithm to partition the MAWS, we calculate the actual energy cost of the obtained application allocation, whose results are shown in Fig.7. As can be seen in Fig.7, when the execution time is relative short, the results of this algorithm are intolerable. Its energy cost is much greater than that of remote execution. However, with the increase of execution time, its energy cost gradually decreases, and even is close to or equal to that of our algorithm with component migration.

(3) Component migration with migration cost (CMWMC)

We employ the proposed min-cut algorithm with component migration to find the optimal component allocation. Fig.7 shows the simulation results of this algorithm. When N=1, the curve CMWMC coincides with the curve RE. When N=10, the CMWMC can save a little more energy than RE. However, when N=100, the former can save much more energy than the latter. Compared with CMWOMC, CMWMC

Table 2. Component allocations and percentages of saving energy of CMWMC when N=100

α	Client	Server	Percentages Saving Cost
1~4	*V6*	*V0, V1, V2, V3, V4, V5*	0%
5~23	*V2, V3, V6*	*V0, V1, V4*	2%~19%
24~100	*V1, V2, V3, V4, V5*	*V0*	20%~43%

obtains much better performance when N is relative small. When N is very large, however, both the two algorithms can save same much energy. Table 2 shows the component allocations and the percentages of saving energy of the CMWMC in comparison with RE under different network conditions when N = 100.

(4) Component replication (CR)

We use the software partitioning algorithm with component replication to allocate the MAWS, whose relationship curves between α and P are shown in Fig.7. When N=1, since expensive migration cost makes most components stay on the server except UI component, the three curves of RE, CMWMC and CR coincides with each other. When execution time is not very long, such as N =10, saving energy by CR is much more than that by CMWMC. Only if the execution time is very long, such as N =100, the saving energy by CR is close to that by CMWMC. Table 3 shows the component allocations and the percentages of saving energy of the CR under different network conditions when N = 100, where node V_i' denotes a replica of node V_i.

Table 3. Component allocations and percentages of saving energy of CR when N = 100

α	Client	Server	Percentages Saving Cost
1~6	*V6, V3'*	*V0, V1, V2, V3, V4, V5*	5% ~ 19%
7	*V6, V3', V1'*	*V0, V1, V2, V3, V4, V5*	20%
8~100	*V6, V3', V1', V2*	*V0, V1, V3, V4, V5*	20% ~ 49.5%

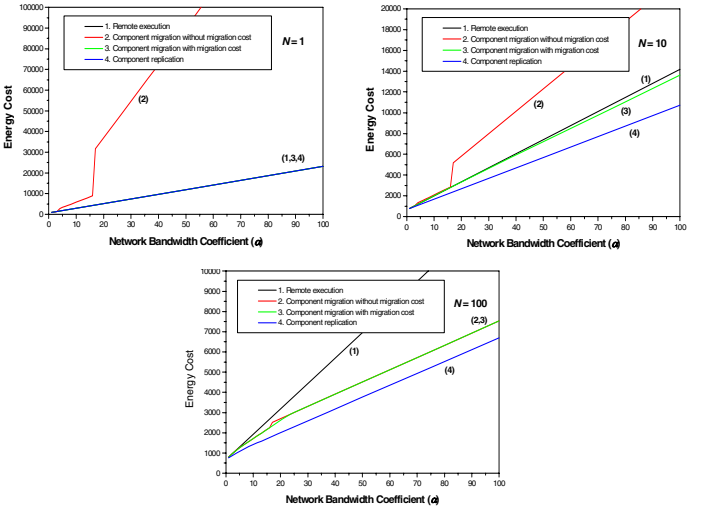


Fig. 7. The relationships between network bandwidth coefficient and energy cost

When the proposed algorithms are applied to partition other applications with a large amount of communication among their components, their simulations results are same as or similar to that of the Aegis system.

6 Conclusion

This paper presents two software allocation algorithms to save power of mobile devices by moving and replicating the appropriate components between the mobile device and the server. Due to consider the component's migration cost, the two algorithms can be dynamically used to adapt to the ever-changing environment in pervasive computing. To my knowledge, this is the first to make use of component replication between machines to save battery power. It is worth emphasizing that the two proposed algorithms are the general min-cut algorithms, so they can be easily applied to partition an application in order to reduce the usage of network bandwidth, improve service performance, and even save multiple resources at the same time. Finally, a series of simulations prove the two algorithms are significantly effective to conserve energy. The simulation results reveal, when an application runs for a long time under a bad network condition, both component migration and replication can save significantly more energy than other approaches, while when it runs for a short time under an excellent network condition, they save only a little energy. Compared with component migration, component replication can save obviously more energy in most of cases.

References

1. W. Mark. Some computer science issues in ubiquitous computing. Commun. ACM, 36(7): 75-84, 1993.
2. M. Satyanarayanan. Avoiding dead batteries. IEEE Pervasive Computing, 4(1): 2-3, 2005.
3. A. Rudenko, P. Reiher, G. J. Popek, and G. H. Kuenning. Remote processing framework for portable computer power saving. In: Proc. of the ACM Symposium on Applied Computing, San Antonio, TX, USA, pages 365-372, 1999.
4. O. Mazliza and H. Stephen. Power conservation strategy for mobile computers using load sharing. SIGMOBILE Mob. Comput. Commun. Rev., 2(1): 44-51, 1998.
5. L. Zhiyuan, W. Cheng, and X. Rong. Computation offloading to save energy on handheld devices: a partition scheme. In: Proc. of international Conf. on Compilers, architecture, and synthesis for embedded systems, Atlanta, Georgia, USA, pages 238-246, 2001.
6. L. Zhiyuan, W. Cheng, and X. Rong. Task Allocation for Distributed Multimedia Processing on Wirelessly Networked Handheld Devices. In: Proc. of 16th International Symposium on Parallel and Distributed Processing, 2002.
7. G. Chen, B.-T. Kang, M. Kandemir, N. Vijaykrishnan, M. J. Irwin, and R. Chandramouli. Studying energy trade offs in offloading computation/compilation in Java-enabled mobile devices. IEEE Transactions on Parallel and Distributed Systems, 15(9): 795-809, 2004.
8. J. Flinn and M. Satyanarayanan. Managing battery lifetime with energy-aware adaptation. ACM Transactions on Computer Systems, 22(2): 137-179, 2004.

9. K. Lahiri, A. Raghunathan, and S. Dey. Efficient power profiling for battery-driven embedded system design. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 23(6): 919-932, 2004.
10. S. Han, S. Zhang, and Y. Zhang. A Generic Software Partitioning Algorithm for Pervasive Computing. In: The International Conference on Wireless Algorithms, Systems, and Applications, Xi'an, China, 2006.
11. A. Robert John. A formal approach to software architecture, Ph.D Dissertation, Carnegie Mellon University, pp. 231,1997.

Mobile Agent Enabled Application Mobility for Pervasive Computing

Ping Yu[1,2], Jiannong Cao[1], Weidong Wen[1], and Jian Lu[2]

[1] Internet and Mobile Computing Lab
Department of Computing
Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
{cspyu, csjcao, cswwen}@comp.polyu.edu.hk
[2] Department of Computer Science and Technology
Nanjing University, Nanjing, China
lj@nju.edu.cn

Abstract. Applications that can follow mobile users when they change to a different environment are in high demand by pervasive computing. In this paper, we describe a mobile agent based paradigm for enabling an application to migrate with the user in pervasive computing environments. Compared with existing efforts on application mobility, our approach has the following distinctive features: (1) Applications are supported by a middleware with a reflective architecture that helps separate business functions from context-awareness logic; (2) Mobile agent is used to manage the mobility of an application and help the application adapt to its new context; (3) The advantages of mobile agent, such as reactivity, autonomy and intelligence, are naturally incorporated into the pervasive computing environment. Our experience shows that mobile agent is a promising technology for pervasive and mobile computing where mobile agents can act as a bridge connecting the cyber world with the physical world.

1 Introduction

In a pervasive computing environment, users can access to computing and information services everywhere at anytime, thanks to the rapid advance of technologies in mobile computers, handheld devices, wireless networks, and various resources and software services. Moreover, users are almost unaware of the computing technologies they are enjoying, as the technologies have been weaved into the fabric of everyday life [1].

However, currently, the development of software for pervasive computing falls behind the technology advance in hardware and communication infrastructures. The pervasive computing community is expecting applications that can make better use of the underlying networking facilities and computing devices without requiring complicated interactions with the users. Among the others, the following requirements are of particular importance:

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 648 – 657, 2006.

- Context-awareness: In the pervasive computing environment, applications and the underlying supporting system should be sensitive to the context information which is of interest to the best performance of the applications [2].
- Self-adaptation: Being context-aware, applications and the underlying supporting software should be able to adapt themselves to better serve the users. Adaptation includes using different computing techniques, changing application and system behavior, and switching to different processing states.
- Mobility: Traditional application-to-computer association will disappear in pervasive computing environments [3]. The user will benefit from application mobility in at least two aspects: (1) User can transfer application to any available device to make a good use of various hardware resources; (2) Application can migrate itself to any computing device to continue the computation, following with its user.

In this paper, we will focus on application mobility. Applications that can follow mobile users when they change to a different environment, especially with the change of device and location, are in high demand by pervasive computing. Implementation of application mobility also depends on context-awareness and self-adaptation techniques. In our previous work, we have described a platform, called "PDAgent", to support the dispatching and management of mobile agent-enabled mobile applications on wireless devices [4]. Mobile agents are self-contained and identifiable computer programs that can move autonomously within the network and act on behalf of the user or other entities. A mobile agent can execute at a host for a while, halt execution, dispatch itself to another host, and resume execution there [5]. Based on the PDAgent platform, we have designed and implemented a middleware "MDAgent" to further support mobile agent-enabled applications with context-awareness, adaptability, and the "follow-me" capability, incorporating mobile agent technology into pervasive computing. In MDAgent, traditional applications are encapsulated or wrapped by mobile agents and thus acquire migration capability from the agents. The application is no longer bound to a certain machine. Instead, it can freely migrate in wired or wireless networks following the user to provide continuous service. Furthermore, with different parameter settings or configurations it can also adapt itself to the new environment.

The rest of this paper is organized as the following. In Section 2, we classify different kinds of mobility with a scenario for pervasive computing. Section 3 describes our MDAgent middleware in detail. Section 4 introduces a developing case of example application and performance evaluation. Section 5 discusses some related work. The last section discusses our future work and concludes this paper.

2 Mobility in Pervasive Computing

We first describe a scenario for our later discussion. Media player is a widely used desktop application, which can run on PC, laptop, PDA, or even mobile phone. It must be a great thing if the media player you are using can migrate to another available device and continue playing the song from exactly where it is suspended. This is desirable if you don't want to be disappointed by power exhaustion when you are enjoying a beautiful song on your PDA, or if you need to move to another room.

You surely want to enjoy a complete song as if you have never left the original machine. This scenario also applies to many other traditional desktop applications, such as notepad, word processor, instant messenger, and so on.

Is it possible to combine entity mobility of the cyber and physical worlds to form a smart computing environment where software entities (especially applications) can move with the physical entities (users or devices)? This is exactly what we would like to investigate with the scenario described above. Not only the music file being played and application execution states (e.g. the point where the music is suspended) will migrate, but also the whole application (executable code) may also move if the application is not available on the new device the user will just change to.

3 Design of MDAgent

"MDAgent" middleware is based on the prototype of "PDAgent", which is a lightweight and highly portable platform for developing and deploying mobile agent-enabled mobile applications on wireless handheld devices. "MDAgent" extends PDAgent by introducing context-awareness and adaptability into the middleware support, especially the support for application mobility.

As a paradigm for code mobility, mobile agent introduces several advantages, including reducing network load by accessing resources locally, executing task asynchronously and autonomously, and being hardware and transport-layer independent [7]. There are plenty of applications that can benefit from using mobile agent. Especially, the mobile agent paradigm suits very well to mobile computing environments, where disconnection from the network often occurs [4, 8]. Naturally, mobile agent also helps in building applications for pervasive computing, notwithstanding mobile agent should be provided with more capabilities such as context-awareness and adaptability to tackle more complicated application mobility. The MDAgent middleware described in this paper is a novel middleware to enable application mobility by using mobile agent.

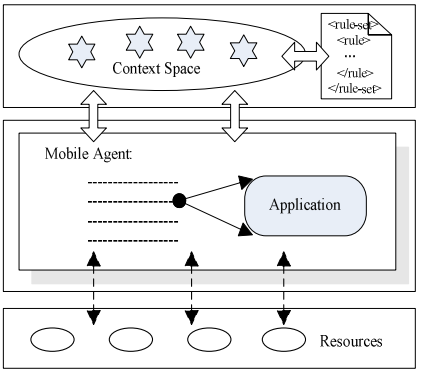


Fig. 1. Abstract Model of Mobile Agent Enabled Application

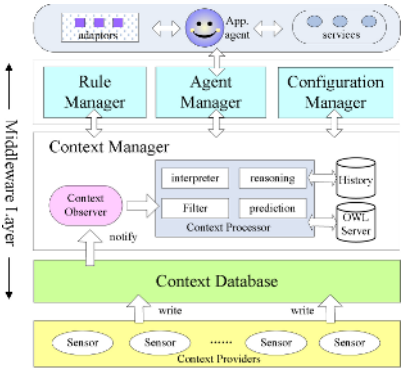


Fig. 2. MDAgent Middleware Architecture

Generally speaking, an application is wrapped by a mobile agent which has the full control (e.g. suspending, migrating, resuming) of the application. Aided by the underlying middleware services, mobile agent-enabled applications can be made sensitive to the user's execution context, including location, activity, preference, and the profile of the device the user is used. With regard to self-adaptation, each mobile agent is attached with a rule set, which consists of some adaptation rules. Figure 1 shows the abstraction of such a mobile agent enabled application.

3.1 The MDAgent Middleware Architecture

The overall structure of MDAgent middleware consists of four main parts: *Agent Manager*, *Context Manager*, *Rule Manager* and *Configuration Manager*. MDAgent relies on a publish/subscribe system to obtain context information from sensors or other external context providers. The *Agent Manager* is in charge of agent life cycle, application snapshot and related resources management. It also communicates with the context and rule managers to exchange relevant information about the agent-enabled application. The *Context Manager* includes sub-components such as context interpreter, context filter, and context reasoning and prediction modules to process raw context information acquired by the context observer. An OWL (Web Ontology Language) server and context history database are also indispensable for context processing. The *Context Manager* also takes charge of disseminating context information to remote hosts which could not access the original context sources. The *Rule Manager* is responsible for loading, interpreting and executing rules with a built-in rule inference engine. It will trigger mobile agent's adaptation when certain rules are fired. During adaptation, some adaptors will be dynamically loaded to help agent adapt its behavior and states. The *Configuration Manager* takes charge of configuring services for the agent according to different contexts, to ensure the proper quality of service.

3.2 Application Model

In order to be moved with the agent, an application should be delicately designed with a reusable pattern. First of all, the application must be device-independent so that it can freely migrate between different devices. However existing experiences tell us that it is difficult to achieve. Although Java enables device-independent code that can be shared across platforms, different Java subsets like J2ME MIDP 1.0/2.0 reduce the portability of Java. Web service is another popular technology which is proposed to support interoperable machine-to-machine interaction over a network by using standard protocols such as XML, WSDL and SOAP. But web services do not support automatic session migration with the user. If the user changes the device, the session opened in the previous device is lost. In our opinion, a mobile application model should be built on mature technologies such as Java and web services, but should also overcome shortcomings we mentioned above. Consequently, we propose a reflection-based application model shown in Figure 3.

In this model, a reflective application is divided into two levels: *meta level* and *base level*. Base level includes context-free functions which can be invoked in different platforms, such as *play()* and *stop()* functions in the media player

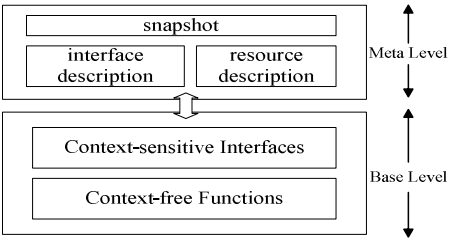


Fig. 3. Application Model

application. For those context sensitive functions such as the media player GUI, we separate them from context-free functions with interface descriptions in XML format. When the application moves to a different kind of device, the middleware will notify the application wrapper - the agent to load appropriate functions suitable to the current context. For example, for a desktop PC, the media player will show in a full Winamp style by loading the function *showAsFullWinamp()*, while for a mobile phone, the media player will be opened in the minimal format by loading the function *showAsMinimal()*.

Meta level includes XML-based interface description files for reconfiguring the application's context-sensitive interfaces, and a snapshot for storing application's high level states such as the music file being played and the frame where music is suspended. Resources refer to the external resources the application will use. Some resources can move with the application, such as the document currently under editing or the music file being played. Others resources like the database cannot be moved and different strategies like "by copy" or "by reference" should be provided for these resources. Through a resource description file, the agent will manage resources with finer grained mobility control.

3.3 Mobile Agent Semantics

As Figure 1 shows, mobile agent functions as an application wrapper to enable application mobility. Hereafter we will not distinguish between mobile agent and mobile application any more, since application has been hidden in the agent.

In MDAgent, mobile agent's semantics is more complicated. First, it should be aware to the user's context of the computing environment. Second, it should be able to adapt the application when context changes. Third, it should be flexible enough to support mobility by different strategies.

To preserve the compatibility with previous the PDAgent platform which enables mobile agent running in PDA and other handheld devices, we extend mobile agent meta-data's DTD as shown Figure 4. The meta-data contains the following parts.

- *Basic Information*: Each mobile agent has a unique "AgentName" and "ClassName" for naming itself. "Description" is an option for providing more information about the agent and its functionalities. "AgentCode" and "AgentCodeURI" are alternatives that specify the source of the code for the agent. Middleware can directly load agent code from "AgentCode" element or obtain the code from the URI specified by "AgentCodeURI".

- *Subscribed Context*: Based on different application requirements, agents are definitely interested in different types of context, although the user's location is the primary context for application mobility. Through analyzing this element, middleware will create corresponding filters for accurately returning the context useful for the agent. Moreover, the user can define context himself, such as font color or size. That is to say, the user can customize the agent during its whole life cycle.
- *Rule Set*: This part defines the rules for adaptation. Although we do not have the limit on the format of the rules (each rule engine implementation has its own rule expression), we recommend a declarative programming approach. Presenting the current design, <condition, consequence> pairs are adopted by MDAgent for the sake of simplicity and extensibility. Alternatively, an external rule file can be imported into the agent by using the "RuleRef" option.

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Agent (AgentName, ClassName, Description, OtherInfo?,
(AgentCode | AgentCodeURI), SubContext?, (rule-set | RuleRef)? >
<!ATTLIST Agent xmlns CDATA #REQUIRED>
<!ELEMENT AgentName (#PCDATA)>
<!ELEMENT ClassName (#PCDATA)>
<!ELEMENT Description (#PCDATA)>
<!ELEMENT AgentCode (#PCDATA)>
<!ELEMENT AgentCodeURI (#PCDATA)>
<!ELEMENT OtherInfo (#PCDATA)>
<!ELEMENT SubContext (Context* )>
<!ELEMENT Context (Type,Value?,Source)>
<!ELEMENT Type (#PCDATA)>
<!ELEMENT Value (#PCDATA)>
<!ELEMENT Source (#PCDATA)>
<!ELEMENT RuleRef (#PCDATA)>
<!ELEMENT rule-set (rule+)>
<!ATTLIST rule-set name CDATA #REQUIRED>
<!ELEMENT rule (parameter*,condition+,consequence)>
<!ATTLIST rule name CDATA #REQUIRED>
<!ELEMENT parameter (class)>
<!ATTLIST parameter identify CDATA #REQUIRED>
<!ELEMENT class (#PCDATA)>
<!ELEMENT condition (#PCDATA)>
<!ELEMENT consequence (#PCDATA)>
```

Fig. 4. Mobile Agent Meta-data's DTD

With this XML-based meta-data, an agent is able to change adaptation logic without modifying its byte code at runtime. The context being subscribed and the adaptation rules can be changed dynamically. The *Context Manager* and *Rule Manager* are both aware of such changes. They will update the corresponding context filters and the rule engine's working memory for the agent when necessary.

3.4 Support for Application Mobility

The general life cycle of a mobile agent-enabled application is illustrated in Figure 5. Whenever the middleware detects that the user is leaving the current site, the agent will automatically be suspended and then cached to the secondary storage (being

deactivated). When the user logs in another device, the middleware will activate the agent which will check out from the current site and migrates to the device where user is working. When arriving at the new site, the agent first checks in (or register) at the middleware. After successful registration (e.g. authentication, authorization), it will restore from the application snapshot and resume execution from exactly where it was suspended.

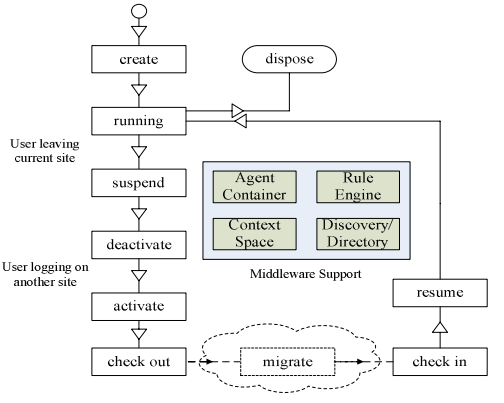


Fig. 5. Life Cycle of "Follow-Me" Application

4 Performance Evaluation

A prototype of MDAgent has been implemented. In implementing the MDAgent prototype, we make use of some open source packages such as Aglets-2.0.2 (a mobile agent container implemented by IBM) [9], Jena (a programmatic environment for OWL) [10], and Drools (a rule inference engine) [11]. For XML encoding and parsing, kXML [12]—a lightweight XML API package is used.

On top of MDAgent, based on the application and agent model described earlier, we have developed some example applications, such as "SmartNotepad" and "SmartMediaPlayer". These applications are enabled to automatically follow their users to provide continuous services.

We deployed wireless locating sensors in the Hong Kong PolyU's Department of Computing building and installed the MDAgent middleware in our location-area network systems. A PC of 1.4Ghz with 256MB of RAM and a laptop of 1.7Ghz with 512MB of RAM were placed in different rooms. "SmartNotepad" and "SmartMediaPlayer" are both intelligent enough to move with the user whenever he/she changes to another machine. As expected, these applications can both resume from where they are suspended. Users are really enjoying the benefit of pervasive computing—compute anywhere anytime!

Figure 6 and Figure 7 presents some of the screen shots of the application execution, running on different devices following with the movement of the user.

Fig. 6. SmartNotepad running in PC

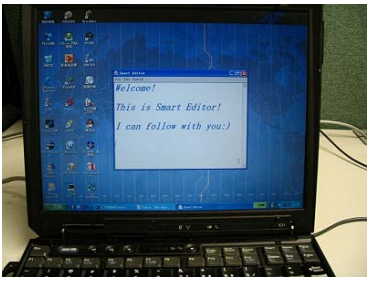


Fig. 7. SmartNotepad migrates to laptop

Performance is evaluated by running these two applications in the PC and laptop mentioned above. Table 1 shows the average latency in suspending, migrating and recovering the mobile applications. These performance figures indicate that the application mobility process is fast enough to be practical and up to the user's expectation. Table 2 describes the network traffic used for migrating the applications. SmartNotepad involves an in-memory file which is being edited, while SmartMediaPlayer needs to refer to an external music file (e.g. "Hero.mp3" in this experiment) which will be transferred to another device. Subtracting the resource size from the traffic occupied, the overhead of migrating the applications is very small.

Table 1. Latency evaluation

Application \ Latency(ms)		Suspend	Migrate	Recover
SmartNotepad	PC	421	921	187
	laptop	110	320	20
SmartMediaPlayer	PC	514	2175	1640
	laptop	321	1752	872

Table 2. Network traffic for application migration

	External Resource Size (Byte)	Throughput (Byte)
SmartNotepad	0	31980
SmartMediaPlayer	5206524	5237271

5 Related Work

Several approaches have been proposed to make applications movable in pervasive computing environments. We present some of them and compare them with our work.

Aura: Aura aims at building a distraction-free ubiquitous computing environment [13]. A task layer is introduced to represent user's intent, adapt and anticipate user needs. Aura supports migrating tasks between Windows and Linux platforms. Task is

similar to mobile agent of MDAgent in that they both involve execution and resource migration. We argue that mobile agent is a more principled paradigm for managing migration and adaptation.

Gaia: Campbell et al. designed a meta-operating system Gaia which brings the functionality of an operating system to physical spaces [3, 14]. By Gaia, physical spaces turn into active spaces where applications can move around. They extended the Model-View-Control framework to establish the mobile application model. Compared with their work, our application model is more general by a separation between the base level functions for business logic and the meta-level data for adaptation and migration. The application is loosely coupled with mobile agent because it can also be run even without the agent wrapper. Only when mobility is concerned, the application needs to be wrapped into a mobile agent.

MobiDesk: MobiDesk is an infrastructure for centralized hosting of desktop computing sessions which can move within virtualized private environments [15]. MobiDesk decouples a user's desktop computing session from the end-user device by moving all application logic to hosting providers. End-user sessions are stored in back-end session servers via a proxy. The main difference between MobiDesk and MDAgent is that, in MDAgent, applications are running locally without interacting with remote application or session servers. Accordingly, network bandwidth is greatly saved especially for those wireless devices.

6 Conclusion

The primary idea of MDAgent is the mobile agent based paradigm for providing traditional stationary desktop applications with migration capability without requiring user's intervention. By the causally connected two-level architecture, traditional applications can be easily deployed to the MDAgent middleware without much modification. Mobile agent acts as a wrapper of the application for providing mobility and context-awareness. Agent is reactive to user's context and adaptable when the context changes. Moreover, mobile agent is also customizable and extensible through the meta-data it carries with. This feature makes the application more flexible to the changing requirements of pervasive computing. MDAgent contributes to pervasive computing research in at least three aspects:

(1) It satisfies new application requirements of context-awareness, self-adaptation and transparent mobility in a smart world.
(2) It provides a promising paradigm for mobile applications with the support of reflection and mobile agent technology.
(3) It is portable and extensible for being deployed in most of the mobile devices and operating systems.

In our future work, we will try to develop and deploy more applications in handheld devices to evaluate the performance and adaptability of our proposed MDAgent middleware. We also want to introduce some machine learning methods for agent adaptation. Context prediction (especially user mobility prediction) is another focus in the next step.

Acknowledgement

This work is supported by the University Grant Council of Hong Kong under the CERG grant PolyU 5183/04E, the Hong Kong Polytechnic University under the ICRG grant G-YD63, and China National 973 Program Grant 2002CB312002.

References

1. Weiser M: The Computer for the Twenty-first Century, Scientific American 265:94-101, 1991.
2. Anind K. Dey, Gregory D. Abowd: Towards a Better Understanding of Context and Context-awareness, In the workshop on The What, Who, Where, When, and How of Context-Awareness, as part of the 2000 Conference on Human Factors in Computing Systems (CHI 2000), 2000.
3. Manuel Roman, Herbert Ho and Roy H. Campbell: Application Mobility in Active Spaces, In 1st International Conference on Mobile and Ubiquitous Multimedia, Oulu, Finland, 2002.
4. Jiannong Cao, Daniel C.K. Tse and Alvin T.S. Chan: PDAgent: a Platform for Developing and Deploying Mobile Agent Enabled Applications for Wireless Devices, In Proc. of the 2004 International Conference on Parallel Processing, pp.510-517, 2004.
5. David Kotz and Robert S. Gray: Mobile Agents and the Future of the Internet, SIGOPS Oper. Syst. Rev., 33(3):7–13, 1999.
6. Alfonso Fuggetta, Gian Pietro Picco, Giovanni Vigna: Understanding Code Mobility, IEEE Transaction of Software Engineering. 24(5):342-361, 1998.
7. Danny B. Lange and Mitsuru Oshima: Seven Good Reasons for Mobile Agents. Commun. ACM, 42(3):88–89, 1999.
8. Paolo Bellavista, Antonio Corradi, and Cesare Stefanelli: Mobile Agent Middleware for Mobile Computing. Computer, 34(3):73–81, 2001.
9. IBM Aglets, http://www.trl.ibm.com/aglets/
10. Jena, http://jena.sourceforge.net/
11. Drools, http://drools.codehaus.org/
12. kXML, http://www.kxml.org/
13. Garlan, D., Siewiorek, D., Smailagic, A., Steenkiste, P.: Project Aura: Toward Distraction-Free Pervasive Computing, IEEE Pervasive Computing, 1(2):22-31, 2002.
14. Roman, M.; Hess, C.; Cerqueira, R.; Ranganathan, A.; Campbell, R.H.; Nahrstedt, K.: A Middleware Infrastructure to Enable Active Spaces. IEEE Pervasive Computing, 1(4):74-83, 2002.
15. Ricardo A. Baratto, Shaya Potter, Gong Su, and Jason Nieh: MobiDesk: Mobile Virtual Desktop Computing. In Proc. of MobiCom'04, pp.1-15, 2004.

Towards Summarized Representation of Time Series Data in Pervasive Computing Systems

Faraz Rasheed, Youngkoo Lee, and Sungyoung Lee

Computer Engineering Dept. Kyung Hee University
446-701 Suwon, Republic of Korea
faraz@oslab.khu.ac.kr, {yklee, sylee}@khu.ac.kr

Abstract. Ubiquitous computing systems are connected with a number of sensors and devices immersed in the environment, spread throughout providing proactive context aware services to users. These systems continuously receive tremendous amount of information about their environment, users and devices. Such a huge amount of information deserves special techniques for efficient modeling, storage and retrieval. In this paper we propose the modeling of context information as time series and applying the time series approximation techniques to reduce the storage space requirements and for faster query processing. We applied an algorithm based on non-linear interpolation to approximate such data and evaluated the approximation error, storage space requirements and query processing time.

1 Introduction

Ubiquitous Computing envisions [1] an environment where devices and sensors can interact seamlessly to provide proactive services to users. There are a number of projects currently being done in ubiquitous computing field [2][3][4]. Context awareness is an integral feature of ubiquitous computing systems. In order to provide proactive services to user, a system must be aware of the users environmental context and user preference. We take context as anything that helps in situational understanding.

Usually a ubiquitous system is connected with a number of sensors which sense the physical and computational environment, the users present in the environment and their activities. Hence various types of data are continuously injected to the ubiquitous computing system. The size of this data grows tremendously with time slowing down the query processing and other database operations. This huge amount of data requires careful management to address the issues in ubiquitous data management [5].

We proposed in our earlier papers the idea of knowledge aggregation which we call Context Summarization [6][7]. The idea is to represent the information in such a form that it takes comparatively less storage space and can still answer queries with appropriate precision. We have proposed several techniques that can be applied on different type of data.

Our summarization process is totally hidden from the user and she still queries the data as she is using the original data. Since the summarization process change the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 658 – 668, 2006.

representation of the data transparently, there is a need for query answering system which can translate the queries on original data so that they can be answered using the aggregate information store. In this paper we are going to describe this query answering component of our summarization system. We will explain how this query answering system can be applied to time series data which includes temperature values, humidity values, available network bandwidth at particular time, network bandwidth usage by particular user, light intensity, etc.

2 Context Summarization

Context Summarization is a process of approximation or the compact representation of data such that it consumes relatively less storage space and can still answer user queries with acceptable degree of confidence. Most of the data received by pervasive system is for the software system's (middleware and application) internal use so that they can process it and provide the proactive services to users promptly and efficiently. Thus we can change the internal representation of this data such that most of the queries can be answered using the summarized information.

Ubiquitous computing systems are equipped with a number of sensors, most of which continuously emit data at different constant rates. Such sensors include temperatures, humidity, light intensity, noise level, signal strength, available network bandwidth, network utilization, etc. This kind of data which is received continuously can be modeled as time-series. The approximate representation of time series can be achieved by selecting the representative points which provides the best approximation to achieve the desirable degree of confidence value. The confidence value is the amount of precision provided by a particular summary of the data. The precision can be modeled by the overall error or the amount with which the summary representation differs from the original time series. Different error measures can be used like Euclidean distance, root mean square error, averaged error. For our experiments, we have used the root mean square error as it is more widely used in approximation theory. The queries could be answered by interpolating through these points or by evaluating the fitted curve to a selected portion of the data.

Such a summarization not only reduces the storage space consumption but also improves the time taken to answer queries. As the ubiquitous systems are essentially distributed, such a summarized representation also improves the network usage for data migration among different nodes. It also improves the efficiency of knowledge reasoning, user preference and machine learning and data mining by minimizing the size of available data and thus the query processing time.

We are using our project called CAMUS (Context Aware Middleware for Ubiquitous Systems [4]) to apply the summarization techniques. Hence, all these techniques are implemented inside the middleware which also host the knowledge repository and provides query interface to user. In the next section we will explain the architecture of our system and in the subsequent section we will show how do we apply the summarization techniques and answer queries over numerical type context. The architecture of our summarization module has been discussed earlier in [7].

3 Applying Query Answering Interface (QAI)

Now we will describe the application of summarization and query answering using time series data (temperature values). A worth mentioned point is that the ubiquitous systems are very resource hungry as they are performing a lot of operations; receiving continuous stream of data from sensor, data filtering, modeling, storage, data processing, reasoning, situational understanding of current context, finding and providing the appropriate service and so on. Hence, the summarization module should be very smart and efficient. That is the reason why we have selected to apply only simple, optimized and less complex techniques to summarize the information and to calculate and return the query result.

We will present a summarization algorithm to approximate time series data based on non-linear Interpolation with equi-spaced points. For interpolation, we used cubic-spline and piece-wise hermite interpolation techniques which we found better in terms of query response time and error. In these techniques an important task is the selection of equi-spaced points which provides the best approximation of data. We have developed an algorithm to select such points. This smaller number of representative points is stored in database and used later to reply queries by interpolating through these points.

The algorithm attempts to achieve the approximation with a specific precision value (which is defined as root mean square error: rmse). But it does not guarantee that such an approximation would be achieved as it also considers the storage space requirement by approximation compared to the original dataset.

3.1 Time Series Numerical Valued Information – Non-linear Interpoaltion

Interpolation is a method of constructing new data points from a discrete set of known points. While linear interpolation connects the discrete points by straight line, non-linear interpolation techniques connects the discrete points using polynomials. We have utilized two of the most famous interpolation techniques namely spline and hermite interpolation. Let we have a temperature sensor reporting the temperature of location 'X' after every 5 minutes. Table 1 presents a sample data set

Table 1. Temperature values received after every 15 minute

Time	Temp.
2:05	35.5
2:10	34.5
2:15	38 °C
2:20	38.5
2:25	35.5
…	

Now let we summarize the information by selecting the points sampled at larger interval than 5 minutes (the original data arrival rate) as in figure 1.

SummaryMasterTable		
ID	**Interval_Start**	**RMSE**
23	2004-06-14 00:00	1.43
24	2004-06-14 03:00	1.42
25	2004-06-14 06:00	1.34
…	…	…

SummaryChildTable		
ID	**PointX**	**PointY**
23	1	35.5
23	9	39
23	17	35
23	25	36.5
23	33	37.5
23	36	34
24	1	34
…	…	…

Fig. 1. Sample Summary Tables (Non-linear interpolation)

Here the selected points for approximation are variable. Each record in summary table has the interval start time and the approximation error. The summary child table contains the selected points for each of these intervals.

The input to this algorithm is a set of 'n' data points and the maximum approximation error (max_error) that can be tolerated. The output is the set of 'k' points where k < n and the actual approximation error as root mean square error.

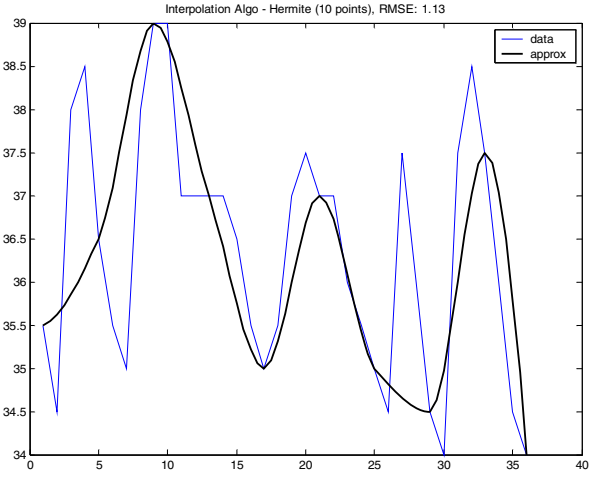


Fig. 2. Time series approximation using non-linear (Hermite) interpolation

The target is to find the set of 'k' data points representing the given-time series so that when interpolated through these points, the error value (rmse) is less than or equal to the max_error. The algorithm proceeds with 'c' which represents how many points should be approximated by each selected points. Practically, it is the gap

between selected points. Hence if c = 4, it would pick 1, 5, 9, 13, 17,… elements of time series. It then uses these points for interpolation and generating the approximate time series. As 'c' increases, the number of selected points 'k' decreases which results in the increase in 'rmse'. The algorithm attempts to find the optimal value of 'c' using which the desired precision can be achieved. The algorithm only attempts to find the c in a particular limit such that 'c1 <= c <= c2, where c2 is the minimum and c2 is the maximum value of 'c'. Above c2, the algorithm would select too many points which is against the spirit of summarization to save the storage space. Below c2, the algorithm would select too fewer points to deteriorate the rmse. Hence a careful selection of c, c1 and c2 is required. Figure 2 demonstrates the approximation done by non linear interpolation. The algorithm to find 'k' points is presented below:

```
1. start with c = v, mode = 'start' // v: arbitrary value
2. Find k-points with c
3. Find rmse with these k-points
4. Switch(mode)
5.a mode = start
      rmse > max_err
            c = c - 1
            set mode 'dec'
            goto 2
      rmse <= max_err
            c = c + 1
            set mode 'inc'
            goto 2
5.b mode = dec
      rmse <= max_err
            save points
            return
      rmse > max_err
            if c = c1          // minimum value of c
                  save points
                  return
            else
                  c = c - 1
                  goto 2
5.c mode = 'inc'
      rmse >= max_err
            c = c - 1
            set mode 'dec'
            goto 2
      rmse < max_err
            if c = c2          // maximum value of c
```

```
            save points
            return
      else
            c = c + 1
            goto 2
```

3.2 Query Answering and Discussion

Suppose we receive a query for attribute ‘v’ at timestamp ‘t’ in data table ‘dt’

SELECT 'v' FROM 'dt' WHERE timestamp = 't'

Query processing starts with searching the interval, in summary table, which contains the desired timestamp. Then it retrieves the points for interpolation from the child table. These points are then interpolated to find out the attribute value for ‘v’ at timestamp ‘t’.

The advantage of this technique is that it reduces the storage space significantly. The query processing is also simple and since the interpolation techniques generate a continuous curve, the summary representation can even answer for the timestamps for which there is no attribute value in the original data table. For example, if original data is received after every five minutes 2:00, 2:05, 2:10, … and there is a query for timestamp 2:13, we can still interpolate and find out the approximate value. This could be important in certain scenario processing when the application wants some information at the time when there is interesting activity sensed by some other sources. Since query processing using summary tables involve two table lookups (compared to one table lookup in original data table), for the query to be processed faster there should be significant number of records in original table. In other words, the interpolation based summary would generally perform faster if the original data table contains very large amount of data compared to the summary tables.

4 Experiments and Evaluation

We have evaluated the summarization of time series data from three perspectives. First of all we have compared the storage space consumption by raw data and information summarized by different techniques. Secondly, we applied the similar queries on both raw data and summarized information and compared the time taken by both of these representations. Finally, we calculated the root mean square error incurred while answering the queries using various summarized representations.

4.1 System Specification

We performed all the experiments on Toshiba notebook using Pentium Mobile using Intel Centrino technology with 1500 MHz processor and 512 MB RAM. We are using Windows XP Professional OS. Our summarization module is implemented using Java programming language. We used mySQL v4.0 as DBMS (running on localhost) and also used MATLAB 6.5 for interpolation techniques. The system was not

considerably busy with other process while we performed the experiments and we did not use multiple threads for processing.

4.2 Time Series Numerical Valued Context Information (Temperature)

We used the data recorded by Cornell Lab of Ornithology for Bird population studies (USA). They installed a temperature logger in bird's nest and recorded the temperature after every 5 minutes. We used the data recorded in 88 days which amounted to around 22,000 records. We summarized this data over 2 hours interval using algorithm described in section 3.1. For query processing we interpolated through these points using cubic spline and piecewise hermite interpolation techniques. The effect on the storage space consumption by the summarization process is presented in the Table 2

Table 2. Storage space consumption by raw and summary data

	Raw data	**Summary data**
Rows	21,978	924
row size	48	97
table size	1035 KB	90 KB
Days	88	88
Interval	5 min.	120 min.

Obviously, the summary table takes less storage space than raw data. The row size in summary table is larger than that of raw data table as it contains more fields. The same information is presented in graphical form in Figure 3.

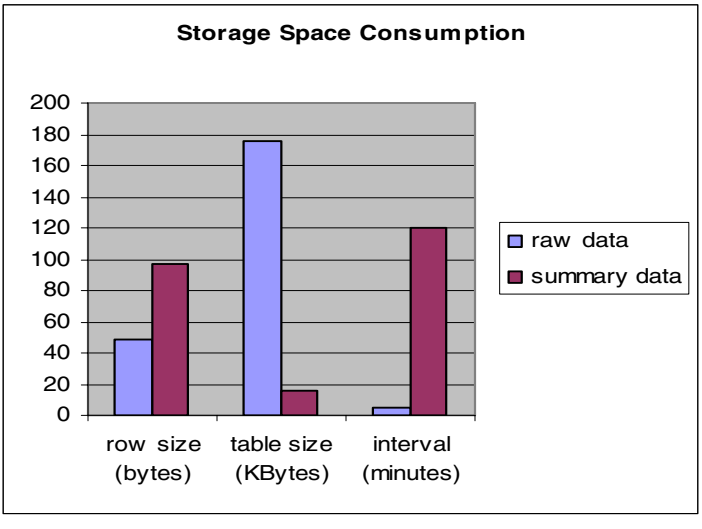


Fig. 3. Comparison of storage space consumption

Now for the time taken by query processing, we applied the simple query to find the temperature at a specific time over 22000 raw data records and 900 summary records. We calculated the time consumed to reply this query over raw data and the summary data calculated by 2 techniques described earlier. Chart in figure 4 compares the time taken to answer query by raw data table and tables using summarized representation.

The time taken to answer query using the interpolation (60ms and 46 ms) is less than that of raw data (140 ms). This is because the raw data is querying over much larger table (22000 records versus 900 records of summary table). The hermite interpolation is processing queries a little faster than cubic spline interpolation.

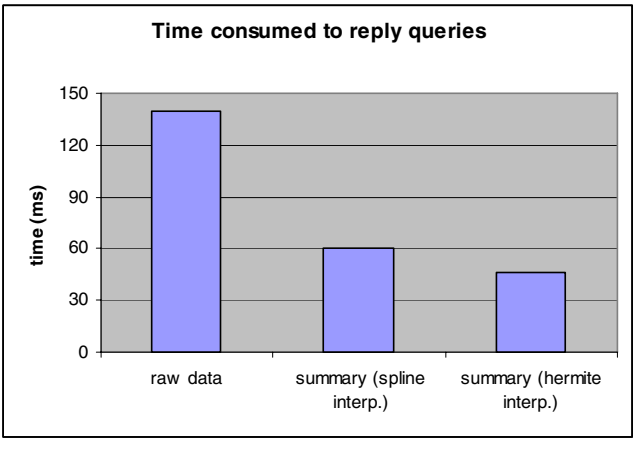


Fig. 4. Time taken to reply queries

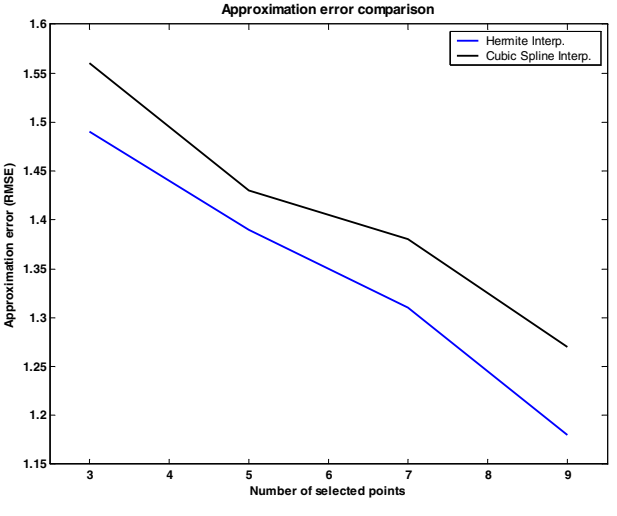


Fig. 5. Comparison of approximation error

Finally, to calculate the lost in precision caused by these queries, we used the root mean square error. The approximation error is calculated by keeping the storage space constant that is by taking the same number of points to interpolate using cubic spline and piecewise hermite interpolation. Figure 5 presents the comparison of approximation error:

It is obvious that as we increase the number of points, the approximation error is reduced. It is also important to note that the approximation error for hermite interpolation is less than the cubic spline interpolation for the same number of selected points.

5 Issues and Challenges

As we mentioned earlier that ubiquitous computing systems are resource hungry because of their distributed nature, a number of diverse sensors and devices they are to deal with and for the smart service selection and service delivery. Hence, performance is reasonably important issue for each of its components. That is the reason why we are keeping our summarization system as thin as possible which can work with very limited resources efficiently. There are quite a few mathematical and graph based techniques like curve fitting with limited points and interpolation to find the missing point in the graph but we are very selective because of the inevitable complexity. Precision and accuracy is another important issue. The tasks that are usually performed using context repository like preference and machine learning, logic reasoning, intention prediction and activity recognition; all of them require data with higher degree of precision. The more we summarize the information, the greater is the loss of precision. To reduce this precision loss we can use more comprehensive techniques but then there is a trade-off between precision and performance. But the important thing is that as the size of data reduces, all the ubiquitous system components that use the context repository tend to perform faster. Security is another significant issue. Since the summarization techniques change the representation of data, hence if the summarization is not done properly, the results produced would be misleading. Hence, summarization components are required to be designed and implemented with great care.

6 Related Work

Several existing systems support techniques like feature extraction and generalization [8][9][4] but we want to formally make summarization as part of the ubiquitous system's data management. Our idea is to generate summaries so that later we don't need the raw data any more and can reply to most of the queries with this summarized information with acceptable degree of confidence.

In DBMS, data mining and data ware housing [10] use the concept of histogram and multidimensional views of database and work on the aggregate, consolidated data instead of raw data to support the higher level decision making and to identify the hidden patterns in the data. The goal of data mining and OLAP is somewhat similar

but we want to transform the raw context to summarized form taking less storage space and provide improved and efficient reasoning and machine learning.

Researchers in DBMS have also analyzed the time series data streams for very large databases [11]. Here, they analyze the data coming in continuous streams with time. They have proposed solutions on how to manage, represent and store the time series data streams. This is also highly related to the context summarization.

7 Future Work and Conclusion

For the future work, we are currently trying to summarize the location map of wifi based location awareness system. This map contains the signal strength received at different locations by the device from access points in the surrounding. The size of this data is very huge; just for few floors the number of records grows in multiple of 10,000! We are also optimizing our Query Answering Interface (QAI) so that it can efficiently use multiple summary representations of different strengths to answer queries with different precision requirements accordingly. In the conclusion, we would say that data aggregation and context summarization is an interesting research area which needs to be further explored. We have presented in this paper the summarized representation of numerical valued time series information, also how and which kind of queries they can answer. The proper application of summarization process does conserve storage space and improves the query processing and data migration in distributed ubiquitous computing environment as supported by our experimental results.

References

1. M. Weiser, The computer for the 21st century. *ACM SIGMOBILE 1999 Review*
2. Chen Harry, Tim Finin, and Anupam Joshi: An Intelligent Broker for Context-Aware Systems. In: *Ubicomp 2003*, Seattle, Washington
3. Gaia: A Middleware Infrastructure to Enable Active Spaces. Manuel Román et al., In *IEEE Pervasive Computing*, Oct-Dec 2002
4. Hung Q. Ngo, Anjum Shehzad, Saad Liaquat, Maria Riaz, Sungyoung Lee: Developing Context-Aware Ubiquitous Computing Systems with a Unified Middleware Framework. *EUC 2004*: 672-681
5. Michael J. Franklin, Challenges in Ubiquitous Data Management. . Informatics: 10 Years Back, 10 Years Ahead, LNCS #2000, R. Wilhiem (ed)., Springer-Verlag 2001
6. Faraz Rasheed, Young-Koo Lee, Sungyoung Lee, Context Summarization & Garbage Collecting Context, UWSI 2005, In the proceedings of ICCSA 2005 page 1115-1124, Singapore, published by Springer Verlag
7. Faraz Rasheed, Young-Koo Lee, Sungyoung Lee, "Towards Using Data Aggregation Techniques in Ubiquitous Computing Environments," *percomw*, pp. 369-372, Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06), 2006
8. Mike Spreitzer, Marvin Theimer, Providing location information in a ubiquitous computing environment, *ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles* Dec 1993, Volume 27 Issue 5

9. Jason I. Hong, James A. Landay, Support for location: An architecture for privacy-sensitive ubiquitous computing, *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, June 2004
10. Alex Berson , Stephen J. Smith, Data Warehousing, Data Mining, and OLAP, McGraw-Hill, Inc., New York, NY, 1997
11. Lin Qiao et al, Data streams and time-series: RHist: adaptive summarization over continuous data streams, *Proceedings of the eleventh international conference on Information and knowledge management*, Nov 2002

Adaptive Bridging with Portable Interceptor for Efficient Integration of Reflective Middleware*

Hyun Ko and Hee Yong Youn**

School of Information and Communications Engineering
Sungkyunkwan University, Suwon, Korea
osthein@hanmir.com, youn@ece.skku.ac.kr

Abstract. Pervasive computing paradigm is based on much more dynamic environment than the existing computing paradigm. Therefore, we need a middleware framework for efficiently integrating the pervasive objects. In this paper we propose an adaptive bridging approach for the interoperability of heterogeneous middleware. Here the objects are classified into hot or cool objects according to the requested frequency reported by the portable interceptors, and the cool spot objects and hot spot objects are assigned dynamic bridge and static bridge, respectively. The proposed approach is evaluated using a test bed comprising of three different platforms - JADE, MICO, and CALM - on nine servers. The experiment reveals that the proposed adaptive bridging approach significantly reduces the response time compared to the case of using only either static bridge or dynamic bridge, especially when the number of activated objects is relatively large.

Keyword: Adaptive bridging, heterogeneous, middleware, objects, pervasive computing, portable interceptor.

1 Introduction

Pervasive computing paradigm is based on much more dynamic environment than the existing computing paradigm. For the pervasive computing we need the introduction of a middleware supporting dynamic operations. Such middleware provides efficient connectivity between the objects and also reduces the software development cost [1]. In the pervasive computing environment there exist a number of objects having distinct characteristics, and thus different middleware might be required for manipulating them separately. For example, some objects need real-time middleware platform and the others need component-based middleware platform. One single middleware platform which can efficiently support various types of objects is hard to develop because of the complexity and overhead of the implementation. Therefore a composite middleware which is flexibly interoperable with various middleware will be more effective than one generic middleware supporting all the objects.

* This research was supported in part by the Ubiquitous Autonomic Computing and Network Project, 21st Century Frontier R&D Program in Korea and the Brain Korea 21 Project in 2005.

** Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 669 – 678, 2006.

In this paper we introduce an adaptive bridging approach which allows transparent and efficient connectivity between heterogeneous middleware. We have developed a framework of reflective middleware called CALM (Component-based Autonomic Layered Middleware) [9], and the proposed bridging approach is embedded in the CALM. The proposed approach combines static bridge effective for the objects requested frequently (hot objects) with dynamic bridge which uses DSI/DII (Dynamic Skeleton Interface/Dynamic Invocation Interface) for infrequently requested objects [8]. We also apply the bridging approach for the connectivity of the component-based platform and agent platform.

Middleware, agent, and component software are important topics worldwide in the research and development of pervasive computing fields. The middleware integrates various pervasive systems, and the agent platform provides flexible and intelligent services to the users. The component-based platform makes the software development tasks more efficient and flexible. It also provides efficient agent collaboration environment through the agent platform. For these reasons, we apply the proposed adaptive bridging approach to integrate the agent and component platform.

The agent platform and component platform are important entities in the ubiquitous computing middleware. A number of researchers have thus proposed various schemes and approaches for them. A dominating standard for the platforms, however, has not yet appeared [4]. As a result, integration of the agent, component, and middleware techniques is an urgent and indispensable task to support various existing standards. For the middleware of general distributed computing there exist some popular standards such as CORBA and DCOM. Especially, CORBA is the industrial standard managed by OMG[2], and it is accepted as a rational and reliable standard [7]. Before mid 1990's, CORBA was regarded as a heavy middleware framework. Nowadays, however, it is not considered to be heavy because of the significant improvement in the computing power. CORBA has been well applied in the enterprise business system and government public system. The traditional CORBA concept, as an expanded RPC, can be efficiently adapted to the existing large scale static computing paradigm such as the integrated system for government and aviation control system. However, the original form of CORBA is not appropriate for pervasive computing since its environment has various fastly changing contexts and consist of many different frameworks [3]. Hence there exists the need of the customization of CORBA for efficiently satisfying the requirement of pervasive computing. This motivated the development of a middleware architecture targeting the pervasive computing environment, which will be introduced in Section 2.

The proposed adaptive bridging approach is for the integration of the objects using portable interceptors. It allows transparent and efficient integration of heterogeneous middleware by providing two main functions as follows.

First, it supports the connectivity between the developed CORBA-based platform (CALM), agent platform (JADE), and component platform (MICO). The CALM objects, JADE objects (agent), and MICO objects (component) can transparently connect to each other by using the generic dynamic bridge with DII/DSI. For example, the agent objects in the JADE can transparently use the components in the MICO or the function of MOM (Message Oriented Middleware) in the CALM.

Second, the proposed bridging provides not only transparent bridging but also efficient and adaptive bridging using portable interceptors. The statistical information on

the object request frequency is used to alternatively select the static bridge or dynamic bridge based on it. The dynamic bridge, which uses DII/DSI, shows relatively lower throughput than the static bridge. However, the generic mapping of the dynamic bridge offers a smaller memory footprint than the interface-specific mapping of the static bridge. Here a bridge pool is maintained which has many optimized bridges appropriate to each type of objects. The main bridge makes decisions by using the information provided by portable interceptors. It allocates static bridge to hot objects for high throughput, which are requested frequently. The detail mechanism will be introduced in Section 3.

The proposed approach is evaluated using a test bed comprising of three different platforms - JADE, MICO, and CALM - on nine servers. The experiment reveals that the proposed adaptive bridging approach significantly reduces the response time compared to the case of using only either static bridge or dynamic bridge, especially when the number of activated objects is relatively large.

The rest of the paper is organized as follows. Section 2 describes the related work and Section 3 presents the proposed adaptive bridging approach. An experiment with the proposed approach is presented in Section 4. Section 5 concludes the paper with some remarks.

2 The Related Work

2.1 CORBA and Component Model(CCM)

The CORBA Component Model (CCM) is a standard component middleware technology that addresses the limitations in the earlier versions of CORBA 2.x middleware based on the DOC model. The CCM standard defined by the CORBA 3.x specification extends the CORBA 2.x object model to support the concept of components, and establishes standards for specifying, implementing, packaging, assembling, and deploying the components [2].

The component technology can overcome various limitations of conventional Object Request Brokers (ORBs) in developing distributed, real-time, and embedded (DRE) applications. It has particular advantages for building large-scale DRE systems. The CCM enables the composition and reuse of software components and the configuration of key non-functional aspects of DRE systems such as timing, fault-tolerance, and security.

2.2 CALM

The CALM (Component-based Autonomic Layered Middleware) [4] adopts both the reflective middleware and adaptive middleware concept to construct a flexible platform for the agents and provide efficient development tools. It consists of two internal layers, one external layer, and various tools forming an efficient agent-based platform. The internal layer consists of the communication platform layer and agent platform layer. The communication platform layer is composed such that it can provide various services based on situation and location using efficient communication protocols with light-weight devices in wired and wireless environment. The agent platform

layer is composed by components so that it can maximize the efficiency of the service, adapt itself to the environment, and accommodate the advantage of diverse agent systems. The external layer is composed of self-growing engine and ontology-based situation-awareness engine for providing intelligent services.

One of the most important characteristics of the CALM is the implementation of the MOM (Message Oriented Middleware). MOM lets the service consumers physically and temporally decouple from the service providers [3]. Communication between the service providers and their consumers is asynchronous, and they do not need to be available at the same time since they send and receive messages from the designated message queues. In contrast, RPC is a synchronous method of requesting remote service execution. Here the consumers must suspend the service execution until they receive a reply from the provider. We have implemented the function of the MOM in the CALM along with the agent system in JADE and MICO's CCM.

We next present the proposed adaptive bridging approach for transparent interconnection of heterogeneous middleware.

3 The Proposed Scheme

3.1 Bridging Heterogeneous Platforms

Here we integrate the agent platform (JADE) and CCM platform (MICO) transparently with the CALM. JADE does not fully support CORBA 3.0. Hence we need a bridge to connect it with CORBA-based middleware. JADE is a software framework implemented in Java language, and JAVA ORB is fully supported in JADE. If the applications of the CALM do not use CCM, JADE and CALM can communicate with each other through IIOP without a bridge. However, the agents in JADE cannot directly access to the CORBA components without a bridge because JAVA ORB does not support full CCM interoperability. In the CORBA specification, the CCM supports low version compatibility using the IDL compiler without a bridge. The ORB platform vendors, who can support the low version compatibility, however, are rare yet. Generally speaking, using the low version compatibility is not feasible [5]. Other heterogeneous agent platform or CCM platform cannot directly use the function of MOM or agent in the CALM without a bridge. However, the agents or the CORBA components can use the function of MOM or agents in the CALM with the proposed bridging approach [3].

There exist three kinds of bridging technique; static bridge, dynamic bridge, on-demand bridge. A static bridge requires static proxy and stub implementation to perform marshalling for each interface. If any change is made to the interface, the proxy and stub must also reflect the change. The interface thus has to be recompiled to generate new proxy and stub codes to encode, decode, and invoke a new interface. Consequently, having only static bridge is not enough in the pervasive environment, which shows fast context change.

On the other hand, the dynamic bridge frees the user from the recompilation task since it has a generic proxy capable of marshalling all data types based on some form of a run-time type library. This type of information look-up can be costly in terms of performance. Generally, dynamic bridges are not as efficient as static bridges. Despite

the potential negative performance factor, the generic mapping of dynamic bridge does offer a smaller memory footprint than the interface-specific mapping of static bridge. Therefore, dynamic bridge is appropriate for fast context change.

Yet a third kind of bridge is on-demand bridge. Here the bridge factory automatically generates the source code for static bridge, usually based on the interface definitions of the client and server, which are to be bridged. Based on these definitions, the bridging code must translate the values between the two middleware domains, and the mapping rules for the respective types are coded into the bridge factory. In this paper we propose a method combining dynamic bridge and static bridge generated by on-demand bridge for efficient integration of different platforms.

3.2 The Proposed Adaptive Bridging Strategy

In the proposed approach all the objects implemented in the server are classified into two objects, cool spot object and hot spot object, according to the request frequency measured by the portable interceptor. For cool spot object, the requests are bursty and the request frequency is not high. On the other hand, for hot spot object, requests are relatively persistent and the request frequency is high.

The objects in cool spot use dynamic bridge based on DII/DSI. On the other hand, the objects in hot spot use static bridge based on the on-demand bridge mechanism. By employing this approach we can achieve good balance between throughput and flexibility. In this scheme portable interceptors take a significant role. The portable interceptors transparently route the request for an object to the main bridge. Also, they provide the information on the request frequency to the main bridge. The interceptors, which are provided to the objects, directs the objects to use either the static bridge or dynamic bridge using the orders generated by the main bridge.

As we described above, the objects of hot spot use their own static bridge. The static bridge is made by the main bridge in run-time. The bridges already made are reserved in the bridge pool. The main bridge activates a bridge in the pool or makes an optimized bridge in run time based on the information provided by the portable interceptors [8].

3.3 The Structure of Adaptive Bridging

In this paper three different platforms are bridged for the performance evaluation of the proposed approach. Here CALM is the backbone middleware, JADE is the agent platform middleware, and MICO is the CCM middleware. We explain the proposed approach using an example.

The main bridge catches the request frequency information of the objects provided by the portable interceptors. The main bridge classifies the objects into two using this information; cool spot objects and hot spot objects. The threshold used for the classification is not always same. In this paper we empirically decide the threshold. For the classification of the objects as cool or hot, we use Poisson distribution.

$$P[K = k \text{ in } T] = \frac{(\lambda T)^k}{k!} e^{-\lambda T}$$
$$F(k) = \sum_{i=0}^{k} \frac{(\lambda T)^k}{i!} e^{-\lambda T} \quad (1)$$

The main bridge gets the λ value of each object, the average number of requests per unit time, from the portable interceptors. The threshold, *k*, is not always same as we mentioned above. The main bridge uses the CDF of Poisson distribution as a classification criterion. When the *F(k)* is larger than 0.7, the object is classified as a cool spot object. Otherwise, it is classified as a hot spot object.

The main bridge provides the function of dynamic bridge for cool spot objects. All inter-platform communication are routed by the main bridge. Figure 1 shows the flow of the dynamic bridging operation in the main bridge [8].

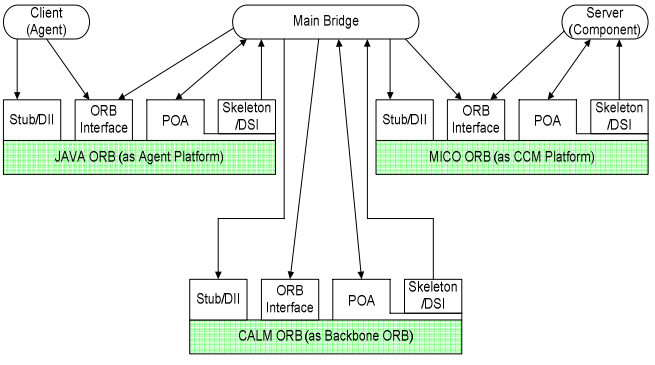


Fig. 1. The flow of the dynamic bridging operation in the main bridge

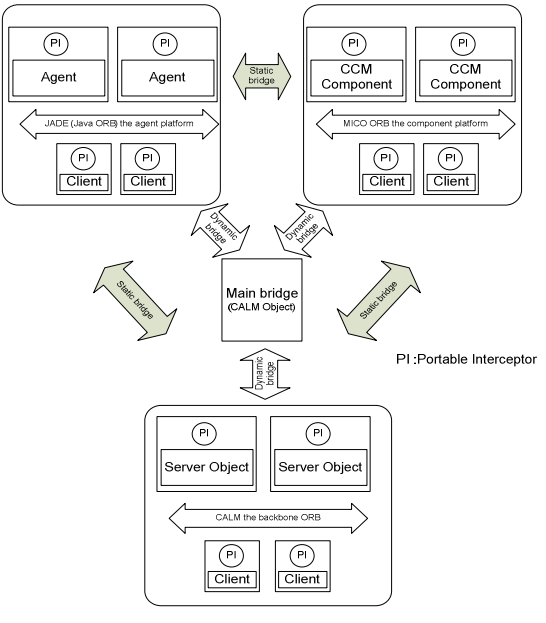


Fig. 2. The structure of the proposed bridging

The main bridge is an object of the CALM. As we see from the figure, it uses not only the CALM ORB but also MICO ORB and JAVA ORB. With this structure the main bridge can provide generic and dynamic bridging function. Generally speaking, however, dynamic bridges are not as efficient as static bridges.

The main bridge provides a static bridge to the objects moving from cool spot to hot spot. Hence the hot spot objects in the system use static bridges which are made by the main bridge in run-time. By using the static bridge, the objects of hot spot can achieve high throughput. However, all the objects cannot get their own static bridge. Then, very large memory and computing resources will be needed. Thus, the objects of cool spot, which are bursty and have low request frequency, just use the dynamic bridge.

Interceptor is an optional extension that allows additional services such as security to be inserted in the invocation path. For example, using the interceptors, we can be notified of the communication between a client and server, and modify the communications, if we wish, by effectively altering the behavior of the ORB. Here we use the portable interceptors for measuring the load of each object, which then send the information to the main bridge. Figure 2 shows the structure of the proposed bridging.

4 Performance Evaluation

Figure 3 shows the experiment environment implemented for the performance evaluation of the proposed approach. Jade, the agent platform, comprises of two Pentium 4 (3.0 GHz, 1GB RAM) servers and one SUN (SUN SPARC dual CPU 800MHz , 2GB RAM) server. The agent objects are loaded in this platform. MICO, the CCM platform, comprises of three Pentium 4 (3.0 GHz, 1GB RAM) servers. The CCM objects are loaded in this platform. The CALM platform comprises three Pentium 4 (3.0 GHz, 1GB RAM) servers. Here the COS objects are loaded. The agent, CCM, and COS objects are simple ones implemented for only the performance evaluation. These objects are for showing the connectivity and interoperability, and activated by the load generator for the experiment of heterogeneous middleware.

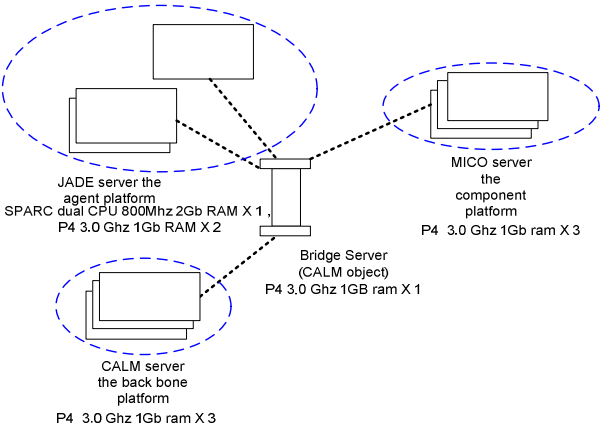


Fig. 3. The experiment environment

The experiment is performed with three different scenarios. In Scenario 1, the entire objects use only dynamic bridges. In Scenario 2, the entire objects use only static bridge of their own. Each object accesses the objects in other platform with its own stub and skeleton. We vary the number of activated objects while measuring the response time of each object. In Scenario 3, the main bridge classifies the objects as cool spot object or hot spot object by measuring the CDF of Poisson distribution. Scenario 3 is thus the case what we propose.

The number of requests of the activated objects is randomly decided. The rate of bursty requests (for the cool spot object) and persistent requests (for the hot spot object) are set to be same by controlling the load generator. In this experiment the threshold value k in the PDF of Poisson distribution is empirically decided by iterative experiments. The value of k is affected by the system clock speed and the size of RAM. We collect the result of experiment using the value of k showing the best result for each scenario.

Figure 4 shows the response times of the objects, which are measured by the portable interceptors. In Scenario 1 the response time using one dynamic bridge is shown. Here the entire objects use the dynamic bridge implemented by DII/DSI. Therefore, the system like Scenario 1 is generic and simple to implement. Since the dynamic bridge uses DII/DSI, the throughput is not high. Also, there exists a saturation point because the entire objects use only one bridge. In Scenario 2, the entire objects use their own static bridge. Thus the throughput is much higher than that with the dynamic bridge. Also, using multiple bridges avoids saturation. However, providing static bridge to every object is impossible in the pervasive computing environment, which has fast context changes. Moreover, letting the object with bursty requests use a static bridge can be a waste of resource since each static bridge consumes some memory and computing resource. If a system has lots of bursty objects that use a static bridge individually, it can even display a lower throughput than using only one dynamic bridge. In Scenario 3, the system uses a mixture of dynamic bridge and static bridge. This scheme allows the system to efficiently adapt to the pervasive computing environment of dynamic context changes. The result in Figure 4 well demonstrates the tradeoff between the static and dynamic bridging method.

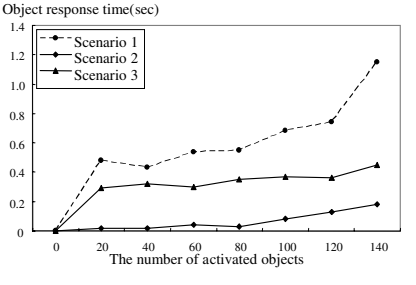


Fig. 4. The comparison of response times

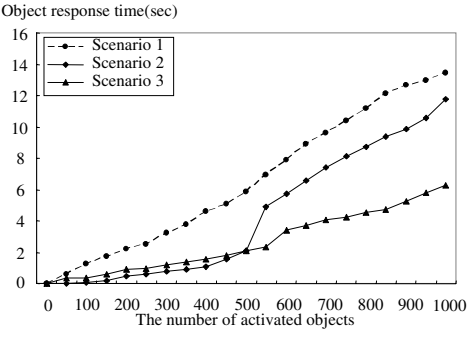


Fig. 5. The comparison of response times

Figure 5 compares the response times when the number of activated objects increases up to 1000. Scenario 1 treats the entire objects using the dynamic bridge without considering the context of the objects. In this case there is no serious problem if the system uses only bursty function or RPC. However, a problem occurs when the objects use the interface for transferring large size data. In Scenario 1, the response time increases linearly. In Scenario 2, each object has its own static bridge. In this case the response time is small when the number of activated objects is under 550. However, the response time drastically increases when the number exceeds 550 because of the resource overhead. Observe from the figure that the proposed Scenario 3 displays much smaller response time than Scenario 2 when the number of activated objects is relatively large.

5 Conclusion and Future Work

The pervasive computing paradigm is much more dynamic than the existing computing paradigm. Therefore, we need a new middleware framework for effectively integrating the pervasive objects. Because of the diversity of the objects in the pervasive computing environment, a single middleware is hardly sufficient. We need a transparent and efficient interconnection scheme for the objects. In this paper we have proposed an adaptive bridging approach for the interoperability of heterogeneous middleware. The proposed adaptive bridging approach is implemented in the CALM (Component-based Autonomic Layered Middleware), which have been developed by the authors.

We proposed a method of adaptively using dynamic bridge and static bridge. The portable interceptor of each object provides the information on the number of requests of the object to the main bridge. The main bridge then classifies the objects as cool spot objects or hot spot objects based on the interceptor's report. It allocates static bridges for hot spot objects having many requests using the on-demand bridging approach. For cool spot objects the main bridge allocates dynamic bridge. The bridges are stored in the bridge pool and are adaptively used as they are needed.

By employing the proposed approach we can take advantage of both the static bridge and dynamic bridge. The performance evaluation with a real experimental setting shows that the proposed scheme allows much higher performance than using

only static or dynamic bridge, especially when the number of activated objects is relatively large.

In this paper the experiment was performed with the CALM, JADE, and MICO. More research will be performed with other middleware such as light weight CORBA and real-time CORBA. We will also design more transparent and efficient bridging technique. A formal method finding the threshold used for deciding an object to be cool spot or hot spot will also be developed.

References

1. ORB interoperability architecture, Available from http://www.omg.org/, 2002
2. Object Management Group. The Common Object Request Broker: Architecture and Specification. Revision 3.0.3, Object Management Group, 2004.
3. Menasce, D.A.:MOM vs. RPC: communication models for distributed applications. Internet Computing, IEEE Volume 9, Issue 2, (2005) 90-93
4. Jennings, N.R.:An agent-based approach for building complex software systems. communications of the ACM, 44(4), (2001) 35-41
5. JADE, Java Agent Development framework web site : http://jade.cselt.it.
6. IBM Japan Research Group “Aglets Workbench” web site: http://www.trl.ibm.com/aglets.
7. Kim, J.H., Ramakrishna, R.S., Kim, Y.S.:LODIN: load distribution mechanism in CORBA using interceptor. TENCON. Proceedings of IEEE Region 10 International Conference on Volume 1, (2001) 61-64
8. Gunwani, R. V., QiWang, Zheng ,C.:Performance evaluation of middleware bridging technologies Performance Analysis of Systems and Software. ISPASS 2000 IEEE International Symposium, (2000) 34-39
9. Han, S.W., Song, S.K., Youn, H.Y.:CALM: An Intelligent Agent-based Middleware Architecture for Community Computing. Proceeding of SEUS-WCCIA'06, (2006) 89-94,

A Simulation Study Comparing the Performance of Two RFID Protocols

Mamatha Nanjundaiah and Vipin Chaudhary

Institute for Scientific Computing
Wayne State University,
Detroit, MI 48202
mamatha@wayne.edu,
vipin@wayne.edu

Abstract. This paper presents a comparison of version 1.0 Protocol Specification for 900MHz Class 0 RFID Tag with that of Class-1 Generation 2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz for large number of tags. Although the Generation 2 protocol has been released for Class 1, it is intended to be used by all classes of RFID tags. Using detailed simulation we compare their performance and security features. If security is a lower priority over speed in cases where one can be sure that the risk of presence of an eavesdropper is low, Class 0 draft protocol should be used as it provides a definite advantage over Generation 2 protocol. In application areas where the risk of consumer identity/privacy theft is high (consumer goods area), Generation 2 provides the security that eliminates the vulnerability of the RFID EPC structure.

1 Introduction

Radio Frequency Identification or RFID is the latest technology that has taken the supply chain industry by storm in the past few years. Although profitable and effective RFID systems have been in operation since decades, they were generally restricted to less complex – closed loop environments. The hype these days surrounds high-resolution supply chain applications of the future.

While the major concerns related to hardware costs and investment in a yet-to-be proven technology (on the scale it has been hyped to perform), are subsiding to a great extent with successful pilot projects along the length and breadth of the technology-hungry supply chain industry, the software components have only just begun to hold some solid ground.

Several protocols to singulate RFID tags, which are generally presumed to be used in significant numbers, have been put forth. We intend to analyze one such protocol that has been published by the EPCglobal Inc., the organization leading the development of industry – driven standards for the Electronic Product Code (EPC). EPCglobal had previously published a draft of standards for the RFID industry. Recently they have released the Generation 2 protocol specifications that have been intended to be more compatible with the prevailing industry standards.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 679 – 687, 2006.

The rest of the paper is organized as follows. Section 2 of this paper describes related work. Section 3 describes the anti-collision protocol as given in the Generation 2 specifications [2]. Section 4 explains the simulation of the Generation 2 protocol. Section 5 gives a comparison between the Generation 2 Protocol and the Draft Protocol and Section 6 discusses the result. We conclude the paper in section 7.

2 Related Work

Several anti-collision algorithms for RFID tags have been proposed in the research literature [4-9]. Hernandez *et. al.* [4] discuss a technique where each tag sends out its ID data continuously with a pause between two consecutive transmissions, where the pause is independent for each tag. Here the probability of all tags being read increases with reading time.

Herald Vogt [5] formulated the reader to broadcast a request, where the message contains an address range, which determines what data the tags should return, and a random number to be used as a seed by the tags in choosing a time slot. After the broadcast, N slots are provided for the tags to answer in. Here, an optimum N is determined so as to maximize the throughput.

Jacomet *et. al.* [6] proposed a technique similar to the binary protocol of [3], but here the tags respond with their next bit in the 1st or 2nd slot following the reader's command, depending on whether the bit's value is '0' or '1'. Hence there will never be a clash in response from tags.

Law *et. al.* [7] describe a query tree protocol that consists of rounds of queries and responses. In each round, if there is more than one tag that has the same prefix requested by the reader, then the reader appends a 0 and 1 to the same prefix and continues the queries. When a tag's ID matches the prefix uniquely, it is identified.

Zhou *et. al.* [8] compare the protocols given in [3] and [7] and provide an improvement to the algorithm in [7] by way of shortcutting the responses of tags that clash.

3 Description of the Generation 2 Protocol

Figure 2 gives the flow of the protocol in the singulation of tags.

The Generation 2 protocol provides three basic operations on how the reader manages tag populations:

- Select: Provides the operation of choosing a tag population for inventory and access
- Inventory: The operation of identifying tags
- Access: The operation of communicating with (reading from and/or writing to) a tag.

When a tag powers up in a reader's field and is not killed it enters the *Ready State*. It remains in this state until it receives a QUERY command whose parameters match its current flag settings. Tags that get selected load their slot counter with a Q-bit random number and transition to the *Arbitrate State* if that number is nonzero or

Reply State if the number is zero. If the tag loses power and is not killed it returns to the *Ready State.* The period of time between two QUERY commands is called an Inventory Round.

Arbitrate State holds tags that are participating in the current inventory round with non-zero values in their slot counters. This counter value is decreased for every QUERYREP command. When the value reaches zero, tags transition to the *Reply State*.

When tags enter the *Reply State,* they backscatter a 16-bit random number. If the tag receives a valid acknowledgement from the reader, it transits to the *Acknowledged State* and backscatters the PC (Protocol Control), EPC and the CRC-16 bits.

Tags in the *Acknowledged State* whose access password is nonzero shall transition to the *Open State* upon receiving a REQ_RN command and backscatter a new 16-bit random number. The reader uses this number along with subsequent commands to the tag.

Tags in the *Acknowledged State* whose access password is zero shall transition to the *Secured State* upon receiving a REQ_RN command and backscatter a new 16-bit random number. The reader uses this number along with subsequent commands to the tag. Tags in the *Open State* will transition to the *Secured State* upon receiving a valid ACCESS command maintaining the same handle that was exchanged with the reader while transitioning from the *Acknowledged State* to the *Open State.*

Tags can be permanently disabled with the KILL command, which transitions them to the *Killed State* when received with a valid nonzero kill password and a valid handle.

4 Simulation

For simulation purposes, the complexity of the above protocol has been reduced by the following assumptions:

- No miscommunications between readers and tags have been allowed. Therefore there will be no variations between transitions from one state to another apart from the path taken for the simulation as shown in the Figure 1.
- The simulation ends when all tags have been identified. There are no 'ACCESS' commands simulated here.
- The Electronic Product Code has been assumed to be 32 bits long and has been generated using CSIM inbuilt uniform, beta and geometric random number generators.

The exact algorithm followed for the purpose of simulation can be explained with the help of Figure 2.

The simulation has been designed to run 16 inventory rounds. Each QUERY command uses one of the 16 combinations of the first 4 bits of the tag IDs (0000, 0001, 0010 …). This way, all the tags are divided into 16 groups, irrespective of the distribution of the tag IDs.

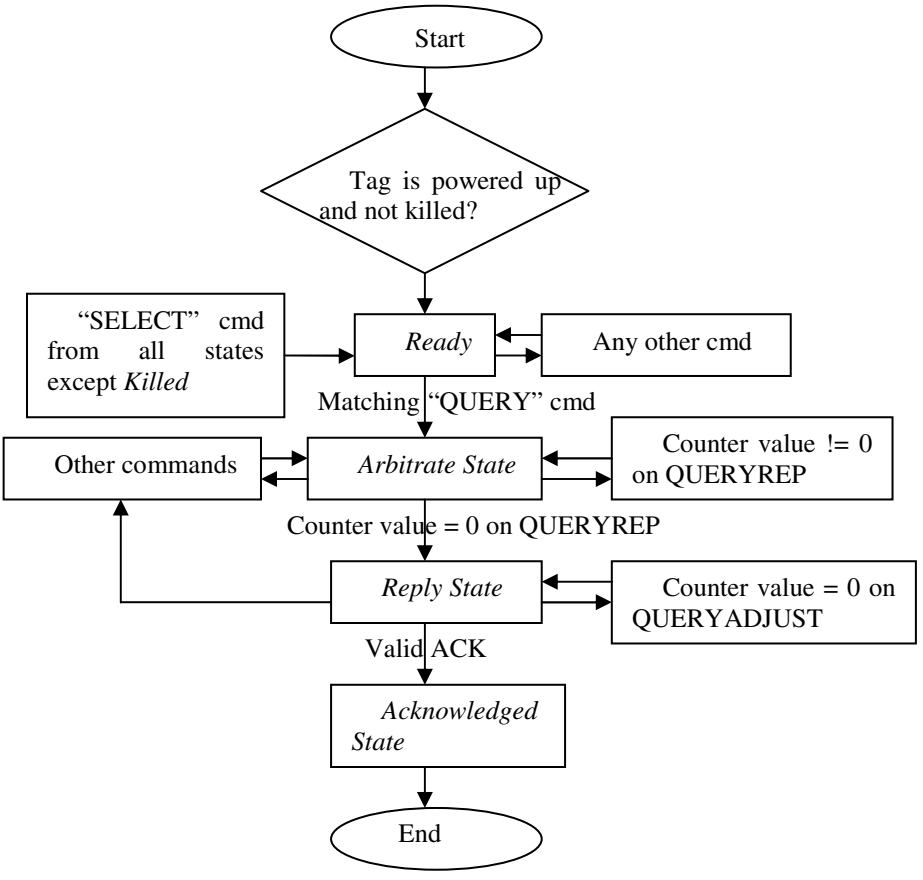


Fig. 1. Algorithm followed in the simulation of the protocol

Tag IDs are generated randomly using "Uniform Distribution", "Beta Distribution" and "Geometric Distribution", over the entire range of 0 to (2^32-1).

All tags power up in the *Ready State.* Tags selected by a matching QUERY command, generate random numbers for their slot counters, based on the value of Q and transition to the *Arbitrate State*. Tags reduce their counter value at every QUERYREP command. At zero counter value, the tags transition to the *Reply State*, backscattering a 16-bit random number. If the reader at this state detects a single tag, the reader acknowledges the tag with the same 16-bit random number.

When the tag receives a positive acknowledgement from the reader, it transitions to the *Acknowledged State* backscattering its PC, EPC and CRC-16 bits. If a QUERY, QUERYREP or SELECT command follows this transmission, it means that the reader has identified the tag. The simulation ends when all tags reach the *Acknowledged State.*

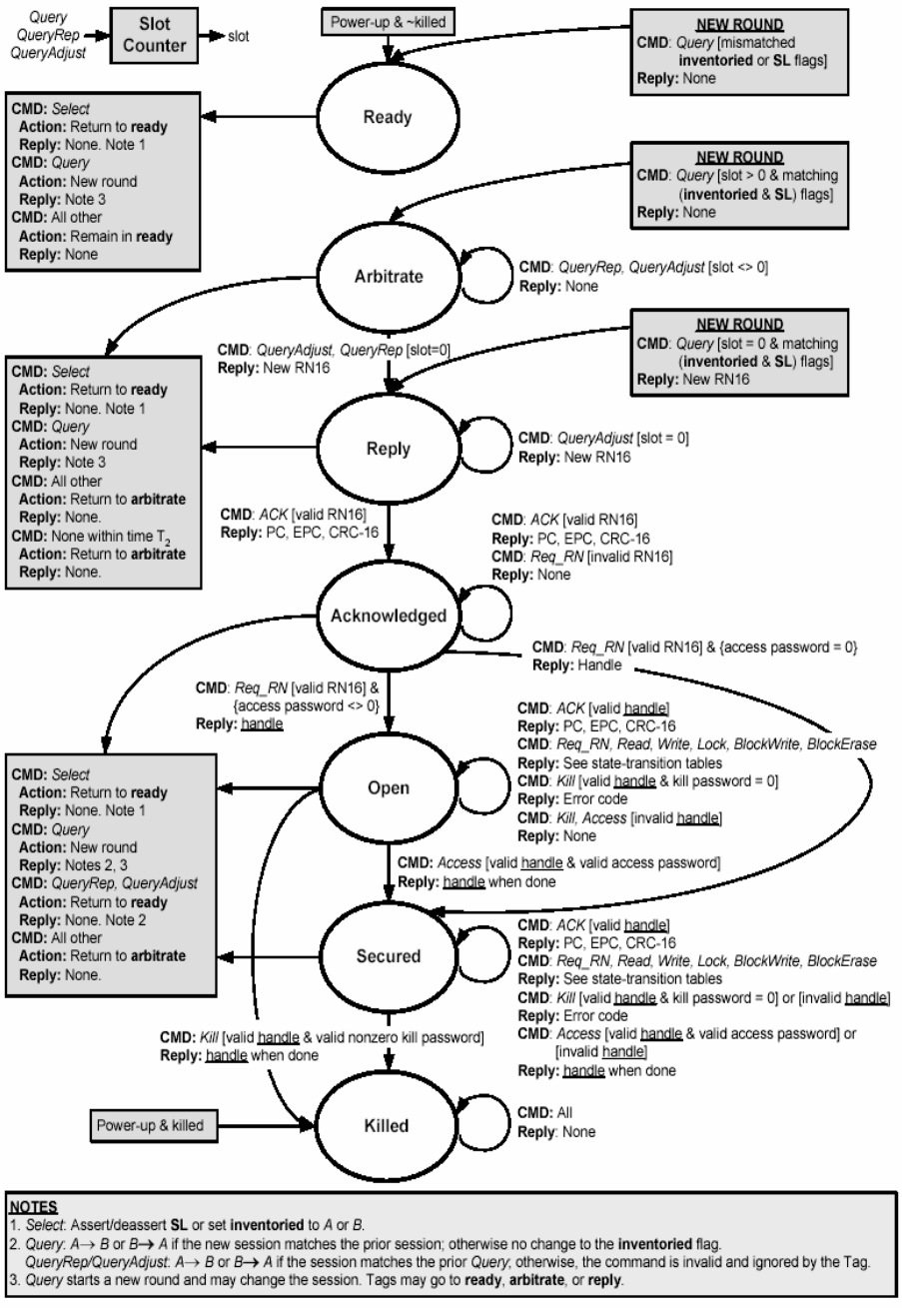


Fig. 2. Tag State Diagram. Source - "EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz", EPCglobal Inc., [2].

5 Comparison with Draft Protocol

The pros and cons of the Generation 2 Algorithm as compared to the Draft protocol can be categorized as follows:

Simplicity:

- Draft Protocol - Simple binary tree approach. Tags go through minimal number of "States" in order to be recognized. The acknowledgment from the reader is a single bit.
- Gen 2 - The number of states the tags need to go through is higher. Reader's acknowledgment consists of 16 bits.

Speed:

- Draft Protocol - Due to reduced number of tag "States", and single-bit acknowledgments, this protocol is more speed-efficient. There is no exchange of handles between the tags and reader.
- Gen 2 - The larger number of states the tags need to go through, coupled with the 16-bit handles and acknowledgments slows down the protocol

Security:

- Draft Protocol - Since the reader acknowledges every bit the tag sends out, an eavesdropper could easily know the IDs of all the tags recognized from listening to the reader, though it cannot listen to the tags directly.
- Gen 2 - Here, the reader never repeats the ID of the tag. The only thing an eavesdropper reader can hear is the 16-bit acknowledgement from the reader. When information is sent to the tag from the reader, it is cover-coded with the handle. (EXORed with 16-bit random number generated by the tag)

Compatibility:

- Draft Protocol - The tag structure is not compatible with the current ISO standards followed by the industry.
- Gen 2 - Tag structure is such that existing ISO standards can also use the protocol easily.

Tag Structure:

- Draft Protocol - Simple 64/96/256 – bit continuous memory with header, first bits, domain manager, object class and serial number
- Gen 2 - Tag memory is larger and complicated being divided into 4 banks each with different subdivisions

Time:

- Draft Protocol - Minimum, optimal time is utilized in the recognition of tags.
- Gen 2 - Significantly greater time is consumed to incorporate the greater security measures.

6 Results

In order to compare the performance of the Generation 2 protocol with the Draft Class 0 Protocol analyzed in our previous paper [1], we have simulated both the protocols for the maximum case of 19 tags. The results of the simulation are shown in Figures 3-5.

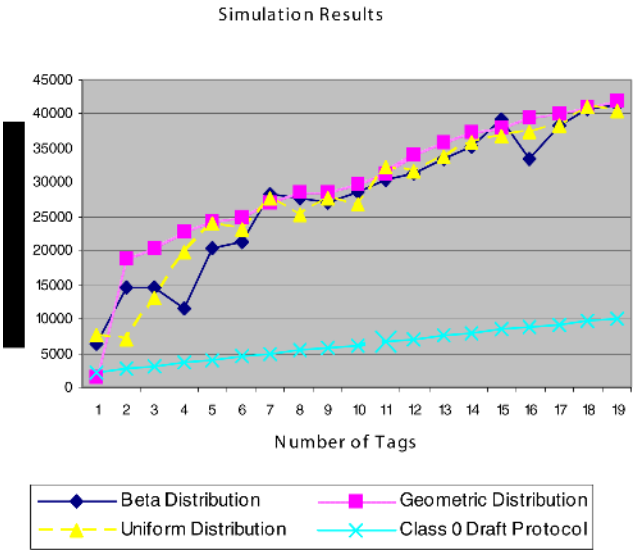


Fig. 3. Comparison of Class 0 Draft Protocol and Generation 2 protocol (for Beta, Geometric and Uniform distribution cases of generation of tag IDs)

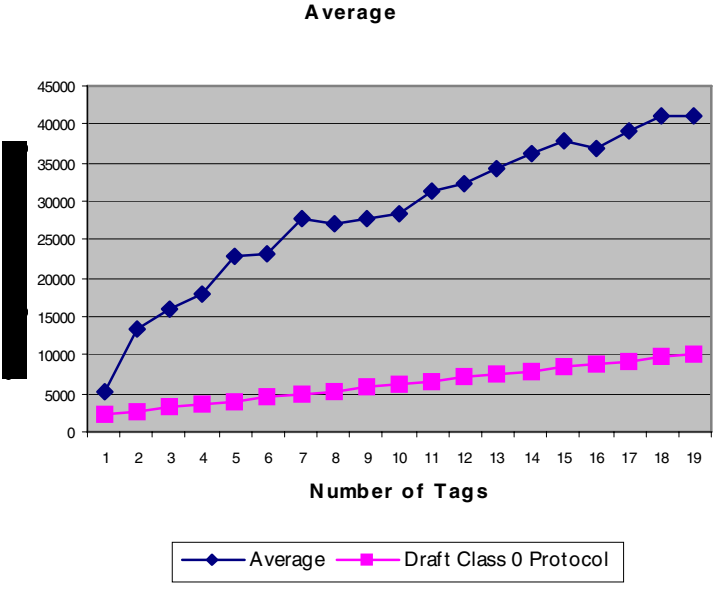


Fig. 4. Illustration of comparison with average values

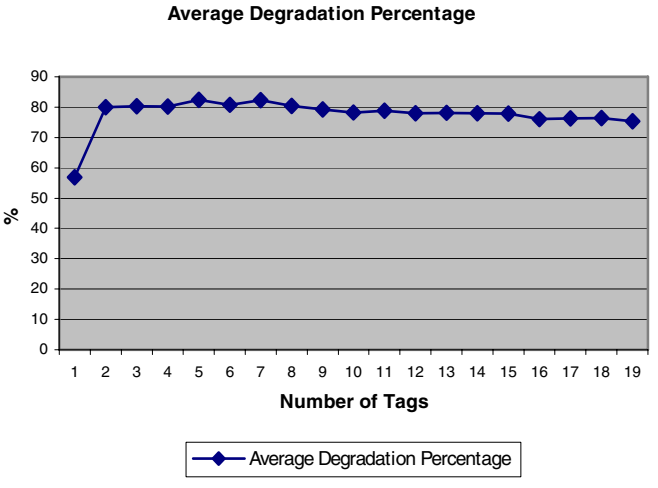


Fig. 5. Degradation of Gen 2 protocol over the Draft protocol

The degradation in performance over the Draft Class 0 protocol is apparent. For the case of 19 tags simulated here, an average degradation of 77.6% was observed.

7 Conclusion

The Generation 2 protocol has a very high standard of security. This comes with a heavy trade off against the rate of recognition of tags. At the same time RFID application in consumer goods area may not have much of a choice in considering an alternative anti-collision protocol as the risk of consumer identity/privacy theft is high and Generation 2 protocol provides the security that eliminates the vulnerability of the RFID EPC structure. From the point of view of a less rigorous application of the RFID system, where risk of presence of an eavesdropper is low, Class 0 draft protocol should be used as it provides a definite advantage over Gen 2.

References

1. Nanjundaiah, M., Chaudhary, V.: Improvement to the anticollision protocol specification for 900 MHz Class 0 Radio Frequency Identification Tag, IEEE Computer Society Press, proceedings the First International Workshop on Ubiquitous Smart Worlds (2005) 616-620
2. EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz, EPCglobal Inc., 31st January (2005), www.epcglobalinc.org
3. Draft Protocol Specification for a 900MHz Class 0 Radio Frequency Identification Tag, MIT Auto-ID Center, 23rd Feb (2003), www.epcglobalinc.org
4. Hernandez, P., J.D. Sandoval, J.D., Puente, F., Perez, F.: Mathematical Model for a Multiread Anticollision Protocol, IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, (2001) 647-650

5. Vogt, H.: Multiple Object Identification with Passive RFID Tags, IEEE International Conference on Systems, Man and Cybernetics, (2002) 6-9
6. Jacomet, M., Ehrsam, A., Gehrig, U.: Contactless Identification Device with Anticollision Algorithm, IEEE Conference on Circuits, Systems, Computers and Communications (1999) 269-273
7. Law, C., Lee, K., Siu, K.Y.: Efficient Memoryless Protocol for Tag Identification, Proceedings of the 4th ACM International workshop on Discrete algorithm s and methods for mobile computing and communications, (2000) 75-84
8. Zhou, F., Jin, D., Huang, C., Hao, M.: Optimize the Power Consumption of Passive Electronic Tags for Anti-collision Schemes, ASIC Proceedings of 5th International Conference, (2003) 1213-1217
9. R. Hush, D.R., Wood, C.: Analysis of Tree Algorithms for RFID Arbitration, Proceedings of International Symposium on Information Theory, (1998) 107

FreeSpeech: A Novel Wireless Approach for Conference Projecting and Cooperating*

Wenbin Jiang, Hai Jin, Zhiyuan Shao, and Qiwei Ye

Cluster and Grid Computing Lab
Huazhong University of Science and Technology, Wuhan 430074, China
{wenbinjiang, hjin}@hust.edu.cn

Abstract. This paper presents a novel wireless projecting and cooperating approach named FreeSpeech based on Ad Hoc network, which makes conference exchange more convenient and ubiquitous. All attendees can access this system by their laptops or other mobile computing devices all around the conference venue and make the projections of their presentations without connecting to the projector physically. Moreover, they can make cooperative display simultaneously with others. FreeSpeech takes a mixed *Virtual Networking Computing* (VNC) C/S mode based on some improvements on the traditional ones. A novel IP multicast-based transfer and control mechanism is presented to make large numbers of attendees obtain the wall screen contents by their laptops locally without aggravating the overhead of bandwidth remarkably. An efficient remote screen synchronization method is employed to transfer screens of some attendees to the projector-connected computer, which also can save the consumed bandwidth significantly. All above methods use the limited wireless bandwidth more efficiently. Performance evaluation shows that this approach can work perfectly for ubiquitous conference occasions.

1 Introduction

Now, making presentations by projectors to show some slides and other materials has been the major means for many conferences, sessions, seminars, etc. Traditionally, some boring physical switches have to be done to let speakers copy their materials to some specific computer connected to the projector or directly connect their computers to the projector physically. These inefficient actions always slow down the progress of conference and make audiences whiny. Moreover, it is hard for speakers to make some collaborative displays with others.

To overcome the above shortages of traditional approaches and bring more convenience to all conference attendees, this paper presents a novel wireless approach for conference projecting and cooperating. First, Ad Hoc network based on IEEE 802.11 protocol [1] is employed here as the fundamental network infrastructure due to its mobility and flexibility. Based on Ad Hoc mode, we construct a new approach named FreeSpeech to make all speakers present their speeches without any physical switches. What they need to do is just to register their laptops to this system through

* This paper is supported by National Science Foundation of China under grant No.90412010.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 688 – 697, 2006.

Ad Hoc network. Then they can do their presentations all around the conference venue with their own laptops, projecting the screens of their laptops to wall screen freely. The attendees can also get the contents of the wall screen by their laptops wherever they are, which is very useful for remote audiences who can not see the wall screen clearly.

2 Related Work

There are several approaches for remote computer screen sharing such as *Virtual Network Computing* (VNC) [2], pcAnywhere, Netmeeting. Here, we develop FreeSpeech based on improving VNC because VNC is open source, platform-crossing and stable. VNC uses *Remote Frame Buffer* (RFB) protocol as the means of the data transfer from the server to the clients. It has been used for many applications such as remote medical diagnosis, embedded control systems and remote conferences. A remote control mechanism of an MR imaging study via tele-collaboration tools based on VNC was presented [3]. VNC was applied as the remote display technology for remote graphical sessions in an environment for enabling interactive grids [4]. In [5], an approach named μVNC was proposed for the light-weight simple screen process. The above applications take VNC as a mature screen transfer way with traditional VNC C/S mode.

For conference-related applications, there also have been many approaches developed. A desktop screen sharing system was proposed for remote conferences [6]. VNC was used for the *remote screen synchronization* (RSS). A VNC-based teleconference system was presented [7]. It allowed multi-users to control one same remote computer screen through Internet by *Face Mouse* that is a mouse pointer followed by some users' faces. However, most of these conference-related applications are based on TCP-based RFB protocol and the number of the clients who require the screen of the VNC server at the same time is limited. For the real conference occasion of the FreeSpeech, there are often hundreds of attendees who want to connect to this system and share the wall screen locally. TCP can not endure this heavy overhead totally over Ad Hoc network. This paper presents a novel transfer mechanism by applying IP multicast, which can be considered as the extension of RFB.

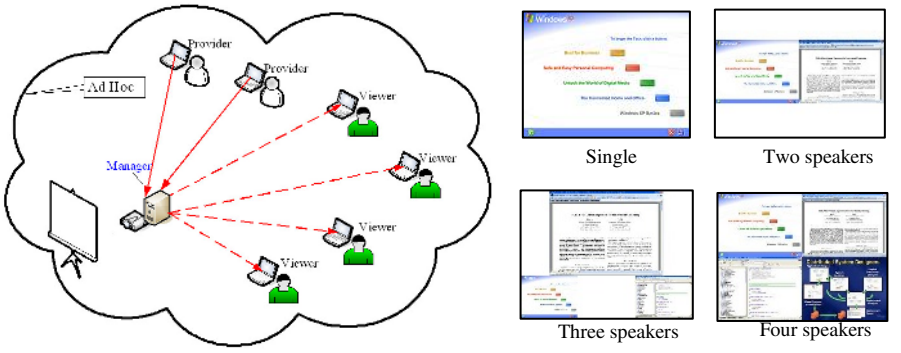
Additionally, to reduce the data of the encoded screen to be transferred, there have been many approaches presented. However, all these approaches encode source screen by some static image compression ways such as Tight [8], OLI [9], FCE [10], which still bring heavy overheads to the networks, especially to Ad Hoc network, with limited bandwidth. To save the bandwidth further, we present a new RSS method based on the *association of spatio-temporal redundancy reductions* (ASTR) for FreeSpeech.

The remainder of the paper is organized as follows. Section 3 gives the system design of the FreeSpeech. Section 4 proposes the IP multicast-based transfer and control mechanism. Section 5 describes the remote screen synchronization approach. Finally, we conclude this paper in section 6.

3 System Design

Here, we present a characteristic framework for conference presentation based on 802.11 Ad Hoc network (in Fig. 1 (a)). A novel mixed VNC C/S mode is designed to let all attendees send their presentations to the wall screen as the providers. And also if they want, they can receive the contents of the wall screen by their laptop locally as the viewers.

The construction of the 802.11 Ad Hoc is a relative mature work. Here we assume that the network has been set up. Then we pay main attention to the construction of the mixed VNC C/S mode for the screen data process and transfer.



(a) Framework (b) Some collaborative display modes

Fig. 1. The framework and display modes of FreeSpeech

To make the following discussions more clearly, some roles in the FreeSpeech are defined as:

- Provider: The laptop that sends its screen to the manger.
- Viewer: The laptop that receives wall screen from the manger.
- Manager: The projector-connected computer responsible for receiving providers' screens, as well as sending the wall screen to all viewers.

A reverse VNC C/S mode is applied here to transfer the providers' screens to the manager. In the traditional VNC structure, The VNC clients receive the screen contents from the VNC server. The flow direction of the screen data is from one to one or one to many. However, in FreeSpeech, the data flow direction is from many providers to one manger. Here, the providers take the role of VNC servers as they provide screen contents and the manger takes the role of VNC client as it receives the screen data. So it is a reverse VNC C/S mode with data flow from many to one. Moreover, the screens from different providers are needed to be combined and displayed by several modes for free cooperative presentation, shown in Fig. 1 (b).

For other attendees who just want to get the content of the wall screen as the viewers, the manager transfers the combined screen to them. This is a VNC C/S mode. In traditional VNC applications, this kind of transfer is always implemented over TCP

connection, which is suitable for the applications that have limited viewers. However, in some real conference environments, the number of attendees is sometimes hundreds or more. It is obvious that it is unacceptable to do such screen transfer work by hundreds of traditional TCP connections to manager. In section 4 we will present a new way to resolve this problem.

Moreover, if the number of the providers is considerable, the bandwidth occupied by the data transfer from the providers to the manger will be considerably wide. In some traditional wired network environments, it is acceptable. However, for the less bandwidth Ad Hoc networks, it is hard. In section 5, we will discuss it in more details.

Fig. 2 shows the architecture of the FreeSpeech. The manager mainly consists of Display Component, User Management and Communication Channel. The Display Component is in charge of decoding the received screen contents, combing them if necessary and displaying them finally. At the same time it prepares the contents for viewers according to what it has received. User Management is responsible for registrations of all attendees and the schedule management of all speeches. Communication Channel provides general TCP connection for the communication of user management and the extended RFB channel for screen content transfer. The latter is also the main focus of the section 4. Besides some common communication components, the provider also has User Registration, Screen Process. The Screen Process is the one of the most important components, which takes care of the screen capturing and encoding. For the viewers, the Screen Process just decodes the screen data from the manager and displays it. The function of User Registration is similar to one of the provider but makes the users registered as viewers.

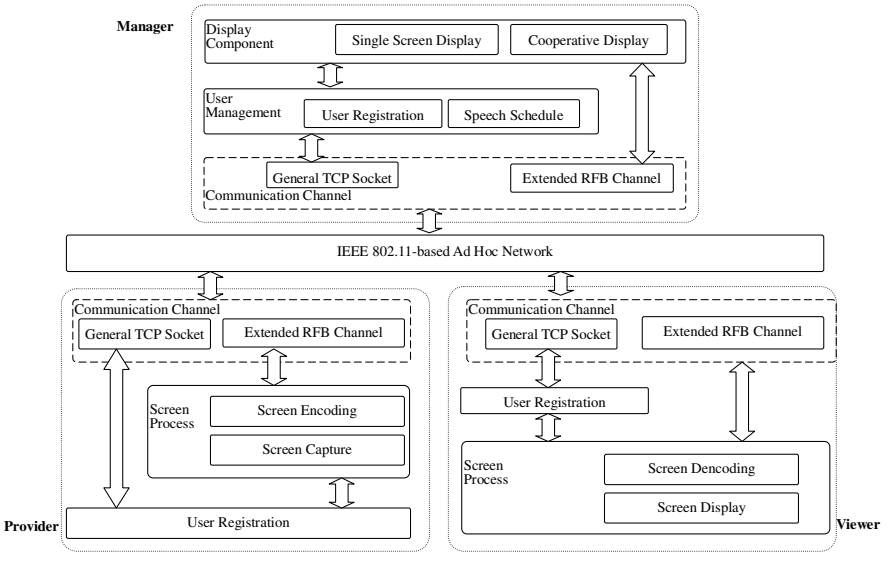
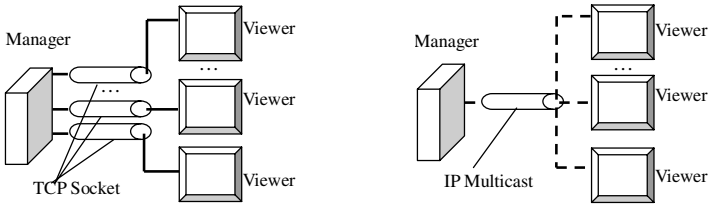


Fig. 2. The architecture of the FreeSpeech

4 IP Multicast-Based Transfer and Control Mechanism

As mentioned in section 3, the TCP-based traditional VNC method of RFB is not suitable for FreeSpeech to transfer screen contents from the manager to large numbers of viewers. Here, we use IP multicast to solve this problem.

Fig. 3 shows the difference between using TCP-based approach and the IP multicast-based approach. It is obvious that, IP multicast can save much bandwidth because it just sends one copy of screen data to all viewers. However, in most of traditional VNC-related applications, TCP-based approach has been employed. The main reasons are that, on one hand, the number of users for these applications is not very large and the traditional network can undertake the bandwidth overhead; on the other hand, it is relative hard to construct the IP multicast channel over global Internet, which needs all routers to support IP multicast. Moreover, the IP multicast is not a reliable channel. In FreeSpeech, it is hard to complete this screen transfer to so many views by TCP, since the bandwidth required exceeds the capacity of the Ad Hoc network totally. The IP multicast must be taken into account. Fortunately, the topology of 802.11-based Ad Hoc network is relative simple. Most of them are configured as single hop network. Moreover, if multi-hops are required, it is also easier to configure them to support IP multicast.



(1) TCP-based approach of VNC (2) IP multicast-based approach of FreeSpeech

Fig. 3. The advantage of IP multicast

Since IP multicast does not provide any reliable data transfer guarantee, we must make some control strategies to reduce the ratio of data loss and improve the display effect of the viewers' laptops.

One fundamental idea is to use one IP multicast packet to carry the data of one independent rectangle required to be updated in the screen. Of course, the size of this kind of data packet must fit for the capacity of the IP multicast. Since the rectangle is independent, the loss of the corresponding data packet does not affect the processes of other rectangles. However, we still want to know which rectangle has been lost on the viewer sides so that more actions can be employed to fix this problem with best effort. On the manger side, every data packet representing one rectangle is assigned a sequence number.

Fig. 4 gives the brief structure of one data packet that carries the data of one rectangle to be updated. IP_M_HEAD and UDP_HEAD are used for the construction of the IP multicast group and the UDP data transfer over IP multicast. SEQ_NUM is the sequence number specified by the manger to make the viewer detect which packet has been lost. RECT_POS is the position of the rectangle and the RECT_DATA is the

encoded rectangle data. The 16-bit SEQ_NUM is the key parameter for making best-effort guarantee for viewers. The following is the brief description of this process.

When the manger receives screen data from the providers, it combines them into one image and shows on the wall screen. Then it knows exactly which areas of the wall screen have been changed and the update data should be sent to all viewers who have subscribed this service. Since the size of IP packet must be less than 2^{16}=64K theoretically, and we want every IP multicast packet can carry the whole data of one rectangle, all areas to be sent are divided into some smaller rectangles. Experiments show that if the IP packet size is larger than 20KB, the ratio of losing packets will increase obviously with the increase of packet size. This is caused by the loss of some link layer fragments of some big IP packets, which always consist of a number of such fragments. Here we set the size of one IP packet less than 20kB.

IP head	**UDP head**	**Application layer**					
IP_M_HEAD	UDP_HEAD	MSG_TYPE	SEQ_NUM	RECT_POS	ENC_WAY	…	RECT_DATA

Fig. 4. The IP packet of one rectangle transfer

Fig. 5 shows more details about this process. R_1 and R_2 are two rectangle areas to be updated. r_1-r_n is the smaller rectangles produced by this division. Of course the division algorithm should take all possible relationships of the two areas into account and make sure the size of every small rectangle should be less than 20KB. The encoding method also should be included to make this decision. A mechanism called *Limited Reliable Multicast* (LRM) is presented here to give a limited best-effort guarantee for the screen data transfer from the manager to the viewers.

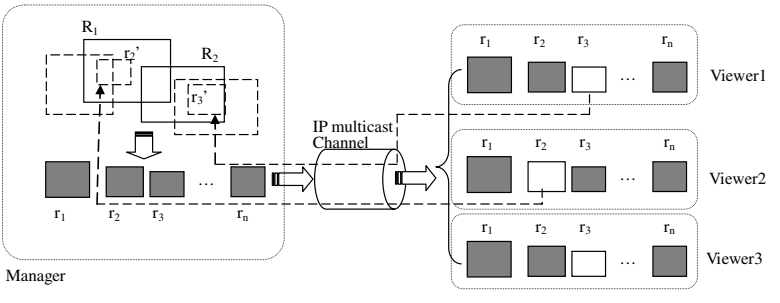


Fig. 5. The mechanism of the Limited Reliable Multicast

The main steps of LRM are:

1. After getting r_1-r_n, the manage assigns everyone a sequence number (SN) and sends them to viewers one by one through the IP multicast channel.
2. Each viewer tries to receive these IP packets and encodes them. Through SN, the viewer can detect if there is any packet lost. If not, the viewer will combine these small rectangles into some larger ones and update the corresponding areas accordingly.

3. Otherwise, if the viewer finds any packet lost, it will be added into the list of the lost packets and a timer is set accordingly.
4. Before the timer is over, the viewer will compare the rectangle with every current area to be updated. If the rectangle has been covered, the lost IP package will be ignored. Otherwise if the timer is over, the viewer will reply the manger with ACK carrying the SN and the position of the lost rectangle and will delete this record. Of course, other viewers also will receive the ACK. They will compare SN from the ACK with their lists of the lost packets. If they find the SN in it, they will remove the corresponding one from the list since the manager has known this case and will resend it.
5. When the manager receives the ACK, it will combine the lost rectangles (denoted by r_2' and r_3' in Fig. 5) into its current areas to be sent and then divide them into a new set of smaller rectangles and send them.

Here, it is unnecessary and unreasonable for viewers to ask the manager to resend lost packets for many times, since the following coming areas will cover them soon and repetitious resending lost packets will aggravate the overheads of the bandwidth obviously.

5 Remote Screen Synchronization

LRM can save much bandwidth of transferring screen contents from the manger to all viewers. However this means is totally unsuitable for the screen data transfer from the providers to the manager since the contents from different providers are different completely. To save the bandwidth occupied by this process, the best way is to reduce the encoded screen data produced by every provider respectively. To achieve this target, in FreeSpeech, we present a new screen process approach by employing the ASTR. In some traditional VNC applications, several efficient approaches such as Tight, FCE have been presented for screen process. However, they all just take spatial redundancy into account and the compression ratio is limited. Here, we take the temporal redundancy between the current changed screen area and the previous frame into account and improve the compression ratio further.

Fig. 6 shows a scenario of cooperative presentation of three providers. For explanation purpose, we name the providers as P1, P2 and P3 as the representatives of the ASTR mechanism.

If P1 has a changed area (we call it sub-region for convenience) to be transferred, we should analyze it and find some temporal redundancy first by comparing with the previous frames. By this comparison, the sub-region can be divided into three kinds of sub-areas: The sub-area with no change (such as D), the sub-area with only translation movement (such as A, C) and the sub-area with more complex movement (such as B). After this analysis, the data that devotes the sub-region will decrease obviously. For D, nothing is needed to do. For A, C, just some metadata (denoted by a block point) including the movement vector, the position are needed to be transferred. For B, more elements are needed. Besides metadata, the compensated block (denoted by hexagon) is also needed to be compressed and transferred together. Here the compensated block is an image containing the difference between the B sub-area and its corresponding

counterpart in the previous frame. Generally, the data of this sub-region will be reduced largely by this approach. More details of the ASTR can be seen in [11].

When the manger receives the data encoded by ASTR, it decodes them. At the same time, the providers, P2 and P3, will transfer similar sub-regions to the manger. Finally the manger scales all of them into corresponding places.

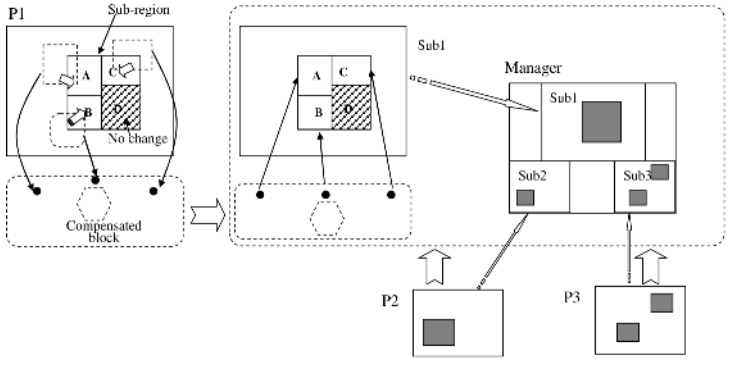


Fig. 6. The mechanism of the ASTR

6 Experimental Evaluation

We have developed FreeSpeech by C++ and taken the UltraVNC as one of the major references. Table 1 shows some differences between the traditional VNC-like applications and the FreeSpeech. Here, FreeSpeech takes a mixed VNC C/S mode to meet the special requirement of the real conference occasions. Moreover, to fit for the limitation of bandwidth of Ad Hoc network, LRM based on IP multicast and ASTR for RSS are presented here to save bandwidth consumption efficiently. Additionally, it provides a set of collaborative display modes for cooperative presentations for multi-speakers (described in Fig. 1(b)).

Table 1. Differences between TVAs and FreeSpeech

	TVAs*	FreeSpeech
Mode	C/S	mixed C/S
Network	Traditional Internet	Ad Hoc
data transfer	TCP	TCP + LRM
Screen process	Tight, FCE, etc.	ASTR
Display mode	Single	Cooperative

*TVAs: traditional VNC-like applications.

To evaluate its performance, we build a test environment. The Ad Hoc network is constructed with 11Mbps (802.11b). The manger is a laptop with CPU Pentium M 1.7GHz, Memory 512MB, and Windows XP as OS. Three providers and 10 viewers work together with CPU from 1.4~1.7GHz and Memory 256~512M.

First, we discuss the performance of LRM compared with the traditional TCP. Fig. 7 shows the bandwidth consumed by LRM and TCP. The solid line is the curve of the consumed bandwidth by the screen data transferred from the manger to 10 viewers by LRM and the dashed line is the one by the screen data transferred from the manger to 3 viewers by TCP. Here just one provider sends its screen to the manger with playing a flash-like tour named "tour.exe" (single mode seen in Fig. 1(b)). It is obvious that the LRM takes less bandwidth than TCP, although LRM serves more viewers. Moreover, when the number of viewers increases to 10 for TCP, the network becomes congested, and the delay between the manger and the viewers becomes considerable long. The screens of viewers are updated very slowly and brokenly, which is unacceptable.

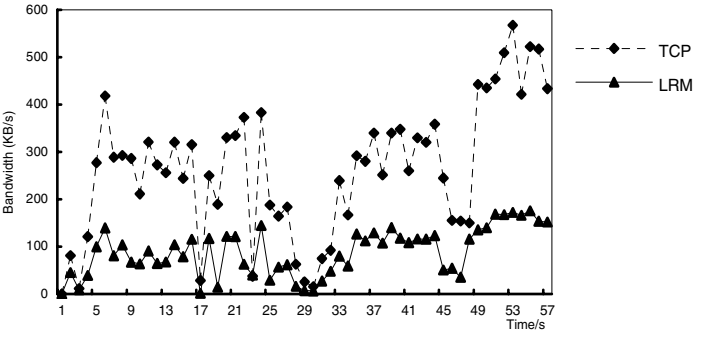


Fig. 7. Performance of LRM

To evaluate the performance of RRS based on the ASTR, we let two providers send their screens to the manger (the mode of two speakers shown in Fig. 1(b)). Fig. 8 shows the curves of the consumed bandwidth by Tight, FCE and ASTR. It is obvious that the ASTR can save considerable bandwidth compared with other approaches.

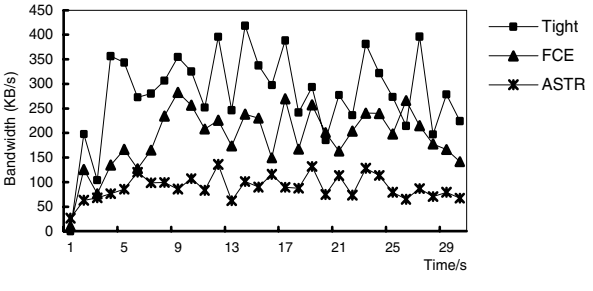


Fig. 8. Performance of ASTR

More experiments have been conducted and the results show that the FreeSpeech is competent and efficient for many real conference occasions.

7 Conclusion

Ad Hoc network has become more and more popular in real applications. More novel applications are desired to provide more convenience for people. Here we propose and develop a new conference projecting and cooperating system named FreeSpeech based on VNC-like mode. Particularly, some innovations and improvements have been implemented. First, a mixed VNC C/S mode is applied to make FreeSpeech work smoothly for all attendees. Second, to save the bandwidth consumed by FreeSpeech, two renovations of LRM and ASTR are presented. The experimental results demonstrate that the FreeSpeech not only can provide high efficiency for the real conference occasions but also can offer much more freedom for attendees.

References

1. Granelli, F. and Kliazovich, D.: Performance improvements in data transfer over 802.11 WLANs. Proceedings of Global Telecommunications Conference Workshops, 29 Nov.-3 Dec. (2004) 253-257
2. Richardson, T., Stafford-Fraser, Q., Wood, K. R., and Hopper, A.: Virtual network computing. IEEE Internet Computing, Vol.2(1), (1998) 33-38
3. Sullivan, J. M., Jr., Mullen, J. S., Benz, U. A., Schmidt, K. F., Murugavel, M., Chen, W., and Ghadyani, H.: Remote control of an MR imaging study via tele-collaboration tools. Proceedings of the SPIE, Vol. 5748(1), (2005) 557-564
4. Talwar, V., Basu S., and Kumar R.: An environment for enabling interactive grids. Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing (HPDC'03), Washington, DC, USA, (2003) 184-193
5. Haraikawa, T., Sakamoto, T., Hase, T., Mizuno, T., and Togashi, A.: μVNC over PLC: a framework for GUI-based remote operation of home appliances through power-line communication. IEEE Transactions on Consumer Electronics, Vol. 48(4), (2002) 1067-1074
6. Negishi, Y. and Kawaguchi, N.: A desktop screen sharing system based on various connection methods. Transactions of the Institute of Electrical Engineers of Japan, Vol.125-C, No.12 (2005) 1882-1890
7. Jin Hak Kim, Sang Chul Ahn, and Hyoung-Gon Kim: Teleconference system with a shared working space and face mouse interaction. The 5th Pacific Rim Conference on Multimedia (PCM 2004), Tokyo, Japan, LNCS Vol.3332, (2004) 665-671
8. Kaplinsky, K. V.: VNC tight encoder-data compression for VNC. Proceedings of the 7th International Scientific and Practical Conference of Students, Post-graduates and Young Scientists, Tomsk, March, (2001) 155-157
9. Li, F. and Nieh, J.: Optimal linear interpolation for server-based computing. Proceedings of the IEEE International Conference on Communications, New York, NY, USA, April (2002) 2542-2546.
10. Li, Q. and Li, F.: FCE: a fast content expression for server-based computing. Proceedings of IEEE International Conference on Communications, Vol.3, June (2004) 1426-1430
11. Jiang, W., Jin, H., Guo, M., Shao, Z., and Zhu, Z.: A novel remote screen synchronization mechanism for ubiquitous environments. Proceedings of the First International Symposium on Pervasive Computing and Applications, Xinjiang, August 3-5 (2006)

Performance Analysis of Unified Data Broadcast Model for Multi-channel Wireless Databases

Agustinus Borgy Waluyo[1], Bala Srinivasan[1], David Taniar[1],
Wenny Rahayu[2], and Bernady O. Apduhan[3]

[1] Clayton School of Information Technology, Monash University, Australia
{Agustinus.Borgy.Waluyo, Bala.Srinivasan,
David.Taniar}@infotech.monash.edu.au
[2] Department of Computer Science and Computer Engineering,
La Trobe University, Australia
W.Rahayu@latrobe.edu.au
[3] Kyushu Sangyo University, Japan
bob@is.kyusan-u.ac.jp

Abstract. The use of data broadcasting in wireless environment has been of much interest especially to deal with the exponential increase of mobile users due to its scalability. In this paper, we present a unified broadcast model in multi-channel wireless databases and its comprehensive performance analysis. This model aims to minimize query access time, tuning time and power consumption of mobile users when obtaining broadcast database items. This scheme also concerns with single and multiple data items request. A prototype and simulation-based experiment has been developed to evaluate the performance of the broadcast model. We compare the performance of the proposed model against the conventional scheme and we found that the proposed unified model provides substantially better performance in every aspect of the evaluation. It is also shown that the results of our simulation are very close to those obtained from the prototype system.

1 Introduction

In wireless computing, applications such as accessing airline schedules, stock activities, traffic conditions, and weather information using PDAs on the road are expected to become increasingly popular. It is noted, however, that several wireless computers including laptops and palmtops use batteries of limited lifetime for their operations and are not directly connected to any power source. As a result, query response time and energy saving are very important issues to resolve before the potential of wireless computing can be fully realised. Wireless data broadcasting is an effective mechanism for disseminating database information to mobile clients due to its scalability [2]. However, data broadcasting requires sequential access of data items. Thus, the increasing number of broadcast database items causes mobile clients to wait for a substantial amount of time before receiving their data items of interest. Consequently, the advantages of broadcast strategy will be diminished. In wireless databases, query response time and energy saving issues are represented by three

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 698 – 707, 2006.

performance metrics namely: query access time, client tuning and power consumption. *Access time*: Elapse time from the time a request is initiated until all data items of interest are received. (ii) *Tuning time*. Amount of time spent by the client to listen for the desired broadcast data item(s). Two modes exist for tuning time; active and doze mode. Active mode is when the client listens into the channel for the desired data item, hence costly to power consumption, while doze mode is when clients simply turned into a power saving mode.

In this paper, we *present* unified data broadcast model for multi-channel wireless databases. The unified model consist of data organisation and indexing scheme for multi channels mobile broadcast environment, which is designed to minimise query access time, client's tuning time, and power consumption. A prototype and simulation model has been developed to provide a comprehensive evaluation performance of the proposed scheme.

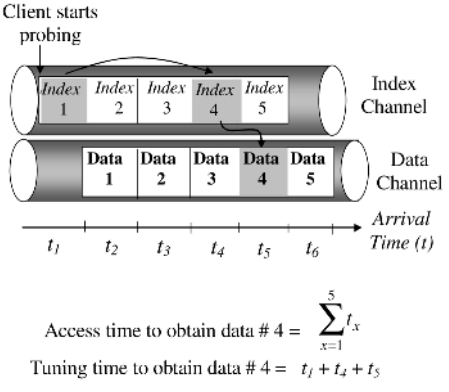


Fig. 1. Data broadcast scheme

In general, broadcast indexing causes a trade-off between optimizing the client tuning time and the query access time. The consequence of minimizing one of them is the increase of the other. For instance, to minimize the access time is to broadcast the index once in each cycle but it will make the tuning time suffer since the client will have to wait for the index to arrive which happens only once in each broadcast cycle. On the other hand, when the index directory is frequently broadcast, the access time will be greatly affected due to the occupancy of the index in the cycle. In this paper, we allocate the data and index segment in different broadcast channel. Figure 1 shows the access and tuning time to retrieve data item # 4, which starts from the time client probes into the beginning of the index channel until the desired data item has been obtained.

The rests of the section in this paper are organized as follows. Section 2 contains the related work of the proposed technique. It is then followed by the description of the proposed scheme and its application in section 3. Section 4 describes the experiment-based simulation and prototype system of our unified broadcast model as compared to conventional method. Finally, section 5 concludes the paper.

2 Related Work

Prabhakara et al presented a broadcast ordering scheme, where the hot items or the most frequently accessed data items are broadcast more often than cold items [8]. This basic technique will be used for comparison in the later section. Huang and Chen proposed some algorithms to identify the most effective organization of broadcast data items [3]. They concern with broadcast data organization scheme in the context of multiple broadcast channels. However, they do not apply indexing scheme in the broadcast program. This situation may lead to wasteful power consumption as mobile clients needs to keep listening into the channel and filtering the data items until the desired ones arrived in the channel.

Broadcast indexing technique is able to minimize the amount of time client listening into the channel and thus less power consumption [4]. The tree-indexing based on *B*+-tree structure is first introduced in [4]. It is then expanded and modified in [5]. However, these techniques integrate index and data segment in a single channel. When the index and data segments are integrated into a single channel, the length of broadcast cycle increases. Consequently, the average access time improves substantially. Another indexing technique is used by [7]. This technique is also based on B^+-tree structure, and it incorporates a separate index channel to locate the data segment in multi data channel. However, this technique does not concern with broadcast ordering issue. Our proposed technique considers single and multiple data items retrieval and allocates the data and index segment in separate broadcast channel. This fact distinguishes our paper from the existing works.

3 Unified Data Broadcast Model for Multi-channel Wireless Databases: Proposed Scheme

3.1 Preliminaries

In this paper, we concern with request that returns multiple data items. A number of applications in this category include a situation when mobile client wants to obtain more than one stock prices information at the same time (i.e. to list the stock price of all car companies under General Motors corp.) To accommodate this type of request, we consider the relationship between one data item and the others based on the query access patterns of mobile clients. Some analysis on query patterns and access information has been reported by Microsoft research group in [1]. Once the access information is received by the server, the statistics will be compiled and the broadcast organisation scheme will be implemented. This paper concerns with broadcast program without replication whereby all data items are broadcast with equal frequencies or uniform frequencies.

The unified data broadcast model is illustrated in Figure 2. The server periodically broadcasts data items according to a predetermined broadcast program. When a user submits a query to his or her mobile device, the mobile device will retrieve the required data item from the broadcast channel by referring to the data index information. The server retrieves the database items from data repository, the results are then processed by the broadcast program generator. The broadcast program

generator concerns with the ordering and allocation of data items in the broadcast channels. To obtain optimum placement of broadcast data items, the broadcast program generator needs to be informed about query access patterns of mobile clients. The broadcast program is then passed to the data indexing generator. The indexing generator constructs the index based on the information from the broadcast program generator. The final broadcast program and the index structure are then transmitted to the server cache. The broadcast scheduler will have the final broadcast program and the relevant index structure from the server cache. It is then responsible for the scheduling issues to broadcast the data items and their indexes before they are transmitted to the wireless channels.

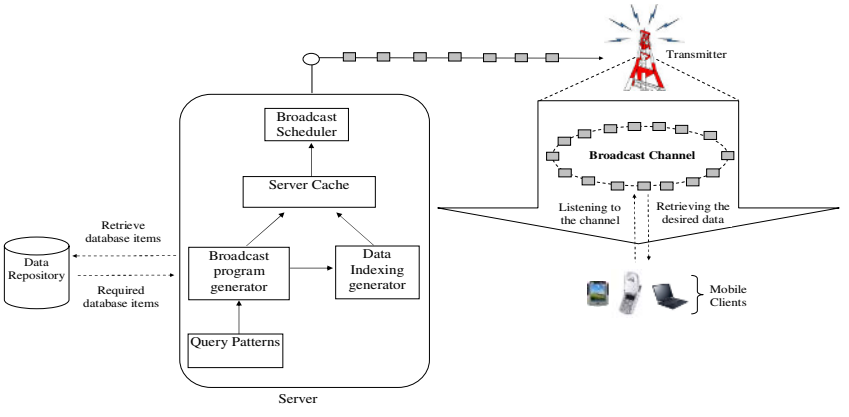


Fig. 2. Unified data broadcast model

3.2 Data Organisation Scheme

Let us denote $D = \{d_1, d_2 \ldots d_m\}$, which is a set of data items to be broadcast in the server, and Q as a set of queries $\{Q_1, Q_2, \ldots, Q_n\}$. In this case, we assume the data item has an equal size and the order of the retrieval can be arbitrary, which means if any of the required data by Q_i arrives in the channel, it will be retrieved first. Each query, Q_i, accesses a number of data items j, where $j \subset D$. The broadcast channel is indicated by C, and the length of the broadcast cycle in a channel is given by BC. We denote the broadcast schedule as $S = \{d_x, d_y, \ldots . d_z\}$. Similarly, the broadcast program for each channel is defined by SC.

Our data broadcast organization scheme is depicted in Figure 3. The first stage is to list the data items in a sequence order. The second stage is to analyse the relationship between data items and calculate the access frequency of each pair of data item according to the given query patterns. The final stage is to order the pairs of data items based on the value of the access frequency in a descending order. Subsequently, they are placed into one or more broadcast channels. The number of broadcast channels will follow the requirement of the optimum number of channels required to broadcast the data items [10].

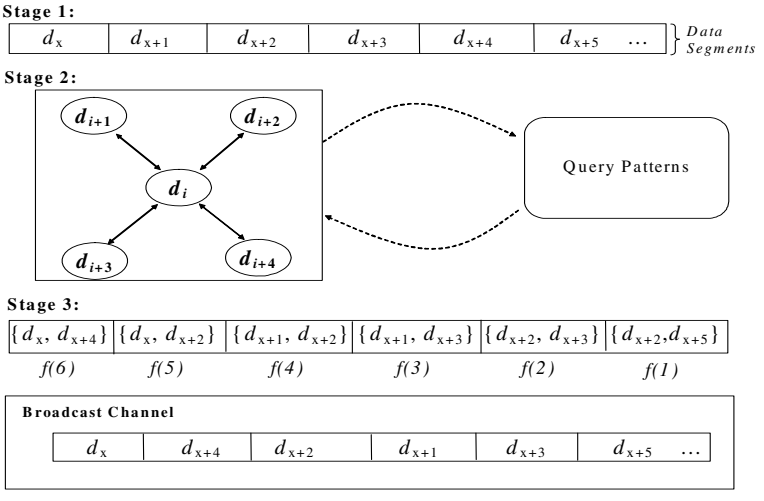


Fig. 3. Proposed data broadcast ordering scheme

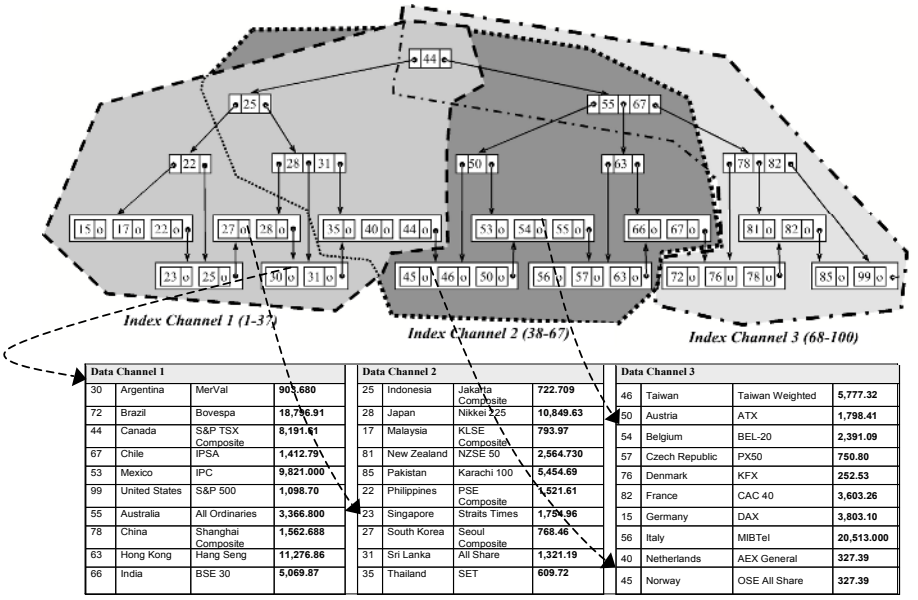
The next issue to consider is to specify the broadcast program. The broadcast program indicates the schedule of the first item in *S* at the broadcast cycle, and the sequential items follow the order that has been generated. To achieve the most effective broadcast program, one must note the statistical patterns of users when they start listening or probing into the channel For instance, this is done by allocating the first item in each *SC* at point 3/4 in the broadcast program. In this way, the majority of requests can be served within a single broadcast cycle. However, this means about 50% of requests have to wait for a short time period but the overall performance will be better since only a fraction of requests needs to wait for the next cycle. The broadcast program can then be generated at regular intervals.

3.3 Global Indexing Scheme

Global Indexing scheme is designed based on *B+* tree structure. Global index consists of non-leaf nodes, and leaf node. Leaf node is the bottom most index, where each key point to actual data items. When being broadcast, each physical pointer to the neighbouring leaf node as well as actual data item are replaced by a time value, which indicates when the leaf node or data item will be broadcast. Global index scheme is partitioned into a number of channels and each channel has some degree of replication. Each index channel has a different part of the entire index structure, and the overall structure of the entire index is still preserved. As such, Global indexing scheme has the same behaviour as single channel model. Global index is exhibited in Figure 4. Further details on our Global Index model can be found in [11]. Query processing of mobile clients in this scheme can be described as follows:

– Mobile client tunes in one of the index channel (i.e. can be of any index channel).
– Mobile client follow the index pointer to the right index key. The pointer may lead to another index channel that contains the relevant index. While waiting for the index to arrive, mobile clients can switch to doze mode.

– Mobile client tunes back on at the index segment that has the right index key, which point to the data channel that contains the desired data item. It indicates a time value of the data to arrive in the data channel.
– Mobile client tunes into the relevant data channel, and switch back to doze mode while waiting for the data item to come.
– Mobile client switches back to active mode just before the desired data item arrives, and retrieves the required data items.



Data Channel 1

30	Argentina	MerVal	903.680
72	Brazil	Bovespa	18,706.91
44	Canada	S&P TSX Composite	8,191.61
67	Chile	IPSA	1,412.79
53	Mexico	IPC	9,821.000
99	United States	S&P 500	1,098.70
55	Australia	All Ordinaries	3,366.800
78	China	Shanghai Composite	1,562.688
63	Hong Kong	Hang Seng	11,276.86
66	India	BSE 30	5,069.87

Data Channel 2

25	Indonesia	Jakarta Composite	722.709
28	Japan	Nikkei 225	10,849.63
17	Malaysia	KLSE Composite	793.97
81	New Zealand	NZSE 50	2,564.730
85	Pakistan	Karachi 100	5,454.69
22	Philippines	PSE Composite	1,521.61
23	Singapore	Straits Times	1,794.96
27	South Korea	Seoul Composite	768.46
31	Sri Lanka	All Share	1,321.19
35	Thailand	SET	609.72

Data Channel 3

46	Taiwan	Taiwan Weighted	5,777.32
50	Austria	ATX	1,798.41
54	Belgium	BEL-20	2,391.09
57	Czech Republic	PX50	750.80
76	Denmark	KFX	252.53
82	France	CAC 40	3,603.26
15	Germany	DAX	3,803.10
56	Italy	MIBTel	20,513.000
40	Netherlands	AEX General	327.39
45	Norway	OSE All Share	327.39

Fig. 4. Global index model

4 Performance Evaluation

This section presents the performance analysis of the unified broadcast model. The performance results are obtained from simulation as well as prototype-based experiments. The prototype system provides a small-scale real-world implementation. The performance of the proposed model is analysed and compared with the existing one. In the existing scheme, the data items are ordered and broadcast based on the access frequency without considering the relationship with other data items. This existing data ordering method was discussed in [8]. The indexing scheme of the existing method follows the B^+ tree index structure, which is broadcast in an index channel. Subsequently, the performance results obtained from the simulation and the prototype system will be compared. In this evaluation, the statistical patterns of users who start listening or probing into the channel follows the behaviour of Gaussian distribution and each query pattern has equal access probability. The index structure is broadcast following a top-down approach and the index distribution model employs a

non-replication model similar to that introduced in [5]. Table 1 shows the parameters of concern for these experiments.

The development of our prototype system has been reported in [12]. In the prototype model, the broadcast data is retrieved from a central database server and they relate to the share price indices context. The hardware technology used for both the client and the server devices is one desktop computer as a server and a notebook computer as a client. The server device is configured with Intel Pentium 4 CPU 2.4GHz , 1 GB RAM, 120 GB HDD. While, the notebook used is a Fujitsu P series Ultra-light Notebook with Transmeta Crusoe™ TM5800 (800MHz) and an integrated WLAN, 256MB SDRAM and 30GB HDD. The server device communicates to the client device over a wireless LAN. The standard wireless Ethernet networking technology 802.11b is utilized. The Microsoft® Winsock control is applied to enable wireless communication in a UDP/IP network [6].

Table 1. Parameters of concern

Parameters	Value
Size of data Items	420 Kb
Bandwidth	11Mbps
Query Patterns	5
Number of Dependent Items in Query	1-4
Number of Broadcast Channel	2
Number of Broadcast Data Items	8
Number of Requests	10-30
Global Index	
Node Pointer Size	24-32 bytes
Data Pointer Size	15 bytes
Indexed Attribute Size	8 bytes
Index Arrival Rate	1 index node per 5 sec interval
Non-Global Index	
Node Pointer Size	19-22 bytes
Data Pointer Size	10 bytes
Indexed Attribute Size	8 bytes
Index Arrival Rate	1 index node per 5 sec interval

The prototype system utilises the share price indices context for data broadcast services. The data content is retrieved from a data source, Microsoft® Access® database. The table stores records of the share prices details including company name, ASXcode, abbreviation, category, and price. The size of each record amounts to about 420 Kb. For simplicity, there are 8 records altogether and 5 query patterns in accessing the records. Each query pattern may request up to 4 share prices at the same time. The bandwidth is determined from the standard bandwidth for IEEE802.11b. The number of requests ranges from 10 to 30 request.

4.1 Simulation Model

In this simulation, two software packages are deployed, namely Visual Basic 6.0 and *Planimate* or animated planning platforms [9]. The algorithm to determine the best

broadcast program is coded in Visual Basic 6.0. The simulation is carried out using *Planimate* simulation tool. The simulation environment is set to apply exponential distribution for index and data item inter-arrival rate given an average value. The simulation is run for thirty iterations, and derives the average result accordingly. In the query patterns, it considers requests that queried up to four numbers of broadcast items, which is reflected in the number of dependent items in the query.

4.2 Performance Analysis

The analysis involves two cases. The *first* case is to compare the query access time of the proposed method with the existing method. The comparison analysis is carried out using a simulation program and prototype system. The performance results of the two schemes are then provided. In the *second* case, the tuning time of Global Index in the proposed method is evaluated against Non-Global Index in the existing method. The results are based on the prototype system. The average power consumption of mobile clients for listening to the channels can then be determined accordingly.

Case 1. As can be seen from Figure 5(a), the simulated-based query access time performance of the proposed method outperforms the existing method.

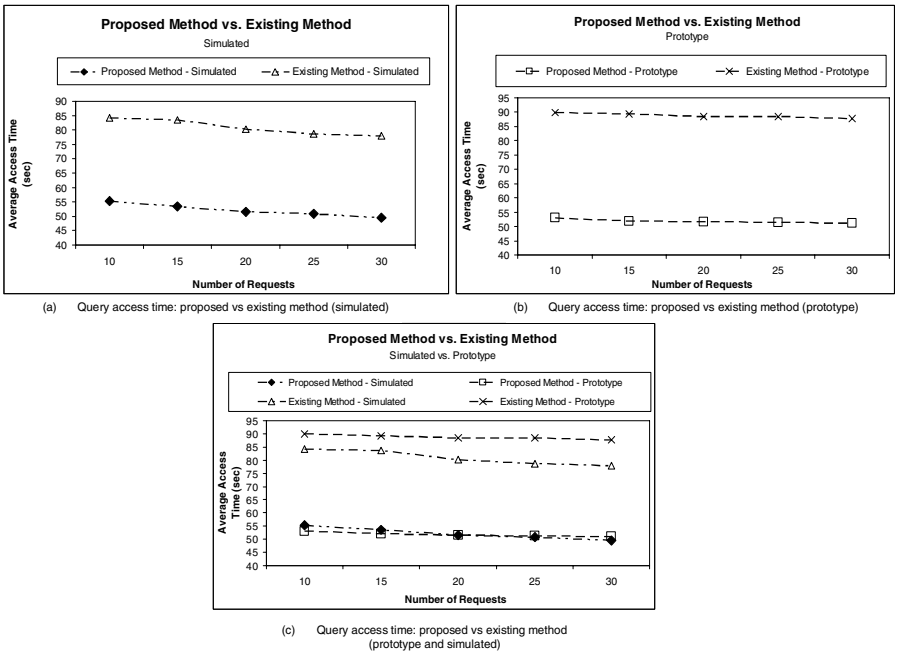
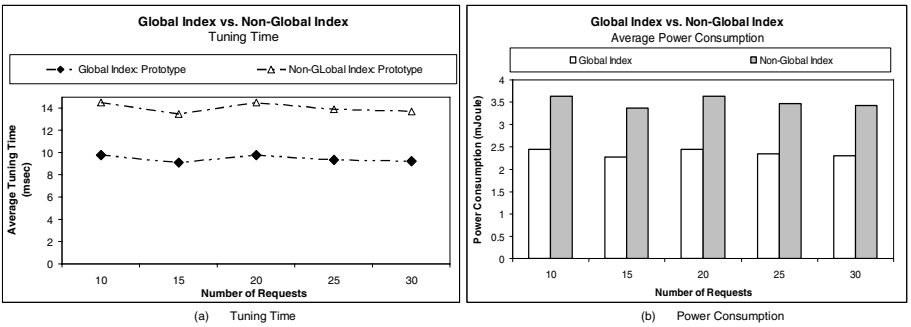


Fig. 5. Query access time: proposed vs existing method

The graph indicates that the proposed method provides a better access time compared with the existing approach by about one and a half times lower access time. The prototype based query access time performance as shown in Figure 5(b) confirms

the superiority of the proposed method. The simulation result provides a slightly higher average access time for each measurement as compared with the prototype results. The graph comparison of the query access time performance obtained from simulation and prototype experiments are given in Figure 5(c). It can be seen that the results from the two types of experiments are very close to one another. This proves the confidence of the accuracy of the simulation model. A more detailed comparison can be found in Table 2. It indicates that the simulated results have an average of 5.572% error in comparison with the prototype ones.

Case 2. The amount of the client's tuning time is depicted in Figure 6(a). It shows that the tuning time of clients with Global Index in the proposed method is substantially lower than for the Non-Global Index in the existing method. Based on this result, we can calculate the power consumptions of mobile clients accessing the desired data items. To obtain the client's tuning time, it is necessary to determine the client's processing time per Index node. In this case, the processing time is derived from the prototype model. The results are averaged from 10 iterations and the client's



(a) Tuning Time (b) Power Consumption

Fig. 6. Tuning time and power consumption: global index vs non-global index (prototype)

Table 2. Simulated vs prototype experimental results

Average Access Time (s) Simulated vs Prototype System					
	Average Access Time (s)				
Number of Request	10	15	20	25	30
Proposed Scheme (Simulated)	55.25	53.48	51.53	50.92	49.58
Proposed Scheme (Prototype)	52.98	51.95	51.67	51.32	51.13
Existing Scheme (Simulated)	84.14	83.58	80.21	78.67	77.92
Existing Scheme (Prototype)	89.88	89.31	88.53	88.42	87.73
Average Error Rate	**5.572%**				

processing time per Index node is found to be 0.0019 sec/index node. The power consumption is measured based on the formula given in [5]. Power Consumption = (250 x *Time during active mode*) + (0.05 x *Time during power saving mode).* Figure 6(b) shows the power consumption comparisons between the Global Index and Non-Global Index methods.

5 Conclusions and Future Work

In this paper, we have presented a unified broadcast model for multi broadcast channel environment. We have developed a simulation as well as prototype model to analyze the performance of the proposed scheme. We compare the performance of the proposed model against the conventional scheme and we found that the proposed unified model provides substantially better performance in every aspect of the evaluation. It is also shown that the results of our simulation are very close to those obtained from the prototype system.

For future work, we will incorporate a certain degree of data replication in the broadcast program. We will investigate the most effective degree of the replication as well as the structure and scheduling of the broadcast program.

References

1. Adya, A., Bahl, P., Qiu, L.: Analyzing the Browse Patterns of Mobile Clients. In Proc. of the 1st ACM SIGCOMM Workshop on Internet Measurement (2001) 189-194.
2. Barbara, D.: Mobile Computing and Databases-A Survey. IEEE TKDE, 11(1) (1999) 108-117.
3. Huang, J.L., Chen, M.-S.: Broadcast Program Generation for Unordered Queries with Data Replication. In Proc. of the 8th ACM SAC (2003) 866-870.
4. Imielinski, T., Viswanathan, S., Badrinath, B. R.: Energy Efficient Indexing on Air. In Proc. of the ACM Sigmod Conference (1994) 25-36.
5. Imielinski, T., Viswanathan, S., Badrinath, B. R.: Data on Air: Organisation and Access. IEEE TKDE, 9(3) (1997) 353-371.
6. Jones, A., Ohlund, J.: Network Programming for Microsoft Windows. Microsoft Press, Redmond, Washington, U.S.A (2002).
7. Leong, H. V., Si, A.: Data Broadcasting Strategies over Multiple Unreliable Wireless Channels. In Proc. of the 4th CIKM (1995) 96-104.
8. Prabhakara, K., Hua, K.A, Jiang, N.: Multi-Level Multi-Channel Air Cache Designs for Broadcasting in a Mobile Environment. In Proc. of the ICDE (2000) 167-176.
9. Seeley, D. et al: Planimate^{tm}-Animated Planning Platforms. InterDynamics Pty Ltd (1997).
10. Waluyo, A.B., Srinivasan, B., Taniar, D.: Optimal Broadcast Channel for Data Dissemination in Mobile Database Environment. In: Zhou, X., Jähnichen, S., Xu, M., Cao, J.(eds): Advanced Parallel Processing Technologies. Lecture Notes in Computer Science, Vol. 2834.Springer-Verlag, Berlin Heidelberg New York (2003) 655-664.
11. Waluyo, A.B., Srinivasan, B., Taniar, D.: Indexing Schemes for Multi Channel Data Broadcasting in Mobile Databases. International Journal of Wireless and Mobile Computing, 1(6) (2005).
12. Waluyo, A.B., Goh, G., Srinivasan, B., Taniar, D.: On-Building Data Broadcast System in a Wireless Environment. International Journal of Business Data Communications and Networking, 1(4) (2005) 14-36.

Real-Time Human Tracker Based Location and Motion Recognition for the Ubiquitous Smart Home*

Jonghwa Choi, Soonyong Choi, DongkyooShin**, and Dongil Shin

Department of Computer Science and Engineering Sejong University,
98 Kunja-Dong Kwangin-Gu, Seoul, Korea
{jhchoi, artjian}@gce.sejong.ac.kr, {shindk, dshin}@sejong.ac.kr

Abstract. The ubiquitous smart home is the home of the future that takes advantage of context information from the human and the home environment and provides an automatic home service for the human. Human location and motion are the most important contexts in the ubiquitous smart home. We present a real-time human tracker that predicts human location and motion for the ubiquitous smart home. We used four network cameras for real-time human tracking. This paper explains the real-time human tracker's architecture, and presents an algorithm with the details of two functions (prediction of human location and motion) in the real-time human tracker. The human location uses three kinds of background images (IMAGE1: empty room image, IMAGE2: image with furniture and home appliances in the home, IMAGE3: image with IMAGE2 and the human). The real-time human tracker decides whether the human is included with which furniture (or home appliance) through an analysis of three images, and predicts human motion using a support vector machine. A performance experiment of the human's location, which uses three images, took an average of 0.037 seconds. The SVM's feature of human's motion recognition is decided from pixel number by array line of the moving object. We evaluated each motion 1000 times. The average accuracy of all the motions was found to be 86.5%.

Keywords: Real-time Human Tracker, Smart Home, Ubiquitous Computing, Pattern Recognition.

1 Introduction

The ubiquitous smart home provides an automatic home service through analysis of the human's and the home's contexts [1]. The human location and motion that is present in this paper are the most important context in ubiquitous smart home. For example, when the human has sat on a sofa to see TV programs, the ubiquitous smart home analyzes human's preference channel and provides automatically a home service (TV). The ubiquitous smart home receives a lot of contexts from the home environment and the human, but the most important context is information about the human's location and motion [2]. We present a real-time human tracker that tracks the

* This work was supported by the Korea Research Foundation Grant (KRF-2005-013-D00052).
** Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 708 – 713, 2006.

human's location and predicts the human's motion in the home. In the ubiquitous smart home it is important to decide with which furniture (or home appliance) the human is associated. To decide this, we analyze three images (IMAGE1: empty room image, IMAGE2: image with furniture and home appliances in the home, IMAGE3: image with IMAGE2 and the human) in real-time and determine the human's location. We use the SVM (support vector machine) to predict the human's motion. The feature for classification is selected by using the pixel distribution of the human image obtained by the real-time human tracker.

2 Related Works

Human tracking for a smart space was studied by various methods. The Pfinder is a real-time system for tracking a person, which uses a multi-class statistical model of color and shape to segment a person from a background scene [3]. It finds and tracks people's head and hands under a wide range of viewing conditions. Tominaga and Hongo proposed a method for extracting human movement and hand gestures from multi-channel motion images captured in the Percept-Room [4]. The human position and the rising hand gestures are estimated by integrating the silhouettes from multiple cameras by a background image subtraction and a frame subtraction. KidRooms is a tracking system based on "closed-world regions", these are regions of space and time in which the specific context of what is in the regions is assumed to be known [5]. Guohui Li and Jun Zhang presented an effective approach to detect a moving object from a video stream based on a background template, with the integration of multiple techniques of addressing illumination changes, shadow and noisy disturbance [6]. However, some issues remain to deserve further exploitations. One is the threshold choice. Automatic or adaptive estimation of the threshold is better. The EasyLiving uses two sets of color stereo cameras for tracking multiple people during love demonstrations in a living room [7]. The stereo images are used for locating people, and the color images are used for maintaining their identities. Unlike previous studies, the real-time human tracker can decide whether a human is associated whit which furniture (or home appliance), and can predict the human's motion.

3 The Architecture of Real-Time Human Tracker

Figure 1 shows the architecture of the real-time human tracker. The real-time human tracker that is presented in this paper takes charge of the prediction of the user's location and its motion for the ubiquitous smart home.

To decide the human's location, the camera handler acquires color images (720 X 486) from the digital network camera every two seconds, and takes charge of status information of network camera and detection of transmission error and recovery. The moving object detector calculates the moving object's area using the difference value between background image (that is stored in the real-time human tracker) and acquired image (that is acquired from the network camera in real-time). The position recognizer estimates the human's location using information of furniture and home appliances in the ubiquitous smart home. The object classifier extracts feature values from the moving object's area that are analyzed by the moving object detector.

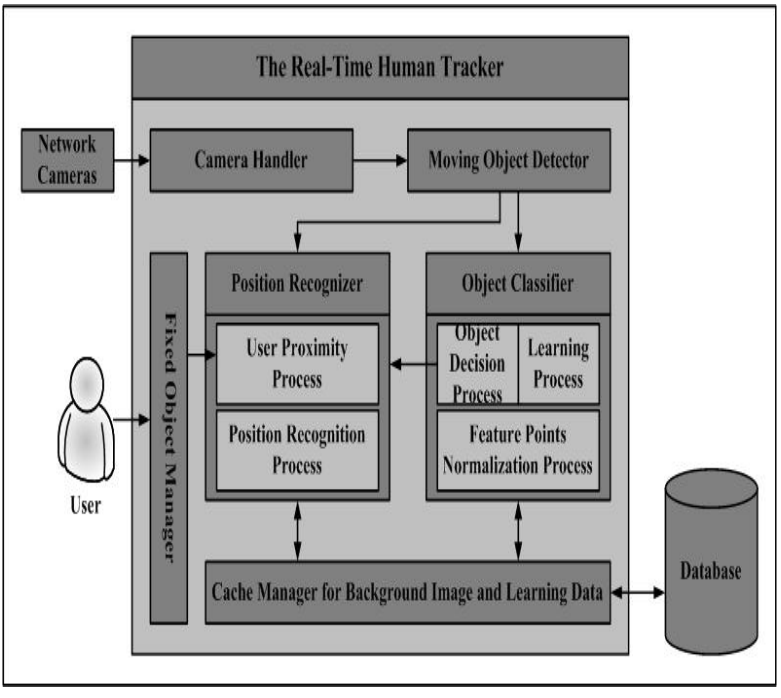


Fig. 1. Architecture of the Real-time Human Tracker

4 Internal Processes of the Real-Time Human Tracker

The real-time human tracker uses four digital network cameras to acquire a real-time image of the human. The camera system used in an experiment used a 10 base-T LAN and consisted of one server camera (32 bit RISC CPU) and three client cameras. Figure 2 shows the camera arrangement in our ubiquitous smart home. View scopes of four cameras were established to be limited by an angle to the top, bottom, right and left of 90 degrees.

We used three images to decide whether the human is included with which furniture (or home appliance). IMAGE 1 is an image which the user and furniture (and home appliances) are excluded. IMAGE 2 is an image in which furniture and home appliances are arranged in the ubiquitous smart home and IMAGE3 is an image that includes the human inside IMAGE2. The real-time human tracker detects the image difference between IMAGE1 and IMAGE2. And, it determines the location of the furniture in the ubiquitous smart home. The calculation of location coordinates (furniture, home appliance and human in ubiquitous smart home) used the silhouette method [8]. Through subtraction between IMAGE1 and IMAGE2, the human tracker acquires the edges of the furniture and home appliances in the home, and calculates absolute coordinates of furniture (or home appliance) using the pixel's array index (x, y) from the edge information. If the human enters the home, through subtraction between IMAGE2 and IMAGE3, the human tracker analyzes the pixel's array index (human image), and compares it with pixel's array index of furniture (or home appliance). If the position recognition analyzes human's location, the object classifier

recognizes human's motion using human image's pixel distribution. We used the SVM (support vector machine) for the human's motion recognition [9]. We recognize four human's motion (motion that lie down, motion that sit, motion that stand-up and motion that walk). The feature that is extracted from four images is created from the moving object detector and is used in four linear-SVMs to recognize each action. Figure 3 shows the structure of the object classifier. Figure 4 shows feature values (image's pixel distributions) that can be selected for human motion recognition and that presents the result of recognition which is analyzed by the object classifier.

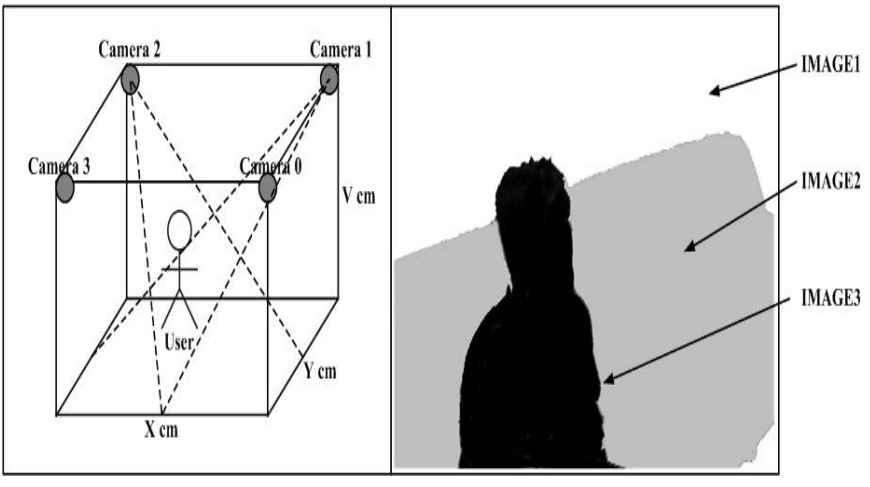


Fig. 2. Structure of Camera Arrangement

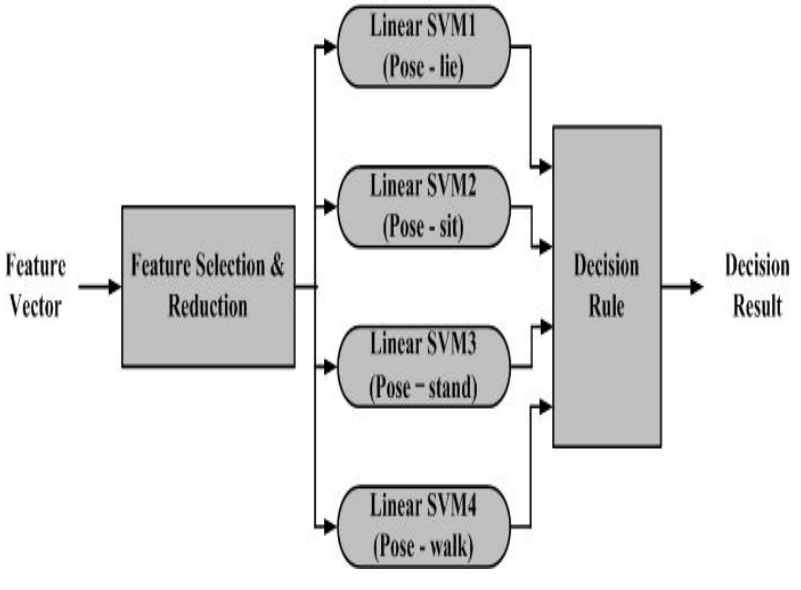


Fig. 3. Structure of Object Classifier

Number / Index	1	2	3	4	5	6	7	8	9	10
Lie1	0.013	0.019	0.021	0.032	0.037	0.040	0.044	0.053	0.056	0.066
Lie2	0.006	0.009	0.010	0.012	0.013	0.014	0.016	0.017	0.020	0.019
Lie3	0.008	0.014	0.016	0.016	0.018	0.021	0.016	0.023	0.030	0.031
Sit1	0.001	0.001	0.001	0.004	0.002	0.005	0.003	0.004	0.008	0.004
Sit2	0.013	0.016	0.016	0.017	0.027	0.035	0.038	0.051	0.041	0.065
Sit3	0.003	0.019	0.036	0.045	0.047	0.049	0.051	0.046	0.048	0.038
Stand1	0.022	0.021	0.014	0.018	0.021	0.019	0.025	0.023	0.031	0.026
Stand2	0.043	0.049	0.063	0.045	0.046	0.042	0.034	0.032	0.035	0.033
Stand3	0.028	0.026	0.031	0.032	0.037	0.037	0.030	0.040	0.041	0.038
Walk1	0.032	0.045	0.061	0.058	0.067	0.068	0.063	0.055	0.050	0.045
Walk2	0.012	0.018	0.014	0.017	0.025	0.018	0.024	0.025	0.022	0.020
Walk3	0.042	0.046	0.048	0.059	0.059	0.086	0.093	0.097	0.088	0.089

Fig. 4. Feature map for SVM

5 Experiments and Evaluation

First, the real-time human tracker decides human location. If human's location is included with the furniture (or home appliance), which is arranged in the ubiquitous smart home, the real-time human tracker recognizes the human's motion through an estimation of motion that is learned by the real-time human tracker. Table 1 shows results of performance experiment for recognition of human location and motion.

Table 1. Results of Performance experiment for recognition of human location and motion

	object location	object motion			total precision		
	time (second)	number of support vector	norm of longest vector	number of kernel evaluations	total	correct	precision rate
Lie	0.03699	171	1.75761	14204	1000	930	93.0%
Sit	0.03719	125	2.02598	13085	1000	911	91.9%.
Stand	0.03687	102	2.01333	13377	1000	821	82.1%
Walk	0.03695	98	2.14278	13335	1000	793	79.3%

In the table 1, performance experiments of human's location, which use three images, took an average of 0.037 seconds. The human's motion recognition is decided from pixel number by array line of moving object as a SVM's feature. We evaluated each motion 1000 times. And average accuracy of each motion, shown in table 1, is 86.5%.

6 Conclusions

We present a real-time human tracker that predicts human location and motion for the ubiquitous smart home. We used four network cameras for real-time human tracking. This paper explains the real-time human tracker's architecture, and presents an algorithm with the details of two functions (prediction of human location and motion) in the real-time human tracker. The real-time human tracker decides whether the human is included with which furniture (or home appliance) through an analysis of three images, and predicts human motion using a support vector machine. A performance experiment of the human's location, which uses three images, took an average of 0.037 seconds. The SVM's feature of human's motion recognition is decided from pixel number by array line of the moving object. We evaluated each motion 1000 times. The average accuracy of all the motions was found to be 86.5%. We are currently studying algorithms of shadow processing and of changed light processing in the real-time human tracker.

References

1. Das, S,K. Cook, D,J.: Guest Editorial - Smart Homes. Wireless Communications, IEEE. vol 9. Issue 6. (2002) 62 - 62
2. Jonghwa Choi. Dongkyoo Shin. Dongil Shin.: Research and implementation of the context-aware middleware for controlling home appliances. Consumer Electronics, IEEE Transactions on. vol 51. Issue 1, (2005) 301 - 306
3. C,Wren. A,Azarbayejani. T,Darrell. A,Pentland.: Pfinder: Real-time Tracking of the human Body. Pattern Analysis and Machine Intelligence, IEEE Transactions on. vol 19. Issue 7. (1997) 780 - 785
4. Tominaga M. Hongo H. Koshimizu H. Niwa Y. Yamamoto K.: Estimation of human motion from multiple cameras for gesture recognition. Pattern Recognition. (2002) 401–404
5. S,Intille. J,Davis. A,Bobick.: Real-time recognition of activity using TemporalTemplates. In Proc. Third IEEE Workshop on Application of Computer Vision. (1996) 1233-1251
6. Guohui Li. Jun Zhang. Hongwen Lin. Tu D. Maojun Zhang.: A moving object detection approach using integrated background template for smart video sensor. Instrumentation and Measurement Technology Conference, IMTC 04. Proceedings of the 21st IEEE. vol 1. (2004) 462 - 466
7. Krumm J. Harris S. Meyers B. Brumitt B. Hale M. Shafer S.: Multi-camera multi-person tracking for EasyLiving.Visual Surveillance. Proceedings, Third IEEE International Workshop on. (2000) 3-10
8. T, Matsuyama.: Cooperative Distributed Vision: Research Achievements and Future Directions. The 7th Symposium on Sensing via Image Information (SSII2001). (2001) 187-198
9. C,J,C,Burges.: A tutorial on support vector machines for patten recognition. Data Mining Knowl, Disc. (1998) 1-47

Automatic Updating of a Book Storage Database in a Ubiquitous Library Information System

Hideaki Araki, Hirohide Haga, and Shigeo Kaneda

Graduate School of Engineering, Doshisha University
1-3 Miyakodani-Tatara, Kyotanabe City, 610-0321, Japan
hede@ishss10.doshisha.ac.jp

Abstract. This article proposes the Augmented Library (AL), which is a library system augmented by ubiquitous computing (UbiComp) technology. In UbiComp, users interact with computers without being aware that they are actually using a computer. We applied the UbiComp concept to a library system. In the AL, the action of removing and returning books, which are the natural actions in a library, become the library system's input. This function is called a "real-time book database update." The system uses a combination of infrared sensors and CCD cameras to identify which book is moved from the shelf. After developing a prototype a prototype system, we conducted experiments that investigated the effectiveness and effectiveness and efficiency of AL.

Keywords: library system, real-time processing, book retrieval, CCD camera, infrared sensor, image data processing, multi thread processing.

1 Introduction

In traditional computer usage, people use a keyboard and a mouse to enter information and data, and results are displayed on a screen. In this situation, users must be aware that they are using a computer. On the other hand, in an ubicomp environment, users do not necessarily think that they are using a computer. A ubiquitous environment provides input/output services without traditional I/O devices such as a keyboard or display. In an ubicomp environment, contextual information such as locations and actions/gestures are input to the computer. The output results of any task execution are "displayed" using everyday artifacts.

Next, let us consider some of the issues conventional library information systems face. To clarifyit, let us consider the typical way in which these systems are used. When retrieving a book, users usually access a book database to get information on availability and ID data. The ID data includes each book's ID number and information about where the book is located. The user searches for the book using this information. However, users often have difficulty in finding the book they need. Even if the book is not on loan, it may sometimes be temporarily moved or incorrectly shelved. Conventional library database systems assume that all books are properly shelved. Furthermore, the system cannot detect when books are taken from the shelf. When a book is incorrectly shelved, or

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 714–723, 2006.

temporarily removed, users may find it difficult or even impossible to locate the book they need. Even when correctly shelved, a book can still be difficult to find on a shelf that contains hundreds of books. ID data does not intuitively indicate the physical location of each book. In a real library, there are many bookshelves, sometimes more than one hundred, in a large space and they are often arranged in a complicated way. Therefore, even if users have the correct bookshelf ID, they may still have trouble locating the actual bookshelf they need.

The avobe paragraph pointed out the following three issues faced by current library information systems:

1. The exact location of a book sometimes differs from the retrieval results returned by the database;
2. It is not easy to locate a book on the shelves;
3. It is difficult to find a particular bookshelf when there is a large, complex arrangement of many bookshelves.

We propose the following methods to solve these problems:

1. **Real-time updates of the book database:** Current book databases are usually updated when a library user borrows or returns a book. In our proposed system, however, additional detailed location information such as bookshelf IDs are also stored in the database. Furthermore, the action of taking a book from the bookshelf or returning it triggers a real-time update of the database.
2. **Physical visualisation:** This method visualises digital information using physical objects. In our prototype, the bookshelf itself displays retrieval results that users can easily and intuitively recognise.
3. **Guidance system:** A guidance system should be implemented to help library users find their way around the library.

Let us consider the following scenario that is an use case of our system.

Mr. H goes to the library to get a book. He doesn't know where the book is, so he decides to use the book retrieval terminal in the library. First, he places his library card on the card reader next to the book retrieval terminal, and enters the book's name. The system then shows the book information and a map showing his current location and the location of the bookshelf where the book is housed. Mr. H takes his library card and walks toward the book shelf displayed on the retrieval terminal map. On the way to the bookshelf, he finds himself lost. He finds a card reader, which is located on the side of each bookshelf, and holds his library card over it. Then, the liquid-crystal display (LCD) that is attached to the card reader indicates the direction to the correct bookshelf. By repeating this action, he eventually reaches the correct bookshelf. The LCD displays the book's name when he holds his library card over the card reader on that bookshelf. Thus, he confirms he has arrived at the correct bookshelf. When Mr. H arrives in front of the appropriate bookshelf, some coloured light emitting diodes (LEDs) that are embedded at the front edge of the bookshelf indicate the exact location of the desired book.

2 Proposal of the Augmented Library

2.1 Basic Concept of the Augmented Library

The term "Augment" means "to increase the size or value of something by adding something to it." [1] As stated in section 1, we consider one of the essential points of ubicomp to be the extension of input/output devices. By extending I/O devices, users will be able to communicate naturally with computers. They do not need to be aware that they actually using computers. AL, which is the culmination of our understanding of ubicomp, implements above-mentioned three functions: real-time book database updates, physical visualisation, and a guidance system. These three functions are used to support book retrieval by users. A schematic illustration of AL is shown in Fig.1.

AL is comprised of a book database and input/output processing subsystems. The book database contains information about each book. In addition to the conventional book information stored in typical library information systems, such as title, author, and book ID, this book database contains additional information such as the physical location information in the form of bookshelf IDs. The input subsystem updates the physical location of each book in real time. When users remove or return a book to the shelves, this action is detected by the input susbystem, and the physical location of the book database is automatically updated. The output subsystem visualises the retrieval results. Such visualisations are performed using a physical object. Our prototype system uses LEDs. The guidance system is part of the output system. We implemented a location-aware guidance system using contactless IC cards and liquid-crystal displays (LCDs).

This article describes "real-time book database updates" in detail, while the other two functions are described in detail in the references [2][3].

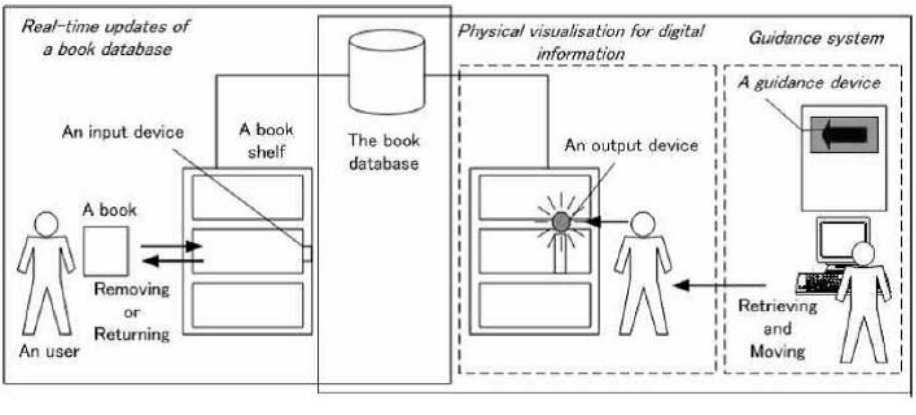


Fig. 1. Conceptual Illustration of AL

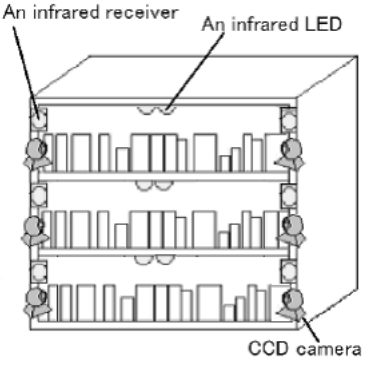


Fig. 2. Installation of Infrared sensors and CCD cameras

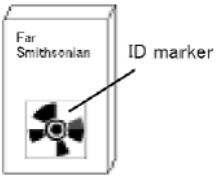


Fig. 3. ID marker on front cover page

2.2 Real-Time Updates of Book Database Subsystem

The input subsystem updates the book database in real time. To update in real time, AL has to recognise the physical location of each book at all times. Changes in the book situation must be immediately reflected in the database. A user may return a book to a different location. To track the location of each book, AL must detect each action of removing and returning a book to the shelf. In our system, infrared LEDs are installed on the top center of a bookshelf and infrared sensors and CCD cameras are installed on both vertical sides of the bookshelf, as shown in Fig.2. The infrared sensors detect when books are taken out and returned and the CCD cameras recognise the ID code of each book. Our original ID markers, which are placed on the front and back covers of each book, as shown in Fig.3, are captured by CCD cameras, and the captured image data are analysed to extract the book's ID data. Library users don't need to do anything to update the book database. Removing and returning books, which are natural actions in libraries, become the input of library system. This function can be considered as an extension of input to the system.

3 Development of a Prototype System

3.1 The Structure of the "Real-Time Updates of a Book Database" Subsystem

Fig.4 shows the structure of the "real-time updates of a book database" subsystem. The system consists of the following seven items:

1. *Infrared sensors (Infrared LEDs and Infrared receivers):* Infrared sensors detect the movement of an object on the shelf. An infrared LED is installed on top center of a bookshelf and an infrared receiver is installed on both vertical sides of the same bookshelf, as shown in Fig.2. An infrared receiver is installed on the top of the bookshelf stack to receive the infrared rays from the infrared LEDs.

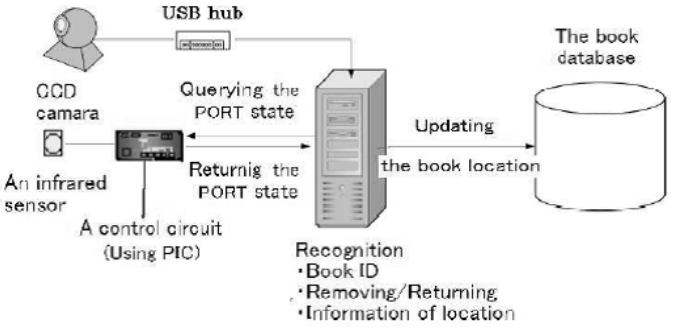


Fig. 4. The structure of the "real-time updates of a book database" subsystem

2. *Infrared sensor control circuits:* Signals from the infrared sensor are received by a control board built-up by using the one chip microprocessor PIC.
3. *CCD cameras:* CCD cameras are installed on both vertical sides of the bookshelf, as shown in Fig.2. These CCD cameras send the image data to the server.
4. *USB hub:* This is used to collect images from the CCD cameras.
5. *Camera server:* The presence of the shelf that detects the movable body is examined from the receiving optical situation sent back from a control base. If an object is moved from or to the shelf, a camera equipped on the shelf captures an image as movement occurs. By analyzing the captured image, it is possible to determine what the moving object is. If the image is a book, the book ID will be extracted from the captured image. The distinction whether it is a book; if it is a book, it can only be identified if the ID marker can be captured. After identifying the book ID, the book database is updated.
6. *he book database:* We used free DBMS MySQL to construct the book database. The table structure of the database is shown in Table.1. The table updated by the "real-time updates of a book database" system only includes the "current table" field.
7. *ID marker:* In order to identify each book, a different ID marker is used. The ID marker is put on the front and back cover of each book. An example of an ID marker is shown in fig.3.

3.2 The Method of Detecting Book Movement Using Infrared Sensor

In the "real-time updates of a book database" subsystem, it detects book movement including which book, which shelf and, exactly what action is being performed (removing/returning the book), and then updates the information on the data base. This is possible due to the continuous image data acquired from CCD cameras. However, this method consumes a lot of computer resources. Therefore, we adopted another method to detect book movement. We decided to use infrared sensors instead.

3.2.1 The Device

The device is composed of an infrared sensor and a USB board. The sensor consists of a luminescence component and an optical receiving component. The luminescence component keeps always emitting the infrared rays to the optical receiving component, and it detects the interception of infrared rays by the shield. That is, when a book is removed or returned, it is detected by the shield. The number of sensors that can be connected with PIC is 25 or less. The optical receiving situation of each sensor appears as a state of PORT of PIC. The sensor that obtains the optical receiving situation demanded by the host sends the information using the reply method through the USB board. The processing of the device only replies to the PORT state.

3.2.2 The Host

The host is composed only of a server. Data is received from the infrared sensors and CCD cameras, the book ID and the state (remove/return) are recognized, and the data base is updated. The processing procedure is shown below. First, the server recognizes the number of connected CCD cameras, and generates the marker reading thread for the same number of CCD cameras at the start of the system. All threads are put into a state of suspension immediately after starting the system. Processing by the host is a repetition of the PORT inquiry by the device. When a state change is recognized (the removal/returning of a book), the host resumes the camera thread that is paired with the relevant infrared sensor. After confirming that the thread has resumed once, the thread is put into the state of the suspended again.

3.3 The Method of Book ID Recognition Using CCD Cameras and ID Markers

The ID marker is our original design. To correctly recognise the book, two ID markers are pasted on the front and back covers of each book. When the infrared sensor detects the movement of an object on the bookshelf, the CCD camera captures an image of the object. After analyzing the captured image, if the moving object is determined to be a book, its ID information will then be extracted.

3.3.1 Structure of the ID Marker

Our original ID marker is shown in Fig.6. The ID marker is a circle because a circle is a rotation-free figure. Each ID marker is divided into four areas.

1. The first area is called the marker identification area and indicates the central location of the ID marker, which helps the image processing software to identify the ID marker.
2. The second area is the ID code area, which represents the ID code itself. The ID code is represented in a binary form.
3. The third area is the ID code starting area, which is colored red. This area represents the starting point of the ID code.
4. The fourth area is a parity check area.

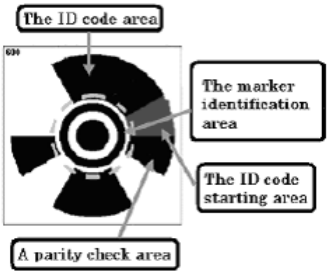


Fig. 5. Structure of ID marker

Fig. 6. Structure of vertical method marker (left) and horizontal method marker (right)

The same two ID markers are pasted on the front and back covers of each book, as shown in Fig.3. When books are removed or returned to the shelves, the front and back covers are captured by the CCD cameras installed on either side of the bookshelf.

3.3.2 Method of Extracting ID Data from ID Markers

There are two main steps in the extraction process. The first step is the recognition of the ID marker area, while the second step extracts ID data from the recognised ID marker area. ID marker area recognition is executed as follows. First, a captured image is transformed to a binary image. Then the marker identification area is searched. When this area is identified, the system tries to find the starting point of the ID code. Next, the ID marker is extracted by turning the image counterclockwise.

3.3.3 Problems Associated with the ID Marker

The number of ID areas applied to the marker shown in Fig.5 is ten bits. That means the ID code area can represent 1024 (2) different ID codes. On the other hand, the number of books housed in an actual library may be more than one million Therefore, in order to represent more than one million different codes, at least 20 bits are necessary for the ID area. We consider two different ways to increase the ID code area. The first method (vertical method) is to narrow the angle of each ID code bit, and to expand the number of partitions (Fig.6 left). The second method (horizontal method) is to represent the ID area as a double circle (Fig.6 right). The most preferable ID marker method is discussed in the experiment outlined in Chapter 5.

4 Evaluation of the Prototype System

4.1 Relationship Between the Recognition Ratio of ID Markers and Enhancing the ID Area

The recognition rate is measured in the actual experiment. Additionally, it relates to the problem of whether the number of IDs can be increased without dropping the recognition rate if the area of the ID marker is divided in the way as described in section 3.3. In this article, we consider which method can increase the number of IDs by comparing the vertical marker method with the horizontal marker method.

In the experiment, we examine whether a recognition rate of 90% or more being can be maintained at a position away from CCD cameras and by how many centimeters when the number of partitions of the code area are expanded. In this case, what is the recognition rate? It is probable that the ID will be accurately read or that the error reading will be correctly detected from the ID area after the ID marker is detected by "the marker identification area (Fig.5)".

As a result of the experiment, we established the following. Using the horizontal method marker, a recognition rate of 90% or more was not able to be maintained at any position away from CCD cameras, because the reading position of the code area shifted when the marker inclined even slightly. On the other hand, the vertical method marker was able to recognize IDs at the recognition rate shown in Fig.7.

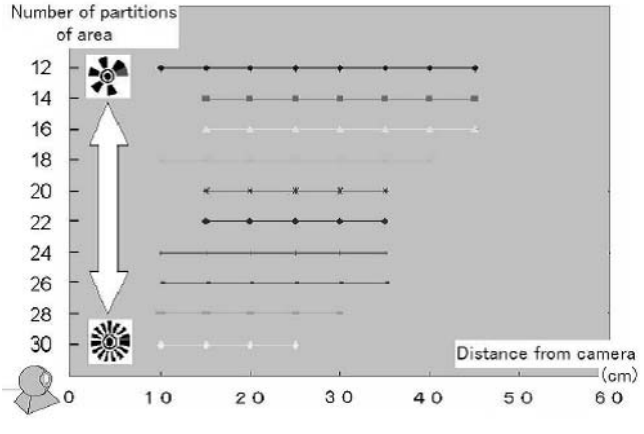


Fig. 7. ID marker recognition(It explains details by section 4.1)

4.2 Measuring the Number of Uniting Sensors That Can Be Connected Near Host PC

Uniting sensors are connected to one PC, which can starts up to four sensors from just one, and we examine the CPU use rate in such cases. In terms of removing or returning a book, it is possible to see changes in the use rates. As

a result, we can calculate the number of cameras and infrared sensors that can be operated at the same time.

The monitor results of the CPU use rate are shown Fig.8. The horizontal axis is time, and the spindle is the CPU use rate. These figures show changes in the CPU use rate from the start of the ID marker recognition process to its end, and when the book was removed and/or returned. The ID recognition thread resumes when a book is removed or returned, and it is suspended at other times.

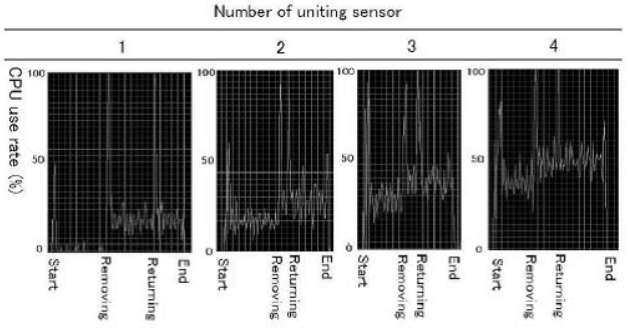


Fig. 8. CPU use rates from uniting sensors 1 (left end) to 4 (right end)

5 Discussion and Conclusion

5.1 Lessons Learned

5.1.1 Recognition Ratio of the ID Markers

When the vertical method marker was compared with the horizontal one, it was clear that the vertical one performed far better. Thus, we should use the vertical one in our system. Fig.7 illustrates that recognition up to 35cm is possible even when using an ID marker that has 26 areas. This one has about 67 million IDs. Although there are limits when the width of the shelf is 70cm or less, this method is enough to cover the number of books housed by many libraries.

5.1.2 Performance Evaluation of the Camera Server

The CPU use rate temporarily reaches 100% when the ID marker is recognized, but the CPU use rate is suppressed at the other time. Therefore, the CPU offers the same recognition degree when two or more threads start, or when a single thread starts. But CPU use rates increase as the number of uniting sensors increases. About 10 can be operated with one server because it changes the CPU use rate by almost 10%.

5.2 Conclusion and Future Works

This article proposed the Augmented Library, which applies an ubicomp concept to library information systems. The system includes real-time database updates,

physical visualisation, and a guidance system. The real-time book database updates enable users to efficiently get information about books. By using physical visualisation, users can find the book they need more quickly and easily. The guidance system helps users to find the bookshelf where their desired book is stored.

Our prototype system uses CCD cameras, infrared sensors, and original ID markers for real-time book database updates. After developing the prototype system, we conducted some experimental evaluations. These experiments investigated the effectiveness and efficiency of the prototype system. However, they also revealed its limitations. One major problem is the recognition ratio of the book IDs. Our vision-based recognition method exhibited limitations in its ability to achieve a good recognition ratio. This was strongly influenced by external environmental conditions. Improving the recognition ratio is our next task in improving the practical usage of AL in real environments.

Acknowledgement. The authors would like to express their sincere thanks to Mr. Makoto Yoshida and Mr. Akira Takahashi, who were former graduate students of Doshisha University, for their efforts of developing the prototype system.

References

1. Collins COBUILD Advanced Learner's English Dictionary, 4th Edition, Harper Collins Publisher, Glasgow, 2003
2. Akira Takahashi: Extension of the TAC paradigm and the description of the Augmented Library by the Extended TAC paradigm, Master's thesis, Doshisha University Graduate School of Engineering, 2006 (in Japanese)
3. Hiromasa Sakashita et al: Physical visualization for location information and guide to bookshelf with a mobile phone in Augmented Library, Proceedings of the 67th Annual Conference of the Information Processing Society of Japan, 3V-2, 2005 (in Japanese)
4. Diego Lopez de Ipina, Paulo Mendonca, and Andy Hopper: TRIP: a Low-Cost vision-Based Location System for Ubiquitous Computing, Personal and Ubiquitous Computing, Vol.6, No.3, pp.206-219, 2002
5. Norihiko Kato and Katashi Nagao: BookSpider - Integration of Information Space and Physical Space in Library information, Proceedings of the 65th Annual Conference of the Information Processing Society of Japan, 5ZA-2, 2003 (in Japanese)

Context-Aware Dynamic Personalised Service Re-composition in a Pervasive Service Environment

Yuping Yang[1], Fiona Mahon[2], M. Howard Williams[1], and Tom Pfeifer[2]

[1] School of Maths and Computer Sciences, Heriot-Watt University, Riccarton, Edinburgh, EH14 4AS, UK
yang_yuping@hotmail.com, mhw@macs.hw.ac.uk
[2] Telecommunications Software & Systems Group, Waterford Institute of Technology, Waterford, Ireland
{fmahon, tpfeifer}@tssg.org

Abstract. A pervasive environment needs to take account of a user's context and preferences in determining which services to provide to the user. Moreover, one of the important features of a pervasive service environment is its dynamic nature, with the ability to adapt services as the context of a user changes, e.g. as he/she moves around. This paper describes how these requirement changes can be sufficiently accounted for by using a personalisation component to 'decide' what a user needs, and a composition component to continuously monitor services and the changes associated with them. The paper presents how services can be recomposed dynamically if the changes in context require it. This approach has been incorporated into a platform to support pervasive services. The service composition process used is described, and the way in which personalisation is incorporated into this process is shown. Finally the paper provides a brief account of two prototypes built as a proof of concept for these ideas.

1 Introduction

In developing solutions to handle the requirements of mobile users, the environment is becoming increasingly complex. The range of different services available to the user is growing rapidly. So too is the number of devices that can be used to access different kinds of services, and the different networks that can be used to communicate between users and applications. The user is therefore presented with a myriad of choices - too many to make the use of the environment a pleasant one, without some intervention.

The goal of pervasive computing is to provide an intelligent environment to support the user in this increasingly perplexing task. Where possible, such an environment should help to select the most appropriate services for any particular user, and to employ the most relevant devices and supporting networks.

The situation becomes more complex when the mobility of the user is accounted for. As the user moves around the environment, the services, devices and/or networks that he/she is using may cease to be available, or new services, devices and/or networks may become available that the user would prefer to use. In fact, location is

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 724 – 735, 2006.

one of many attributes defining the user's current context and other context attributes including time or current activity can also affect the choice of service, device or network. Thus one of the most important functions of a pervasive environment is the dynamic discovery, selection and composition of services in a planned order to satisfy a request from a client [1, 2]. Service composition should be context aware and personalisable to meet a user's needs and preferences, and their reselection and recomposition if circumstances change [3, 4].

This paper presents a solution to the problems described above. It concerns itself with how personalisation is involved in service composition and re-composition, and how personalisation adapts services to user context and requirements in this dynamic process. The approach is based on the use of rules and policies for making the composition decisions [5, 6]. On the other hand personalisation itself is developing rapidly. With the growing interest in context awareness and pervasive systems, personalisation functionality has evolved to take account of different aspects of the users' context, in an environment populated by many small, networked devices that can sense users' situation anytime, anyplace [7-9].

The research has been carried out as part of Daidalos, a large European research project that is developing a pervasive system for a mobile environment [10]. The goal of the research reported in this paper was to develop the approach required for a general system to handle personalised service composition and re-composition dynamically depending on the preferences and changing context of the user. Based on this a sequence of prototypes with increasing functionality have been created and used to demonstrate the suitability of this approach. The next section gives an overview of efforts made in the areas of Personalisation and Composition. Section 3 describes the platform, to which the components described in the paper belong. Sections 4 and 5 detail the Personalisation and Composition components and how they work together to achieve dynamic service re-composition. Section 6 details the implementation of the prototypes built as a proof of concept for the ideas described in this paper. Overall, the paper shows how important personalisation is to the whole process of service management in a pervasive service environment.

2 Related Work

Automatic composition of services is an important aspect of pervasive environments (e.g. GAIA [11], AURA [12]). However, much of the work on composition to date is based on a static composition of services, where requirements of the composition do not change and the state of the constituent services remain the same.

Chakraborty [13] describes a static composition where a particular composition is attempted and achieved. However, there is no accounting for continuous change of the component services during the lifetime of the composed service. Tosik [14] describes a service composition with some level of management. This management focuses primarily on usage privileges and Quality of Service (QoS). Since Tosik deals with web services, which are inherently static, he does not expand into an area of composition management of services with volatile availability.

Personalisation adds further challenges to the problem. Casati [15] in his paper on 'eFlow' describes a dynamic service composition, which includes personalisation. The personalisation involved assumes user input of their requirements. The composed services are not technically dependent on each other, although they do complement one another. For example, in Casati's service composition, a service to organise a charity ball would be composed of services to book the banquet, order the invitations from the printers, and start the advertising campaign. Although the paper describes 'dynamic composition' it focuses more on the ability to change the definition of the composition, and so the dynamic composition is over many compositions over a period of time, and not within a single composition. The 'dynamicity' described does not include monitoring of the composition during the lifetime of the composite service, although the composition definition in 'eFlow' can be changed during the lifetime of a composite service. Real dynamic re-composition based on continually changing user requirements is not addressed.

Sheng et al. [16] define a personalised composite service specification architecture, based on which users can specify their needs by adjusting existing process templates. However, this approach requires a user to locate process templates and annotate them with contextual information. Thus, quite a few user interactions are involved when orchestrating a composite service, which is not very realistic in a pervasive environment where a large number of service compositions may occur dynamically.

The Tivoli Personalized Service Manager [17] developed by IBM provides an integrated infrastructure of software products for Internet service provisioning. It offers the ability to generate web pages for specific devices, allows users to personalise portal home pages, and provides services with functionality such as calendar, agenda and address book which can be used by ISPs (Internet Service Providers) to develop their own additional services. Its localization feature provides the capability to translate into different languages. However, it only takes into account user's profile and preferences but not more general context information such as time or current activity. In addition, it only allows ISPs to utilize the simple services supplied by the product itself and does not provide the essential function of composing any existing services/resources provided by other parties.

The SPE (Secure Persona Exchange) framework described by Brar and Kay [18] provides personalized services to users in ubiquitous computing environments based on user preferences stored on mobile devices. Like [17], it does not take account of dynamic contextual data while achieving personalization.

In summary, although a lot of work has been done independently in the areas of service composition, and in personalisation, there is little work done in exploiting the capabilities of one for the benefit of the other. Real dynamism seems to be missing from the composition solutions proposed to date. Dynamism is essential for a mobile environment, since the mobile environment itself is never static.

3 The Daidalos Pervasive Service Platform

The Personalisation and Composition components described in this paper, are part of the Daidalos overall platform architecture. The Pervasive Service Platform (PSP) is

the platform that adds pervasiveness to all services deployed in the Daidalos network. The architecture of the PSP comprises six main components [19], as follows:

Context Manager (CM): The task of the CM is to collect, process and provide context information. Context is the set of information that describes an identity's preferences, profiles and current situation.
Rule Manager (RM): The RM focuses on the management and processing of rules. A rule describes a set of events that have to occur and conditions that have to be met in order for some actions to be triggered. Users employ rules as part of personalisation while services use them to be notified about changes in the user's environment.
Event Manager (EM): The EM collects and distributes events. Events occur when context changes or when they are created by rules. Typically, events are used by the Rule Manager or by services that want to be notified about changes.
Personalisation (P): This tailors the services to the user's preferences. Personalisation takes into account preferences that the user has stated explicitly and also infers preferences to make services more personalised.
Pervasive Service Manager (PSM): This is responsible for service discovery, selection and composition. It is used to find available services, select them based on the user's context and compose them to a service session that can fulfil the user's task. It also continuously monitors composed services during their lifetime based on service availability and contextual relevance to the user.
Security and Privacy Manager (SPM): The SPM hides the user's real identity from untrustworthy parties by using virtual identities. It also provides access control to user information (such as context) and the use of services.

All of these components interact and co-operate to form the PSP. The combination of their functions creates a unique pervasive service environment, and provides a fully functional, integrated solution for context-aware, personalised, rule-based and event-driven service discovery and composition. However, since the main topic of this paper is how the Pervasive Service Manager and Personalisation work together in performing service composition, we will not be considering the functionality of the other components in any detail.

4 The Role of Personalisation in Dynamic Service Management

Service Composition involves combining a set of services together, to give a complete service offering. As discussed in section 2, this is not a novel idea. However, the solution outlined in this paper goes one step further by introducing 'dynamic service composition'. This means that even after the service has been composed, its constituent services continue to be monitored based on the applicability to the overall composition. This is done with the assistance of Personalisation [20].

4.1 Personalised Service Selection

Personalisation is first used at the initial stages of composition. A composed service is defined by its 'Service Model'. The Service Model defines a set of service types that

are required to make a composed service. Some services are compulsory for the functioning of the composed service and some are optional. The Composer in PSM attempts to retrieve services of the service types defined in the Service Model using Service Discovery.

When a request for a service is passed to the Service Discovery component, it will in general return more than one candidate service, each of which is able to fulfil the task required by a user. Due to the diversity of services, expecting the user to determine the most appropriate one among them would be unrealistic and time consuming for the user. Thus, the system needs to make a decision to select one for the user. This occurs during the process of service selection. In general for a composite service $S_c = S_0 + \ldots + S_t$, a list of services $S = \{S_0,\ldots,S_n\}$ may be discovered, any of which could be used as a component service S_k of S_c. Given the user preferences and context, a personalised selection is performed that best suits the user's specific goal. Selection criteria used in our system include:

- User specified criteria: Users may have specific requirements on the cost, speed, QoS, location, mobility, etc. of a service. These may depend on where the user is located, what devices/networks are available, what mode the user is operating in, and so on. These requirements provide guidelines for finding an adequate match. The list of discovered services are ranked according to the criteria (e.g., rank the services from the lowest price to the highest one) and the highest ranked service is chosen as a component service (e.g., the cheapest service). Moreover, a user might have an explicit preference for a specific service (e.g., a specific provider's wireless network). In this case, if the specific service is available, it can be simply selected as a component service.
- System criteria: These criteria are used to improve the performance of a composite service. For example, selecting component services that are located close to each other would reduce the amount of data transferred and reduce communication time among component services.

As described further in section 6, a limited set of simple criteria are used to choose services in the current prototype. The situation can become complicated when several criteria need to be combined together. How to make a rational compromise among conflicting criteria (e.g., the fastest service may not have the best QoS) is an issue that needs to be resolved in the next phase of this work.

4.2 Personalised Service Parameterisation

Once the service is selected, the next interaction with Personalisation is to personalise the service itself to the user requirements. This involves adapting the service itself in some way to suit the user e.g. large font for visually impaired users.

In order to personalise services selected for a composition, the appropriate attributes need to be passed to each service as parameters. There are two categories of service parameters:

- Operating Parameters: These are used by a service to control its functioning as well as running process implicitly. They are used to describe inherent properties of a service and thus their values cannot be modified by others.
- Personalisable Parameters: In order to cater for different users, a service may allow some parameter values to be configured according to user related aspects. These parameters are called personalisable parameters and are used to characterise a service and improve its performance. Compared with operating parameters, personalisable parameters can be configured with new values.

In order to distinguish between operating parameters and personalisable parameters, a service parameter needs to have a property indicating whether or not it is personalisable. Each parameter p_k has the form

$$p_k = (pname, pvalue, pcategory)$$

where *pname* and *pvalue* are the name and value of a parameter respectively, and *pcategory* indicates whether or not the parameter is personalisable.

A service that allows itself to be personalised, needs to provide interfaces for personalisation. In order to know what features of the service can be customized, a standardized interface is required which tells the personalisable parameters of this service. To determine the parameter values, user context and his/her preferences related to this service are analysed and appropriate values are decided accordingly (e.g., the QoS of the WLAN network service is set high when the user is watching an important football game). A parameter may have a default value, which applies to the situation, for example, the user has no special requirements. It can be overridden by the specific value derived from user preferences/context. An interface which allows setting the parameter values is also needed from the service.

Service parameters, including their names, meanings and types, may vary with services. Due to the variety of services and their degree of dependence on each other, it would be very difficult to interpret service parameters without a standard definition. Thus, for each service type we need a list of general parameters that have common definitions and shared meanings across all services with this type. This parameter list acts as an ontology to be referred to by services that want their parameters to be personalised.

Personalised service parameterisation can be static or dynamic depending on when it takes place. When the personalisable parameters of a service are configured at its starting point, this is referred to as static parameterisation. Due to the change of user context or preferences during the service execution time, some service parameters may need to be adjusted to suit the user requirements dynamically. Dynamic parameterisation usually happens in relatively unstable environments (e.g., the resolution of an image may be lowered if it is transferred from a user's computer to his/her mobile phone).

4.3 Role of Personalisation in Dynamic Re-composition

Personalisation continues to be paramount to the dynamic capabilities of the Composer once the service is fully composed. The applicability of a service to the

composition might change for reasons such as: the service is no longer available, a service with a higher preference becomes available, QoS available on a service changes, or changes to the service itself (e.g. cost changes based on time). When any of these guards are triggered, then a re-composition will occur transparently to the user.

The PSM, which contains the composition component, uses the Personalisation component to determine the user's preferred service. Personalisation (using the RM and EM) will trigger events to inform PSM when services of higher preference become available. Service preferences can be based on such attributes as price and QoS, and so when any of these attributes change, Personalisation will inform PSM.

In addition, even if these service attribute values remained the same, a user's preference for a service may change. In a mobile world, the user preferences are context sensitive and so change as the user's context changes. These preference changes are passed to PSM to allow a service re-composition when Personalisation deems a user requirement for a currently running composed service to have changed.

5 Service Composition in Daidalos

Service Composition is part of the Pervasive Service Management (PSM) of the PSP. This PSM consists of five major components that cooperate to provide composite services. These components are: the Priority Processor, Service Discovery, Service Selection, the Service Composition Manager and the Service Actuator. The interactions of these components can be seen in Figure 1 and are described below.

1. The Service Composition Manager (SCM) in the PSM is responsible for performing reasoning on the selected composite service. It obtains the detailed service information from the composite service description, which includes the specific requirements for component services.
2. The SCM calls Personalisation to adapt the composition process of the selected composite service.
3. Personalisation refers to the Context Manager and the Rule Manager for user context and preference rules respectively. The composite service is adapted (e.g., adding/removing a component service, setting appropriate starting time for each component service or changing the order of component services) according to the user context/preferences.
4. The SCM then issues the requests for component services to the Service Discovery (SD) component in the PSM.
5. The SD searches the appropriate service directories for all possible services that could be used to meet the user request and returns a set of possible candidate services.
6. The candidate services are then evaluated and ranked in order based on non-functional preferences. Personalized Selection determines which services are most appropriate to be composed together to make a composite service that best meets the user's preferences.

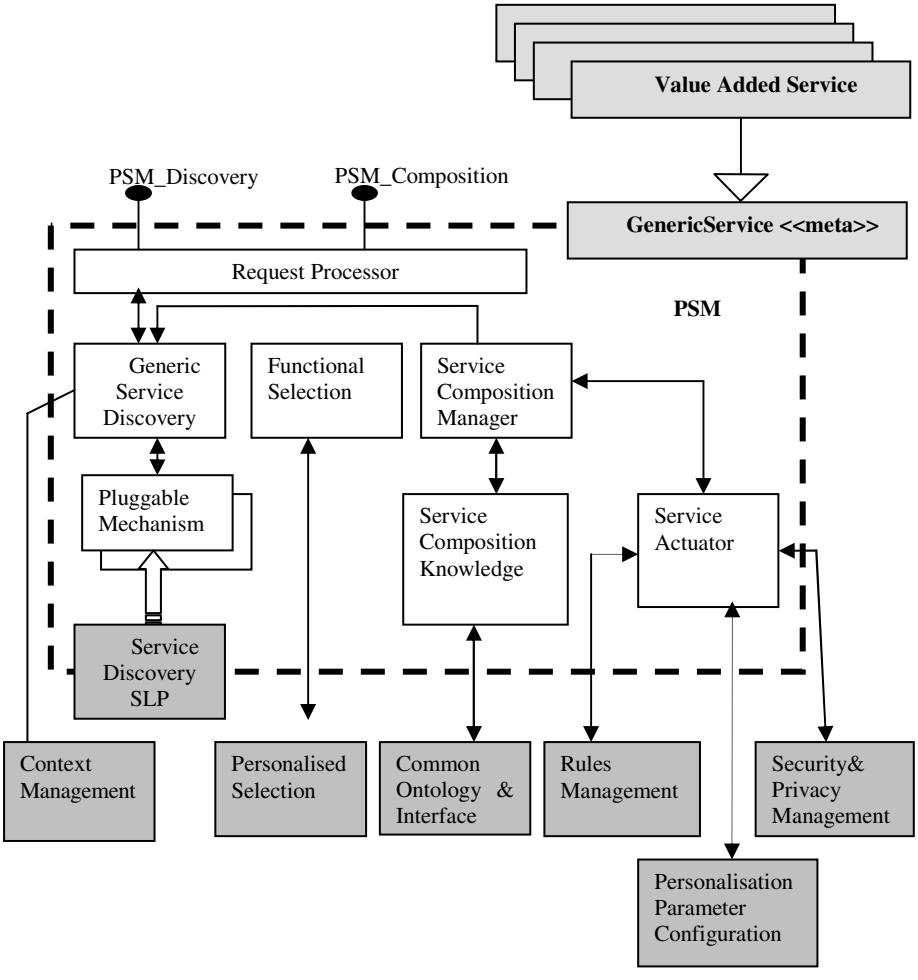


Fig. 1. Pervasive Service Management

7. In order to decide the most appropriate services from the ones available, Personalisation refers to the Context Manager and the Rule Manager for user context and preference rules respectively. Service selection is carried out based on the user context/preferences.

8. The chosen service is fed to the Service Composition Manager, which retrieves the "service model" for the service. This service model describes the template for composing the various contributing services of the composite service.

9. The Service Composition Manager looks for component services that match the functional and technical criteria (specified in the service model) of the composite service. This process is repeated until suitable component services have been found.

10. The list of component services making up the composite service then needs to be personalised. This is achieved through adjusting the parameters to these services.

The Service Actuator in the PSM calls Personalised Parameterization to configure personalisable parameters of the services.

11. Personalisation refers to the Context Manager and the Rule Manager for user context and preference rules respectively in order to determine appropriate values for the parameters. Service parameters are configured accordingly.

12. Finally, the Service Actuator instantiates service instances if necessary, and returns a handle to the composite service, for use by the service consumer. The Service Actuator is the main component responsible for monitoring constituent services of the composed service during its lifetime. It is the component that adds the *real dynamicity* to the service composition process. A composite service is never in a stable state i.e. it is constantly open to service adaptation. The Service Actuator monitors the service through its lifetime. If specific context guard conditions are violated, the service actuator may request a re-composition of the service, giving an indication of the violating service component.

6 Implementation and Testing

The system has been implemented and demonstrated at two public meetings so far. Three separate scenarios have been used to drive the development and to demonstrate the resulting system. The first two are relatively small but demonstrate different aspects of the problems of pervasiveness. The final scenario used to demonstrate the feasibility of the PSP platform is more complex and involves a number of different events and actions. The basic flow of the relevant parts of this scenario is as follows:

Bart is at home watching his newscast on his home monitor. A VoIP call comes in for him from his boss. Based on his current preferences the call is directed to his PDA. The newscast is automatically paused while he is talking to his boss. His boss asks him to go to the airport to pick up a customer. When Bart hangs up the call, the newscast continues where it left off. Bart walks to his car. The newscast is automatically transferred to his car PC. When Bart starts driving, the newscast goes into sound only mode. It runs to the end and finishes. Additionally another service on the car is the Traffic Information Service that takes traffic information off the dvb-t network that services the area. As he reaches the airport, the Information Service is recomposed to become an Airport Information service, and now takes information off the Airport WLAN. When Bart sees the customer's plane has landed, he gets out of the car causing the Airport Information Service to transfer to his PDA. Bart goes to meet the customer.

The set of applications and their component services required for this scenario were a Newscast Service, a VoIP service and an Information Service including Traffic and Airport specific services. These were implemented following the requirements of the PSP.

In the scenario, there are examples of context triggers that cause certain behaviour within the system. This was implemented using rules that were triggered when certain contexts occurred, e.g. *Newscast* will pause when Bart's context is 'BUSY'. This context value can be set by any other application, in this case the *VoIP* application.

An example of Personalisation is evident with the *Information Service*, which will display information based on the known native language of the user. In the

demonstrated scenario, this was limited to German or English. To use this feature of Personalisation the application developers simply need to provide a method which allows Personalisation to ask what parameters need to be personalised, 'language' in this case. They also need to provide a method to allow Personalisation to tell the application what the value of the parameter should be, in this case either German or English.

PSM's control of Service Discovery and Composition is also evident in the scenario. The most obvious example of composition is when the Information Service recomposes from a Traffic Information Service to an Airport Service. The service itself is required to be composed of services of types 'InformationGUI' and 'TransportInformationService'. On the car, PSM composes the BMWInformationGUI with the TrafficInformationService. On reaching the airport, a re-composition occurs based on a context trigger. PSM listens for events from the rules triggered and will recompose accordingly, swapping the TrafficInformationService, that was listening to data coming off the dvb-t, for the AirportInformationService that listens to data coming off the airport WLAN. PSM also controls the service transfer, which is seen in two places in the scenario - when Bart leaves the house for the car, the newscast follows him to the car, and when the Airport Information Service follows him from the car to the PDA. Again, this is possible by having the applications provide some simple interface methods that the PSP can use to make the service pervasive.

This particular scenario was demonstrated on the current prototype at a demonstration in December 2005. In addition to the PSP, the applications and the PSP itself were integrated with lower layer network components, also being developed inside Daidalos. The context triggers were simulated; however, the rest of the demonstration was real. The test site included a BMW with a display monitor, a dvb-t network for the traffic information and several lower level components responsible for such things as QoS, billing, authentication and network handover. This prototype has shown how services can be made pervasive in a simple way by the PSP platform. Additionally, this has been shown on a real network with all the issues associated with a network accounted for. These were real services giving real value in a pervasive manner.

In this section the progression of Daidalos has been illustrated, in particular the progression of the Context, Personalisation and PSM components. Other aspects of the platform such as the Privacy and Security component have been ignored as outside the scope of this paper. It has been shown that it is not only possible to create a pervasive environment of services using the PSP, but also that the PSP makes this task an amenable option for Service Providers.

7 Conclusion

This paper has described a dynamic method of service composition that uses the capabilities of personalisation to achieve its requirements in pervasive environments. It argues that this goes one step further than the static service composition that is described in many research papers. Continuous monitoring of the user and their environment enhances the user's experience of services and networks by truly minimising the need to interact with them. The implementation of a prototype has shown that the

ideas set forth in this paper are indeed feasible. Although the scenarios are merely a slice of the potential of a pervasive network, they have given us a peek at its potential.

However, there are a number of challenging problems remaining. In the second phase of Daidalos we will be addressing these. In particular, we will be looking at monitoring user behaviour and inferring user preferences. It is essential that the system itself should monitor the user and infer user preferences as far as it is able in order to build up user preferences without requiring too much of the user. Secondly, we will be addressing the handling of security and privacy in relation to dynamic personalised service composition and re-composition. Thirdly, we will be incorporating full context awareness. In the first phase context information and triggers, such as location information or location changes, were simulated to verify the ideas but in the second phase we will move to a fully integrated system in which such information and triggers are provided by the underlying infrastructure.

Other interesting issues raised by the research and which need to be addressed include the development of appropriate context-aware user preferences. The first (and simplest scenario) illustrated several of these. How does the system know that when the professor approaches it should switch back to using the PDA? How do we know that the professor's proximity is relevant – he/she may be a few metres away but there is a wall between them? How does the student get information on the professor's location without violating the professor's privacy? And so on.

These ideas will continue to be explored and further developed and prototyped over the next three years as the project progresses in its second phase. Due to the integrated nature of the whole project, these developments will add value to the overall results produced. For example, new more innovative service discovery technologies are to be identified, new methods of inferring preferences and predicting required behaviour are to be investigated. In this phase, we have combined the steam and the pistons to provide forward motion. The next phase will focus on finding out how to put more power in the engine itself.

References

1. Huang, A.-C., Steenkiste, P.: Building Self-configuring Services Using Service-Specific Knowledge. The Thirteenth IEEE International Symposium on High-Performance Distributed Computing, Honolulu, Hawaii USA, June 4-6. (2004)
2. Hirschfeld, R., Kawamura, K.: Dynamic Service Adaptation. The 4th International Workshop on Distributed Auto-adaptive and Reconfigurable Systems, Tokyo, Japan, IEEE Computer Society, March 23-26. (2004)
3. Huang, C., Garlan, D., Schmerl, B., Steenkiste, P.: An Architecture for Coordinating Multiple Self-Management Systems. Proceedings of the 4th Working IEEE/IFIP Conference on Software Architecture (WICSA-4), Oslo, Norway, June 12-15. (2004)
4. Davy, A.: Task Driven Service Composition for Pervasive Computing Environments. M-Zones White Paper, http://www.m-zones.org. (2004)
5. Filman, R. E., Friedman, D. P.: Aspect-Oriented Programming is Quantification and Obliviousness. Proceedings of the ECOOP 2001 Workshop on Advanced Separation of Concerns, Budapest, June 17-18. (2001)

6. Araniti, G., De Meo, P., Iera, A., Ursino, D.: Adaptively Controlling the QoS of Multimedia Wireless Applications through User Profiling Techniques, IEEE Journal on Selected Areas in Communications, Vol. 21(10). (2003) 1546-1556
7. Lewis, D., O'Donnell, T., Feeney, K., Brady, A., Wade, V.: Managing User-Centric Adaptive Services for Pervasive Computing. Int. Conf. on Automatic Computing (ICAC'04), New York, May 17-18 (2004) 248-255
8. Wagner, M., Balke, W.-T., Hirschfeld, R., Kellerer, W.: A Roadmap to Advanced Personalization of Mobile Services. In Proceedings of the 10th Int. Conf. on Cooperative Information Systems (CoopIS) Industry Program 2002, Irvine, CA, USA, October 30 – November 1 (2002)
9. Maamar, Z., Mostefaoui, S. K., Mahmoud, Q. H.: Context for Personalized Web Services. Proceedings of the 38th Annual Int. Conf. on System Sciences (HICSS'05), Big Island, Hawaii, January 3-6 (2005)
10. Daidalos. Daidalos EU Framework Programme 6 Integrated Project, http://www.ist-daidalos.org. (2005)
11. Roman, M., Hess, C., Cerqueira, R., Ranganathan, A., Campbell, R., Nahrstedt, K.:A Middleware Infrastructure for Active Spaces. IEEE Pervasive Computing, 1(4). (2002) 74-83
12. Garlan, D., Siewiorek, D., Smailagic, A., Steenkiste, P.:Project Aura: Towards Distraction-Free Pervasive Computing. IEEE Pervasive Computing, 1(2), (2002) 22-31
13. Chakraborty, D., Yesha, Y., Joshi, A.: A Distributed Service Composition Protocol for Pervasive Environments, WCNC 2004 - IEEE Wireless Communications and Networking Conference, 5(1), March 2004, 2579-2584
14. Tosik, V., Pagurek, B., Esfandiari, B., Patel, K.: Management of compositions of E- and M- business web services with multiple classes of service, NOMS 2002 – IEEE/IFIP Network Operations and Management Symposium, vol. 8, no. 1, April 2002, 935-938
15. Casati, F., Ilnicki, S., Jin, L. et al: Adaptive and Dynamic Service Composition in eFlow. HP Labs 2000 Technical Reports, www.hpl.hp.com/techreports/2000/
16. Sheng, Q. Z., Benatallah, B., et al: Enabling Personalized Composition and Adaptive Provisioning of Web Services. The 16th International Conference on Advanced Information Systems Engineering (CAiSE), Riga, Latvia, June 7-11. (2004)
17. IBM. Tivoli Personalized Services Manager, Version 1.2. ftp://ftp.software.ibm.com/ software/pervasive/info/tech/tpsm_ss.pdf, 2002.
18. Brar, A., Kay, J.: Privacy and Security in Ubiquitous Personalized Applications. User Modelling Workshop on Privacy-Enhanced Personalization, Edinburgh, UK, 25 July. (2005)
19. Farshchian, B., Zoric, J.,Mehrmann, L., Cawsey, A., Williams, H., Robertson, P., Hauser, C.: Developing Pervasive Services for Future Telecommunication Networks. Proceedings of WWW/Internet 2004, Madrid, Spain, October 6-9. (2004) 977-982
20. Williams, M. H., Yang, Y., Taylor, N., McBurney, S., Papadopoulou, E., Mahon, F., Crotty, M.: Personalized Dynamic Composition of Services and Resources in a Wireless Pervasive Computing Environment. Proceedings of First International Symposium on Wireless Pervasive Computing, Phuket (2006) 377-382

A Context-Aware Multi-agent Service System for Assistive Home Applications*

Yong Kim, Yoonsik Uhm, Zion Hwang, Minsoo Lee, Gwanyeon Kim, Ohyoung Song, and Sehyun Park**

School of Electrical and Electronics Engineering, Chung-Ang University, 221, Heukseok-dong, Dongjak-gu, Seoul 156-756, Korea
{ykim, neocharisma, zhwang, lemins, cityhero}@wm.cau.ac.kr, {song, shpark}@cau.ac.kr

Abstract. In this paper, we present an Ontology-based Context-aware multi-Agent Service System(OCASS) architecture to provide these context-aware services in a smart home. To model various contexts, we design Ontology which supports to share context knowledge, detect and resolve inconsistency of the knowledge. In addition, we classify context-aware services into three layers - Session, Task, Subtask - to make sure definition of service conflict problems and to solve the problems easily. With our context model and the classification of service conflicts, our system supports autonomic tasks including recognizing and learning user's formal/informal activity pattern, and resolving conflicts between services in different situations of users invisibly.

Keywords: context-aware system, multi-agent architectures, Ontology, ubiquitous computing, pervasive computing, smart home.

1 Introduction

A lot of work has been done in trying to make applications in ubiquitous computing environments context-aware so that they can adapt to different situations and be more receptive to users' needs [1] [2] [3]. A number of context-aware systems, which provide users with relevant services and information based their situational conditions[1], have been developed to demonstrate the usefulness of various contexts. However, most of the systems for an interaction between context-aware services categorize them in respect of not their functions nor goals but controlling devices, so they have not been adequately considered conflict problems between services. Also, most of studies have not totally been considered the recognition of user's informal activity pattern, and the context pattern learning, including a solution of service conflict problems in respect of intelligence in a smart

* This research was supported by the MIC(Ministry of Information and Communication), Korea, under the Chung-Ang University HNRC(Home Network Research Center)-ITRC support program supervised by the IITA(Institute of Information Technology Assessment).

** The corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 736–745, 2006.

home. Moreover, context-aware services have not been widely available to everyday users and developing such systems is still a complex and time-consuming task.

A context-aware system is required to perform the following tasks: 1) to sense, reason and mine about the various contexts including user's formal/informal activity pattern, 2) to share the contextual information between agents or systems that are located in an open, dynamic, and distributed environment through providing the semantics of the context, 3) to manage service units to solve conflict problems between services when the system provides users with relevant multi-services by making a decision which actuators are in accordance with the current context and by scheduling a collaboration of actuators, 4)to acquire additional contexts on demand whenever to require more contexts than the first contexts.

In this paper, we propose an Ontology-based Context-aware multi-Agent Service System(OCASS) architecture that performs intelligent context-aware services to satisfy the requirements. To evaluate the effectiveness of our architecture, we also created a testbed and comprehensive scenarios that resolve conflicts between services in different situations. The rest of this paper is organized as follows. Section 2 gives related works about context-aware systems. Section 3 presents definition of conflict problems between services in a smart space. In section 4, we describe the OCASS architecture with our OWL based ontology models and the interaction of its agents. Section 5 presents our implementation testbed. Finally, we state our future work and conclusion in section 6.

2 Related Works

A number of context-aware systems have been developed to demonstrate the usefulness of context-aware technology. Recent research work of context-aware systems has focused on providing infrastructure support for context aware systems. In the Context Fabric [2] infrastructure, Hong et al. took a database-oriented approach to provide context abstraction by defining a Context Specification Language; and a set of core services. However, the design of a proprietary context specification language may lead to the lack of a common model. In the CoBrA [4] project, Chen et al. proposed an agent-oriented infrastructure for context representation, sharing knowledge and user's privacy control. They developed common ontology for developing context expression ontology to independent system and common policy language, shared context model to providing all kind of devices, services, and agents. Service-Oriented Context-Aware Middleware (SOCAM) architecture [5] provides efficient support for acquiring, discovering, interpreting and accessing various contexts to build context-aware services. SOCAM proposes a formal context model based on ontology using WebOntology Language to address issues of semantic representation, context reasoning, context classification and dependency. However, these researches use only the first contexts that are sensed by context providers[6] at first and that can be cause to provide services, so they may lead to lack of more intelligent services for recent users who request extremely various needs in a smart space.

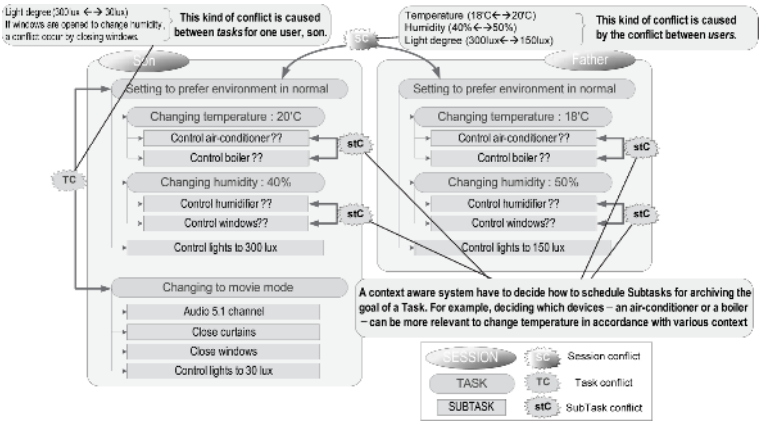


Fig. 1. A classification of context-aware services and service conflicts in an example

3 Context-Aware Services and Service Conflicts

Most of studies for interaction between services categorize the services in respect of controlling devices. Interaction between services can be more complex in especially ubiquitous environments. Context-aware services are intelligent, flexible, multiple, and dynamic services which are depending on user's situational conditions. In smart home environment, which there are various and complex requirements of home members, it is important that context-aware system manages service units to solve conflict problems between services in these aspects.

We first classify context-aware services into three layers - *Session*, *Task*, *Subtask* - to make sure the definition of conflict problems between services. *Session* is a group of Tasks. One *Session* is opened(active) when a user comes in a service-available location - including a virtual location, in which a user is located when he connects from outdoor to the home network for being provided services - and is closed(retired) when the user goes out the location if there are no remain *Task*s. The *Session* that exists per one user and one location may be able to transmit its context information to the other one. A *Task* is a service unit to contain a goal. To archive the goal, the *Task* consists of one or more *Subtask*s that are the smallest service units like controlling a device or setting up its properties. A user can be provided automatically multiple *Task*s in a *Session*. These concepts of classified service layers can provide the base for the interaction between services. Especially, to solve conflict problems between services we classified the problems with our definition of service layers as follow.

Session Conflict. In one location, multi-user can be provided same or similar Tasks simultaneously. Because the Tasks consist of Subtasks to target controlling the same device or changing service environment, the conflict between users can be occurred. We define this conflict as a *Session conflict*.

Task Conflict. A device can receive different commands at the same time when two or more tasks need to be performed for one user. This kind of conflict

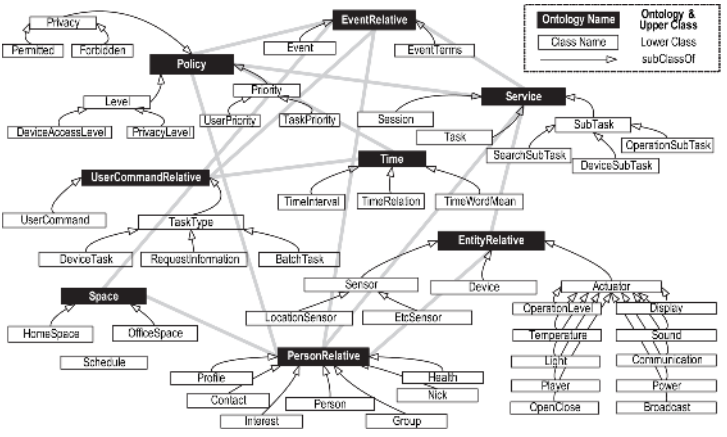


Fig. 2. Context model in a smart home (HomeNet Ontologies[7])

is not the problem between users but Tasks. We define this kind of conflict as a *Task conflict*.

Subtask Conflict. Subtasks that are organized to achieve a goal of a task have to be ordered with contextual information. At this process, it is possible to conflict between Subtasks because two Subtasks intend to control one device at the same time or to control similar devices that change same environment. We define this kind of conflict as a *Subtask conflict*. A *Session conflict* and a *Task conflict* are finally caused by one or more *Subtask conflict*s, but a *Subtask conflict* is not based on them.

Figure 1 shows context aware services newly defined by us and conflict problems with an example. We believe that this new definition will help the conflict problems between services be addressed. Context-aware systems can find or solve service conflicts which can be occurred frequently in the ubiquitous environment easily by modeling ontologies and making rules with our definition.

4 OCASS: Ontology-Based Context-Aware Multi-Agent Service System

4.1 Context Model – HomeNet Ontologies

Context-aware systems need recognizable contextual information and should be able to acquire more information or knowledge through inferencing the relations among them. Moreover, the system should support a context model to be able to share those relations with other systems which are in various connected domains. We have developed the context model(*HomeNet Ontologies*[7]) that is based on ontology expressed using OWL.

The set of ontologies consists of vocabularies to express concepts that are associated with following like person, device, space, time, service, user's activity,

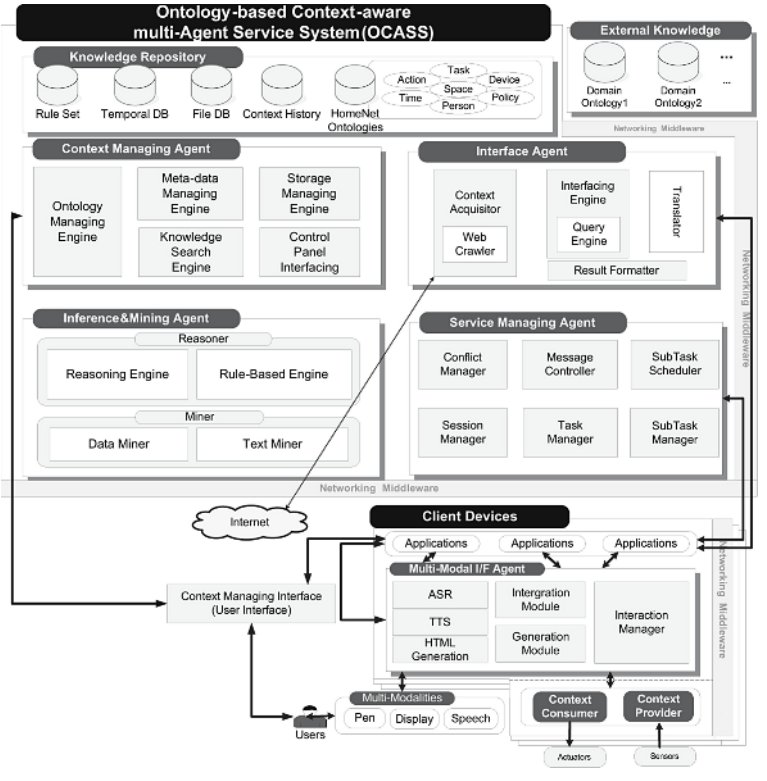


Fig. 3. OCASS architecture and client devices

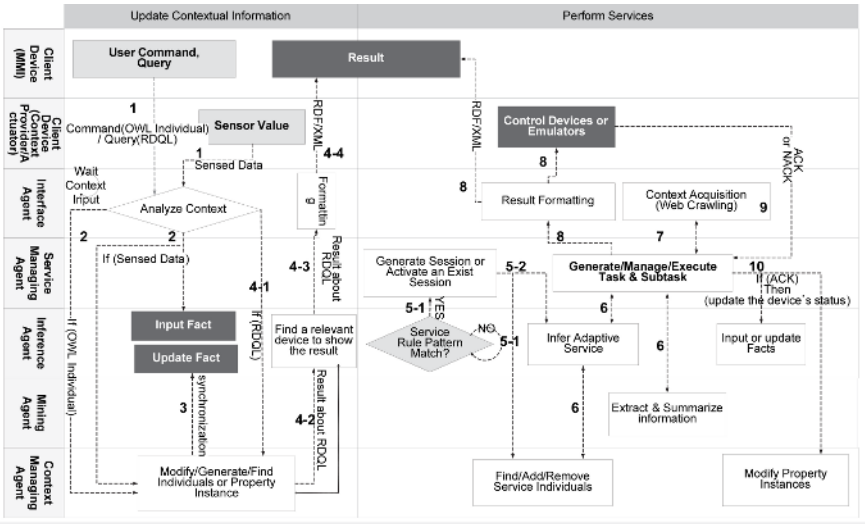
event, and policy. The ontologies are grouped into eight distinctive ontology documents. Figure 2 shows the ontologies, their properties, and their associated relations.

4.2 OCASS Architecture

Our OCASS(an Ontology-based Context-aware multi-Agent Service System) architecture is designed to satisfy the requirements that are specified in Section 1 above for a smart home. Figure 3 shows the system diagram of OCASS design. OCASS consists of several big components - *Knowledge Repository*, *Context Managing Agent*, *Interface Agent*, *Inference & Mining Agent* and *Service Managing Agent*.

* **Knowledge Repository.** The *Knowledge Repository* provides various contexts and information, structured and unstructured data, rules. Also, it stores static and dynamic patterns as knowledge. Using RuleML and JESS[8] script language, *Inference Engine* accesses and stores, deletes its rules.

* **Context Managing Agent.** Context-aware system can efficiently manages number of contexts for supporting context-aware services well. That is, it has



1. First of all, MMI(Multi-Modal Interaction framework) or Context Provider recognizes and collects some contextual data from sensors or users.
2. After Interface Agent translates, analyzes and formats the contexts from Client Deivces into RDF/XML form, Inference Agent updates facts - a collection of information nuggets - to reason the situational conditions.
3. During Inference Agent updates the facts, Context Managing Agent also finds, modifies and generates OWL individuals or property instances for synchronization between the facts and the OWL individuals.
4. If the format of the contextual data from Client Devices is RDQL, Interface Agent forwards the RDQL to Context Managing Agent and Context Managing Agent gives a result of the query. After the result is formatted to RDF/XML, Interface Agent transmits the information to a relevant Client Device that is found by rule-based reasoning with the last facts.
5. If some sensed information matches service rule patterns that a user has written in RuleSet or that is automatically made by learning mechanisms, Inference Agent orders Service Managing Agent to generate a Session or to activate an exist Session. At the same time Inference Agent infers adaptive services based on several facts (for example, user id, user's current location, default tasks, proper actuators, etc.).
6. Then, Service Managing Agent manages multi service layers. At this time, it tries to solve several conflict problems between services that we defined in the above. For instance, it performs scheduling Subtasks so that it prevents service conflicts like turning on a boiler and an air-condition simultaneously if a user didn't order purposely two devices to turn on. Services provided by OCASS can be both controlling devices and extrating or summarizing useful information.
7. If there are more contexts needed for supporting intelligent services, Service Managing Agent requests Context Acquisitor to acquire needed contexts from Context Providers or internet.
8. Finally, Interface Agent makes a proper format of actuator control commands that the actuator can understand and can be operated.
9. In addition, our system estimates whether it executes right services or not for self-learning and for determining to support other available services. For example, if main light in a room is out of order when a service includes the operation of it, the system tries to turn on an alternate device through reasoning of Inference Agent.
10. If the services correspond to an user's intend, OCASS updates the current status that is changed by the services.

Fig. 4. Interaction of OCASS agents

to perform several functions: 1) to find the contexts that devices require and to support them in some available format, 2) to validate and to design the context ontology class, and 3) to store and to manage information related with users, devices, space, location, and etc. *Context Managing Agent* is an agent to perform these three functions in connection with *Knowledge Repository*. We implemented a part of this *Context Managing Agent* related OWL using Jena[9].

* **Interface Agent.** *Interface Agent* performs representing of contextual data from MMI(Multimodal Interaction framework)[10] or sensor agents, creating queries, transforming and transmitting the result about the queries. *Context Acquisitor* collects and selects contexts needed for adaptive service from web or *Context Providers* (MMI or Sensor Agents).

* **Inference & Mining Agent.** *Reasoner* is responsible for checking class consistency and implied relationship and for asserting inter-ontology relations when the system needs to integrate domain specific ontologies. It is implemented using a rule-based reasoning engine. This agent deduces current situation from relations between classes and individuals of context model. We implemented the core of *Inference Engine* using modified JESS[8].

Miner generates contexts and patterns to fit a user's interest and design. *Data Miner* generates or modifies OWL individuals from a useful tendency with statistic information in *Context History*. These individuals are used for *Reasoner* to decide proper services with environmental information for users. We implemented this learning mechanism of *Data Miner* using HHMM(Hierarchical Hidden Markov Model)[11]. *Text Miner* has a function of summarizing and drawing, clustering with unstructured data in a text form. As *Text Miner*, we use the core modules of IN2 Platform [12].

* **Service Managing Agent.** *Service Managing Agent* is an agent that administrates various services. It has three modules to manage three service layers(Session, Task and Subtask), to find service conflicts, and to solve the conflict problems. Especially, *Subtask Scheduler* schedules multiple Subtasks so that solves Subtask conflict problems.

Figure 4 shows how the agents in OCASS architecture interact with each other included MMI(Multi-Modal Interaction framework)[10], Context Providers, and Actuators. In fact, the agents parallelly make their operations caused by ondemand services.

5 Implematation Testbed

5.1 Testbed Setup

In order to test our OCASS architecture for home networks, we created the testbed shown in figure 5. For OCASS gateway host, we used a standard PC (Intel Pentium4 2 GHz) with a JVM from the Java 2 Enterprise Edition. Agents open a remote-method-invocation (RMI) connection to send requests to the appropriate service agents.

For location tracking, we deploy a location tracking system based on IR sensors, pressure sensors and image sensors. The image based location tracking system consists of CCD cameras and a pan-tilt-zoom (PTZ) camera with feature fusion-based people tracking algorithm [13]. Installing the sensors, labeling the coordinates, and manually adjusting sensitivity of sensors, entering this data into our software took about 168 person-hours.

5.2 Service Scenario

We demonstrated our approach with comprehensive scenarios. We trained our system recognizing and learning user's formal/informal activity patterns on *Knowledge Repository* in several weeks.

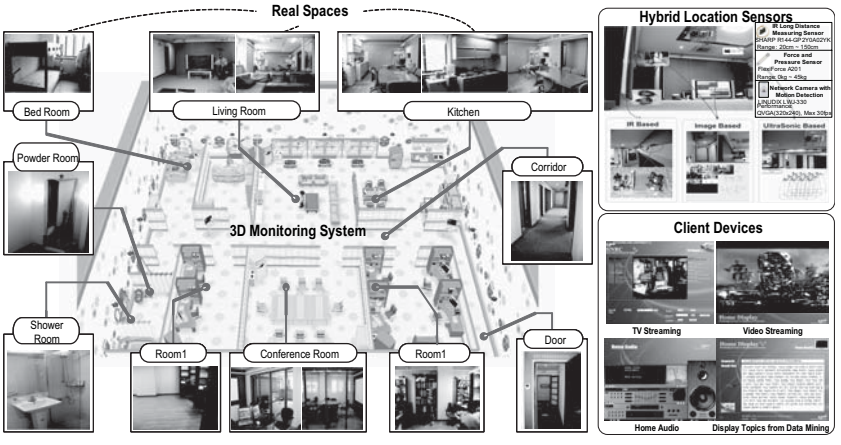


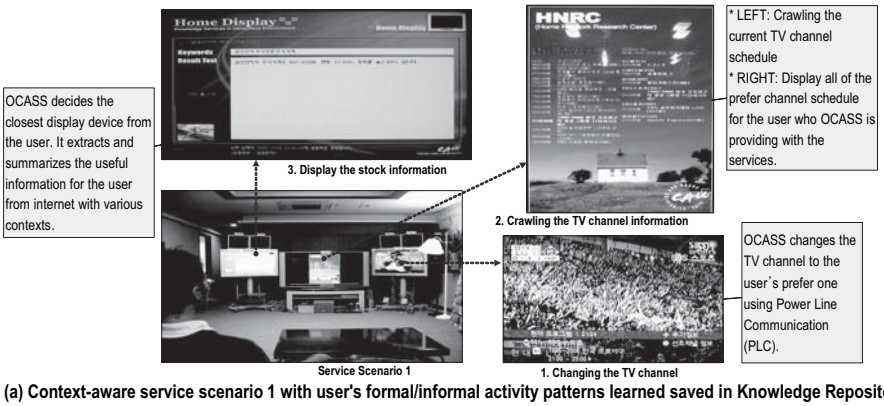
Fig. 5. Our Implementation Testbed in Home Network Research Center

* **Service Scenario 1.** *The context-aware service based on the learned user profiles* (fig. 6(a)): The father enters the living room at 7 PM and sits on the sofa. OCASS reasons the father is at home after his work. OCASS evaluates the dynamic context to provide the father with more adaptive and optimal service environment. OCASS turn on the TV on the display 1 and tunes the preferred channel with the program guide of the channel on the display 2. OCASS also performs the web crawling and show the stock prices of his interests on the display 3.

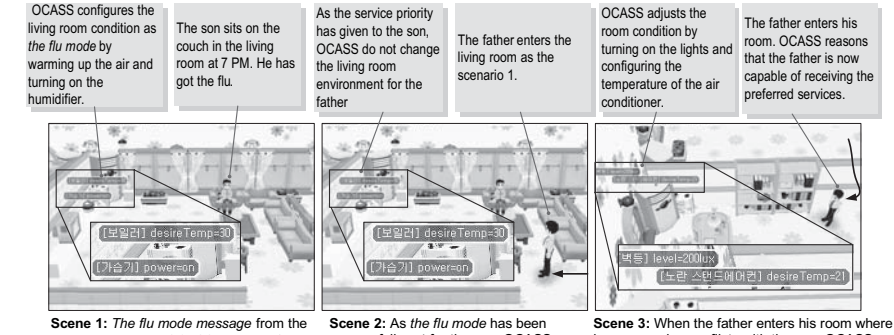
* **Service Scenario 2.** *The intelligent service resolving conflicts between two user services in different situations* (fig. 6(b)): The son laid himself on the couch in the living room at 7 PM. OCASS configures the living room condition as *the flu mode* by warming up the air and turning on the humidifier (Scene 1). When the father enters the living room as the scenario 1, and then the *session conflict* occurs. As the service priority has given to the son, OCASS do not change the living room environment for the father (Scene 2). When the father walks through the corridor and enters his room, location sensors indicate the father's new location to OCASS. OCASS reasons that the father is now capable of receiving the preferred services. OCASS adjusts the room condition according to his preferences by turning on the lights and configuring the temperature of the air conditioner (Scene 3).

5.3 Lessons Learned

In this section, we discuss several findings from the OCASS deployments. In a similar fashion to [14] we share the participants' general feedback on the home network environment with OCASS, where they used it in their homes, and how they interacted with it. We also discuss the OCASS's impact on the lives of the home network service members. The results of our deployments suggest that



(a) Context-aware service scenario 1 with user's formal/informal activity patterns learned saved in Knowledge Repository



Scene 1: *The flu mode message* from the son is delivered to OCASS as a RDF/XML expression

Scene 2: As *the flu mode* has been successfully set for the son, OCASS sends ACK then deletes the command individual.

Scene 3: When the father enters his room where has no session conflicts with the son, OCASS adjusts his room condition according to his preferences by turning on the lights and configuring the temperature of the air conditioner.

(b) Intelligent service scenario 2 resolving conflicts between two user-services with screenshots of 3D monitoring system

Fig. 6. Service Scenario

OCASS deployments can be an effective tool in helping home network members with the tasks of service and information sharing. The intelligent services with OCASS was well received both by the home network members and the elders. In all cases, the home network members who participated said that they would use such a service if it were given to them, and in most cases, they would deploy the system if it were commercially available and affordable. Participants thought that the display was pleasing and blended in nicely with their home appliances, though some complained that the services were "somewhat frightening." They tended to receive the services in often used, common areas of their homes. For example, the services were provided in the family/TV room, living room, home office, or kitchen. When asked about what the elders thought of these deployments of the services, most thought them to be acceptable. Based on our OCASS experience, we think a certain set of service strategies might be generally applicable to the design of smart home environment.

6 Conclusion

In this paper, we describe an Ontology-based Context-aware multi-Agent Service System architecture to provide context-aware services in a smart home. The development of the Ontology-based context model - HomeNet Ontologies - and OCASS architecture are still at an early stage of research. We believe that our system architecture, with the context model, should support tasks including recognizing user's informal activity pattern, learning context patterns, and resolving conflict problems between services. We will continue to develop our system to be able to interact with other agents or systems in an extended smart space. We will also work to develop for supporting a detail resolution of service conflict problems including potential services in the space. In addition, we plan to prototype adjusted context model considering user activity and emotion, activity pattern.

References

1. Dey, A.K.: Providing architectural support for building context-aware applications. Human-Computer Interaction (HCI) Journal **16** (2001)
2. Hong, I., J.: An infrastructure approach to context-aware computing. HCI (**16**)
3. Pascoe, J., Ryan, N., Morse, D.R.: Issues in developing context-aware computing. In: HUC. (1999) 208–221
4. Chen H and Finin T: An ontology for a context aware pervasive computing environment ijcai workshop on ontologies and distributed systems. (In: Acapulco MX 2003)
5. Tao Gu: A service-oriented middleware for building contex-aware services. (In: Journal of Network and Computer Applications 28)
6. Biegel, G., Cahill, V.: A framework for developing mobile, context-aware applications. In: PerCom. (2004) 361–365
7. Yong Kim, EunYoung Hwang, and Sehyun Park: A context aware multiagent system architecture for a smart home. (In: Net-Con'2005)
8. JESS, the Rule Engine for Java Platform: (http://herzberg.ca.sandia.gov/jess/)
9. Jena2, A Semantic Web Framework: (http://www.hpl.hp.com/semweb/jena2.htm)
10. Multimodal Interaction Framework: (http://www.w3.org/tr/mmi-framework)
11. Fine, S., Singer, Y., Tishby, N.: The hierarchical hidden markov model: Analysis and applications. Machine Learning **32** (1998) 41–62
12. IN2 Platform,Saltlux: (http://www.in2web.co.kr)
13. Lee, J., Kim, S., Kim, D., Shin, J., Paik, J.K.: Feature fusion-based multiple people tracking. In: PCM (1). (2005) 843–853
14. Consolvo, S., Roessler, P., Shelton, B.E.: The carenet display: Lessons learned from an in home evaluation of an ambient display. In Davies, N., Mynatt, E.D., Siio, I., eds.: Ubicomp. Volume 3205 of Lecture Notes in Computer Science., Springer (2004) 1–17

Profile Processing and Evolution for Smart Environments

Robbie Schaefer[1], Wolfgang Mueller[1], and Jinghua Groppe[2]

[1] Paderborn University/C-LAB
Paderborn, Germany
[2] University of Innsbruck
Innsbruck, Austria

Abstract. Ubiquitous systems use context information to select and adapt multimodal user interfaces and appliances for individual users in certain situations. However, in order to enable true reactive environments, context information has to be adequately collected, filtered, and processed and combined with user, device and other profiles. In this article, we present how an XML-based transcoding system can be applied for advanced profile processing and evolution. We demonstrate how to encode domain knowledge into sets of rules, which perform adaptations of user, device and context profiles for smart environments.

1 Introduction

Context aware computing [2] is a hot topic in the application of ubiquitous computing [18]. As context awareness and natural user interaction requires smart intelligence, this is also often denoted as Ambient Intelligence (AmI) [1]. To implement context awareness, context data retrieved from various sensors and cameras have to be processed by a system and combined with user, device etc. profiles. Context data as given by context profiles are by no means static and can also have different instances. For examples, a person may walk continuously changing current location and may have varying eyesights over years or even months. Thus, in smart systems dynamic and individual aspects must be taken into account when making decisions, for example, to select an adequate output device and to encode content for best presentation to a user.

In this paper, we consider advanced *profile processing* and *evolution*, which denotes the automated modification of profiles in the course of operations and the construction of temporary profiles. In profile evolution, rules are defined based on domain knowledge under which conditions what parts of profiles are modified, deleted, or newly created. In the domain of expert systems [9], this process is known as forward chaining, i.e., defined rules evaluate existing knowledge (facts) optionally creating new facts until a final conclusion is reached. In our application not just the final result is of importance but the history of the rule selection, i.e., the decision process, to explain the decision to a potential user, which is managed by the explanation component in an expert system. We focus on profiles for ubiquitous computing systems and proof-of-concept verification

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 746–755, 2006.

in a smart home environment mainly considering user and device profiles and combine them with temporary existing context profiles. We demonstrate how an XML-based transformation system can process production rules following the principles of an expert system for profile processing and evolution. Example code in this paper is given in the RDL/TT language as its production rules are more comprehensive and can be processed faster compared to XSLT. However, though we use RDL/TT here, same principles also apply to related approaches like XSLT without loss of generality.

The remainder of this article is structured as follows. The next section discusses related work and gives a short overview of basic technologies followed by an introduction to profile based customization of ubiquitous computing systems. The main part elaborates on steps to create a forward chaining profile evolution system. Thereafter, we present examples and evaluation scenarios before concluding our work.

2 Related Work

For related work we focus on different languages and systems for profile description and processing.

For profile description, RDF (Resource Description Framework) [6] is widely applied. RDF is an XML-based format for the description of arbitrary resources through definition of their attributes. Several other formats like CC/PP (Composite Capability / Preference Profile) [7] apply RDF for their specific domain. CC/PP, for instance, is for the definition of general device properties and user preferences. UAProf (User Agent Profile) [19] is for the definition of vocabularies of device-related information and can apply CC/PP. For the exchange personal data, several systems are based on vCard [4] instead. Another general alternative is Dublin Core (DC) [3], a metadata standard with simple value pair elements for the description of networked resources. DC is not based on a specific format, so that also HTML can be used for metadata definition along the lines of the Dublin Core syntax as given in [8].

For profile processing in Web applications, recommender systems [17] implement filtering and processing of user preferences. For instance, Movielens (www.movielens.umn.edu) applies collaborative filtering for recommendations. Hereby, it captures user preferences to automatically generate a profile by rating movies and searches for similar profiles generating recommendations from them. However, in filtering based approaches no explanations can be generated [17]. Other approaches apply content-based technologies based on Bayesian text categorisation with machine learning, e.g., for book recommendations [10]. They analyse the contents of the recommended items and support the generation of explanations.

In the area of general profile processing and evolution, [14] presents a solution based on fuzzy sets, which focus on the evaluation of profiles with no modification of profile data. Sparacino [16] uses Bayesian networks to construct a profile for museum visitors customizing information of stories. [13] presents an

approach to merge multiple user profiles in context-aware environment considering conflicts of users' interests. They address properties in the context of smart home environments and focus on filtering user profiles due to current user locations, available services, and technical limitations of terminals. Rentto et al. [12] describe user's attitudes and how they interact with their environment. Their studies demonstrate that seamless integration into the environment as well as correctly locating the user is essential for the introduction of the technology.

We have introduced a fuzzy-based profile processing [14] with rule-based description in RDL/TT [15] for transcoding of XML files. In this article, we introduce a dedicated architecture for context-aware profile processing and outline how our system compares to basic principles of expert systems with respect to rule description and processing [9], where human knowledge is represented as data and rules and rules follow *IF-THEN* patterns and operate on so-called facts. In our applications, facts are given by different profiles and rules refer to more or less complex implications like $rain \rightarrow close\_shutter$. We have developed an architecture based on a specific set of profiles dedicated to context-aware computing in ubiquitous environments. In contrast to experts systems, our approach directly processes the intermediate DOM-tree representation of XML files and are thus most effective and efficient with respect to rule representation and runtime [11]. The execution of our rules compares to principles of *forward chaining* enhanced by fuzzy rules starting with initial facts concurrently executing possibly recursive rules until a conclusion is reached.[1] Additionally, our approach supports explanations of decisions in rule derivations. This is of utmost importance when the approach is applied in smart home environments, i.e., to explain non obvious decisions to a user on demand.

3 Profiles for Context-Aware Computing

We have developed a framework for automated customization of smart home environments. We presume that a smart home integrates multiple distributed embedded devices and sensors, which continuously send data and are to be monitored and controlled. Figure 1 sketches the architecture of our customization system, which is further outlined in the next paragraphs.

A set of pre-defined profiles provides information with user preferences, capabilities, and device properties. The monitoring system captures the environment of the smart home, detects users and devices, and continuously generates a *soft-context* profile from the obtained information. The profile processing system retrieves necessary information from input profiles and optionally applies conflict resolution strategies to handle conflicting information finally generating a customization profile. The evolution system uses the customization profile to operate the devices and to generate recommendations as additional customizations. Additionally, it monitors the status of the devices by loading *hard-context* profiles.

[1] Fuzzy rules were introduced because user preferences and capabilities are often given as fuzzy sets in such applications, like $\{high, medium, low\}$.

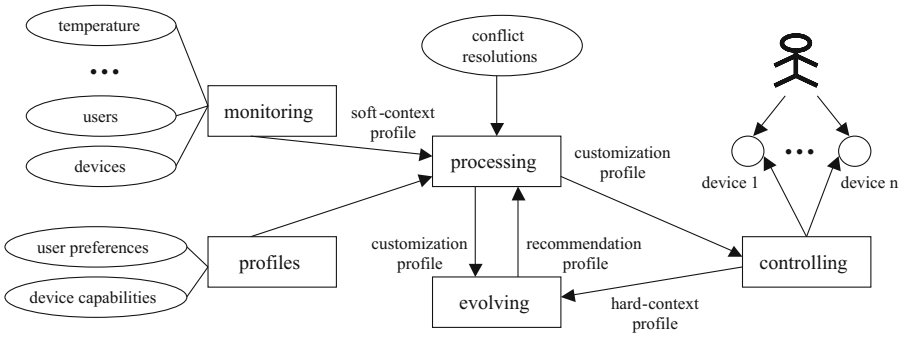


Fig. 1. Customization System Architecture for Smart Homes

Users can individually adapt the customization by optionally operating devices manually. The adaptation is captured by the controlling system and is reflected in the hard-context profile. The evolution can capture the manual operation to the environment by comparing the customization profile with the hard-context profile. When new preferences are available, the evolution system filters the preferences and automatically evolves the user profiles.

The customization system manages two categories of profiles:[2] a set of persistent profiles, which contain basic information independent of the concrete context; a set of temporary profiles, which are constructed on-the-fly and mainly keep information related to the current context like location, temperature etc.

Persistent Profiles

- *User profiles* capture users' general data, preferences and abilities. Beyond general user information like name, preferences specify, e.g., the favourite TV channels and the preferred temperature. Abilities describe the information about a user's physical abilities, like eyesight.
- *Device profiles* describe information related to device properties including the software and hardware as well as general information. A device can support different I/O interfaces.

Temporary Profiles

- *Soft-context profiles* have current information from sensors received via the monitoring system from the so-called soft environment, e.g., temperature, humidity, brightness, and locations of people.
- *Hard-context profiles* describe the current status of devices, e.g., whether a shutter is close or open, which are controlled and captured by the controlling system.

[2] We consider a simplified system here. For broader application, we can additionally combine them with network, service, and application profiles.

- *Recommendation profiles* define a set of new settings, i.e., mainly conclusions for user preferences.
- *Customization profiles* are computed from persistent profiles and soft-context profiles and include modified settings.

4 Forward Chaining Evolution

The implementation of a forward chaining profile evolution takes the four steps, which are outlined in the next subsections.

4.1 Merging Profiles to a Working Memory

The evolution system first creates a master profile denoted as working memory. All relevant profiles are merged into that working memory (wm) on which all further transformation and evaluation steps are applied. In the domain of smart homes, for example, one frequently needs to process the combination of user, device and context profiles. In the working memory, each instance of a profile has a unique ID, which allows the processing of replications of individual profiles. The following example, which uses a language neutral type-values XML representation of profiles, shows how they merge into one working memory.

```
<wm>
  <profile id="xyz123" type="user">
    <meta> ... </meta>
    <property type =firstName> <value>Peter</value> </property>
    <property type="lastName"> <value>Pederson</value> </property>
    <property type="visCap"> <value>high</value> </property>
    <property type=location> <value>Room 35</value> </property>
    ...
  </profile>
  <profile id="dvx432" type="device">
    <meta> ... </meta>
    <property type="screenX"> <value>320</value> </property>
    ...
  </profile>
  ...
</wm>
```

4.2 Specifying Production Rules

Our system is based on RDL/TT, which restructures an incoming XML document to a new XML document by means of transformation rules. For each XML element we can define a set of rules in a Java-like syntax. The following example sketches the rough structure of an RDL/TT definition for the previously introduced wm-element.

```
global:
   <initialization code>
wm:
  <rule 1>;
  ...
  <rule n>;
```

RDL/TT rules can be just simple statements, e.g., for renaming elements or more complex programming constructs such as loops and branching. To build a forward chaining system, we branch when a set of conditions evaluates to true as it is given in the next example.

```
wm:
  if (valueOf(xy123.visCap)=="Low")
  ||(valueOf(dvx432.screenX)<= "640")
  {
   // increase font size
   replace(dvx432.property.fontSize.
           content, "20pt")
  }
  else if ....
```

Note here, that the above excerpt is simplified. In order to retrieve the required profiles, we first have to navigate the tree of XML elements to the right identifier and then to return to the parent profile. This can be, e.g., achieved in one statement through a complete path: "valueOf(this.profile.attribute.id.content.xy123.parent.parent.parent.parent.property.visCap)". Since such long paths are quite cumbersome to describe, we are using a pre-processor that expands the id to the required pre- and suffix and reduces the statement to valueOf(xy123.visCap). Once the condition is true, changes in the working memory are executed. The above example increases the font size to 20 to increase readability for people with poor visual capabilities.

RDL/TT allows the modification of the structure of the working memory, which corresponds to the creation of new facts for expert systems. This is a quite powerful feature but can also be harmful in practice, since the resulting profiles then might no longer be compatible with the applications. Therefore, complex restructuring has to be applied with care. In order to be on the safe side it might be helpful to restrict on modifying only contents of existing properties. Similar to expert systems, which typically have an explanation component, our system stores explanations in a log file for all successful operations on the working memory.

4.3 Executing Rules

Having defined a rule set in combination with a master profile (working memory) we now can start the transcoder with the rule set and the working memory as parameters. The resulting document is a modified version of the working memory, in case of cascaded if-statements multiple transcoder runs can be necessary. The

process terminates when there are no more differences between the input- and output documents. Since we can not guarantee that there are no recurrences in the production rules, an additional upper limit of allowed transcoding runs has to be defined. This number depends on the actual number of rules and the individual application.

4.4 Reconstructing Profiles

After the completed transformations we break up the working memory into individual profiles. For each profile we check for differences with respect to the original profile and, in case of changes, those are marked in a history file because

1. some anomalies might have occurred during processing or an unlikely result has been generated. In such a case, the user or system has to restore the original profile and additionally ask the explanation component why the changes were made. As a consequence, the production rules can be a subject of improvement.
2. the history of changes has to be processed by another component to learn from past behaviour and predict future settings in order to adapt automatically to the user's preferences.

5 Examples and Evaluation

Our main target applications are in the home automation domain. In this context the software is currently evaluated in our Ambient Computing Laboratory (AC-Lab), which is equipped with several sensors as well as means to remotely control different appliances such as lights, shutters, displays etc. In this context we consider a realistic scenario, which can already be partly controlled by our current software.

5.1 Scenario

Consider that it is 5pm and due to previously collected data, the ambient intelligence system concludes that a person - let us call him David – will be at home in half an hour. At or just before the expected arrival time, the smart home system starts with David's preference settings. One action is, for instance, to match the current indoor temperature with the preferred value. Since the temperature is about 18 degrees Celsius and the preferences define 20 degrees Celsius, the heating is turned on. After having authenticated himself at the door terminal, David enters the TV room. The system automatically switches the TV on and checks for David's preferred TV programs in the order of his priorities. Since the sun is standing low and still shining through the windows and since David defined that he likes having daylight at home as long as possible, the system increases the contrast and brightness of the display. However, since David considers the maximum contrast as uncomfortable, he manually selects a lower value. Later, the sun sets and as his preferences define that he prefers

more privacy at night, shutters close and some indoor lights are switched on at low levels. As a consequence, TV contrast and brightness are re-adjusted.

Later, David's wife Sarah arrives, so that the smart home system also has to concurrently process her profiles. Since her temperature preferences are two degrees above David's preferred temperature, the conflicting parameters are easily resolved by applying the average value. However, her TV preferences are in principle on a different channel but since David is enjoying a specific show he has always watches in the past history, the system rates David's preference higher than Sarah's. This is a typical situation, which sometimes cannot automatically be resolved by a system and some manual user interaction is often required to step back and reselect the channel. Nevertheless, processing the history of manual decisions can help to provide a list of good recommendations here.

5.2 Rules

This simple scenario shows two important aspects for profile evolution that need the application of rule based systems: (i) combination of different properties and (ii) conflict resolution. Since even simple scenarios require the definition of a considerable set of rules, an advanced generation of new rules with learning facilities are of high importance for the acceptance of such a system. To outline basic principle, we give a short example with rules and facts in pseudo code.

```
R1: IF(current temperature < preferred temperature)
    THEN switch on heating
R2: IF(someone is scheduled to be present)
    THEN retrieve the preference profile
R3: IF(there is a preference profile)
    THEN retrieve the preferred temperature from the profile

F1: current temperature is 16 degree Celsius
F2: David is arriving in 30 minutes
```

Initially, the system checks for rules with true conditions, given the two facts (F1 and F2). The only matching rule is rule R2, which is selected and fires. The action "*retrieve the preference profile*" is an instruction to load the user profile from the profile database "*there is a preference profile*" and add it to the working memory, so that a new fact F3 is added.

```
F3: there is a preference profile
```

In the next phase, the condition of the third rule evaluates to true, since there is a preference profile available. The fired action informs the system to retrieve the preferred temperature data from the preference profile in the working memory, e.g., the profile says that the preferred temperature is 20, so yet another fact is added to the working memory as F4.

```
F4: preferred temperature is 22 degree Celsius
```

The next cycle begins and Rule R1 fires switching the heating on. After the system has automatically customized the environment according the rules and facts, a user might be not satisfied with the result. The reason for that could be that the user's preference has changed over time or a user has temporarily different preferences as she/he prefers a higher room temperature than usual. In order to modify the environment, the user either informs the system about changes or controls the devices directly. Finally, the system recognizes the manual change, updates profiles, and automatically modifies settings.

5.3 Implementation

In order to implement the scenarios, our proof-of-concept lab is equipped with several sensors for determining locations, temperatures, brightness, humidity etc., which help to generate temporal context profiles. The lab is additionally equipped with home automation facilities like electrical shutters and doors. Finally, we have different interaction devices and modalities installed, e.g., graphical user interfaces, speech control, and gesture control. All appliances as well as monitors and entertainment devices can be switched and controlled by our software. The diversity of sensors and devices allow many different configurations. Different users can work with different preferences and capability profiles, so that scenarios can be evaluated in reasonably complex settings.

Our current implementation is based on RDL/TT. Single runs on complexer rule sets with up to 85 rules execute with up to 590ms, where approximately 80% of the processing time is due to file I/O operations and copying and merging of profiles, which is a clear potential for improvements in our current software. This gives promising results so that we expect to keep system reaction below one second, which is completely sufficient for our applications. In our first evaluations, we just have tested single user scenarios. This was mainly due to the fact that we currently have no precise user localisation system available, which can manage more than one user.

6 Summary and Conclusion

In this article, we introduced advanced concepts for profile processing and evolution and investigated how XML transformation systems can be applied as a forward chaining production system. We identified several different types of temporary profiles and demonstrated how to merge them into a working memory, which was processed by RDL/TT production rules finally splitting them to individual profiles again. Though we have demonstrated principle by means of RDL/TT, concepts also apply to other XML transformation systems like XSLT.

In applications we are mainly interested in context aware computing for home automation, where decisions have to be dynamically taken based on different sets of profiles. Though we have identified a runtime bottleneck in our current implementation when processing large data and rule sets, this is not a critical issue as we expect that the software will be sufficiently reactive after little optimizations

of file I/Os and operations on internal tree structures. Thus, after that, our approach seems to be a good trade-off between performance and rapid prototyping of context aware applications in Ambient Intelligence scenarios.

References

1. Aarts, E., Marzano, S.(eds.): The New Everyday – Views on Ambient Intelligence. 010 Publishers, Rotterdam, The Netherlands (2003)
2. Dey, A. K.: Understanding and Using Context. *Personal and Ubiquitous Computing*, 5(1), (2001) 4–7
3. Dublin Core Home Page. purl.oclc.org/metadata/dublin_core
4. Howes, T., Smith, M., Dawson, F.: *RFC 2425: A MIME Content-Type for Directory Information (vCard Specification 3.0).* IETF (1998)
5. Kay, M.: *XSL Transformations (XSLT) Version 2.0. W3C Working Draft.* World Wide Web Consortium (2005)
6. Klyne, G., Carroll, J.J.: *Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation.* World Wide Web Consortium (2004)
7. Klyne, G., Reynolds, F., Woodrow, C., Ohto, H., Hjelm, J., Butler, M. H., Tran L.: *Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation.* World Wide Web Consortium (2004)
8. Kunze, J.: Encoding Dublin Core Metadata in HTML. IETF-RFC2731. www.ietf.org/rfc/rfc2731.txt
9. Levine, R., Drang, D., Edelson, B.: *Artificial Intelligence and Expert Systems.* McGraw-Hill Book Company, USA (1996)
10. Mooney, R. J., Roy, L.: *Content-Based Book Recommending Using Learning for Text Categorization.* Proceedings of the Fifth ACM Conference on Digital Libraries, San Antonio, TX, USA (2000)
11. Plomp, J., Schaefer, R., Mueller, W.: Comparing Transcoding Tools for Use with a Generic User Interface Format. In *Proceedings of Extreme Markup Languages 2002, Montreal, Quebec, Canada.* (2002)
12. Rentto, K., Korhonen, I., Vtnen, A., Pekkarinen, L., Tuomisto, T., Cluitmans, L., Lappalainen, R.: Users' Preferences for Ubiquitous Computing Applications at Home. In *Proceedings of EUSAI 2003*, *LNCS*, Springer (2003)
13. Salem, B., Rauterberg, M.: Multiple User Proifle Merging (MUPE): Key Challenges for Environment Awareness. In *Proceedings of EUSAI 2004*, *LNCS*, Springer (2004)
14. Schaefer, R.: Fuzzy Evaluation of User Profiles. In *Workshop on User Profiling at CHI 2004, Vienna, Austria*, (2002)
15. Schaefer, R., Mueller, W., Dangberg A.: RDL/TT – A Description Language for the Profile-Dependent Transcoding of XML Documents. In *International ITEA Workshop on Virtual Home Environments, Paderborn, Germany* (2002)
16. Sparacino, F.: Sto(ry)chastics: A Bayesian Network Architecture for User Modeling and Computational Storytelling for Interactive Spaces. In *Proceedings of Ubicomp 2003, Seattle, WA, USA*, Springer, (2003)
17. Ujjin, S., Bentley, P.J.: Building a Lifestyle Recommender System. In *Proceedings of the 10th International WWW Conference, Hong Kong* (2001)
18. Weiser, M.: The Computer for the 21st Century. *Scientific American*, 256(3) (1991)
19. Wireless Application Protocol Forum: *Wireless Application Group User Agent Profile Specification.* (1999)

A Context-Aware Smart Home Service System Based on uWDL*

Yongyun Cho, Kyoungho Shin, Jaeyoung Choi, and Chaewoo Yoo

School of Computing, Soongsil University,
1-1 Sangdo-dong, Dongjak-gu, Seoul 156–743, Korea
{yycho, delio}@ss.ssu.ac.kr, {choi, cwyoo}@comp.ssu.ac.kr

Abstract. For a smart home in ubiquitous computing environments, execution of all the home services must be dependent on user's situation contexts, which are dynamically generated in ubiquitous environments. In this paper, we propose a home-network service system that can support home services appropriate to user's situation information in ubiquitous computing environments. The suggested system uses a uWDL workflow service scenario [2] describing user's situation information as service execution constraints and supports a context-aware home service through comparing contexts described in the service scenario with user's situation information generated from ubiquitous computing environments. To do that, the suggested system represents contexts described in a uWDL document as a context subtree, which expresses not only context data but also relation information among services into the fields of its node. The suggested system uses a algorithm for context comparison between context subtrees and user's situation information. The algorithm selects a context that has all together values and types entirely equal to those of user's contexts. Therefore, the suggested system will be useful in the development of a context-aware home-network workflow service application based on uWDL in the ubiquitous computing environment.

1 Introduction

Compared with traditional distributed computing environments, workflow services in ubiquitous computing environments must decide a service transition according to the user's situation information that is generated dynamically [1,2]. uWDL (ubiquitous Workflow Definition Language) is a workflow language based on a structural context model which expresses context information as transition constraints of workflow services [2]. Through a uWDL workflow service scenario, an user can describe what service must be executed according to situation information. For execution of context-aware services, we need a method that can recognize contexts in a scenario and select a service correspondent with situation information. Through a workflow service scenario document in uWDL,

* This research is supported by the Ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 756–765, 2006.

developers can represent context information as workflow's state transition constraints. To make a uWDL workflow service aware of contexts, a developer needs a method that can drive a correct service from workflow services described in uWDL scenario by comparing context with user's situation information.

In this paper, we present a uWDL-based home-network system that represents contexts described in a uWDL workflow service scenario document as rule-based context subtrees, and derives service transition according to user's state information in ubiquitous environments. For that, a uWDL handler in the uWDL-based home-network system represents contexts into node fields of a parse tree that it generates as a result of parsing for a uWDL scenario document. The uWDL mapper uses the sub-tree's node information in comparison with user's situation information.

2 Related Work

2.1 Context-Aware Workflow Services in Ubiquitous Computing Environments

Context in a ubiquitous environment means any information that can be used to characterize the situation of an entity [3]. The subject of context information is called an entity. Profile information is the explicit information on an entity. In ubiquitous environments, a context can be expressed with RDF-based triplet [2]. RDF(Resource Description Framework) [9] is a language to describe resource's meta-data and it expresses a resource as a triplet of {subject, predicate, objective}. For example, let's suppose such a situation as *John sits down a sofa in living room.* It can be expressed as {(UserType, John), (ActivityType, sit), (SofaType, livingroomSofa)}. An application or a system that uses context information or performs context-appropriate operations is called a context-aware application or a context-aware system [4]. The existing workflow languages, such as BPEL4WS [5], WSFL [6], and XLANG [7], are suitable for business and distributed computing environments. However, they do not include any element to describe context information in ubiquitous computing environments as transition conditions of services.

2.2 uWDL (Ubiquitous Workflow Description Language)

uWDL [2] can describe context information as transition conditions of services through the <context> element consisting of the knowledge-based triplet - subject, verb, and object. The uWDL reflects the advantages of current workflow languages such as BPEL4WS, WSFL, and XLANG, and also contains rule-based expressions to interface with the DAML+OIL [8] ontology language. In uWDL's schema [2], the <node> element points to web services in ubiquitous environments. The <context> element contains the <constraint> element in order to specify high-level context information generated by ontology and inference services as a form of structural description. The <constraint> element has the triplet subelement of <subject>, <verb>, and <object> based in RDF.

The <transition> element specifies the state change of a current node. The <condition> element makes a decision to select a proper service by context, profile, and event information. The composite attribute of the <constraint> element has a value of 'and', 'or', and 'not'. The <rule> element means a set of the <constraint> elements.

3 A uWDL-Based Home-Network Service System

3.1 A System Architecture

Figure 1 shows the architecture of the suggested uWDL scenario-based smart-home system, which is aware of user's situation information in ubiquitous computing environments.

As shown in Figure 1, the suggested system supports context-aware home services using a uWDL service scenario, in which an user's situation information is described as a services execution condition. After a service developer or an end-users writes a uWDL workflow service scenario, the scenario is transmitted to a uWDL context handler in Figure 1. The uWDL context handler represents contexts described in a uWDL scenario as RDF-based context subtrees through parsing. A uWDL context mapper in Figure 1 uses a context subtree for context comparison with user's situation information, which are generated

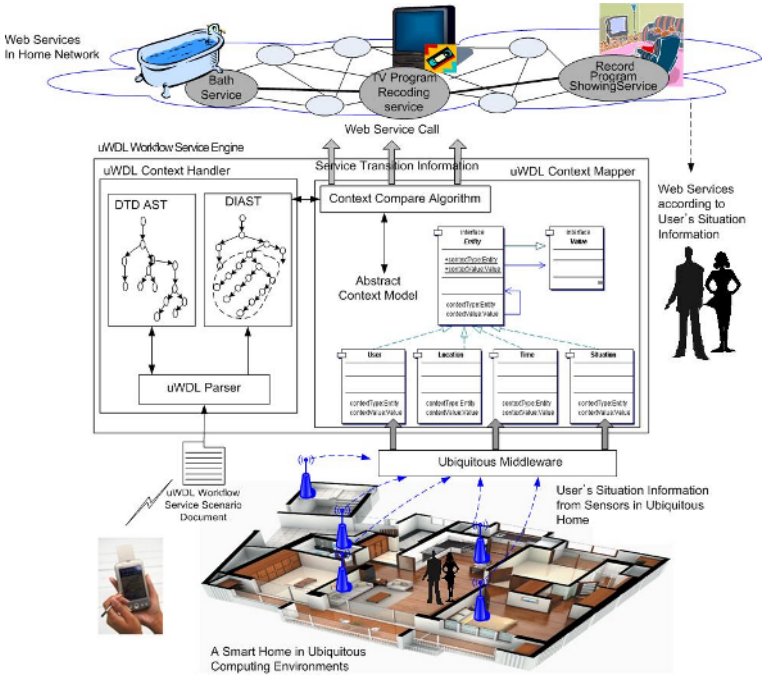


Fig. 1. A smart-home system's architecture based in a uWDL workflow service scenario

dynamically from ubiquitous computing environments and are objectified as the entities through the abstract context model [2]. Through the context comparison, the uWDL context mapper searches home service networks for a home service appropriate to a user's situation information generated from ubiquitous computing environments. If it finds a service corresponding to a specific user's situation information, the uWDL context mapper calls the service to offer context-aware home service to users.

3.2 Editing a uWDL Scenario Through a uWDL Scenario Editor and a Hand-Held Equipment

Service developers or end-users can make a uWDL service scenario through a special scenario editor or such the mobile equipments as PDA or hand-held PC. Especially with hand-held equipments, end-users can easily reserve services that they want in anytime and anyplace. The suggested system offers a method that end-users can edit a uWDL scenario with PDA. Commonly, to edit uWDL scenario in XML is inefficient in PDA's edit environment, which offers less computing resource than general-purpose computers. Therefore, we design a uWDL scenario editing environment for PDA. The environment is based in user interfaces to describe context's constraints with RDF-based triplet, subject, verb, and object. Figure 2 shows the suggested uWDL scenario editing environment in PDA.

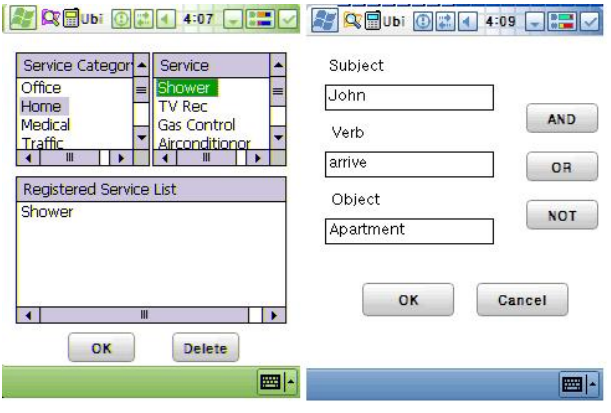


Fig. 2. A uWDL scenario editing environment in PDA

The PDA editing environment consists of a service category window, a service list window, a registered service window, and a constraint info. pop-up window to input constraint information. An end-user can select a service category from the service category window, and then the service list window shows lists, which an end-user can select a service that he want. When an end-user selects a service, the constraint info. pop-up window appears on the PDA screen. Through the text fields in the type of RDF triplet of the pop-up window, an end-user

can input constraint information, which is needed to execute the service that he selected. For example, let's suppose again such a situation as *John sits down a sofa in living room.* It can be expressed as {(UserType, John), (ActivityType, sit), (SofaType, livingroomSofa)}. Therefore, an user has to input the constraint information fields as shown in Figure 2. After finishing the inputting, a uWDL scenario for the information is produced and is transmitted to the uWDL home service engine through Internet for context-aware home services. In the constraint info. pop-up window of Figure 2, 'AND', 'OR', and 'NOT' buttons are for the composite attribute of the <constraint> element in uWDL scenario.

3.3 A uWDL Context Handler

As a result of a parsing, the uWDL handler makes a DIAST (Document Instance Abstract Syntax Tree) that represents the structure information of a uWDL document as a tree data structure. At this time, a context described as RDF-based triplet entity in a uWDL scenario is constructed as a subtree of the parse tree. Figure 3 shows a part of an DIAST that the uWDL context handler creates for a uWDL scenario document.

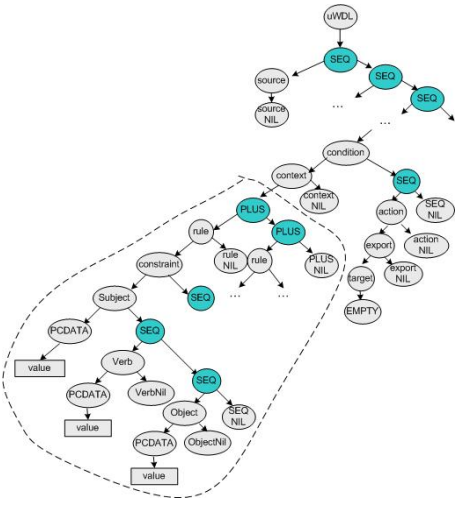


Fig. 3. A part of uWDL DIAST's structure

In Figure 3, transition conditions that can decide workflow service transitions are represented by a <constraint> area that is circled with a dotted line. It means that a subtree whose root node is a <constraint> element represents contexts in a uWDL scenario as the <constraint>'s lower nodes in RDF-based triplet. The uWDL context mapper uses the subtrees whose root node is <constraint> in the DIAST for context comparison with a user's situation information from sensors. For that, the uWDL context handler uses field information of a subtree's

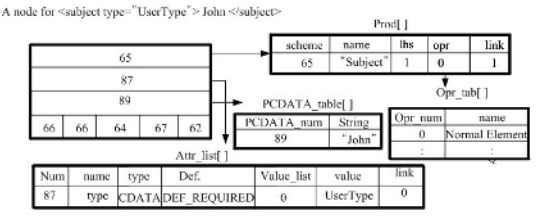


Fig. 4. A architecture of subtree's data tables for <subject type="UserType"> John </subject>

node structure. For example, let's consider a scenario statement, <subject type= "UserType"> John </subject>. Figure 4 shows a subtree and data tables that the handler produced for the statement after parsing.

In Figure 4, the Production Number is a unique element number which distinguishes each element node. The Left, Parent, and Right links indicate for element's order and connection information in the DIAST. Each node in the DIAST is divided into a common element node or an operator node. An operator node displays a meta-character to express a language-specific characteristic of elements in a uWDL's DTD. The Attribute Information PTR is a pointer that indicates a relevant record of Attr_list[ ] to get an element's attribute value. The String PTR is a pointer that indicates a record of the PCDATA_table that contains string information of PCDATA or COMMENT element.

3.4 A uWDL Context Mapper and a Context Comparison Algorithm

Contexts that the context mapper uses for the comparison are described in a triplet based on RDF. Context information from the sensor network can be embodied as a triplet consisting of subject, verb and object according to the structural context model based in RDF. A context described in the <constraint> element in the uWDL service scenario consists of the triple entity based in RDF. The context mapper extracts context types and values of the entity objectified from sensors. It then compares the context types and values of the objectified entity with those of the DIAST's subtree elements related to the entity. In the comparison, if the context types and values in the entity coincide with the counterpart in the DIAST's subtree, the context mapper drives the service workflow. A context comparison algorithm is shown in Figure 5.

In Figure 5, we define a context embodied with a structural context model from the sensor network as OC = (OCs_type, OCs_value), (OCv_type, OCv_value), (OCo_type, OCo_value), and a context described in a uWDL scenario as UC = (UCs_type, UCs_value) (UCv_type, UCv_value), (UCo_type, UCo_value). OC means a context objectified with the structural context model, and it consists of OCs, OCv, and OCo, which mean subject, verb, and object entities, respectively. UC means a context described in a uWDL scenario. UCs, UCv,

```
Boolean MatchContext(UC A, OCS B) {
  int j; /* For the index of context in B each context set */
  for each j in OCS B { /* Repeatedly comparing contexts in A, B context set */
      if ((A.UCs_type == Bj.OCs_type && A.UCs_value == Bj.OCs_value) &&
          (A.UCv_type == Bj.OCv_type && A.UCv_value == Bj.OCv_value) &&
          (A.UCo_type == Bj.OCo_type && A.UCo_value == Bj.OCo_value))
                return TRUE /* Found context match */
  } /* End for */
  return FALSE; /* Return matchresult */
}
```

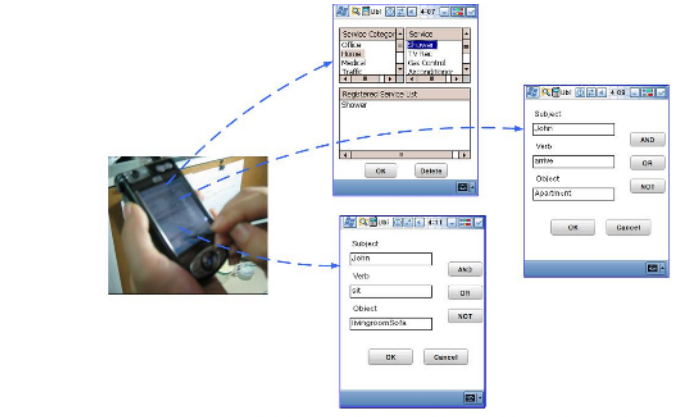
Fig. 5. An algorithm for comparing UC A with OCS B

and UCo mean subject, verb, object entities, respectively. A context consists of a pair of type and value. Also, OCS and UCS that mean each set of OC and UC can be defined as OCS = (OC1, OC2, OC3, ..., OCi) and UCS = (UC1, UC2, UC3, ..., UCi).

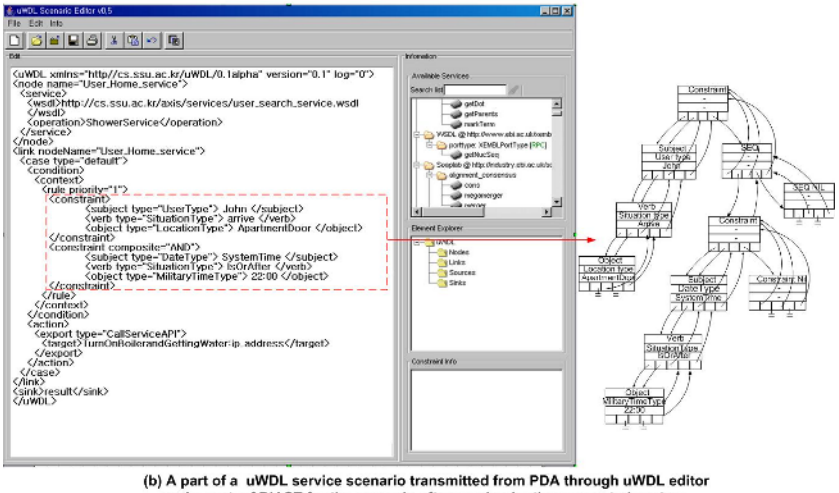
4 Experiments and Results

For an experiment, we will developed an uWDL scenario for home-network services in ubiquitous environments, and show how the suggested uWDL mapper makes the workflow's service perform context-aware transitions. The uWDL scenario will be written with PDA and be transmitted to a uWDL editor in the uWDL home service engine. The example scenario is as follows: John has a plan to go back his home at 10:00 PM, take a warm bath, and then watch a recorded TV program, which he want to see after a bath. When John arrives to his apartment, a RFID sensor above the apartment door transmits John's basic context information (such as name, notebook's IP address) to the uWDL home service engine server. If the conditions, such as user location, situation, and current time, are satisfied with contexts described in the uWDL workflow service scenario, then the server will prepare warm water. When he sits the sofa in the living room after he finishes a bath, the service engine will turn on the power of TV in the living room and play a TV program that was recorded already. Figure 6(a) shows constraints inputted through PDA for the service scenario. Figure 6(b) shows the uWDL service scenario that the service engine takes through Internet and shows a part of the <constraint> subtree of DIAST that the WDL parser produces for the uWDL scenario. Now, if the context mapper receives context data objectified as (ActivityType, sits), (UserType, Michael), (UserType, John), and (SofaType, livingroomSofa), it compares the contexts' types and values with the subtree's elements shown in Figure 6(b). In this case, because the system recognizes Michael as a stranger, the context (UserType, Michael) is not suitable anywhere in the subtree's elements and it is removed.

OCs can be generated innumerably from a sensor network according to a user's situation. Therefore, the uWDL handler must quickly and correctly select an OC coinciding with a UC, which is described in the uWDL scenario from such



(a) A home service registration based in uWDL scenario through inputting constraints with PDA.



(b) A part of a uWDL service scenario transmitted from PDA through uWDL editor and a part of DIAST for the scenario after parsing by the suggested system.

Fig. 6. A uWDL scenario editor for a example service scenario and a DIAST's subtree

innumerable OCs. We generated OCs in random and took two experiments under specific conditions. As conditions for first experiment, we increased the number of OCs incrementally and placed the OCs coinciding with the UCs into the middle and end of the OCs that we produced randomly. Through the experiment, we measured how fast the suggested uWDL handler found the OCs that coincided with the UCs in the uWDL scenario of Figure 6(b) according to the conditions. In the second experiment, when we generated a OC that is not correspondent with a UC, we add new conditions, under which each s, v, and o of the OC respectively become a factor of the discordance among UC and OC. For example, when we generated OCs which are not correspondent to UCs in Figure 6(b), we make v and o of OC's triplet entities to be correspondent with the UCs' counterparts, but s of the OC to be discordant with UC's counterpart. In that case, the s is a

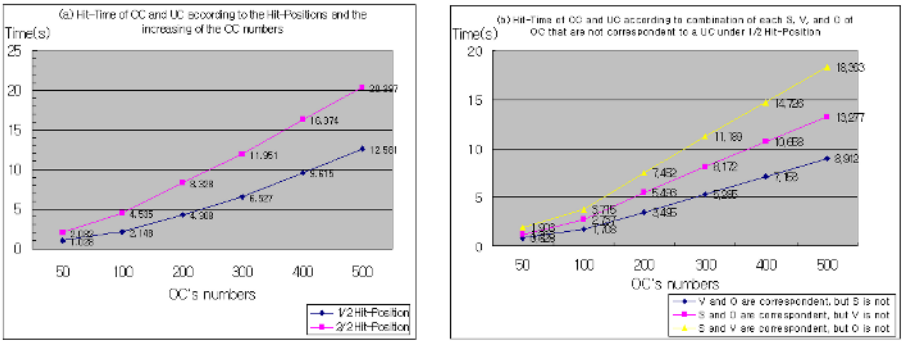


Fig. 7. A result of hit-time of OC and UC according to hit-position and conditions of OC's s, v, and o

factor of the discordance. We use a Pentium 4 3.0 Ghz computer with 1G memory based in Windows XP OS as a uWDL home service engine and a PDA with 512M memory based in Windows CE for the experiment. Figure 7 shows the result. In Figure 7, we increased the OC's amounts by 50, 100, 200, 300, 400 and 500 incrementally. In Figure 7(a), 1/2 hit-position means that the position of the OC coinciding with the UC is the middle of the produced OCs, and 2/2 hit-position means that the position of the OC is the end of the OCs. In Figure 7(b), we generated OCs which include respectively S, V, and O as a factor of discordance with UCs. As shown in the result of Figure 7(a), the hit-time did not increase greatly regardless of the OCS's considerable increase. As shown in the result of Figure 7(b), the hit time in case that a factor of discordance is O of OCs is larger. The reason is that the suggested context comparison algorithm compares OC with UC in order. However, we can see the hit time was not increased a greatly in that case either. Also, we found that the registered services, such as a bath service, a TV program recording service, and a TV program playing service, had executed. It shows that the suggested uWDL handler can sufficiently support context-aware workflow services.

5 Conclusion

For context-aware and autonomic services, contexts in a uWDL workflow service scenario need to be translated into a data structure to express such various and complicate information as user's position or time, and to be manageably used as condition information for service transitions. In this paper, we present a uWDL context comparator that can recognize a uWDL workflow service scenario document, and can drive service transition according to a user's situation information. Through experiments, we showed processes in which the uWDL handler represents contexts described in the uWDL scenario document as subtrees. For context comparison algorithm, we defined contexts described in a uWDL scenario as OC and contexts objectified from ubiquitous computing environments.

We showed an experiment in which the uWDL mapper compared contexts of UCSs and OCSs through the context comparison algorithm, and measured hit-times and service transition accuracy to verify the efficiency of the algorithm. Through the results, we found that the hit-times were reasonable in spite of increment in the OCs amounts. Therefore, this uWDL-based home-network service system will contribute greatly to the development of the context-aware home-network service application programs in ubiquitous computing environments.

References

1. W. Keith Edwards and Rebecca E. Grinter, Ubicomp 2001, LNCS 2201, pp. 256-272, 2001.
2. Joohyun Han, Yongyun Cho, Jaeyoung Choi: Context-Aware Workflow Language based on Web Services for Ubiquitous Computing, ICCSA 2005, LNCS 3481, pp. 1008-1017, (2005)
3. Anind k. Dey: Understanding and Using Context, Personal and Ubiquitous Computing, Vol 5, Issue 1, pp.69-78 (2001)
4. Guanling Chen, David Kotz: A Survey of Context-Aware Mobile Computing Research, Technical Report, TR200381, Dartmouth College (2000)
5. Tony Andrews, Francisco Curbera, Yaron Goland: Business Process Execution Language for Web Services, BEA Systems, Microsoft Corp., IBM Corp., Version 1.1 (2003)
6. Frank Leymann: Web Services Flow Language (WSFL 1.0). IBM (2001)
7. Satish Thatte: XLANG Web Services for Business Process Design, Microsoft Corp. (2001)
8. R. Scott Cost, Tim Finin: ITtalks: A Case Study in the Semantic Web and DAML+OIL, University of Maryland, Baltimore County, IEEE (2002) 1094-7167
9. W3C: RDF/XML Syntax Specification, W3C Recommendation (2004)
10. James Snell: Implementing web services with the WSTK 3.2, Part 1, IBM Tutorials, IBM (2002)
11. Jena2-A Semantic Web Framework,available at http://www.hpl.hp.com/semweb/jena1.html
12. Deborah L. McGuinness, Frank van Harmelen (eds.): OWL Web Ontology Language Overview, W3C Recommendation (2004)

Toward Context-Awareness: A Workflow Embedded Middleware*

Shaxun Chen, Yingyi Bu, Jun Li, Xianping Tao, and Jian Lu

National Laboratory for Novel Software Technology, Nanjing University
Nanjing City, P.R. China, 210093
csx@ics.nju.edu.cn

Abstract. Context-aware computing is widely researched in recent years, but we lack for a powerful context-aware middleware which supports a uniform programming model. So developing context-aware applications is still complex and time-consuming. We introduce a workflow embedded middleware called FollowMe. It supports pluggable context-aware applications. FollowMe includes a workflow engine and sustains applications described with pvPDL, which is a workflow definition language proposed specially for context-awareness. The employment of workflow makes the development of applications simplified and the maintenance much easier. We testify the improvement by realizing an example and the related evaluation.

1 Introduction

The concept of pervasive computing was introduced by Mark Weiser in 1991. He described pervasive computing as the everywhere and anytime computing which is transparent to users [1]. Pervasive computing calls for a new human-computer interaction mode in order to decrease users' attention to computing and make computing invisible to users. To achieve this, context-awareness is developed. Contexts refer to any information that portrays the situation of users and the environments the users in. Computational devices with the ability of context-awareness can follow the situation and provide the users with adequate services without users' intended input.

Although context-aware computing is progressing rapidly, developing context-aware applications is still a complex and time-consuming job. Application developers should learn the details about bottom layers such as the working state of sensors. When an application changes they should totally rewrite the program, and when the domain varies, little work can be reused. Furthermore, applications are self-governed. They do not have a uniform construction so that they cannot share an infrastructure and developers can not reuse the common parts of them. Besides, most context-aware projects pay their attention to providing a single service instead of a flow of services.

In this paper, we establish a middleware called FollowMe, which shields bottom layers details, and provides a workflow based programming model for context-aware applications.

* Funded by 973 of China (2002CB312002) and 863 Program of China (2005AA113160, 2005AA113030, 2005AA119010), NSFC (60233010, 60403014).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 766–775, 2006.

Our system includes a workflow engine. It is event-declarative and RDQL [2] apprehensible. FollowMe supports pluggable applications described with pvPDL (Pervasive Process Definition Language), which is a workflow definition language designed for context-awareness. The development of context-aware applications become more efficient and workload of maintenance is highly reduced in virtue of workflow.

Our programming model is constructed upon the OSGi [3] framework and ontology based context model. OSGi provides modularization and platform independency, and the services organized by OSGi are hot-pluggable. With the help of domain-independent context model, FollowMe can be customized to specific domains by deploying domain related ontology.

The rest of paper is organized as follows. We list related work in section 2 and discuss why workflow is suitable for context-aware middleware in section 3. In section 4, we give an overview of FollowMe's architecture. Then we describe the implementation of the workflow system and the characters of pvPDL in section 5, and show a scenario described with pvPDL as an example in section 6. At last, we evaluate our system and conclude respectively in section 7 and 8.

2 Related Work

This paper is part of work of FollowMe Project. While [13] is an overview of FollowMe system, this paper focuses at the workflow based programming model which facilitates the pervasive application development and distinguishes our system from others.

We notice that there are mainly two methods for context exploitation in existing projects. One is the event-trigger mechanism, such as Context-toolkit [4], Context Fabric [5], and Context Cube [6]. The common ground of them is treating a context as an event and providing services in the callback of the event. The other is the rule-based declarative programming method. There are some prototype systems employing this method to specify context-aware application behaviors, such as [7]. Jose J. Alferes, et al. proposed an more flexible approach using Logic Program Updates, in which the rules for context-awareness changed dynamically [8].

However, these studies focused on one single service rather than a serial of services. In pervasive computing environments, an application may be composed of a chain of services triggered by a sequence of contexts. If we use the simplex event-trigger mechanism to process these scenes, we have to register an event in the callback of the previous event, that is, recursive event registers. It leads to ugly and unreadable programs. Declarative programming is a good choice in some domains, but it is sometime inefficient. When there is a flow of contexts and some context is the precondition of another, rules become complicated and performance goes very low. So, we introduce a workflow model to exploit contexts and develop applications.

Having a workflow system, we also need a workflow definition language to describe applications. Actually, there are quite a few such languages as XPDL [9], BPML [10]. But they have several problems when adapting to context-awareness. First, context information is a complex set of data. Traditional workflow definition languages can not express them felicitously. Second, services of a context-aware application can not simply use another service's result, but instead context information obtained from users

and environments. So we propose pvPDL to describe context-aware applications in our system.

[14] also proposed a workflow language which combined with web services for ubiquitous computing. However, there were few clues indicating that the workflow language they used was optimized and suitable for pervasive applications. In contrary, we endow our workflow engine with event-declaration and RDQL comprehension abilities which strengthen the description competence of contexts and simplify the development of context-aware applications. These will be discussed in part 5.

3 Benefits of Workflow for Context-Awareness

Workflow is the automation of a business process, that is, a serial of related activities [11]. The workflow technology has been successfully applied to the traditional business domain and distributed computing. We use workflow to perform services composition, flow management, task distribution and collaboration.

In pervasive computing environments, people get right services at right time without users' intervention. However, a user may request a flow of services and these services are highly related. What's more, most context-aware applications of smart environments have inherent business flow logic and can be divided into several simple parts (services). In these cases, workflow is highly applicable for pervasive computing. When workflow meets context-awareness, it shows advantages as follows.

Firstly, with the help of workflow, FollowMe can support pluggable context-aware applications and software structure is clearer. We define pluggable applications as those applications which can be deployed, removed and updated dynamically by only modifying configuration files without stopping the system. Workflow decouples business logic and the realization of concrete functions. In our system, a context-aware application appears as a workflow description file (a pvPDL file) and a number of encapsulated services. We can add or update an application by adding or modifying a corresponding pvPDL file at runtime (See section 5). In FollowMe, applications built in the workflow model are pluggable and structured. As a result, it contributes to the flexibility and modularization of FollowMe.

Secondly, workflow simplifies the development of context-aware applications. These applications are usually driven by a chain of events (contexts or variation of contexts). In traditional hardcode mode, we have to register the next event in the codes processing the previous event. It will cause the confusion of software structure when the event chain is very long. This is only an aspect of the improvement on application development; we will discuss it in part 1 of section 7.

Moreover, developing applications in workflow mode facilitates reuse. As mentioned above, an application is divided into several simple parts and some of them can be reused by other applications. For example, "personal identification" can be shared with quite a few context-aware applications. Workflow raises the level of reuse.

The cost of maintenance is reduced in workflow systems as well. Formerly, when an application changes, even just two steps of application exchange, we have to rewrite large quantities of codes. Now, in the workflow model, When the flow changes, we just modify application description files (pvPDL files) and, if function changes, modify a

few codes. The pvPDL files are declarative specifications rather than explicit programming behaviors. So modification is much easier.

In addition, workflow helps us distribute tasks to each user and facilitates the collaboration between them. Actually, most of practical context-aware applications have a flow of services and call for the cooperation between users. Traditional methods can not deal with the collaboration of services in a concise way, and this is the strongpoint of workflow. By the way of organizing services and distributing them to proper users, workflow actualizes cooperation in context-aware applications easily.

Finally, in our system, applications are described with pvPDL which is declarative specifications. It is propitious to turn our system to be data-centric and platform independent. In FollowMe, context-aware applications can run on the any OSGi equipped computational device which has pvPDL files. Services needed can be downloaded from network at runtime.

4 Architecture View

Overview of FollowMe's architecture. As Fig.1 illustrates, our system has three layers: physical, platform and application layer.

Physical layer gathers data from physical world via various sensors and formats them to be computer-readable. We use Crickets [12] to perceive users' positions and Mica2 [12] for light, noise, temperature, etc.

Platform layer is the core of FollowMe. It is built on OSGi framework and manages all services to facilitate the developing and deploying of context-aware applications. Platform layer shields details of the physical layer. It includes an ontology based context model and a workflow system, which will be discussed in the next section.

Application layer uses services and libraries provided by the platform layer. It includes pvPDL files which describe business logic of applications, and a GUI called FollowMe Application Builder for developing context-aware applications graphically (See Fig. 3).

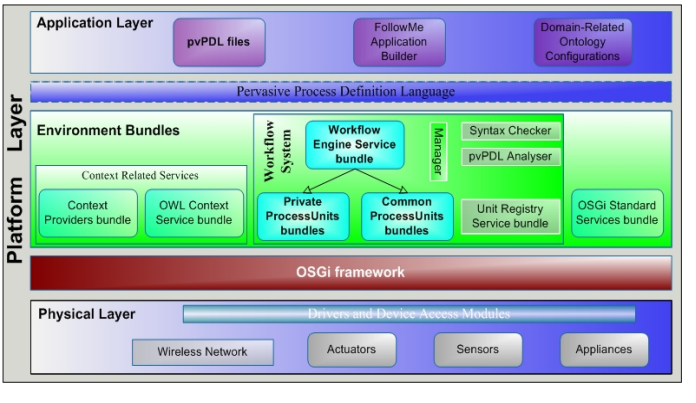


Fig. 1. FollowMe middleware architecture

OSGi platform. OSGi framework supports hot-pluggable modules called bundles, which can be downloaded and deployed at runtime. FollowMe is built on OSGi platform; the services, including workflow services and context services, are encapsulated in bundles (See Fig.1). Another advanced character of OSGi is platform independency. System built on OSGi can be easily ported to PDAs, smart phones, etc.

Ontology based context model. FollowMe's context model is based on ontology and described with OWL. This model is formalized and flexible. FollowMe can be customized to specific domains by deploying domain related ontology. In the scenario of section 6, we customize it to support the office automation domain. In Fig. 1, context providers bundle interacts with sensors and transforms sensor data to raw contexts; context services bundle generates high-level contexts from raw contexts and manages contexts' life cycle and conflicts.

Physical distribution. In our smart environment, sensors, context providers, and context consumers (applications) are physically distributed. The workflow engine and the context reasoner are on the central node. Computers connect through ethernet and wireless network.

5 Workflow System and Pervasive Process Definition Language

The employment of workflow is a remarkable contribution of FollowMe. In this section we will discuss the workflow system in detail.

5.1 Running Mechanism of Workflow System

The business logic of a context-aware application is described in a pvPDL file. This file is firstly parsed by the pvPDL syntax checker. If passed, it will be explained by the pvPDL analyser and the result is stored. The starting condition of the application is registered to the workflow manager. These are done only when the system starts or the application is deployed on FollowMe for the first time. When the condition is satisfied, the analysis result will be read and executed in a new thread by the workflow engine which is on the central node. The workflow engine calls various Process Units, which provide services to perform the functions of activities. Common Process Units can be reused among different applications while Private Process Units are specially for one application. Physically, they are both encapsulated in bundles and located in distributed nodes. The unit registry bundle discovers and registers new Process Units automatically. The modules mentioned above have been shown in the middle layer of Fig.1.

We can see that if we want to add a new application to FollowMe, the only work is creating a pvPDL file and some Private Process Units. Process Units can be reused among applications and downloaded from other nodes at runtime. So only a few Process Units should be developed. Both the file and the units can be deployed to the middleware dynamically, having no influence on other running applications. When we want to change or remove an application, we can modify or deregister the pvPDL file and corresponding Process Units. The workflow system detects the alteration of applications automatically.

5.2 Workflow Engine

The workflow engine is also a bundle on the OSGi platform. Our workflow engine has parallel processing ability, supporting multi-applications at the same time. Besides, the engine has two remarkable characters: event-declaration and RDQL apprehension.

Event-declaration. This is not the simplex event-trigger mechanism. What we should do in FollowMe is just describing events in pvPDL files. All the rest of work such as registry and callback are done by the engine automatically. Moreover, if there are a sequence of events, we just list these events in the pvPDL file but needn't call back recursively, which should be done in simplex event-trigger systems for context-awareness. On the other hand, traditional workflow systems can not do this either. They use a condition judgement for state-transition constraints. This is a query mechanism and has poor performance.

In our system, both the application and the activities can be triggered by events. Meanwhile, state-transition constraints are reserved. Consequently, contexts can be pushed or pulled by the engine as needed, which simplifies application development.

RDQL comprehension. RDQL is a query language which can select matched RDF triples from a triple set. RDQL can describe complex information flexibly by composing triples. And our context information, including value, data type, relations among data types, time stamp and TTL, is a complicated set of data. So we choose RDQL to specify contexts in our system.

The workflow engine is RDQL-apprehensible. Developers can describe demanded contexts by giving an RDQL query sentence in pvPDL files. The events in applications are also represented by RDQL. The engine will parse them and get suitable contexts or register the events to the context service. Traditional workflow systems can not express contexts felicitously and, of course, can not deal with them.

5.3 Pervasive Process Definition Language

The pvPDL is a workflow definition language. We propose it to describe context-aware applications. The pvPDL is represented in the XML format and we definite the syntax of pvPDL in pvPDL-Schema. We will not lay out the schema for it is too long. We give a syntax checker to testify the validity of pvPDL files.

The pvPDL is context-describable and event-declarative. The top level element "Process" stands for a unique application. Within the element "DataFields", constant and variable are defined, which can be shared with all steps in the application. "Activities" is the aggregation of the element "Activity" which points to a specialized step of the application. Subelements of "Activity" direct some Process Unit performing the task and declare the parameters passing to it, actual and formal both acceptable. The elements "Process" and "Activity" both have the attribute "event", which gives the triggering event, represented in RDQL, of its parent element. As the subelement of "Transitions", "Transition" describes the paths between activities, while its attribute "condition" indicates the context constraints of the path in query mode. Here we pass over many other details of pvPDL.

If you feel writing a pvPDL file still tedious, you can use flow diagrams to describe your applications. What you should do is dragging and dropping the graphical elements

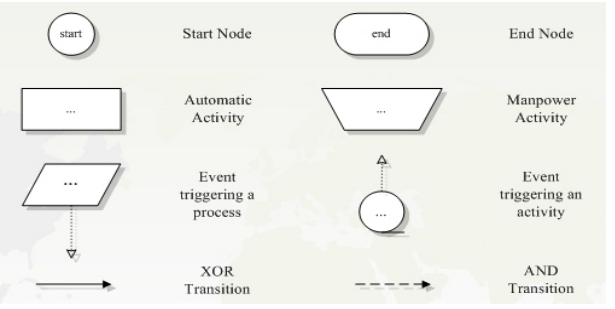


Fig. 2. Meaning of elements in flow diagrams

in FollowMe Application Builder. The diagrams you draw can be translated to pvPDL files automatically. Fig.2 shows the meaning of elements in flow diagrams, and we will give a scenario represented by the flow diagram in the next section.

6 Case Study

Scenario. Imagining you are a member of a research group, and you will give a lecture in a seminar. You should copy the slides to your flash disk, carry it to the meeting room, copy the slides to the computer in the meeting room, and then open them. The work is tedious and much attention is wasted.

In our ideal, the lecturer needn't do anything but editing the lecture notes. The latest version of the slides should be uploaded to the server. And it will be downloaded to the computer in the meeting room and opened when the lecturer enters the room and comes near the lectern. The work above is done automatically following the schedule contexts and user's position contexts. When a stranger comes into the meeting room, a warning will appear on the screen to prompt lecturer to take care of sensitive information. When the lecturer leaves, the slides close automatically.

Implementation. We develop an application named Seminar Assistant which is divided into two parts. One is called User Assistant and runs on the computer of the user. The other called Meeting Assistant runs on the computer in the meeting room. User Assistant uploads the slides when they are modified, while Meeting Assistant is responsible for opening and closing the slides automatically and warning the lecturer when a stranger comes in.

Here we explain Meeting Assistant in detail. Fig.3 shows the GUI of FollowMe Application Builder. The upper right quadrant of the window is the flow diagram of Meeting Assistant and the bottom of right is a segment of the corresponding pvPDL file translated automatically. As the figure shows, Meeting Assistant has four activities. The task of the first activity (marked with A1 in the figure) is opening the slides. It is activated by the speaker entering the meeting room and this time fitting with the calendar in the profile context. The third (A3) activity closes the slides and the fourth(A4) shows a warning. They are triggered by the leaving of lecturer and coming of a stranger respectively, depicted by the circle in the figure. The second activity(A2) is a virtual activity

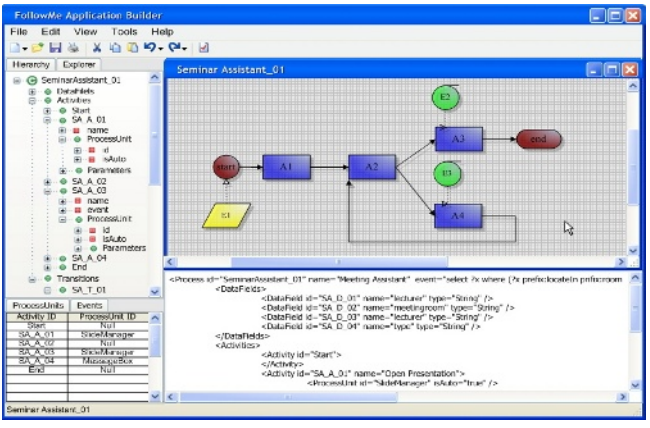


Fig. 3. GUI of FollowMe Application Builder. The upper right quadrant is the flow diagram of Meeting Assistant and the bottom of right shows a segment of the corresponding pvPDL file.

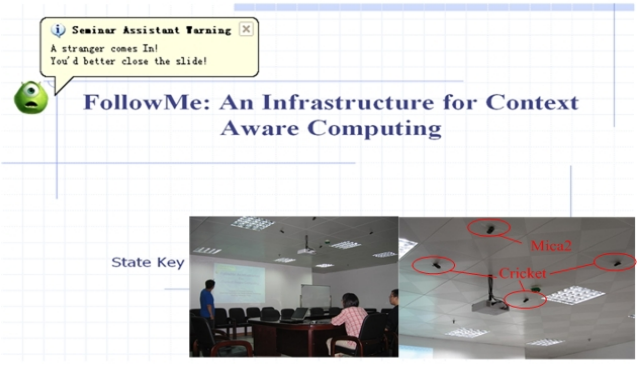


Fig. 4. Effect of Meeting Assistant when a stranger comes in. The bottom of right shows the smart environment.

for the need of flow transition. We develop Process Units "Slides Manager" shared by the first and third activities, and "Message Box" for the fourth. They are parameterized and can be reused in other applications.

Runtime effect. Fig.4 shows the effect of Meeting Assistant when a stranger comes into the meeting room during the seminar.

Other applications. Seminar Assistant is somewhat simple in application logic, which is convenient for us to illustrate the working mechanism of our system. Besides, we have developed several other applications, including Active Map and Security Manager, which embody the advantages of workflow for their more complicated logic.

7 Evaluations

Application development efficiency. We develop the Seminar Assistant in the ad-hoc method and workflow model respectively. By ad-hoc method, we spend about 10 man-days. We interact with a number of sensors and specify the program behaviors by the physical data get from them. Development of the application is complicated and we spend a lot of time in debugging and testing. When using the latter method, two persons use just one day to complete it. Later, we develop Active Map and Security Manager in the same way, and they cost 4 man-days respectively. In this case, bottom layer details are shielded and we needn't care about them. We only develop simple services and then draw a flow diagram to describe the business logic of applications. Data management, flow control, etc. are done by FollowMe automatically. So we can see the time cost is highly reduced.

FollowMe's performance. We test the time cost of Seminar Assistant to evaluate FollowMe's performance. In Fig.5, the first two lines are the time cost of Seminar Assistant developed in the ad-hoc method, and another two lines show the performance with FollowMe middleware. For each method, we collect two sets of data when the file size of slides is 50KB and 500KB respectively. We do this in order to find the effect of network traffic on FollowMe's performance. "Total time" stands for the latency between lecturer's coming and the opening of slides. And we check system time in each part of the program to get detailed information. The data in the form is the average of measured values and the unit is millisecond. The hardware configuration of the central node is 2-XEON-2.8GHz and 4GB-RAM. We also do the test for other applications; the time cost rates of each part are similar.

Time / Cost / Size / Method		Total Time Cost (ms)	Context Comparison and Storing	Context Reasoning Time	Context Filtering Time	Context Serialization Time	Rest of Time (Workflow & Network etc.)
Ad-Hoc	50KB	37	0	0	0	0	37
	500KB	135	0	0	0	0	135
FollowMe	50KB	940	340	146	106	255	93
	500KB	1050	341	148	105	265	191

Fig. 5. Performance evaluation of FollowMe and the workflow system

From the data we can see the application running on FollowMe is slower than the ad-hoc method. However, FollowMe brings about great advantages for context-aware computing and a latency of one second is acceptable to users. And we can see the main cost of FollowMe is context processing. FollowMe provides contexts for a number of applications simultaneously so that the context library is larger. This context library can support Active Map and Security Manager as well, while ad-hoc method can not. A little more time contributes to more abilities.

We also test the response time of our system when three applications mentioned above run simultaneously on FollowMe. The latency of a certain application rises linearly by the increase of the number of applications deployed on the framework. This result manifests that efficiency of this middleware is fine.

Workflow system performance. In Fig.5, the last column shows the time cost including the workflow system and network transport. It only takes a small part of the total time. This result attests that the workflow system does not depress FollowMe's performance obviously. Compared with its advantages, it's completely worthwhile introducing workflow to context-awareness.

And we will develop a distributed context processing mechanism and distributed workflow engines in the next stage of our work. It can be expected that the performance will be even better and time delay will be no trouble for users.

8 Conclusion and Future Work

FollowMe is a service-oriented, pluggable and programmable context-aware middleware. It supports workflow based applications and we propose a new workflow definition language pvPDL for context-awareness. The case study and evaluations show that workflow simplified context-aware application development and has good performance. In the near future we will standardize the service called by workflow engine and work towards a distributed context processing mechanism to improve efficiency and robustness.

References

1. Weiser M.: The Computer for the 21st Century. In: Scientific American, September 1991. (1991)94-100
2. RDF Data Query Language. http://www.w3.org.
3. Open Service Gateway initiative. http://www.osgi.org.
4. A. K. Dey, et al.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Anchor article of a special issue on Context-Aware Computing. In: Human-Computer Interaction (HCI) Journal, 16(2-4). (2001)97-166
5. J. Hong and J. Landay.: An infrastructure approach to context-aware computing. In: Human-Computer Interaction (HCI) Journal, 16(2-4). (2001)287-303
6. Lonnie Harvel, Ling Liu, Gregory D. Abowd, et al.: Context Cube: Flexible and Effective Manipulation of Sensed Context Data. In Proceedings of the Second International Conference on Pervasive Computing, April 2004, Linz/Vienna, Austria. Springer-Verlag, (2004) 51-68
7. Wolfgang Beer, et al.: Modeling Context-Aware Behavior by Interpreted ECA Rules. In Proceedings of Parallel Processing, 9th International Euro-Par Conference, August 2003. Springer, Klagenfurt, Austria (2003)1064-1073
8. Jose J. Alferes, et al.: Computing Environment-Aware Agent Behaviours with Logic Program Updates. In Proceedings of Logic Based Program Synthesis and Transformation, 11th International Workshop, November 2001. Springer, Paphos, Cyprus (2001)216-232
9. XML Process Definition Language. http://www.wfmc.org.
10. Business Process Modeling Language. http://www.bpmi.org.
11. The Workflow Reference Model (WFMC-TC00-1003 Issue 1.1). WfMC (1995)
12. Sensors produced by Crossbow. http://www.xbow.com.
13. Jun Li, YingYi Bu, Shaxun Chen, Xianping Tao, Jian Lu.: FollowMe: A Pluggable Infrastructure for Context-Awareness. In Ubicomp2005, Tokyo, Japan (2005)
14. Joohyun Han, et al.: Context-aware Workflow Language based on Web Services for Ubiquitous Computing. In LNCS3481 - ICCSA2005. Springer (2005)1008-1017

Service Rendering Middleware (SRM) Based on the Intelligent LOD Algorithm

Hakran Kim[1], Yongik Yoon[2], and Hwajin Park[2]

[1] Department of Computer Science, [2]Multimedia Science
Sookmyung Women's University
Chungpa-Dong 2-Ga, Yongsan-Gu, 140-742, Seoul, Korea
{imhera, yiyoon, hwajinpk}@sookmyung.ac.kr

Abstract. In ubiquitous computing environments, the need of contents adaptation and delivery is growing to support multiple target platforms for single source. Since 2D/3D graphics contents deal with a large data set and a high performance, a service adaptation for context changing is required to manipulate graphics contents with a more complicated method in multiple devices such as desktops, laptops, PDAs, mobile phones, etc. In this paper, we suggest a new notion of service adaptation middleware based on service rendering algorithm, which provides a flexible and customized service for user-centric 2D/3D graphics contents. The service adaptation middleware consists of *Service Adaptation* (SA) for analyzing environments, *Service Rendering* (SR) for reconfiguring customized services and *Intelligent LOD Algorithm* for generating an adapted service by processing customized data according to the level of detail. These adaptation services are able to intelligently and dynamically support the same computer graphics contents with good quality, when user environments are changed.

Keywords: graphics contents adaptation, service adaptation, service rendering, middleware, level of detail.

1 Introduction

As wired/wireless integrated environments improve, there have been many attempts to provide extensible quality services over various networks and through devices [1,3,6]. At the beginning, service adaptation techniques for text contents were developed. DIA(digital item adaptation) techniques for audio video information using MPEG4, MPEG7 and MPEG21 have been studied in recent years[6,9].

Nowadays, graphics contents such as games, avatars, etc. have attracted a lot of attention. But, a graphics contents service adaptation has not been considered yet because 2D/3D graphics contents deal with a large data set and need high performance. Therefore, the graphics contents in ubiquitous computing environments should be served with the best quality to clients who use a wide range of devices and are in various environments [5]. In addition, services should be adapted to clients' statements at sub platforms, and the adaptation of all aspects in a middleware system from the application level to the operating system level is required.

Though the application will be able to recognize changes in the execution environment, it is difficult to adapt the changed situation to its middleware. Therefore,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 776 – 785, 2006.

in order to provide solutions in various situations, it is efficient to manage an intelligence adaptation mechanism in the middleware layer. However, the methods referred previously lack in providing the adapted service in real-time from the middleware when environments change.

Therefore, the goal of this research is to suggest an adaptation method that provides an optimal and identical service customized to clients' environments. We suggest a new notion of adaptive service in middleware, which provides a flexible and customized service for user-centered 2D/3D graphics contents. This adaptive service consists of Service Adaptation (SA), Service Rendering (SR) and Intelligent LOD Algorithm. Service Adaptation (SA) is for analyzing, filtering, and triggering clients' environments. Service Rendering (SR) is for replacing current graphics configurations with changed statuses whenever SA is asked. With results from SR, customized reconfigurations of clients' statuses, Intelligent LOD Algorithm is for generating an adapted service by processing the customized data according to the level of detail.

Since 2D/3D graphics deal with a large data set and need high performance, it is very limited in devices with low system requirements such as PDAs, and mobile phones. To solve this problem, we reduce the original vertex data set into multiple levels of vertex data set and process texture images in a similar way. This will be explained in detail in section 3 and 4.

This paper is organized as follows. Section 2 explains related works of the service adaptation for graphics contents and motivations, and section 3 describes a concept of SR middleware with SA module, SR module, and Intelligent LOD Algorithm module to provide optimized services to user centered environments. Section 4 proposes the Intelligent LOD Algorithm module, Section 5 an example and analyzes it using a simple example, and Section 6 concludes our research results and discusses future works.

2 Motivation and Background

In the past few years, needs for service adaptation or service re-purposing have emerged. The most notable facets of service adaptation are adaptation of a navigation model in user interfaces, contents personalization and media adaptation. 2D/3D computer graphics are one of the cornerstones of multimedia contents. Therefore, we focus on SR for graphics contents adaptation among quite a few technologies to provide scalable, seamless multimedia services.

In many cases, devices access the same contents from the same device provider (providing virtual maps/guides, multi-user games, etc) and it is this broadness of contents and the heterogeneity of devices (in terms of performance, capability, network connection, etc) that are the main concerns in a market that is continuously expanding. It is also a concern of users to obtain the best quality for their devices, i.e. to meet general expectations that overall quality of experience will be better when they use any higher performance devices [2]. The paper [2] proposes multi-resolution meshes for a multiple targets, single content adaptation within a MPEG-21 framework for Graphics DIA (Digital Item Adaptation). Multi-resolution models are generated in VRML and are based on benchmarks for the referred devices. These mesh data are defined in advance and stored for adaptation service. But, this paper does not mention

which layer handles the adaptation service and how it does. It means that research on interoperability, one consideration for graphics DIA, should be studied. Because computer graphics contents imply different capabilities including display size, resolution, memory, etc. for each device, an effective result is to be served for a customized and identical service based on user centered Service Oriented Architecture in the middleware layer [7].

3 Service Rendering Middleware (SRM) for 2D/3D Computer Graphics Contents

3.1 Model of SRM

Generally, an adaptation considers all factors that may change quality of services. A usage environment description includes network characteristics, user environment capability, user preferences, etc. In this research, we focus on the most critical factor, user environment capability, i.e. device memory, device display resolution, and device location. We suggest a new concept of adaptive service in middleware, which provides a flexible and customized service for user-centered 2D/3D graphics contents. This is called Service Rendering Middleware (SRM). This adaptive service consists of three modules for adaptation i.e. Service Adaptation (SA), Service Rendering (SR) and Intelligent LOD Algorithm and two modules, service request module for requests and service delivery module for delivery. The concept of SRM is depicted in figure 1.

Each module performs the following processes. Service request module is for accepting inputs from sensors and requesting SA. *Service Adaptation* (SA) is for analyzing, filtering, and triggering clients' environments. *Service Rendering* (SR) is for replacing current graphics configurations with the changed status. *Intelligent LOD Algorithm* is for generating an adapted service by processing customized data according to the level of detail. Service Delivery module is for delivery of the adapted service.

We will explain the adaptation process from service to delivery with an example scenario. Suppose that a user is playing with an avatar on a desktop in a wired or wireless web environment, and he wants to continue to play with the same avatar on his mobile phone after a while. First, a sensor connected to networks recognizes that a user wants to use the same service on a mobile phone. The service request module accepts changing signals from the sensor and informs SA that his device has been changed from a desktop to a mobile phone. Then the SA module analyzes his mobile phone status, and decides whether the service will be provided or not. If SA decides positively, the result and user environment information including the previous and current statuses are transferred to SR module. The SR module sets up a configuration table describing the service adaptation and current status i.e. a mobile phone, an application program, graphic libraries, header files, methods, and resources. The reconfigured table is transferred to Intelligent LOD Algorithm module by SR based on the customized reconfiguration of current status. The Intelligent LOD algorithm gathers suitable programs and the down-sized avatar data from database and generates the final adapted service according to the level of detail. The final adapted service is delivered to him through the service delivery module.

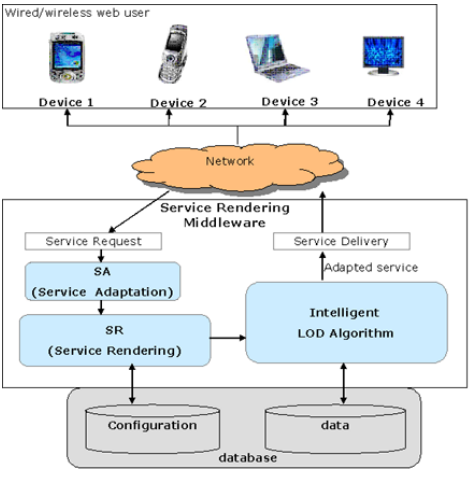


Fig. 1. The Concept of Service Rendering Middleware

3.2 Configuration of Service Rendering (SR)

In this section, we will discuss service sets for an adaptation. A configuration and data necessary to SR and Intelligent LOD Algorithm are stored in a database. Factors for the computer graphics contents service adaptation are as follows. They are device, application program, graphic library, header file, methods and resources necessary to each configuration.

The adaptation service based on user centered service oriented architecture is defined as Si. Elements of S are Device, Application Program, Graphic Library, Header file, Methods and Resources. All elements included in S are defined in the level of detail method, and the set S can be shown as follows;

$$S_i = \{ D_i , A_i , L_i , H_i , M_i, R_i \} , 1 <= i <= n ,$$

n= number of devices

D : Device set, A: Application Program set, L: Graphic Library set

H: Header files, M: Method set, R: Resource set

Configurations of each element that composes the service set S are predicated as follows; First, the device set defined as *Di* includes device name *(dn),* device location *(dl),* memory *(dm),* and display resolution *(dr).*

$$D_i = \{dn, dl, dm_i, dr_i\}, 1 <= i <= n$$

Second, the application program is an execution file made in advance using the SDK which corresponds with user interface environments, and its set is defined as *Ai.* The set *A* includes graphic library *(al)*, header file *(ah),* and method *(am).*

$$Ai = \{ al_i, ah_i, \sum am_i \}, \quad 1 <= i <= n$$

Cases of methods and the number of methods are consequently decided by quality and contents of graphics, which should be created in the application program. The graphic library inside the application program is a module, which is a set of specific

functions. The functions will be able to be used to produce graphics contents that fit to each user environment in the application program.

We define the graphic library as *Li,* and it includes macro *(lm),* preprocessor *(lp),* and methods *(lf).*

$$Li = \{\sum lm, \sum lp, \sum lf\},\ 1 <= i <= n$$

The number of macros, preprocessors and methods could be changed; it will follow graphic library types. Generally, the header file inside the application program is composed of a system library and user functions. We will define it as *Hi,* and it includes macro (hm), type definition (htd) and function (hpf).

$$Hi = \{\sum hm, \sum htd, \sum hpf\},\ 1 <= i <= n$$

The number of macros, type definitions and functions could be changed; it will follow system types. The method includes a lot of modeling and rendering processes that imply lighting, ray tracing, shading and texture mapping, etc. But these processes are limited in some devices such as mobile phones and PDAs, therefore, multiple levels of method are needed to create suitable computer graphics contents for each user environment. The Method set defined as *Mi* includes method number *(mn),* modeling part *(dl),* and rendering part *(dm).*

$$Mi = \{mn, mm, mr\},\ 1 <= i <= n$$

If the resources are used in the Methods, textures images, audio files, video files etc. from light weight to heavy weight ones are offered to the application program. The reason why the resources of multiple levels are necessary follows user environments and it is because of the limitation of resource capacity. The resource set defined as *Ri* includes resource number *(rn),* resource size *(rs),* and resource capacity *(rc).*

$$Ri = \{rn, rs, rc\},\ 1 <= i <= n$$

The sets presented above are necessary for configuration forming and data fetching from a database through the Intelligent LOD Algorithm and making a suitable decision to provide a flexible service in each user environment.

4 Adaptation Algorithm

When a user environment is changed, it cannot be guaranteed that computer graphics contents maintain the same quality of service as before using identical application programs with graphic libraries, header files, and resources. The reason is that each user device carries different capabilities including display size, resolution, memory, etc. To generate an adapted contents service in real time in the current user environment, application programs need to be defined in advance. Explained more concretely, each application program is classified into several levels that fit into some benchmarked environments and is regenerated to be stored at each level. Graphic libraries, header files, methods and resources follow the same process. In this section, we will describe a procedure of SR and Intelligent LOD Algorithm for adaptation.

4.1 Service Rendering

The SR module should organize configurations of the current services for flexible user-centered services. Set the current user device as *device i* and the current

application program as *application program i*, and *i* is from 1 to *n,* and *n* means the number of devices needed. First, *device i* configuration of the current status is fetched from a database and *application program i* is reconfigured corresponding to the level of *device i*. The reconfiguration of *application program i* produces new tables that have listings of suitable graphic libraries, header files, methods and resources. A Reconfiguration model in the SR module for the adapted service is depicted in Figure 2.

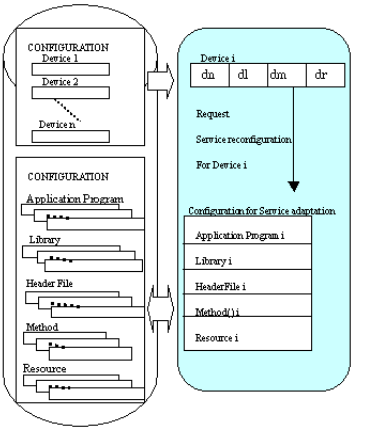


Fig. 2. Service Rendering Module

4.2 Intelligent LOD Algorithm

The main role of Intelligent LOD Algorithm is to generate an adaptation service that satisfies with a reconfiguration created from SR by intelligently and dynamically adapting contents to end device constraints (location, memory, and display resolution).

For example, if a configuration inputted from SR should provide an adapted service for an animation created with openGL in a PDA, in order to create the adapted service, first, the integral part of the application program i suitable in the PDA is decided. In the next step, if the operating system of the platform is using wince,

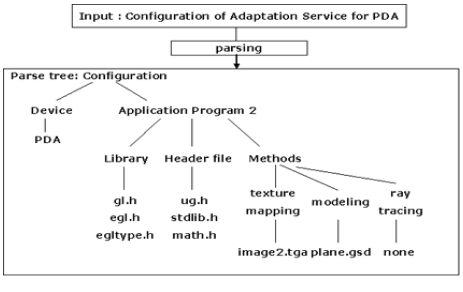


Fig. 3. An example decision tree for an adaptation service for PDA

libraries which are necessary for the application program will be gl.h, egl.h, and egltype.h from the libraries which suit to this system and the case of header files decided identically will be ug.h, stdlib.h, and math.h. At last, for the methods including modeling process, texture mapping process and rendering process such as ray tracing, etc., image2.tga for texture mapping, and a mesh file, plane.gsd are selected. The following Figure 3 is showing an example decision tree using an inference mechanism in AI.

The general procedure for adapted service is as follows.

Step 1: Requesting procedure
At this step, the table S is declared and initialized with the configuration data from SR.
Step 2: Fetching procedure
At this step, fetching procedure occurs.
It is where the user environment is exchanged. Application program data, graphic library, header file, and resources should be fetched from the database for a service that fits to the configuration from SR. Like the example above, these processes base on an inference mechanism in AI.
Step 3: Constructing adapted service procedure
The adapted service is constructed with data from Step 2 and delivered to the service delivery module.

The figure 4 shows the control process of Intelligent LOD Algorithm and interrelation of the configuration and data in the database, and intelligent LOD Algorithm is shown in Algorithm 1 below.

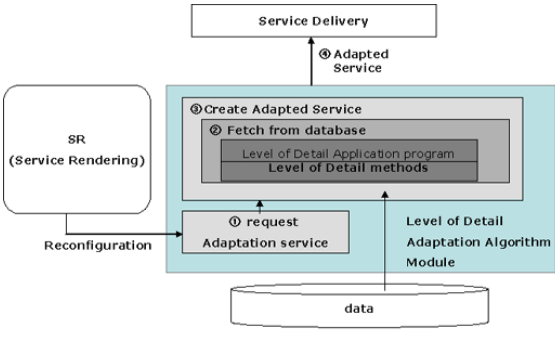


Fig. 4. Intelligent LOD Algorithm Module

<Algorithm 1> Intelligent LOD Algorithm

STEP 1: **Requesting procedure**
Adaptation_Service *S (D, A, L, H, T)* S<-(D1,A1,L1,H1,T1) *// default service comes up to first device environment*

```
//D:device, A:application program, L:library, H:header file, T:resources
User_Demand_On
User-environment_state <- sensor
S' <- from SR;
```

STEP 2: **Fetching procedure**

```
while j = 0; j < number of S' set's elements
for (i = 1; i <= n; i++)  // n is number of devices
 { if A,L, H, and M is in S' set
   A' <- Choose the suitable application program A;
   L' <- Choose the suitable graphic libraries L;
   H' <- Choose the suitable header files H;
  M' <- Choose the suitable methods M;
   If the methods need resources
     T' <- Choose the suitable resources T;
 } return A', L', H', M', T';
```

STEP 3 : **Constructing adapted service procedure**

```
S' = A' + L'+ H'+ M'+ T';
S <- S';
  Return S;
```

5 An Example and Analysis

The Adaptation service of computer graphics contents is necessary to remake an application program with multi-level applications providing the same service and various sizes fitting to multi-level capability of devices. For instances, the number of polygons allowed in a 3D graphics application is very limited on PDAs or mobile phones, while over 1000000 polygons are acceptable in the same application on a desk top. Advanced graphics techniques such as ray tracing, radiosity, and particles do not work in low-performance user environments such as PDAs or mobile phones. Texture mapping, one of effective rendering methods, does not work either if the size of a texture image is large.

In this case, a texture image file can be reduced to an appropriately sized file to suit to device capability.

Such texture image can be reduced to several files with various sizes such as 10k for a mobile phone, 100k for a PDA, 1M for a desktop, etc. Therefore, the level of detail of an application is important for graphics contents adaptation service appropriate to levels of each device environment. This research considers two methods for reducing and combines them to classify an application into multi-level of detail. The followings are the two methods in modeling and rendering for the level of detail.

The first method is the level of detail vertex. Vertexes proper to end device capability are created in advance from the highest detail to the lowest detail level divided in some phases. The second is the level of detail texture image method. If

computer graphics contents executed in each device need texture mappings to a model, the texture image's size is considered in the device capability.

We show an example program for a PDA emulator using openGL-ES, which was downloaded from a site [14]. The example is running under WinCE 3.0 with 16MB RAM with a monitor resolution of 240*320 in the PDA emulator environment. The resources, which are used in our example program are composed with 2 targa files, a raw file for texture mapping and a gsd file for meshes. The targa files are 49k bytes with 256 x 96 resolution, 24k bytes with 64 x 64 resolution and and 64k bytes with 128 x 128 resolution the raw file is 32k bytes with 128 x 128 resolution. (a) in Figure 5 shows processing capability using 2 targa files, one of them is 64k bytes, a raw file and a mesh file of 102KB with 15FPS(frame per second). On the other hand, (b) in figure 5 shows 24FPS processing capability using the same mesh file, 2 targa files, one of them is 49k bytes and raw file as (a). (c) in Figure 5 shows 9FPS processing capability using the same targa files and a raw files as (b), and a mesh file of 468 bytes. As shown in the execution screen shots of the PDA emulator in figure 5, the processing power to deal with the animation falls in the case that the application program controls more vertices and lager texture image.

Generally, a desktop environment equipped with window XP and 512MB RAM and a monitor with maximum resolution of 1280 x 1024, has higher support and processing capability. If user environments are more powerful high-level interfaces as desktops, an animation could be shown more velocity and high resolution with bigger sizes of resources, texture images and mesh files than PDAs.

Fig. 5. Result images in a PDA emulator: (a) the image using a mesh file of 102KB and a 64k bytes texture image, (b) the image using a mesh file of 102KB and a 49k bytes texture image, (c) the image using a mesh file of 468 bytes and, (d) a 64k bytes texture image, (e) a 49k bytes texture image

6 Conclusions and Future Work

Taking high performance computer graphics contents that were mainly available to high-end device users and reprocessing them to make them fit to users' environments, and converting them to suit a broader field of users' demands and service environments made an entrance as the essential subject of this paper. Thereupon we

propose the Service Rendering Middleware (SRM) that includes Intelligent LOD Algorithm and SR modules for a 2D/3D computer graphics contents adaptation method in ubiquitous computing environments. The SRM uses reflection to dynamically adapt to context changes, which takes care of the weaknesses inherent in existing middleware platforms.

In our research, we show an example of applying two levels of mesh files and analyzed it. In the near future, we will propose an efficient algorithm for the level of detail vertex and texture image and considerate network capability and user preference as well as device capability for computer graphics contents adaptation.

References

1. Korva J, Plomp J, Määttä P , Metso M : On-line Service Adaptation for Mobile and Fixed Terminal Devices, Proceedings of the Mobile Data Management '01 (2001) 252-259
2. H. Kim, C. Joslin, T. Di Giacomo, S. Garchery, N. Magnenat-Thalmann : Multi-resolution Meshes for Multiple Target, Single Content Adaptation within the MPEG-21 Framework, IEEE ICME Conference (2004) 1699-1702
3. P. Gioia, A. Cotarmanac'h, K. Kamyab, P. Goulev, E. Mamdani, I. Wolf, A. Graffunder, G. Panis, A. Hutter, A. Difino, B. Negro, M. Kimiaei, C. Concolato, J. Dufourd, T. Di Giacomo, C. Joslin, N. Magnenat-Thalmann : ISIS: Intelligent Scalability for Interoperable Services. IEE CVMP (2004) 295-304
4. Peter Soetens, Matthias De Geyter : MultiStep Media Adaptatio: Implementation of a Knowledge Based Engine. the 14th international conference on World Wide Web (2005) 986 - 987.
5. Alexandre Kotarmanac'h, Renaud Cazoulat : Architecture for multimedia content adaptation delivery over heterogeneous environments. Broadband Europe, session 10. Paper 10-04 (2004)
6. Anthony Vetro, Christian Timmerer : Digital Item Adaptation: Overview of Standardization and Research Activities. IEEE Transactions on Multimedia, VOL. 7, NO. 3 (2005) 418-426
7. Yiqun Hu, Liang-Tien Chia, Deepu Rajan : Region-of-Interest based Image Resolution Adaptation for MPEG-21 Digital Item. Proceedings of the 12th annual ACM international conference (2004) 340 – 343
8. Mulugeta Libsie, Harald Kosch : Content Adaptation of Multimedia Delivery and Indexing using MPEG-7. Proceedings of the tenth ACM international conference (2002) 644-646
9. Metso M, Koivisto A , Sauvola J : Content model for mobile adaptation of multimedia information. Journal of VLSI Signal Processing 29 (2001) 115-128
10. Eric Lengyel : Mathematics for 3D Game Programming and Computer Graphics 2nd Edition, CHRLES RIVER MEDIA, INC.
11. http://www.typhoonlabs.com/index.php?action=developer.htm

Jini-Based Ubiquitous Computing Middleware Supporting Event and Context Management Services*

Seungyong Lee[1], Younglok Lee[1], and Hyunghyo Lee[2,**]

[1] Dept. of Information Security, Chonnam National University, Gwangju, Korea 500-757
birch@lsrc.jnu.ac.kr, dogu@jnu.ac.kr
[2] Div. of Information and EC, Wonkwang University, Iksan, Korea 570–749
hlee@wonkwang.ac.kr

Abstract. The key feature of ubiquitous computing services or applications is that they should be highly adaptive to events and context information. These factors are essential in ubiquitous computing environments. The services and applications must communicate with each other through fixed or ad-hoc networks. In ubiquitous computing, the event and context managers must be provided at the middleware level, for convenient development of associated applications. In this paper, ubiquitous computing middleware supporting event and context management services are proposed. The JavaSpaces service in Jini network technology is proposed to modify and develop the event manager, because JavaSpaces contains various interfaces that can be used to implement event management services, such as write, read and notify. Due to excellent GUI support, Macromedia Flash was used to represent the virtual ubiquitous computing environment, with communications through XMLSocket. It is demonstrated that the developed event and context managers can make it straightforward to efficiently develop ubiquitous computing applications.

1 Introduction

In ubiquitous computing environments, events from entities continuously occur, these events trigger other new actions or operate applications, and the environmental context information of entities always changes. The applications react differently, according to the environmental status and context information. An event can be defined as an object that contains information regarding the external status, which is of interest to other software components. Various contexts information is generated, from low level signals, to valuable high level information. Users or applications in the ubiquitous computing environment should be able to adapt to current environmental information [1,2,3].

In ubiquitous environments, most applications and sensors produce or consume events and context information, and event producers and consumers interact in an

* This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).
** Correspondent author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 786 – 795, 2006.

event-driven mode by transmitting and receiving events. The various events and context information from sensors and devices continuously changes. Therefore, it is necessary to systematically manage events and context information in the ubiquitous computing environment, at a middleware level [10]. In this paper, reference is made to modules managing events and context information as *event manager* and *context manager*. In this paper, developing an event manager and context manager with JavaSpaces and Macromedia Flash technology, which includes registration and lookup services, is proposed.

The remainder of this paper is structured as follows. Chapter 2 presents related work. Chapter 3 presents the design and implementation of the proposed event management system based on JavaSpaces and a middleware context manager. In chapter 4, the degree of effectiveness explains the virtual ubiquitous computing applications and devices that can be developed with ubiquitous computing middleware. Finally, a conclusion is made and further research is presented in chapter 5.

2 Related Work

The CORBA Event Service standardized the transfer of asynchronous notifications between objects [7]. An object generating events is called supplier and an object receiving events is called the consumer. A supplier passes an event to a consumer by invoking an appropriate method on the consumer interface. Suppliers and consumers can be decoupled from each other using event channel objects. The event channel forwards all events received from the suppliers to all consumers that have been registered in the channel. The standardized CORBA notification service developed by the OMG Telecommunications Domain Task Force provides much more flexible event filtering capabilities [8].

One of the ubiquitous middlewares is the Gaia of the University of Illinois under active research. The event manager of Gaia satisfies many general needs of event management [3]. It creates event channel factories remotely on preconfigured machines whenever it decides that the existing event channel factories are already overloaded. In essence, the event manager maintains the state for channels it creates and recreates them if they are crashed or transferred to the event consumers.

The Jini technology, an open architecture for the home network middleware, assists new service components in being connected to the home network over time and assists clients in using them promptly without extra settings. In case of service component upgrades, the existing client services can operate without problems [4, 6]. If the client has interesting external changes, it is necessary for the changes to be asynchronously notified, regardless of whether Jini services are local. Jini technology can notify of changes using the event notification concept used in other Java components. The Jini lookup service enables applications to search for services that the client requests. The client can be registered with the lookup service as a remote event listener so that the client can be notified.

In distribution applications, the JavaSpaces technology acts as a virtual space between providers and requesters of network resources or objects [8,9]. A function of

the JavaSpaces service is to notify entities interested in event objects when the event objects are stored in JavaSpaces. Therefore, a notification service can be implemented using JavaSpaces. The JavaSpaces service included in JINI technology enables clients to share objects. Its goal is to facilitate cooperative distributed computing. It is a reliable distributed storage system. JavaSpaces should be thought of as a "place" where it is possible to store and share information in the form of objects. The objects in JavaSpaces flow around space-based systems, allowing clients to share and store information and, share behaviors via the use of dynamic class loading. In JavaSpaces, an object which is to be an entry must also implement entry interfaces.

3 A Ubiquitous Computing Middleware Supporting the Event Management and Context Management

The different events and contexts are essential to ubiquitous computing environments. Therefore, the events and contexts must be separately managed in the ubiquitous computing environment. In this chapter, the proposed ubiquitous computing middleware, supporting the *event manager* and *context manager* is described. First, the difference between an *event* and *context* are described as follows.

Event: Occurrence of an interesting action and change of states or values. If an event occurs and is published, applications react to the event and perform various actions or operations. The reactive behaviors of the applications are triggered by the event. The event plays the role of a trigger for applications.

Context: Situational Information representing physical states such as location and time or attribute values of an entity such as location of the entity and status of printer. An event occurs from changes in these states.

3.1 A Ubiquitous Computing Middleware

All applications, devices, and sensors in the ubiquitous environment continuously changes and must be configured and adaptive to the changes to make interactions richer. These occurring changes which can be referred to an *event* include movements of entities, executions of applications, sensing, and contextual changes. Efficient management of the events is indispensable to ubiquitous computing environments for applications to be sufficiently adaptive to events. The event manager provides useful services such as registering events, receiving events from devices or sensors, and notifying applications of event occurrence. With the management of various events in ubiquitous computing middleware, it is required to store and retrieve context information. Applications in the ubiquitous computing environment normally use context information to decide the next behaviors. The context manger stores context information from applications, sensors, devices, or entities to its repository and responds to queries from applications for particular context information when it is requested. Due

to these reasons, the event manager and context manager are requisites for ubiquitous computing environments. The event management service and context management service must be supported at a middleware level to be commonly used by context-aware applications, devices, and sensors with convenience and flexibility.

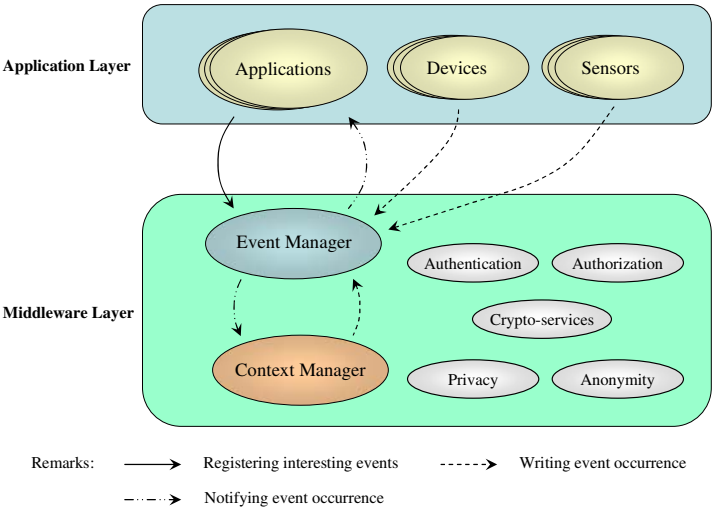


Fig. 1. Layout of the Relationship between Applications and the Managers

Figure 1 describes the layout of the relationship and communications among applications, event manager and context manager. Applications register themselves with an event manager to receive their interesting events and all events from applications, devices and sensors are written to an event manager. The event manager notifies applications or context managers of occurring events. Many other security related services must be supported in middleware, such as a privacy protection service, authentication service and authorization service, which are beyond the scope of this present paper.

3.2 JavaSpaces-Based Event Manager

JavaSpaces within Jini technology is a space where Java objects can be stored and accessed as a distributed computing model. It has been designed to assist application developers solve two related issues: distributed persistence and the design of distributed algorithms. The JavaSpaces service uses RMI and the serialization feature of the Java programming language to accomplish these goals. JavaSpaces has interfaces for *write*, *read*, and *notify*, which are necessary to implement the event management service. JavaSpaces is used and modified to implement event management service, because of the benefits of JavaSpaces and the requirements for the event manager. Applications write an event object to an event manager to transmit an event and the event manager notifies of the occurrence of the event to applications interested in the event.

Applications or devices in a ubiquitous computing environment produce and publish various events and consume the occurring events. Applications or devices which produce various events are referred to as *event producers*. Applications which use events are *event consumers,* because the occurring events are consumed. However, in order to effectively use events without effort, the event producers must write or transmit events to the event manager and allow the event manager to notify all the applications, which have interest in the event, of the occurrence of the events.

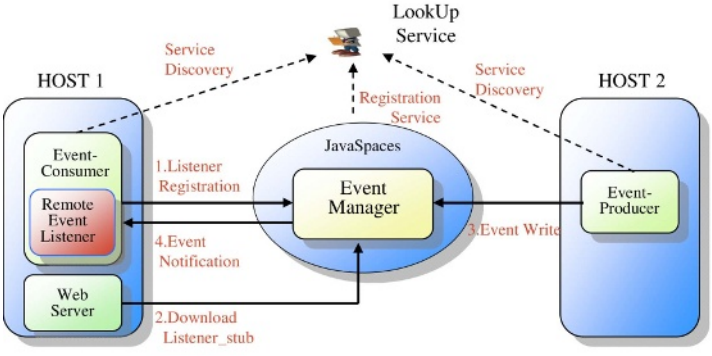


Fig. 2. Procedure of Event Service between Event Consumer and Event Producer

The procedure of the event management service is illustrated in figure 2. The event producer can transmit events to the event consumers through the event manager.

As can be seen in the above procedure, the event manager registers itself to provide event management service using the JINI registration service. The event consumer and producer can find the event management service using the JINI lookup service.

The event manager uses channel to transmit events to all event consumers. An event is kept in an event channel and is transmitted to all the consumers waiting for the events.

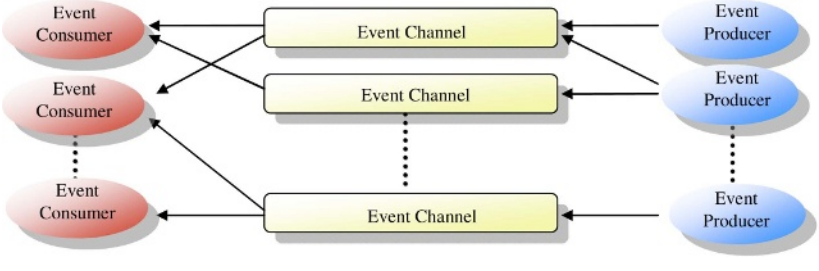


Fig. 3. Events Distribution through Event Channels

Figure 3 illustrates the method at which events for applications are registered in channels and the method at which the event manager distributes the events to actual applications. All event consumers register events of interest to the event manager, in order to receive notifications of occurring events. The event manager creates one

channel for one event type and transmits the event notification to all applications waiting for the occurrence of the event by registering themselves to the channels of the event type. The event channels are created with respect to each event type. If an event occurs by an event producer, the event producer writes and inputs the event to event channels created with respect to the event type. After inserting the event into the specific event channel, notifying of the occurrence of the event, all of the event consumers reconfigure themselves according to the event.

3.3 Context Manager

In ubiquitous computing environments, various context information exists, and can be used to characterize the situation of an entity such a person, place or object[5]. *Context* differs from *event* in many respects, because a context is simply information describing states or values, such as location and temperature, while an event is a description of actions or changes. A *context manager* has been developed because it is necessary to effectively manage the context information at a middleware level for developing ubiquitous computing applications. The context manager aggregates context information from sensors or devices, stores it into context, and may convert it into high level context information, depending on inference rules. The context manager queries requests from event managers and replies to the requester with context information. This reply corresponds to an event for applications or devices waiting for the event. The context manager always examines if rules for occurrence of events are satisfied. If rules are satisfied, it generates an event to the event manager.

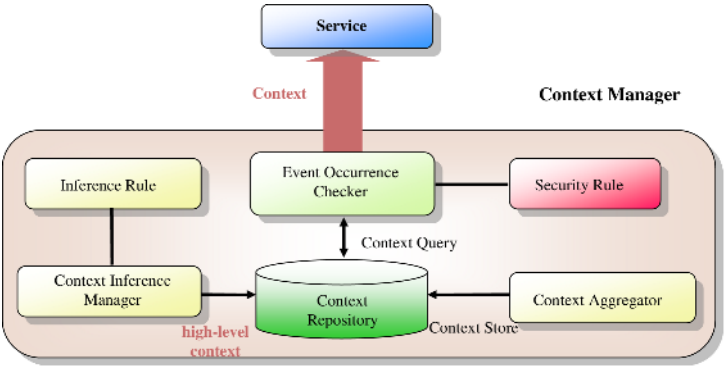


Fig. 4. Architecture of the Context Manager

Figure 4 describes the detailed architecture of the context manager. The context aggregator in figure 4 aggregates physical raw data or signals from sensors and transforms it into the context information stored in the context repository. The context used in the manager consists of context name, context type, subject, object, the occurring time, and so on. The subject of the format is referred to as an owner of the context and the object is referred to as the value of the context. The context manger also has a context inference function that the context manager can use to infer new high-level context from existing context.

4 Prototype Implementation

The prototype of the services in ubiquitous computing middleware has been implemented in the Linux/Windows OS, JDK 1.3 and JINI 1.2 development environment. The event manager has been implemented by utilizing and modifying JavaSpaces services, as described in chapter 3. In order to implement the prototype, first a ubiquitous scenario is introduced, and the overall architecture is presented. Then, virtual applications appearing in the scenario are implemented with Macromedia Flash, suitable for representing the virtual ubiquitous computing environment and initiating communications using XMLSocket.

4.1 A Scenario for the Implementation

A ubiquitous computing environment scenario is presented, for describing the usability of the proposed management services and developing prototype implementation. The scenario is as follows.

> *6 PM, July 10, 2005. Bob heads to home after finishing his daily work. On his way home he stops by a fruit store. When he arrives at his home, the porch light automatically turns on, and he enters after authentication. He changes his clothing in his room, eats dinner, then sits down on the sofa. The context manager which recognizes that Bob sits down on the sofa, transmits his favorite channel information to the TV based on his preferences, and turns on the TV. During watching TV, he recognizes that he has left his wallet in the fruit store. So he leaves his house in a hurry with the TV on. The TV stores the current TV status and turns itself off after receiving an event from the context manager indicating that Bob has left. When he comes back and sits on the sofa, the TV turns on with the TV program which he saw prior to leaving.*

Based on the above scenario, a virtual ubiquitous computing environment is composed, which demonstrates the interactions between the TV and the sensors using the developed event and context managers. In this prototype development, two partial scenarios in the above scenario are considered. Scenario 1: the light automatically turns on when a member of the family enters into the living room. Scenario 2: when the person sits down on the sofa, the TV automatically turns on with the person's favorite channel if sensing that the person sits down to watch TV.

4.2 Overall Architecture of the Prototype for the Scenario

For describing prototype implementation of the scenario, the overall architecture is introduced, as presented in Figure 5. There is a TV application, light application, location sensor, light sensor and pressure sensor in this virtual environment. The dotted lines in the figure demonstrate that the event producers such as the pressure and light sensors write the events to the event manager and the broken dotted lines mean that the event manager notifies the event consumers of occurrence of the events. The TV and light applications register themselves to the event manager in order to listen to events. The sensors write sensing information to the event manager, and the

event manager notifies the TV application, light application and the context manager through event channel 2. The sensor devices are referred to as an event producer and the TV and context manager are referred to as an event consumer. Channel 1 in figure 5 is for checking the occurrence of another event type. The event checker generates an event if a specific rule is satisfied for the event.

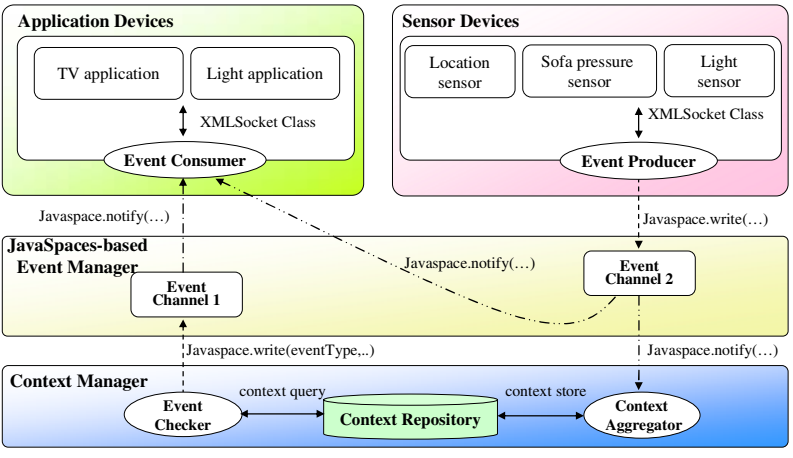


Fig. 5. Overall Architecture of Scenario Prototype Implementation

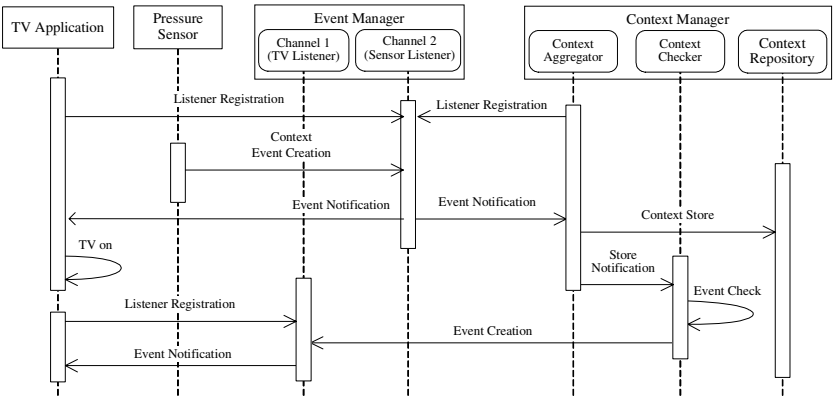


Fig. 6. Scenario Communication Sequence Diagram

The diagram presented in figure 6 depicts the sequence of interactions among managers, applications and sensors. The pressure sensor, which senses that a person sits on the sofa, is an event producer and the TV and context manager are event consumers. The TV application and context aggregator, which is a module of the context manager, register themselves as listeners to the event manager. If the pressure sensor evolves an event and transmits this event to channel 2, the event manager notifies

both the TV and the context manager of event occurrence is through channel 2. The TV application automatically turns itself on, in order to be adaptive to the event after receiving the event information. The context manager stores the received context information to the context repository. After storing this information, the context checker which is a sub module of the context manager, checks whether a new event can occur, due to the pressure sensor context. If satisfied for a new event with existing contexts, the context checker creates a new context. The flows of the diagram are not totally sequential, because applications, devices and sensors in the ubiquitous computing environment operate in an event-driven way.

4.3 Demonstration of the Implementation

In the demonstration, Macromedia Flash is used, because of excellent GUI support, convenient representation of specific states and communications using XMLSocket. If Macromedia Flash together with Sun's Java is used for development, the developers can build systems with convenience and efficiently express a virtual ubiquitous computing environment where the application and device communicate well with managers. Applications have been developed such as a TV, light and sensors with the Java language, and graphical expressions and interfaces of the entities with Macromedia Flash. XMLSocket is used for communications among context manager, applications, and sensors. They interact with the event manager in ubiquitous computing middleware.

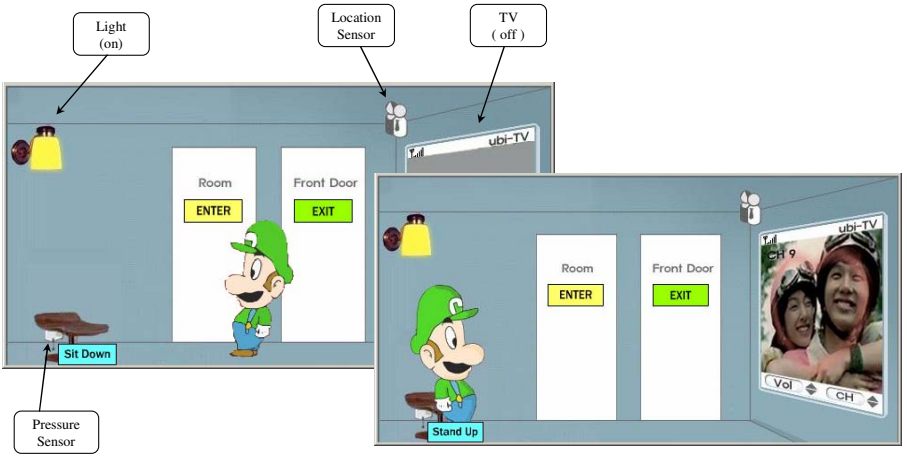


Fig. 7. Demonstration of prototyped implementation of two scenarios

Figure 7 presents the screenshots of the prototyped implementation of the virtual ubiquitous computing environment mentioned in the scenarios. If Bob enters the virtual room, the location sensor notices that he enters, and automatically turns on the light. Inside the programs, the location sensor detects the event that he comes in and transmits the event information to the event manager. The event manager notifies the light and context manager of the occurrence of the event. After taking the information

related to the event, the light turns on to be adaptive to the event and the context manager stores the event information for retrieving. If he sits on the sofa to watch TV, the pressure sensor creates an event that he sits on it and transmits the occurring event to the context manager. The TV is automatically turned on with his preferred TV channel, which is retrieved from the context information repository.

5 Conclusions and Further Work

In the ubiquitous environment, where context-aware applications and devices continuously change, sensors always perceive changes and various events occur. It is imperative to be adaptive to changes and events. For this reason, ubiquitous computing middleware has been developed for supporting event and context management services. The event manager based on Jini JavaSpaces technology, receives events from event produces and distributes them to applications of interest. The context manger aggregates and maintains all context information from applications and sensors, and retrieves and transmits them when other context-aware applications query these applications. It is shown that the scenario presented, operates well with virtual sensors interacting with the event and context managers.

In future work, it is desired to control the access to events and context so that only entities which have the correct authority can take the events and retrieve context, by extending management services. In addition, the location of credentials related to event authorization, is considered. In order to observe events, the prototype is only placed in event producers.

References

1. Anind K. Dey: Understanding and Using Context. Personal and Ubiquitous Computing (2001) 5:4-7
2. Anind K. Dey and Gregory D. Abowd: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. Human-Computer Interaction (2001) 16:97–116
3. Anand Ranganathan, Roy H. Campbell: An infrastructure for context-awareness based on first order logic. Personal and Ubiquitous Computing (2003) 7:353-364
4. Jim Waldo: The Jini Architecture for Network-Centric Computing, Communications of the ACM (1999) 7:76-82
5. Manuel Romn, Christopher Hess, et al.: Gaia - A Middleware Infrastructure to Enable Active Spaces. In IEEE Pervasive Computing, Oct-Dec 2002.
6. Polly Huang, Vincent Lenders, Philipp Minnig, Mario Widmer: Jini for Ubiquitous Devices. ETH TIK-Nr. 137, June 2002
7. CORBA services: Common Object Services Specification, OMG Specification. Object Management Group, Nov. 1997
8. Siegel, J.: CORBA 3 - Fundamentals and Programming. Object Management Group (2000)
9. W.Keith Edwards, W.Edwards: Core Jini. Pearson Education (2000)
10. Tatsuo Nakajima, Kaori Fujinami, et al.: Middleware Design Issues for Ubiquitous Computting. The 3rd international conference on Mobile and ubiquitous multimedia, Oct. 2004

Building a Frame-Based Interaction and Learning Model for U-Learning

Nam-Kek Si[1], Jui-Feng Weng[1], and Shian-Shyong Tseng[1,2]

[1] Department of Computer Science
National Chiao Tung University, ROC
edwardsi@cs.nctu.edu.tw,
roy@cis.nctu.edu.tw, sstseng@cis.nctu.edu.tw
[2] Department of Information Science and Applications
Asia University, ROC
sstseng@cis.nctu.edu.tw

Abstract. With the development of wireless network and embedded system, the Ubiquitous Computing technology has been applied in learning domain, which is called Ubiquitous Learning (U-Learning). We proposed the Frame-based Interaction and Learning Model (FILM) framework to model the context-aware applications of U-Learning for the construction, maintainability and extensibility issues. The FILM contains context-aware frame layer (lower layer) and U-Learning activity script (higher layer). In lower layer, the context information and event process are handled by frame structure and the slot attachment, and the relations of context frames are managed by the context ontology. In higher layer, the sequence of frames are combined as the context-aware services and represented by the XML-based U-Learning Activity Script and High Level Petri Net is applied for events flow control and activities sequence construction. Finally, a context-aware scenario is described and modeled by FILM.

1 Introduction

As Internet usage becomes more popular over the world, the learning technology including online learning, employee training courses, and e-book in the past ten years has been globally accepted. With the development of wireless network and embedded system, the mobile computing technology brings a revolution to free users from the constraints of stationary desktop computing. In recent years, the Ubiquitous Computing (U-Computing) technology has been applied in learning domain which is called Ubiquitous Learning (U-Learning). The main property of U-Learning is using the context information to provide learning services at the right time, right place and right way. Several researches about the development of U-Learning applications And various system architectures about the management and usage of context information were proposed. However, the following issues should be solved in the development of the U-Learning applications: Construction is costly because of complex event handling tasks. System maintainability and extensibility are difficult because learning activities are implemented in control program.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 796 – 805, 2006.

With our observation, the context information of location, person, device, sensor and interaction being discussed in the context-aware services of U-Learning generally have stereotyped attributes. The context-aware mechanism could be seen as the add-on process of event handling cooperated with existing services. Therefore, we apply knowledge based approach to design the framework. We use frame as the knowledge representation to model locations, persons and devices. Each event handling process is represented as the rules or procedure call attached to the slot of the corresponding frame.

Therefore, we proposed the Frame-based Interaction and Learning Model (FILM) framework to model context-aware applications of U-Learning for the construction, maintainability and extensibility issues. The FILM contains context-aware frame layer (lower layer) and U-Learning activity script (higher layer). In lower layer, the context information and event process are handled by frame structure and the slot attachment, and the relations of context frames are managed by the context ontology. In higher layer, the sequence of frames are combined as the context-aware services and represented by the XML-based U-Learning Activity Script and High Level Petri Net is applied for events flow control and activities sequence construction. The FILM framework modularizes the services and activities of context-aware application of U-Learning. Context-aware services are managed by frame knowledge for better maintainability. Context-aware learning activities as XML-based ULAS script for easier construction and service extensibility. The separation of device level system control and events handling level of context-aware services also provides easier system extensibility.

2 Related Work

In this section we review several U-Computing applications. Then, we have an overview to the current state of U-Learning research. Since the context-awareness is the focused feature in ubiquitous domain, we also discussed the related works of how to acquire and process context.

There are some researches of U-Computing applications [1][4] which are related to learning purpose. Abowd [1] described a prototype of a mobile context-aware tour guide which used user's current location and past location history to assist user in visiting scenic spots and writing tour diary. Dey [4] described Conference Assistant, a prototype of mobile, context-aware application, which assisted conference attendees using user's current location and the conference's agenda in order to provide conference guiding and valuable information gathering. In short, we found that existing U-Computing application development focused on how to use and manage context information and the usage of embedded device without considering the concept of learning activity, e.g., Role Playing, Learning situation. Thus, Lindquist [5] stated that it is not easy to port the existing U-Computing application directly to U-Learning scenario.

In recent year, researches [2][8] introduced some creative U-Learning scenario and developed the corresponding implementation. Ogata [8] described a context-aware language learning support system for Japanese polite expression learning. It provides

the right polite expression that is derived from hyponymy, social distance and situation through the identification of the target user and place. Cheng [2] proposed a personalized ubiquitous learning support environment in which an instructor edits learning instructional requirement set. By comparing them with learner behavior sensed, learner's situation can be grasped. Finally, suitable personalized support for learner's situation can be identified by predefined rules. In these researches, during their design process, the learning activity within the scenario is embedded in process flow and event control, as a result, learning activity is hard to maintain. Moreover, cooperation and interoperability between different applications could not be expected. Furthermore, the purposed scenarios are quite simple and only for special purpose. Actually, there exist some construction issues of the development of U-Learning applications and we think that they are important aspect to the development of this domain.

In order to support the development of U-Computing application, various architectures focusing on the ease to manage and use context information have been proposed. Context Toolkit [3] which is a context information architecture consists of three types of components: context widgets, context interpreters and context servers. Widget abstracts the acquiring context details and provides context information to other components and application, the interpreter transforms information between different types of context, the server aggregates related context to obtain high level con-text information. The architecture also provides callback and resource discovery mechanism to support developing process. Similar works are Contextual Information Service (CIS) [9], HIVE [7], etc., which provide a multi-layer software engineering process in order to separate context management from application development. Their process can simplify the context management, but can not support the scenario development. Therefore, the modification, extension or combination with context-aware application based upon context architecture is still not flexible.

In summary, there are three issues in the construction of the U-Learning applications: 1). It is difficult to design a U-Learning application due to complex, low level event handling. 2). Because of combining learning activities with control program, it is difficult to maintain the service activity. 3). The extensibility of U-Learning scenario has not been discussed.

3 Frame-Based Interaction and Learning Model (FILM)

For solving the issues described before, we proposed the Frame-based Interaction and Learning Model (FILM) framework to model the context-aware applications of U-Learning. The FILM system architecture is shown in Fig. 1.

3.1 The FILM System Architecture

Frames. Frames [6][10][11] provide a convenient structure for representing objects that are typical to a given situation such as stereotypes. A frame is basically a group of slot / slot value pairs that define a stereotypical object. Each slot may have several

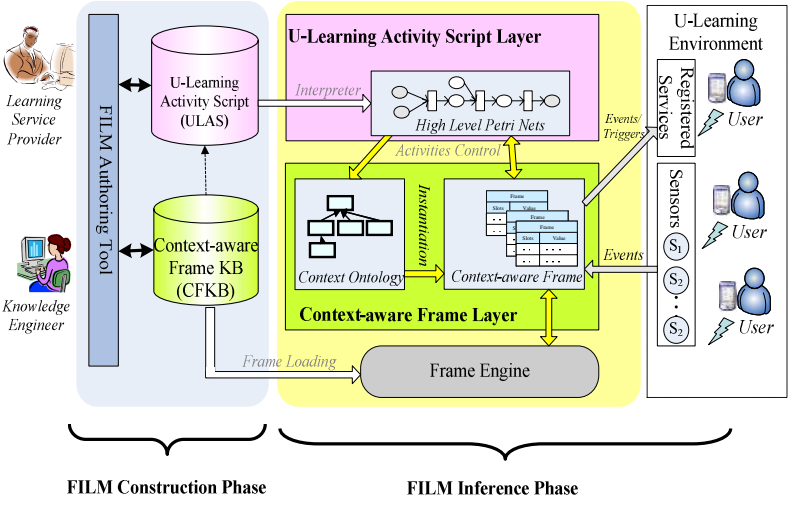


Fig. 1. The FILM system architecture

procedural attachments. The procedural attachment generally consists of four types. The if-needed type is procedure to be executed when a slot value is needed but a default value is not available. The if-added type is run for procedures to be executed when a value is to be added to a slot. The if-removal type is run whenever a value is to be removed from a slot. The if-edited type is run whenever a slot's value is to be changed. Using frames with procedural attachments and its inheritance, a very powerful knowledge representation system can be built.

Petri Net. Petri net is a graphical and mathematical modeling tool, which can be used to express process flow, parallelism, concurrent, causality, synchronization and resource sharing. It has a well-defined semantics allowing formal analysis and consists of places, transitions and arcs that connect them. HLPN is an extension of Petri Nets, which is able to model and validate more complex systems.

FILM system architecture. There are two phases in the FILM system architecture: FILM construction phase and FILM inference phase. In FILM construction phase, the context-aware frames are designed firstly by knowledge engineer. The designed context-aware frames are stored in the Context-aware Frame Knowledge Base (CFKB). With the designed frames, the learning service providers could edit their context-aware services by XML-based script authoring tool. The edited scripts could be stored in ULAS Database.

FILM. There are two layers in the FILM model.
The lower one is Context-aware frame layer. The context information and event process are handled by frame structures and slot attachments. The attributes of each object, such as person, sensor, device and environment are represented as slot/slot value pair of a frame. The relations of context frames are managed by the context

ontology. Finally, the knowledge modeled by the context-aware frames can be processed by a frame engine.

The higher one is U-Learning activity script layer. The sequence of frames is combined as the context-aware services and represented by XML-based ULAS. ULAS engine is responsible for processing the ULAS and interpret the ULAS as HLPN, which is applied for events flow control and activities sequence construction. The places represent the prerequisite objects and transitions represent the actions (The <Act> tag, more detail is in section. 4). An action is an operator set to a group of frames. The guard function Guard(T_{Ai}) of a transition T_{Ai} represents the prerequisite for firing the T_{Ai}. The prerequisite could be context or data available event.

3.2 The Context-Aware Frame Modeling

Context including location, interaction and situation has been commonly used in ubiquitous application. To simplify our discussion, we use frames to model the first two, which utilized the frame interaction mechanism.

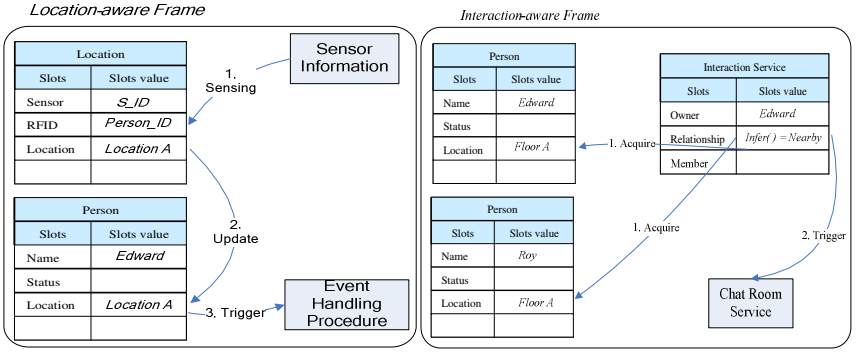


Fig. 2. Two examples of context-aware frames

Location-aware frames. The frame model of location-aware service contains three inference steps: 1) Sensor senses the personal ID. 2) Location frame updates the location slot of Person frame. 3) Person frame triggers specific service by new location slot value.

Interaction-aware frames. The frame model of interaction-aware service contains two inference steps: 1) Relationship rule class in Interaction Service is triggered, and rule class then infers on the location slot value of multiple Person frames. 2) According to the infer result, specific interaction service is triggered.

3.3 The U-Learning Ontology

Ontology, a specification of a conceptualization, is a description of the concepts and relationships that can exist for an agent or a community of agents. A formal logical

ontology consists of the logical elements: concepts and relations (generalization and specialization) restrained by logical axioms (assertions). The frames about service, person, location, time and sensors are managed in U-Learning ontology as shown in Fig. 3. We can further reason about the ontology to utilize the structure of knowledge through using attached rule.

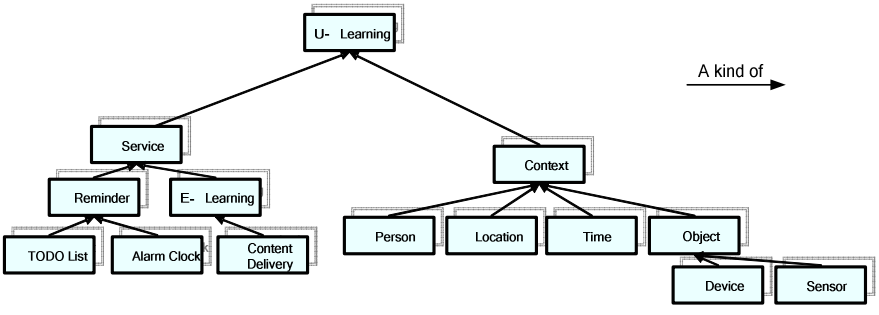


Fig. 3. U-Learning Ontology

4 The U-Learning Activity Script

ULAS consists of four major elements: Environment, Roles, Objects and Play. The scripts of the U-Learning scenarios are represented as an XML-based format. The description and example of the U-Learning Activity Script is shown in Table 1.

Tour Guide Scenario. Student Edward is visiting the Museum of Botanical Garden with RFID attached PDA. The museum was equipped with several sensors embedded in the floor and the placard. While he approaches a plant, the corresponding guidance service would obtain his location, query the topic about the location and finally, show the learning contents of the topic about the plant around in his device. The ULAS of this example is presented in Fig. 4.

Table 1. Elements of ULAS

Tag Name	Multiplicity	Description
<Script>	1 and only 1	The root of script.
<Environment>	1 and only 1	Scene of the script, e.g., location.
<Roles>	1 or more	Characters and their role.
<Objects>	1 and only 1	Objects appeared in the script.
<Device>	0 or more	An instance of device, e.g., PDA.
<Sensors>	0 or more	Group of sensor in an area.
<Play>	1 or more	Compose with several Acts. It is a time-ordered sequence of frame set.
<Act>	0 or more	The activity's frames set.

```
<Script  theme=”tour guidance”>
   <Environment Area=” Botanical Garden”/>
   <Roles><Student Name=”Edward”></Roles>
   <Objects>
      <Device type=” PDA”/>
      <Sensors type=”Location” GroupName= “Botanical Garden”/>
   </Objects>
   <Play>
      <Act>Student approach_exhibition</Act>
      <Act>location query_topic</Act>
      <Act>topic query_content_URL</Act>
      <Act>Device show_content</Act>
   </Play>
</Script>
```

Fig. 4. An U-learning Activity Script example

5 Context-Aware Scenario Modeling Example

In this section, we keep on using the tour guide scenario in Fig.4. The ULAS and frame structure of FILM are described for modeling the context-aware scenarios, and the corresponding HLPN and frame structure in this example are shown in Fig. 5.

The functionality of a context-aware application of U-Learning is embbedded in the procedural attachments, rule class and procedure call. Event process is done by slot attachment; in addition, message flow is done by the interaction of frame and is also realized by the slot attachment. The detail of model related to the execution process is shown in Table 2.

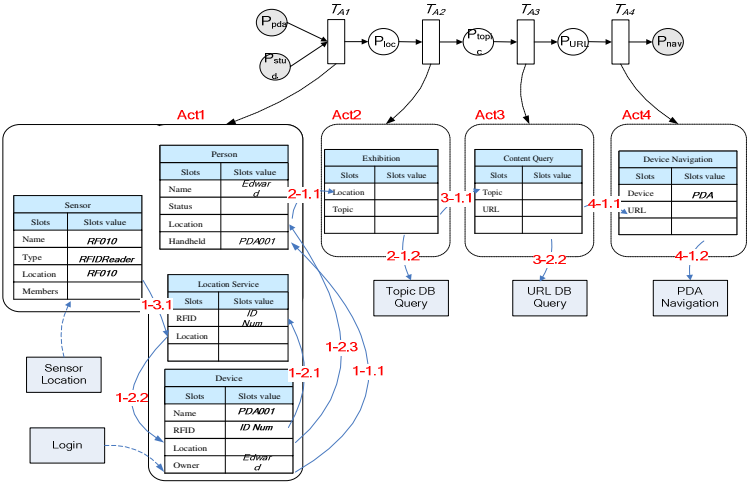


Fig. 5. The FILM of tour guide scenario

Table 2. The execution process of tour guide scenairo

Act no.	Action Description Model Detail
Act1-1	Edward is requested to logon system using current handheld device. The device he used is PDA to which RFID is attached. After logon process, PDA is registered as Edward's handheld device.
	Device.Owner*:* If added: [*Update owner's handheld device*] (1-1.1)
Act1-2	Device use Location Service to query Location. If location is updated, also update caller (device)'s location. In addition, a change to device location also causes owner's location changes.
	LocationService.Location: If needed: [*New Location Service and initials with RFID*] (1-2.1) LocationService.Location: If edited: [*Update Caller's Location*] (1-2.2) Device.Location: If edited: [*Update Owner's Location*] (1-2.3)
Act1-3	Sensor RF010 in exhibition sensed RFID of Edward's PDA and added to Sensor's Members. Location Services could catch the location update event.
	Sensor.Members: If added: [*Acknowledge Location Service*] (1-3.1)
Act2-1	Using TopicDBQuery procedure to retrieve the topic.
	Guard(T_{A2}) : *IF HasValue(Person.Location)* T_{A2}: *Initial Exhibition's Location with Person's Location.* (2-1.1) Exhibition.Location: If added: [*Topic = TopicDBQuery(Location)*] (2-1.2)
Act3-1	Using URLDBQuery procedure to retrieve the URL.
	Guard(T_{A3}): *IF HasValue(Exhibition.Topic)* T_{A3}: *Initial ContentQuery's Topic with Exhibition's Topic.* (3-1.1) ContentQuery.Topic: If added: [*URL = URLDBQuery(Topic)*] (3-1.2)
Act4-1	DeviceNavigation's URL is to be set in order to show the content of the current location, after that, using ExecPDANavigation to execute the display command.
	Guard(T_{A3}): IF HasValue(ContentQuery.URL) T_{A3}: *Initial DeviceNavigation's URL with ContentQuery's URL.* (4-1.1) DeviceNavigation.Device: If added: [*PDANavigation()*] (4-1.2)

6 Discussion

While applying the U-Computing technology to assist the students in learning, the complex events handling processes are difficult for user to provide rich and flexible services. Since the context information can be managed by frame according to domain attributes, the design of event process knowledge is easier; moreover, it could be maintained with existing frame knowledge management tool. Furthermore, the learning activities and scenarios are the most concern for the end users, such as

students and teachers. The model of XML-based ULAS can help service provider easily edit the scenarios of U-Learning based on the implemented context-aware frame. Currently, the ULAS syntax is a simple version, in the near future, by defining more strict syntax, the interpretation between ULAS and HLPN will be done more clearly. The design of FILM provides the ability for constructing the context-aware U-Learning environment in more editable, maintainable and extendsible way.

7 Conclusion

In this paper, we proposed the Frame-based Interaction and Learning Model (FILM) framework to model the context-aware applications of U-Learning. The two layers architecture in FILM can separate the U-Learning system into low level context events process and high level U-Learning activities editing. The context-aware frame and XML-based ULAS representation are applied to model the complex events handling and process flow. Two context-aware services of location-aware service and interaction-aware are presented in context-aware frame layer. With FILM, a tour guide scenario is modeled. It is a more maintainable and extendsible way to construct the context-aware application. In the near future, the FILM will be further implemented and experimented. In addition, for providing better learning efficacy for students, the applying of educational theory with FILM will also be investigated.

Acknowledgement

This work was partially supported by National Science Council of the Republic of China under contracts NSC95-2752-E-009-015-PAE, NSC94-2524-S-009-001, NSC94-2524-S-009-002.

References

1. Abowd, G.D., Atkeson, C.G., Hong, J., Long, S., Kooper, R., Pinkerton, M.: Cyber-Guide: A Mobile Context-Aware Tour Guide. ACM Wireless Networks (1997)
2. Cheng, Z., Sun, S., Kansen, M., Huang, T., He, A.: A personalized ubiquitous education support environment by comparing learning instructional requirement with learner's behavior. Advanced Information Networking and Applications (2005)
3. Dey, A.K., Salber, D., Abowd, G.D.: A Context-Based Infrastructure for Smart Environments. Proc of MANSE, Dublin, Ireland (1999)
4. Dey, A.K., Futakawa, M., Salber, D., Abowd, G.D.: The Conference Assistant: Combining context-awareness with wearable computing. Proc of ISWC (1999)
5. Lindquist, K.: New Applications for Ubiquitous Learning. Open Consultation Process, Report
6. Minsky, M.: A framework for representing knowledge. The Psychology of Computer Vision, P. Winston, Ed. McGraw-Hill, New York (1975)
7. Nelson, M., Gray, M., Roup, O., Krikorian, R., Maes, P.: Hive: Distributed agents for networking things. IEEE Concurrency (2000)
8. Ogata, H., Yano, Y.: Context-Aware Support for Computer-Supported Ubiquitous Learning. Proc of WMTE (2004)

9. Pascoe, J.: Adding generic Contextual Capabilities to Wearable Computers. 2nd International Symposium on Wearable Computers (1998)
10. Riley, G.: Expert Systems – Principles and Programming. PWS (1994)
11. Schank, R.C., Abelson, R.P.: Scripts, plans, goals, and understanding. Hillsdale, NJ: Lawrence Erlbaum (1977)

A Novel Steganographic Technique Based on Image Morphing

Satoshi Kondo* and Qiangfu Zhao

The University of Aizu, Turuga, Ikkimachi,
Aizuwakamatsu, Fukushima, 965-8580 Japan
qf-zhao@u-aizu.ac.jp

Abstract. Steganography is the technology of hiding messages in such a way that no one except the authorized recipient knows the existence of the messages. In steganography, a message is hidden in some cover message. The larger the cover message is relative to the hidden message, the easier it is to hide the latter. When the hidden message is an image, it is difficult to hide the message into another image unless the size (in number of bits) of the hidden image is much smaller than that of the cover image. To solve this problem, this paper proposes a novel steganographic technique based on image morphing. The basic idea is to transform the hidden image into a morphing image, and use the morphing image as the stego message. The authorized recipient can recover the hidden image from the morphing image through demorphing. The morphing image can also be used directly for certain purposes.

1 Introduction

Information security is one of the important problems we must solve to build a ubiquitous computing environment. Steganography or information hiding in general provides a simple but effective way for solving this problem [1],[2]. Steganography is the technology of hiding messages in such a way that no one except the authorized recipient knows the existence of the messages. In steganography, the secret message is often hidden in some cover message. In general, the larger the cover message is relative to the secret message, the easier it is to hide the latter. Images are relatively large compared with texts, and therefore are often used as the cover messages (see [3] and [4]). Usually, the secret message is hidden without changing the visual effect of the cover image. Thus, the existence of the secret message cannot be confirmed from the cover image alone.

In recent years, digital cameras are gaining more and more popularity, and a large amount of digital images are available on the Internet. However, to hide an image using existing steganographic techniques is very difficult unless the size of the secret image is much smaller than that of the cover image. For example, a 1024×768 color image, with 3 bytes per pixel, has the potential to hide 294,912

* S. Kondo was with the University of Aizu when we submitted this paper, and is now with Adin Research, Inc., Japan.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 806–815, 2006.

bytes of information, if we use 3 bits per pixel. In this case, the size of the hidden image should be smaller than or equal to 1/8 of the size of the cover image.

Another drawback in using existing steganographic techniques is that they do not allow partial hiding of the message. As an example for partial hiding, we may consider the case to treat medical images. In this case, we may try to hide the patient's personal information (e.g., characters for recognizing the personality of the patient) and keep the sickness information (e.g., face color) "readable" to other doctors. Using conventional techniques, the image data of the patient must be hidden completely in the cover data, and be recovered completely if the recipients want to see the data.

To solve the above problems, this paper proposes a new steganographic technique based on image morphing. So far, morphing has been used mainly for producing animation movies or special TV programs [5]. In this paper, we apply this technology to image hiding. Using the proposed technique, the above mentioned problems can be solved as follows. First, the morphing image (which is one of the inter-mediate images between the source image and the target image) can be used as the stego data. Here, the source image is the image to hide. Upon receiving the morphing image, we can recover the source image using demorphing (proposed in this paper) based on some stego keys. To hide the source image, we need two images (the morphing image and the target image) of the same sizes plus some morphing parameters. Thus, the capacity of morphing based steganographic technique is much larger than existing ones. The morphing based steganographic technique can also keep part of the information of the source image "readable" or visible on the stego data (the morphing image), and hide the other parts. In the case of medical image, we can hide the patient's personal information through morphing, and keep the sickness information "readable" to other doctors. Of course, this is not possible if we use image morphing directly. We propose a method for this purpose in this paper.

This paper is arranged as follow. Section 2 provides a brief review of morphing and steganography. Section 3 proposes the morphing based steganography and a demorphing method for extracting secret image from the morphing image, and describes the feature of the proposed technique. Section 4 provides some application examples of the proposed technique. Section 5 is the conclusion.

2 Preliminaries

2.1 Morphing

In image morphing, many morphing images are generated from a source image and a target image. The morphing rate $k \in [0, 1]$ is an important parameter to control the "distance" between a morphing image and the source/target image. When $k = 0$, the morphing image is the source image. When $k = 1$, the morphing image is the target image. Any other value of k generates an inter-mediate image that is similar (or dissimilar) to the source/target image.

To generate the morphing image, we also need the feature vectors of the source image and the target image. The feature vector is different for different

morphing technique. For example, the feature vector is specified by the meshes in mesh warping [6]; and specified by several feature lines in field morphing [7]. Generally speaking, the steganographic technique proposed in this paper can use any morphing techniques. In the following, we just consider field morphing.

The first step in morphing is to obtain the feature vector F_M of the morphing image. This is done as follows:

$$F_M = (1-k)F_S + kF_T \qquad (0 \leq k \leq 1) \tag{1}$$

where F_S is the feature vector of the source image, and F_T is the feature vector of the target image.

The second step is to find the warped source image and the warped target image. The warped source image and the warped target image are obtained by

$$(x', y') = f(x, y) \tag{2}$$

$$I_W(x, y) = I(x', y') \tag{3}$$

where (x, y) is the coordinate of a pixel in the warped image (the warped source image or the warped target image), (x', y') is the coordinate of a pixel in the original image (the source image or the target image), and f is a warping function for finding the coordinate of the pixel in the original image when a pixel in the warped image is given. The warping function f is generated by the feature vectors of the source image and the morphing image to obtain the warped source image; or generated by the feature vectors of the target image and the morphing image to obtain the warped target image.

In (3), $I_W(x, y)$ is the image value at the pixel (x, y) in the warped image, and $I(x', y')$ is the value at the pixel (x', y') in the original image. To generate the warped image, the values of the warped image are calculated one by one. For any pixel (x, y) in the warped image, we find the corresponding pixel (x', y') using (2). The image value of the warped image at (x, y) is then found by using (3). This process is called *reverse mapping*.

Finally, the morphing image can be obtained as follows:

$$I_M(k) = (1-k)I_{wS}(k) + kI_{wT}(k) \tag{4}$$

where $I_{wS}(k)$ is the warped source image, $I_{wT}(k)$ is the warped target image, and $I_M(k)$ is the morphing image. Note that all these images depend on the morphing rate k. The warped source image and the warped target image are combined by the ratio $1-k : k$ when the morphing image is obtained using (4). Therefore, the obtained morphing image changes from the source image to the target image when k changes from zero to one.

2.2 Steganography

The general process of steganography is shown in Fig. 1. This process contains three phases, namely, embedding (or hiding) the secret data into the cover data to obtained the stego data; transmitting the stego data (which looks like the

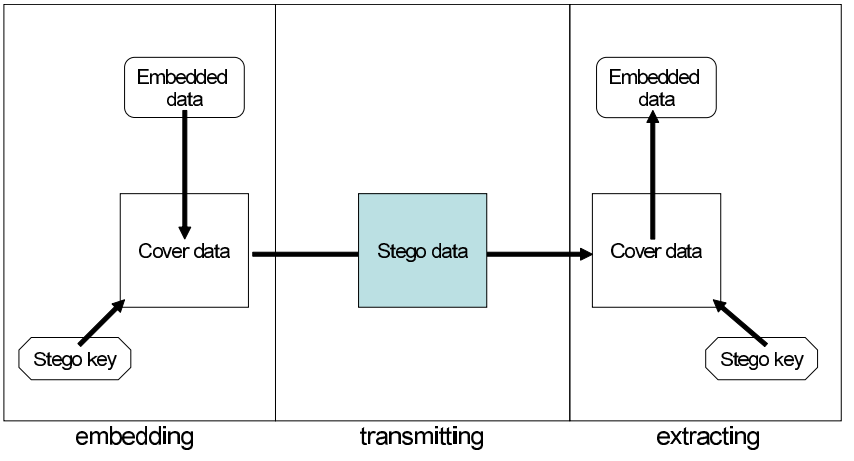


Fig. 1. Process of steganography

cover data but contains also the secret data); and extracting the embedded data [8]. Although any data can be used as the cover data for information hiding, we use images as cover data in this paper. In addition, the stego data and the data to be hidden are also images in this study.

A closely related technology is cryptography. The feature of cryptographic technology is to hide the content of information. On the contrary, the feature of steganography is to hide the existence of information. Of course, these two technologies can be used together to increase the security. For example, the content of information can be hidden first by using cryptography. Then the existence of the encrypted information can be hidden by using steganography.

3 Steganography Based on Image Morphing

3.1 General Considerations

In this section, we propose a new steganographic technique based on morphing technology. The process of the proposed technique is shown in Fig. 2. Let us compare this technique with the conventional steganography.

1. The embedding algorithm: The role of the embedding algorithm is to embed the secret data into the cover data. In the proposed technique, the morphing technique corresponds to the embedding algorithm.
2. The extracting algorithm: To extract the source image from the morphing image, we should have an inverse transform. For this purpose, we propose demorphing in this paper. That is, demorphing corresponds to the extracting algorithm.
3. The embedded data: In the proposed technique, the source image is the datum to be embedded.

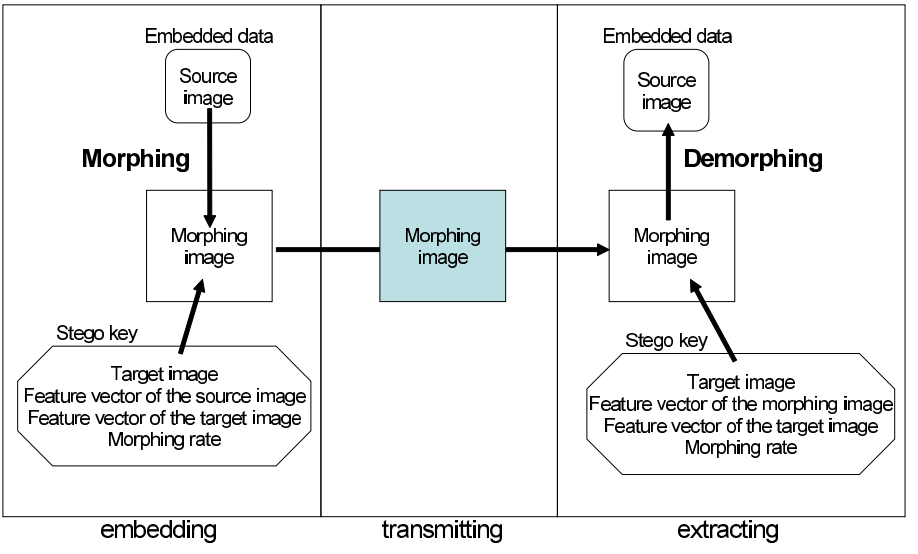


Fig. 2. A steganographic technique based on morphing technology

4. The cover data: There is no cover data in the proposed technique.
5. The stego data: The stego data is a combination of the embedded data and the cover data. The role of stego data is to transmit the embedded data without informing the existence of the embedded data. In the proposed technique, the morphing image can fulfill this role.
6. The stego key for embedding: The stego key for embedding is used for embedding the secret data. In the proposed technique, there are four stego keys, namely the target image, the feature vector of the source image, the feature vector of the target image, and the morphing rate.
7. The stego keys for extraction: The stego key for extraction is used for extracting the secret data. In the proposed technique, there are four stego keys, namely the target image, the feature vector of the morphing image, the feature vector of the target image, and the morphing rate. We cannot extract the source image if we lack any of the keys.

3.2 Demorphing

One of the important points to use the morphing based steganography is demorphing. So far, people just consider how to produce moving pictures using through morphing. No one has tried to transform the morphing image back to the source image. In this section, we propose a demorphing technique for this purpose.

The first step of demorphing is to find the warped target image using (2) and (3). This is the same as finding the warped target image in morphing. The target image, the feature vector of the target image and the morphing rate are used here.

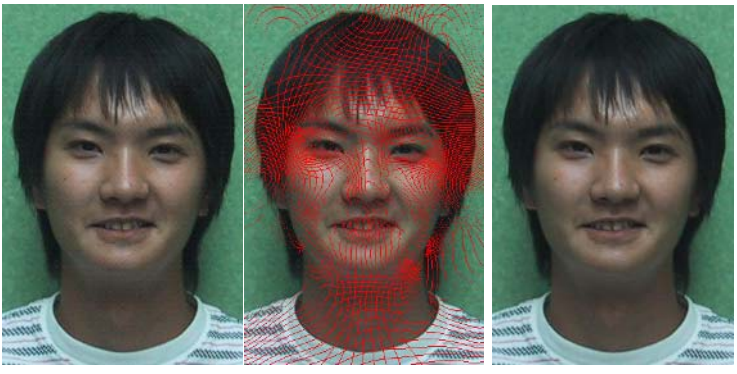


Fig. 3. The source image, the extracted image and the image after filtering

The second step is to find the warped source image as follows:

$$I_{wS}(k) = \frac{I_M(k) - kI_{wT}(k)}{1-k} \qquad (0 \le k < 1) \tag{5}$$

Actually, (5) is obtained from (4) by moving the warped source image to the left side. The information used for finding the warped source image include the morphing image, the warped target image, and the morphing rate.

To find the source image, we should also find the feature vector of the source image. This is done as follows:

$$F_S = \frac{F_M - kF_T}{1-k} \qquad (0 \le k < 1) \tag{6}$$

Actually, (6) is obtained from (1) by moving the feature vector of the source image to the left side. The information used for finding the feature vector of the source image include the feature vector of the morphing image, the feature vector of the target image, and the morphing rate.

Now we are ready to find the source image. In fact, the source image can be obtained by using (2) and the following equation:

$$I_S(x', y') = I_{wS}(x, y) \tag{7}$$

where $I_S(x', y')$ is the pixel value of the source image, and $I_{wS}(x, y)$ is the pixel value of the warped source image. Recall that (2) is the equation for determining the coordinate of the corresponding pixel in the source image when a pixel in the warped source image is given. For each pixel in the warped source image, we can find the corresponding pixel in the source image using (2), and then find the value of the source image at that pixel using (7).

Fig. 3 shows an example of demorphing. The first figure is the source image, and the second one is the extracted image by using demorphing. From the extracted image we can see that it contains many noises (which appear as curves).

In fact, when the warped source image is obtained during morphing, the coordinate of the pixel in the source image corresponding to a pixel in the warped source image is obtained by (2). On this occasion, the same pixel in the source image might be obtained for different pixels in the warped source image. This means that the morphing process is not a one-to-one mapping. Some of the pixels of the source image may not be used in producing the warped source image, and these pixels cannot be reconstructed by demorphing.

To remove the noises, a direct way is to save the pixels corresponding to the noises and use them as another type of stego keys for demorphing. These pixels can be obtained easily by investigating the pixels which are not used when finding the warped source image during morphing. Using this method, the source image can be reconstructed completely.

Another method to remove the noises is to use some kind of filters after extracting the source image from the morphing image. A simple low-pass filter can be used for this purpose. Usually, the low-pass filter is applied to all pixels in the image, and the image will be blurred. Since the positions of the noises are known, we can just apply the low-pass filter to these pixels only. By doing so, the image will not be blurred. The third figure in Fig. 3 is a result of using this technique. The low-pass filter used to obtain this image is simply an average filter. Comparing the extracted image with the source image, we can hardly tell the difference. Theoretically, however, the source image cannot be reconstructed completely through filtering.

3.3 Features of the Proposed Technique

Compared with conventional steganographic techniques, the proposed one can cover (although there is no direct cover data) an image using much less data. In the proposed technique, the morphing image plays the roles of both cover data and stego data. The stego key is relatively big. As described earlier, the stego key for extraction contains the target image, the feature vector of the target image, the morphing image, and the feature vector of the morphing rate. Thus, to cover one image, we need only two images of the same size plus some morphing parameters. In addition, since the stego key is big, we may consider that the morphing based steganography is more secure than conventional techniques.

In conventional steganography, the secret data are completely hidden in the cover data so that the stego data and the cover data are look-alike. Only the recipient who has the stego key can extract the secret data. In the morphing based steganography, The morphing image (the stego data) has certain similarity with the source image (the secret data), and the similarity is controlled by the morphing rate. This seems to be one defect of the morphing based steganography, but it is not. Actually, even if the morphing image has certain similarity with the source image, it is simply another natural image. For example, in the case of face image, the morphing image is just the face of another person, although physically this person may not exist at all. Therefore, from the morphing image alone, one cannot know the existence of the source image.

Conversely, since part of the information of the source image can appear in the morphing image, we can use this property for some special purposes. For example, suppose that we have the face image of a patient. The face image contains two kinds of information. One is related to the personality of the patient, and another is related to the sickness (say, the face color). We may ask some other doctors to see the latter and hide the former only. Morphing based steganography can be used for this purpose.

4 Applications of the Proposed Technique

A direct way to use the proposed technique is to transform all images into morphing images, and put the latter into the database. For authorized users, they can extract some of the source images using the stego keys received beforehand or through other channels. Different users may have different keys, and they can extract different source images from the same database.

By putting the morphing images in the database like this, there is no problem even if the image data are acquired by the third person illegally, because individual information cannot be extracted from the morphing images without the stego keys. Even for the authorized users, they can only access images that are extractable using their personal stego keys.

Now, let us consider another application of the proposed technique. Suppose that a doctor (doctor-A) is going to disclose the features related to the sickness of a patient to another doctor (doctor-B) while hiding the individual information of the patient. In this case, doctor-B can examine the sickness of the patient without being told who he/she is. This is impossible if we use existing steganographic techniques.

Fig. 4 (a) shows the morphing images of a patient for different morphing rates. In this example, the painted parts are considered to contain some important clues for detecting the sickness. While the morphing rate approaches to 1, the morphing image approaches to the target image in which there is no clue of sickness at all. That is, the individual information of the patient can be hidden completely by using a morphing rate close to one, but at the same time, the sickness information of the face image also disappear from the morphing image. The reason can be seen from (4). That is, when the morphing rate is close to one, the morphing image is constructed almost from the target image alone. Even if we use a relatively smaller morphing rate, the sickness information can be weakened significantly, and may no longer be useful for detecting sickness.

This problem can be solved by separating the sick information from the face image of the patient (the separation should be done automatically, but for the time being we just suppose that doctor-A knows some parts of the face might be related to the sickness, and these parts can be separated manually). During morphing, the warped source image and the warped target are combined to form the morphing image for all pixels except the separated parts. These parts in the warped source image are just "added" to the morphing image (there is no contribution from the target image for these parts).

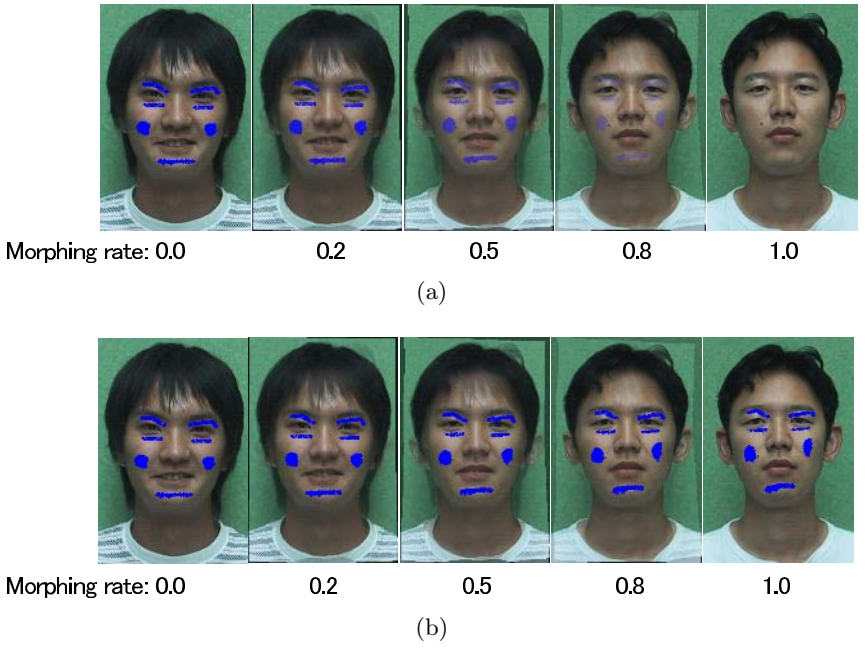


Fig. 4. Morphing images of a sick person with different morphing rates: (a) shows the results obtained using direct morphing and (b) shows the results obtained using the revised morphing technique

Fig. 4 (b) shows the morphing images of a patient generated with different morphing rates using the new method described above. In this figure, the individual information is hidden by increasing the morphing rate. The parts related to the sickness can also be kept because it is added directly to the morphing image after warping. In this way, doctor-A can disclose the sickness information to another doctor while hiding the individual information of the patient.

5 Conclusion and Future Works

This paper proposed a new steganographic technique based on image morphing. This technique is especially useful for embedding image data because the capacity is very high compared with existing techniques. To extract the hidden information, we proposed an algorithm for demorphing. This algorithm can extract the secret image from the stego data using some stego keys. Specifically, the stego keys include the morphing rate, the target image, the feature vector of the target image and the feature vector of the morphing image.

This paper also provided a technique to keep certain (e.g., sickness) information while hiding other information (e.g., the personal information). The basic idea is to separate the two kinds of information, perform morphing for the information to be hidden, and perform warping only for the information that can be

disclosed. In fact, this idea is important not only for doctors. Many researchers now are doing "face" related researches. Examples include face detection, face recognition, expression recognition and so on. In these researches, it might be useful to hide the personal information and disclose the information related to certain research. Unfortunately, existing databases all contain personal information of the contributors that are subject to abuse.

Of course, the proposed technique is useful not only for image hiding. If a proper morphing (or transform in general) technique is provided, we can also use the technique to hide other type of information. Examples may include: music, moving pictures, and so on. We are trying to commercialize this technique in the near future, and apply it to solving some practical problems.

Acknowledgment

The authors would like to thank Prof. Hideyuki Takagi at Kyushu University for his kind discussion in this research.

References

1. Petitcolas, F. A. P., Anderson, R. J., Kuhn, M. G.: Information Hiding: A Survey. Proc. of the IEEE, Vol. 87, No.7, pp.1062-1078 (1999)
2. Venkatraman, S., Abraham, A., Paprzycki, M.: Significance of steganography on data security. Proc. IEEE International Conference on Information Technology: Coding and Computing (2004)
3. Ogihara, T., Koide, S., Kaneda, Y.: Data embedding into bilevel images using the error diffusion method. Electronics and Communications in Japan (Part III: Fundamental Electronics Science), Vol. 85, Issue 11, pp. 36-44 (2002)
4. Niimi, M., Noda, H., Kawaguchi, E.: An image embedding in image by a complexity based region segmentation method. Proc. International Conference on Image Processing, pp. 74-77 (1997)
5. Wolberg, G.: Image Morphing: A Survey. Visual Computer, Vol.14, pp.360-372 (1998)
6. Wolber, G.: Digital Image Warping, IEEE Computer Society Press, Los, Alamitos, CA (1990)
7. Beier, T., Neely, S.: Feature-Based Image Metamorphosis. Computer Graphics (Proc. SIGGRAPH), Vol. 26, No.2, pp.35-42 (1992)
8. IPA (Information-Technology Promotion Agency, Japan): Technology survey of Information Hiding. http://www.ipa.go.jp/security/fy10/contents/crypto/report/Information-Hiding.htm.
9. Makino, K.: Social Aspects of Information Hiding. Journal of Information Processing Society of Japan, Vol.44, No.3, Special Issue on Information Hiding, pp. 260-264 (2003)

A Group-Oriented (t, n) Threshold Signature Scheme Against Replay Attacks

Chin-Chen Chang[1,2], Kuo-Lun Chen[2], Chu-Hsing Lin[3], and Jen-Chieh Chang[4]

[1] Department of Information Engineering and Computer Science,
Feng Chia University, Taichung, Taiwan
ccc@cs.ccu.edu.tw
[2] Department of Computer Science and Information Engineering,
National Chung Cheng University, Chiayi, Taiwan
ckl86@cs.ccu.edu.tw
[3, 4] Department of Computer Science & Information Engineering,
Tunghai University, Taichung, Taiwan
{chlin, g942817}@thu.edu.tw

Abstract. In this paper, a modified signature scheme with (*t, n*) threshold group-oriented for applications in an ubiquitous environment is proposed. In the proposed scheme, any group of more than *t* out of *n* members can cooperate to generate the group signature. The message recovery property is also added into the scheme. By employing the public key of the verifier, the transmission of the signature will be secured. Finally, to detect a removal by an eavesdropper, the linkage property between signed ciphertexts of message blocks is also added into the scheme.

Keywords: Group-oriented signature, (*t, n*) threshold signature, message recovery, data linkage, replay attack.

1 Introduction

Digital signature schemes make the authentication on electrical documents easier. One of the famous digital signature schemes was proposed by ElGamal [3]. After that, many modified signature schemes based on the discrete logarithm problem were also proposed. In recent years, new concepts in this topic including multisignature, group-oriented threshold [4], authenticated encryption schemes [5] are also discussed fervently.

The concept of group signature was introduced by Chaum and Heyst [1]. In 1994, Harn proposed a new group-oriented (*t, n*) threshold digital signature scheme [4]. In the paper Harn published, there are one public key for the whole group and one secret key for each member in the group. The group signature can be generated by at least *t* group members together. And the group signature can be verified by any outsider with the public key of the group.

In 1998, Lee and Chang proposed a more efficient algorithm based on discrete logarithm problem [7].

However, Harn's schemes [4] do not have the property of message recovery, and they are all based on the discrete logarithm problem. In fact, these schemes can be

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 816 – 825, 2006.

viewed as a modification of ElGamal's signature scheme. These schemes have a drawback that at least two parts of the signed ciphertext, r and s, will be generated. During the verification process, the plaintext is also needed to be transmitted to the verifier.

In most of the ordinary signature schemes, the signature can be verified and the message can be recovered by any verifier. But in some situations, the signer may wish the signature to be verified by some specified recipient. Furthermore, the property of message recovery is performed to be associated with the scheme so that the recipient can be chosen by the signer. The signature scheme based on RSA with (t, n) threshold was proposed by Desmedt and Frankel in 1991 [2]. But according to the Desmedt's scheme, the signature generated is not like the one generated by these modified ElGamal's signature schemes. For the same plaintext, the same signed ciphertext is generated, when different ciphertexts are generated for the same document in different signature phase by ElGamal's signature schemes. This may result in the success of replaying attacks.

In this paper, a group-oriented (t, n) threshold signature scheme for applications in an ubiquitous environment [10] is proposed. In the proposed scheme, the same property of the signing and verification processes as Harn's scheme is held. A group signature can only be generated by at least t members in the group. Furthermore, the scheme has the property of message recovery, also named authenticated encryption. The plaintext can be derived from the signed ciphertext during the signature verification. In this way, the transmission can achieve more secrecy and the storage for plaintext can also be saved. Due to the limitation of the ubiquitous environments, we have to design an efficient and secure signature scheme to reduce the cost of computation and promote the security of communication.

2 Preliminaries

Based on the (t, n) threshold secret sharing scheme proposed by Shamir [9], we assume that there is a group secret key and a group public key. Divide the group secret key into a number of shadows according to the rule of (t, n) threshold such that the group secret key can be obtained when more than t shadows are collected, with the shadow of each member as an user's secret key. When a plaintext needs to be signed, the members can generate the partial signatures with their secret keys. The clerk hence can generate the signed ciphertext with these partial signatures.

In Shamir's (t, n) threshold secret sharing scheme, a polynomial of degree t-1 is used, said $f(x)$ mod p, where p is a large prime number. The secret value s is embedded in the polynomial such that $f(0)$ equals s. Let x_i be the identity of the user U_i and $f(x_i)$ be the corresponding shadow for U_i, where $i = 1, 2,\ldots,n$. When the secret value needs to be retrieved, according to the Lagrange Interpolating Polynomial Scheme (LIPS), any t users can reconstruct the polynomial and obtain the value of $f(0)$ by collecting their shadows.

3 Proposed Scheme

In the proposed scheme, there is a system authority (SA) for generating the necessary parameters, the group secret key and the group public key. SA also divides the group secret key into n shadows, and a shadow for each user is used as his/her secret key. Besides, some public information is also published following the key distribution process. In the signature generation phase, any t or more than t out of n members can generate the partial signatures with their own secret key. With these partial signatures, the group signature of the plaintext can be generated by the clerk. The clerk can be any participant in the group. With the signed ciphertext, the group public key and some public information, any outsider can verify the validity of the signature.

3.1 Key Generation and Distribution

First, SA selects N, p and q such that N equals pq. Here p and q are two large prime numbers so that simply knowing the value of N is unable to obtain the values of p and q. Furthermore, let the values of p and q be defined as $p = 2p'+1$ and $q = 2q'+1$, where p' and q' are two large primes. Let g be the primitive element of both GF(p) and GF(q). Hence, $\phi(N) = (p-1)(q-1)$ and $T = \lambda(N) = lcm(p-1, q-1) = 2\,p'q'$ can also be obtained.

Assume that there are n members in the group G, and let G be $\{U_1, U_2, \ldots, U_n\}$. For the registration of group G, SA selects a random number d as the group secret key, where d is relatively prime to $\phi(N)$ and d is an odd number. As the theorem mentioned by Kenneth [6], when g is a primitive root over both GF(p) and GF(q) and GCD(d, $\phi(N)$) = 1, g^d will be a primitive root over both GF(p) and GF(q). With the group secret key d, SA computes the corresponding public key $e = d^{-1} \pmod{\lambda(N)}$ such that $ed = 1 \pmod{\lambda(N)}$.

After obtaining the group secret key and public key, SA divides the group secret key into n shares, and the share for the user U_i in the group G is as follows.

Step 1: SA randomly chooses a polynomial $f(x) \pmod{\lambda(N)}$ with degree t-1 and $f(0) = d$.

Step 2: Let ID_i be the identity of the user U_i chosen randomly. For each member U_i, compute $f(ID_i)$, where $ID_i \neq 0$, ID_i is odd for $i = 1$ to n and $f(ID_i)$ is even.

Step 3: For the member U_i, SA computes the share $s_i \pmod{p'q'}$ as follows.

$$s_i = \left(\left(f(ID_i)/2 \right) \Big/ \left(\left(\prod_{\substack{j=1 \\ j \neq i}}^{n} (ID_i - ID_j) \right) / 2 \right) \right) \pmod{p'q'} . \tag{1}$$

After generating these n shares, distribute the share s_i to its corresponding owner U_i. Publish the public information, including the module N and the group public key e. Keep the values of the group secret key d, the factors p, q, p', q', $\lambda(N)$ and $\phi(N)$ in secret for the reason of safety.

According to the method described by Desmedt and Frankel [2], more than t shares are collected and the polynomial $f(x)$ (mod $\lambda(N)$) can be reconstructed. Assume that these t shares are $s_1, s_2,\ldots,s_t$. The polynomial can be reconstructed as

$$\begin{aligned} f(x) &= s_1 \cdot \prod_{j=t+1}^{n}(ID_1 - ID_j)\prod_{\substack{j=1 \\ j\neq i}}^{t}(x - ID_j) + s_2 \cdot \prod_{j=t+1}^{n}(ID_2 - ID_j)\prod_{\substack{j=1 \\ j\neq i}}^{t}(x - ID_j) + \cdots \\ &\quad + s_t \cdot \prod_{j=t+1}^{n}(ID_t - ID_j)\prod_{\substack{j=1 \\ j\neq i}}^{t}(x - ID_j) \pmod{\lambda(N)} \\ &= \sum_{i=1}^{t}(s_i \cdot \prod_{\substack{j=1 \\ j\neq i}}^{t}(x - ID_j)\prod_{j=t+1}^{n}(ID_i - ID_j)) \pmod{\lambda(N)} . \end{aligned} \tag{2}$$

Therefore, $f(0)$ can be obtained by the equation,

$$f(0) = \sum_{i=1}^{t}(s_i \cdot \prod_{\substack{j=1 \\ j\neq i}}^{t}(-ID_j)\prod_{j=t+1}^{n}(ID_i - ID_j)) \pmod{\lambda(N)} . \tag{3}$$

3.2 Signature Generation and Encryption

For a document M, when t or more than t members in the group G agree to sign the document, they can provide their partial signatures to construct a group signature. With the partial signatures, the clerk can produce the group signature for the document M. During the generation process, the random mechanism is applied. The generation process of the signature is described as follows.

Step 1: For a document $M < N$, SA selects two random numbers a and b for the group. Make a and g^b public. Because g is a primitive root, g^b can be any element in the reduced residue set over module N.

Step2: Without loss of generality, let these t participants who agree to sign the document be $U_1, U_2,\ldots, U_n$. Every participant calculates the partial signature s' with his own secret key s_i and the random numbers a, g^b (mod N) generated by SA. According to the rule described previously, the partial signature generated by User U_i is

$$s_i' = (M^a g^b)^{s_i \cdot \prod_{\substack{j=1 \\ j\neq i}}^{t}(-ID_j)\prod_{j=t+1}^{n}(ID_i - ID_j)} \pmod{N} . \tag{4}$$

Step 3: Every participant provides his partial signature s_i'. After collecting these t partial signatures, the signature for the document M can be obtained as

$$\begin{aligned} \prod_{i=1}^{t} s_i' &= \prod_{i=1}^{t}(M^a g^b)^{s_i \cdot \prod_{\substack{j=1 \\ j\neq i}}^{t}(-ID_j)\prod_{j=t+1}^{n}(ID_i - ID_j)} = (M^a g^b)^{\sum_{i=1}^{t}(s_i \cdot \prod_{\substack{j=1 \\ j\neq i}}^{t}(-ID_j)\prod_{j=t+1}^{n}(ID_i - ID_j))} \\ &= (M^a g^b)^{f(0)} = M^{ad} g^{bd} = Sig_{a,b}(M) \pmod{N} . \end{aligned} \tag{5}$$

Step 4: Send the signature and the document M to the verifier.

3.3 Signature Verification

For a signature of the document M signed by the group G, anyone can verify the validity of the signature with the group public key (e, N) and the published information a and g^b. The verification process is as follows,

$$\mathrm{SG} = (Sig_{a,b}(M))^e \pmod{N}. \tag{6}$$

With the published information a, g^b and the document M, check if the following equation is correct.

$$M^a g^b \stackrel{?}{=} SG \pmod{N}. \tag{7}$$

If the equation holds, the signature is valid and the message M can be authenticated by the verifier.

4 Secure Communication

In some situations, the signers wish that the message and the signature only can be obtained and verified by some chosen verifier. The message recovery property sometimes is employed in the design of the signature scheme. In that way, the additional encryption scheme for communication security only needs to be applied once on the signature. The document M can be recovered from the signature when it is verified.

To embed the message recovery and communication security properties into our scheme, some modification is required as described in the following.

Let the public key of the specified verifier be (e_v, N_v) and the secret key of the verifier be d_v, where $e_v d_v = 1 \pmod{\phi(N)}$. And it is suggested that N always be greater than N_v. In this way, the reblocking problem can be avoided. In the signature generating process described in Section 3, replace the random number a with the public key e_v of the verifier. The signature generating process is modified as follows.

In *Step 2*, each signer computes $M^{e_v} \pmod{N_v}$ and then computes his partial signature s_i' as

$$s_i' = (M^{e_v} g^b)^{s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j)} \pmod{N}. \tag{8}$$

The signature of the document M will be

$$\prod_{i=1}^{t} s_i' = \prod_{i=1}^{t} (M^{e_v} g^b)^{s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j)} = (M^{e_v} g^b)^{\sum_{i=1}^{t} (s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j))}$$

$$= (M^{e_v} g^b)^{f(0)} = M^{e_v d} g^{bd} = Sig_{e_v, b}(M) \pmod{N}. \tag{9}$$

After the signature is generated, $Sig_{e_v,b}(M)$ is sent to the verifier and the necessary information g^b is published.

When the verifier receives the signature, the verifier confirms the signature and gets the message M with his own secret key d_v, the group public key e of the signers, and the public information g^b. At first, the verifier computes the inverse of g^b and gets the message as follows:

$$(Sig_{e_v,b}(M))^e \cdot (g^b)^{-1} = (M^{e_v d} g^{bd})^e \cdot (g^b)^{-1} = (M^{e_v} g^b) \cdot (g^b)^{-1} = M^{e_v} \pmod{N}. \tag{10}$$

The message M can be obtained by

$$M = (M^{e_v})^{d_v} \pmod{N_v}. \tag{11}$$

5 Data Linkage

In general, the document to be signed is usually divided into a number of blocks before it is signed. Especially in authenticated encryption schemes, redundancy schemes are needed to link the message blocks to prevent the eavesdropper from removing some blocks out of the ciphertext blocks without being detected. Now, here is the same problem, "How can the verifier detect if there is any block being removed by some outsider such that the verifier can ask the group G to retransmit the missing block message?"

In the following, the problem can be easily solved by involving a block number. Assume that the document M is divided into m blocks, M_1, M_2,…, M_m. As the schemes described in Section 4, with the random number b, the process of signature generation and verification is modified as follows.

The signature generation process:

Step 1: For the document blocks $M_1, M_2,\ldots, M_m < N_v$, SA selects a random number b for the group, where GCD(b, $\phi(N)$)=1, and publishes b. Same as the parameters in Section 4, let (e_v, N_v) be the public key of the verifier and d_v be the secret key of the verifier, where N_v is always smaller than N.

Step2: Without loss of generality, let these t participants who agree to sign the document be U_1, U_2,…, U_t. Every participant calculates his/her own partial signature s'_{ik} for the message block M_k with random number b and his own secret key s_i generated by SA according to the LIPS rule. For the message blocks M_1, M_2, and M_3, L_1, L_2, and L_3 are equal to $M_1^{e_v} g^{b^m} \pmod{N}$, $M_2^{e_v} g^{b^{m-1}} \pmod{N}$, and $M_3^{e_v} g^{b^{m-2}} \pmod{N}$, respectively. In the same way, L_4, L_5,…,L_m will be $M_4^{e_v} g^{b^{m-3}}$, $M_5^{e_v} g^{b^{m-4}}$, $\cdots$, $M_m^{e_v} g^{b}$, respectively.

Step 3: Every participant U_i provides his/her partial signature s'_{ik} for the message block M_k.

$$s'_{ik} = (L_k)^{s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j)} \pmod{N}. \tag{12}$$

After collecting these t partial signatures, the group signature for M_k can be obtained as follows.

$$Sig(M_k) = \prod_{i=1}^{t} s_{ik}' = \prod_{i=1}^{t} (L_k)^{s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j)}$$

$$= (M_k^{e_v} g^{b^{m-k+1}})^{\sum_{i=1}^{t} (s_i \cdot \prod_{\substack{j=1 \\ j \neq i}}^{t} (-ID_j) \prod_{j=t+1}^{n} (ID_i - ID_j))}$$

$$= (M_k^{e_v} g^{b^{m-k+1}})^{f(0)} = (M_k^{e_v} g^{b^{m-k+1}})^{d} = M_k^{e_v d} g^{b^{m-k+1} d} \pmod{N}. \quad (13)$$

Step 4: The signature for the document M can be obtained as

$$(m, Sig(M_1), Sig(M_2), Sig(M_3), \cdots, Sig(M_m)) \pmod{N}. \quad (14)$$

The result (m, $Sig(M_1)$, $Sig(M_2)$,…, $Sig(M_m)$) (mod N) is the signature for the document M. The generation and transmission of the sequence of the signature generation is $Sig(M_1)$, $Sig(M_2)$,…, $Sig(M_m)$. Besides the signatures of message blocks, the number of message blocks, m, must also be known by the verifier. Based on the factorization problem, it is impossible to obtain $g^{db^i} \pmod{N}$ from $g^{db^{i+1}} \pmod{N}$. This prohibits the situation that there might be some member in the group G signing the next message block without cooperation with others.

The verification process:

For the signature of the document M signed by the group G, the verifier can confirm the validity of the signature (m, $Sig(M_1)$, $Sig(M_2)$,…, $Sig(M_m)$) with the group public key (e, N) of the signers. The sequence of the verification will be $Sig(M_1)$, $Sig(M_2)$,…, $Sig(M_m)$. The verification and message recovery process is as follows.

$$\begin{aligned}
(Sig(M_1))^e \cdot (g^{b^m})^{-1} &= (M_1^{e_v d} g^{db^m})^e \cdot g^{-b^m} \\
&= (M_1^{e_v}) \cdot g^{b^m} \cdot g^{-b^m} = M_1^{e_v} \pmod{N} \\
(Sig(M_2))^e \cdot (g^{b^{m-1}})^{-1} &= (M_2^{e_v d} g^{db^{m-1}})^e \cdot g^{-b^{m-1}} \\
&= (M_2^{e_v}) \cdot g^{b^{m-1}} \cdot g^{-b^{m-1}} = M_2^{e_v} \pmod{N} \\
&\vdots \\
(Sig(M_m))^e \cdot (g^{b})^{-1} &= (M_m^{e_v d} g^{db})^e \cdot g^{-b} \\
&= (M_m^{e_v}) \cdot g^{b} \cdot g^{-b} = M_m^{e_v} \pmod{N}.
\end{aligned} \quad (15)$$

After obtaining the encrypted message, the message block M_k can be decrypted with the secret key of the verifier d_v.

$$M_k = (M_k^{e_v})^{d_v} \bmod (N_v) \text{ for } k = 1 \text{ to } m. \quad (16)$$

In this way, the message $M = M_1 \mid M_2 \mid \ldots \mid M_m$ can be recovered by the verifier.

6 Analysis

In this section, a brief discussion about the security of the proposed method is described as follows.

(1). According to Shamir's secret sharing scheme, it is impossible to recover the group secret key and generate the group signature with less than t members of the group in those schemes described in Sections 3-5.

(2). In the schemes in Sections 3 and 4, if SA chooses the random number b such that GCD(b, $\phi(N)$)=1 and g is the primitive element over module N, according to the theorem proposed by Kenneth [6], g^b will also be a primitive element. Because the group secret key d of the group G is relatively prime to $\phi(N)$, g^d is also a primitive element. It means that the probability for the signer to choose two random numbers b_1, b_2 such that $g^{b_1 d} = g^{b_2 d} \pmod{\phi(N)}$ is very small.

(3). If the partial signature s_i' (mod N) is known, some outsider may try to obtain the secret key s_i of the user. Facing the discrete logarithm problem, it is computationally infeasible to obtain s_i from s_i' (mod N).

(4). In Section 5, when the scheme is used as the linkage property, the generation of the signatures of M_1, M_2,…, M_t will not affect the necessary requirements for generating the following signatures, $Sig(M_{t+1})$, $Sig(M_{t+2})$,…, $Sig(M_m)$. When less than t members try to generate a signature of M_{t+1} without cooperating with others, they still need to solve the factorization problem. Although any member can compute the value of $M_{t+1}^{e_v} g^{b^{m-t}} \pmod{N}$, he also needs to solve the factorization problem and gets the group secret key from the equation $ed = 1 \pmod{\phi(N)}$ to generate the valid signature $Sig(M_{t+1}) = (M_{t+1}^{e_v} g^{b^{m-t}})^d$.

(5). In the scheme in Section 5, when the signature of each block is transmitted to the verifier, an eavesdropper removes one of these signatures. Let these signatures be $Sig(M_1)$, $Sig(M_2)$,…, $Sig(M_m)$ and the signature being removed by the eavesdropper is $Sig(M_t)$. In such case, when the receiver wants to verify the signature $Sig(M_{t+1})$, the verifier will regard that the signature being verified is $Sig(M_t)$ and then confirm the signature with $g^{b^{m-t+1}}$. The verification equation will be

$$\begin{aligned}(Sig(M_{t+1}))^e \cdot (g^{b^{m-t+1}})^{-1} &= (M_1^{e_v d} g^{db^{m-t}})^e \cdot g^{-b^{m-t+1}} \\ &= (M_1^{e_v}) \cdot g^{b^{m-t}} \cdot g^{-b^{m-t+1}} = M_1^{e_v} g^{-b} \pmod{N}.\end{aligned} \tag{17}$$

Hence, when the verifier wants to recover the message M with his own secret key d_v, the recovered message will be meaningless. Therefore, the removal is detected. The verifier can ask the signers to resend the missing signature $Sig(M_t)$.

If the eavesdropper attempts to remove any signature out of those signatures without being detected, the eavesdropper may try to modify the signature into a valid one. For example, when the eavesdropper wants to remove $Sig(M_t) = M_t^{e_v d} g^{db^{m-t+1}}$

without being detected, he may try to modify $Sig(M_{t+1}) = M_{t+1}^{e_v d} g^{db^{m-t}}$ into $Sig'(M_{t+1})$ such that $Sig'(M_{t+1})$ becomes $M_{t+1}^{e_v d} g^{db^{m-t+1}}$. Although the eavesdropper can obtain $M_{t+1}^{e_v} g^{b^{m-t}}$ from $Sig(M_{t+1})$ using the public key of the signers' group and obtain $M_{t+1}^{e_v} g^{b^{m-t+1}}$ by multiplying $(g^{b^{m-t}})^{-1} g^{b^{m-t+1}}$ with $M_{t+1}^{e_v} g^{b^{m-t}}$, the eavesdropper can not obtain the valid signature $Sig'(M_{t+1}) = M_{t+1}^{e_v d} g^{db^{m-t+1}}$ without knowing the secret key d of the signers' group.

(6). By applying the random mechanism to the schemes in Sections 3-5, when the group of signers sends the document M twice, the signatures will be

$$\begin{aligned} &Sig_1(M) = M^{a_1 d} g^{b_1 d} \pmod N \text{ and} \\ &Sig_2(M) = M^{a_2 d} g^{b_2 d} \pmod N \text{ for the scheme in Section 3 .} \end{aligned} \tag{18}$$

Since the numbers a_1, b_1 and a_2, b_2 are all randomly generated, the probability that $Sig_1(M) = Sig_2(M)$ is very small.

The signatures generated by the scheme in Section 5 is performed in the same way. The signatures generated will be

$$\begin{aligned} &Sig_1(M) = M^{e_v d} g^{b_1 d} \pmod N \text{ and} \\ &Sig_2(M) = M^{e_v d} g^{b_2 d} \pmod N . \end{aligned} \tag{19}$$

Since the numbers b_1, b_2 are generated randomly, the probability that $Sig_1(M) = Sig_2(M)$ is very small. In this way, the signature can resist against the replaying attacks.

7 Conclusions

In this paper, a signature scheme for applications in an ubiquitous environment based on factorization problem is proposed. The new scheme provides the property of (t, n) threshold group-oriented signature. When the message recovery and data linkage properties are involved in the scheme, the data secrecy is achieved and a removal of the signatures of message blocks by an eavesdropper can be detected. The necessary security properties are also considered in the paper. By adding one or two nonces into the scheme, the signatures can resist against the replaying attacks. Our scheme only needs two exponential operations to calculate the signatures, and therefore is suitable for ubiquitous environment.

Acknowledgments. This paper is partially supported by the National Science Council, Taiwan, NSC-94-2213-E-029-001.

References

1. D. Chaum and E. Van Heyst: Group signature. Advances in Cryptology-Proceedings of Eurocrypto '91, Springer Verlag, 1991, pp. 257-265.
2. Y. Desmedt and Y. Frankel: Shared generation of authenticators and signatures. Advances in Cryptology-Proceedings of Crypto '91, Springer Verlag, 1991, pp. 457-469.

3. T. ElGamal: A public key cryptosystem and signature scheme based on discrete logarithms. IEEE Trans. on Information Theory, IT-31, no. 4, 1985, pp. 469-472.
4. L. Harn: Group-oriented (t, n) threshold digital signature scheme and digital multisignature. IEE Proc. Comput. Digit. Tech., vol. 141, no. 5, 1994, pp. 307-313.
5. C. L. Hsu and T. C. Wu: Authenticated encryption scheme with (t, n) shared verification. IEE Proc. Comput. Digit. Tech., vol. 145, no. 2, 1998, pp. 117-120.
6. R. Kenneth H.: Elementary number theory and its applications. second edition, Addison-Wesley Publishing Company, 1987, pp. 249-298.
7. W. B. Lee and C. C. Chang: Efficient group signature scheme based on the discrete logarithm. IEE Proc. Comput. Digit. Tech., vol. 145, no.1, 1998, pp. 15-18.
8. R. L. Rivest, A. Shamir and L. Adleman: A method for obtaining digital signatures and public key cryptosystems. Commun. ACM, vol. 21, no. 2, 1980, pp.521-528.
9. A. Shamir: How to share a secret. Commun. ACM, vol. 24, no. 11, 1979, pp. 612-613.
10. K. Homma, Y. Sato, H. Kato, T. Fukumoto, K. Yano, K. Kawakami, T. Sakaki, and K. Funabashi: Systems Technologies for the Ubiquitous Society. Systems, Man and Cybernetics, IEEE, vol. 2, 2005, pp. 1989-1994.

Incorporating Data Mining Tools into a New Hybrid-IDS to Detect Known and Unknown Attacks

Lokesh D. Pathak and Ben Soh

Department of Computer Science and Engineering,
La Trobe University, Bundoora Campus,
Melbourne, Australia 3083
{ldpathak, ben}@cs.latrobe.edu.au

Abstract. Modern network attacks range from fully automated to multilayer attacks. Vulnerabilities in a system are exploited by an intelligent attacker to facilitate to do anything from denial of service (DoS) attacks to the system takeover. This paper addresses the development of an architecture that includes the use of fault tolerance and honeypot technology to provide layered protection to avoid a single point of failure.

Keywords: Network Security, Fault Tolerance, Honeypot.

1 Introduction

Current intrusion detection systems are not intelligent enough to create a situational knowledge required to monitor and protect these networks effectively. History and recent developments illustrate that new mutations of the existing exploits occur more often [1].

The proposed network architecture provides layered protection against the common vulnerabilities and attacks on the network. The network design includes fault tolerant strategies such as redundancy, system reconfiguration and system recovery. Also honeypot technology used in the architecture provides another layer of fault tolerance as the attackers are intentionally invited to attack the network. Once they are inside the network they are allowed to attack in a controlled manner and their steps are monitored and tracked which are further studied to take appropriate countermeasures in the future to make the production server secure. Further the implementation of the hybrid- IDS which is developed using BRAINNE module helps to detect known and unknown attacks in the proposed network layout.

2 Related Work

Numerous network designs and layouts based on different techniques have been suggested so far to provide secure communication over the network. Intrusion detection system (IDS) is an art to detect inappropriate, malicious or incorrect activity [02]. There are two types of methods used in IDS to detect intrusions: anomaly based and misuse detection. In the former, normal system behavior is studied and if there is

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 826 – 834, 2006.

a deviation from the normal behavior alerts are generated indicating something wrong is going on the network. T. F. Lunt developed anomaly based IDS using a statistical approach where the abnormality is determined by a value which is the function of abnormality values of the profile measures [03]. Feature selection method is also based on anomaly based detection of intrusions [04]. Other approaches include Bayesian statistics using belief networks or using first principles [05] and work done by Mizuki Oka to determined the correct user behavior using a method called Eigen co-occurrence matrix (ECM) that models the sequences such as commands and extracts their corresponding features [06].

The second method is based on the known vulnerabilities of the network [07]. For these known attacks there are rules made in the rule base and when there is a match of traffic pattern, the rule is fired and appropriate action is taken. Hence this is also called pattern matching method. Keystroke monitoring technique is one of the first misuse based detection methods used for the detection of intrusions where the sequences of keystrokes are observed [08]. Other methods include the detection using expert systems [09] and neural networks [10] where intrusions are classified based on the examples learned in the past. Following section describes effort to propose and design such a network to provide secure communication.

3 The Network Layout

The new hybrid- IDS is implemented on the management and control computer within the network layout which is having two identical servers connected to the internet by a router, switch and firewalls as shown in Fig. 1.

In normal mode, when there is no attack, Server1 is dealing with all the production traffic while Server2 remains idle. Sever2 is kept synchronized with Server1 using checkpoint level synchronization technique in normal mode at all times.

In attack mode, on the occurrence of an attack, Server2 is activated by the management and control computer and all the production traffic is switched to it

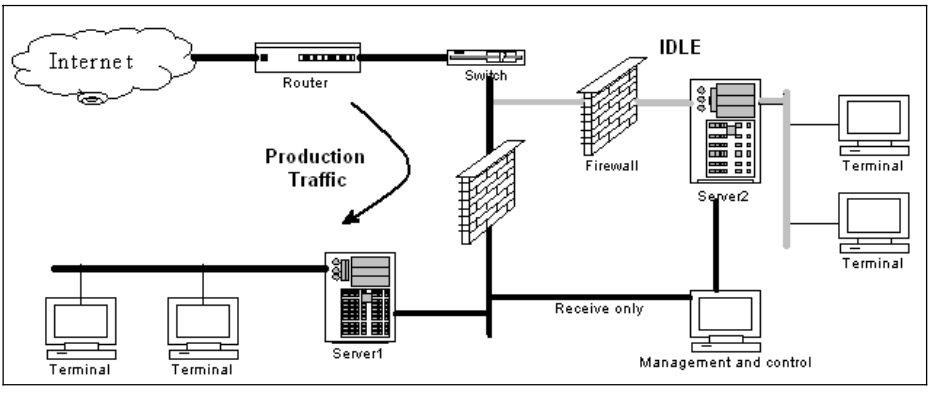


Fig. 1. Normal Mode

while Server1 becomes a honeypot. As shown in Fig. 2, in attack mode the firewall on Server1 becomes a reverse firewall also known as honeywall once an attack is detected.

The management and control computer is connected to Server1 by a 'receive only' cable [11]. So it can receive all the data from network while not transmitting anything back to the network. As Server1 becomes a honeypot after the occurrence of an attack, all the system logs are collected remotely on the management and control computer so that the attacker can not alter these logs.

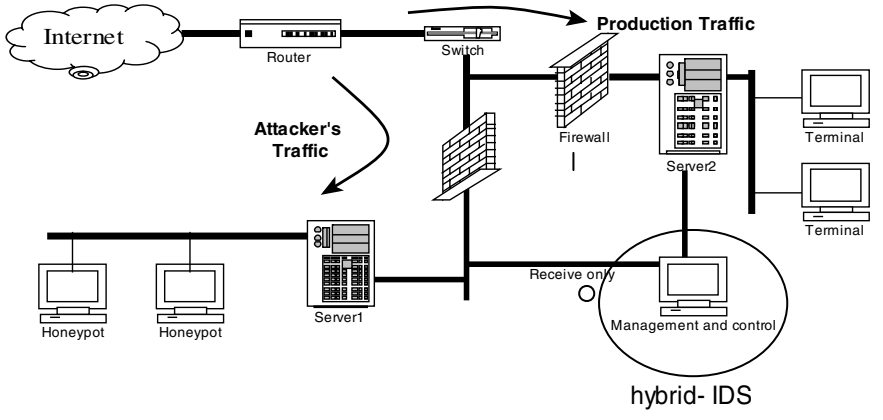


Fig. 2. Attack Mode

4 Design Details of Hybrid- IDS

The design of the proposed hybrid- IDS is shown in Fig. 3 which is implemented on management and control computer in the network layout described before. The idea here is to log all the network activity remotely with respect to the honeypots and by studying those logs to develop the countermeasures for Server2 which is the active server and handling the production traffic in the attack mode. Our model consists of

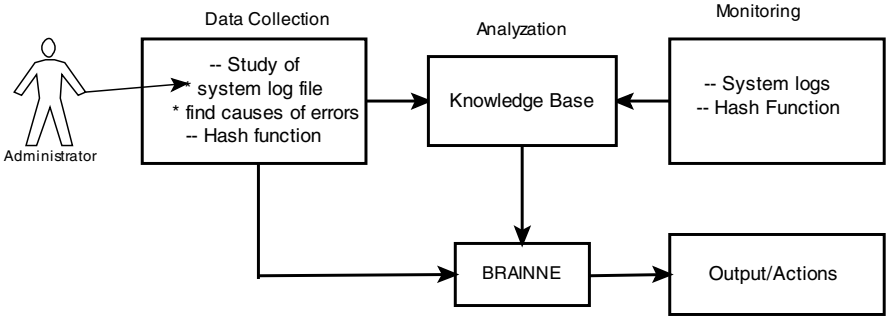


Fig. 3. System Design

three main processes namely: (i) Data Collection Process , (ii) Monitoring Process, (iii) Analyzation Process , (iv) BRAINNE and (v) Output /Actions. Each one of these processes is explained in detail in the following section.

4.1 Data Collection Process

The data collection process maintains all the information regarding the faults and the corresponding causes occurring in the system because of those faults. There are two components collected in data collection process: the faults (error messages) and their corresponding causes. First component is obtained from the system log file. But there is no information regarding the second component. We need to create some mechanism by which we can obtain knowledge of the cause of each fault and match them with their corresponding error messages or group of error messages. Because only the error messages are obtained not their corresponding causes, so the logged information must be studied from different machines in different environments to collect most of the error messages. Once information regarding the errors and their corresponding causes has been collected, a knowledge base (KB) is built by applying a hash function to the error messages while the cause for each error is stored in a separate file.

The final results of the analyzation process depend heavily in terms of correctness and efficiency on data collected in the data collection process. This is just like medical diagnostic system in which all the decisions are based on experience and literature. The output of the data collection system is a complied knowledge which is used to build the knowledge base (KB) [12].

4.2 Monitoring Process

The production server Server2 in attack mode is examined in the monitoring process of the hybrid- IDS. As the users and attackers are interacting with Server2, all the network activity is collected in the log file on management and control computer. These logs are then converted to hash values using same hash function as used in data collection process. This hash value is searched in the KB in the analyzation process as explained below to determine whether it is already there or not. The purpose of using the hash concept is to make the searching faster from a large set of stored error messages in KB so that the required action can be taken in real time.

4.3 Analyzation Process

Analyzation process contains the knowledge base (KB). KB is built using the hash table in which each error message is stored along with its hash value. KB is prepared based on the data collection process where data is collected from Server1 which is acting as a honeypot providing all the services as Server2 but having no production activity. A hash function is applied to convert the error messages to hash values which are stored in the KB. As all the traffic on the honeynet is considered as attack traffic, there are more chances for the administrator to collect as much attack related information as possible and then use it to avoid those attacks in future on the production server.

From the monitoring process, hash values from the production server (error messages) are compared with the hash values stored in the KB made from data collected from honeypots. If the searched hash value is found in the knowledge base that means this error is previously studied in the data collection process and its corresponding cause is already known. It is given as input to the BRAINNE module to create IF-THEN rule and classified as a known attack.

4.3.1 BRAINNE

BRAINNE (Building Representations for AI using Neural NEtworks) is an automated knowledge acquisition tool that extracts knowledge in the form of IF-THEN rules [13]. BRAINNE is capable of determining concept hierarchies by deriving conjunctive and disjunctive rules. Input pattern for BRAINNE consists of an input and a corresponding output class to which each input belongs. The input class here means the error messages and the output class is the fault corresponding to that error message. Supervised learning technique is used to train the BRAINNE module. The rules generated by BRAINNE contain Left hand side, (IF) which is the set of conditions and the right hand side (THEN) is followed by a hypothesis.

4.3.2 Output /Actions

The output from BRAINNE is the rules that are executed once the conditions for them holds true i.e. once the searched hash value from monitoring process is matched with the hash value in the KB. For each type of error message and its cause a particular preventive action is taken. This preventive action conFig.s the firewall and the network for that particular attack and its cause is avoided in this way in future if the same attack occurs again.

5 Implementation Details of the System

Working of the proposed system is described using the flowchart in Fig. 4. Flowchart is divided into three different subsystems to explain its working. Performance of the overall system depends on the other components to produce correct and efficient results as explained below.

5.1 Learning Subsystem

Learning subsystem is the study of the system to collect useful data for a long period from the system log file. The log file records everything from normal data to error messages.

The administrator studies the system log file to find out the errors and their corresponding causes. Once the causes of the errors are determined, they are stored in a separate file with their corresponding errors. Another file is maintained to store only the error messages. This is the process of data collection and it never stops because attackers have the freedom to interact with the honeypot which is providing the dummy services similar to the real production Server2. So honeypots are the continuous source of information regarding attacks for the network administrator.

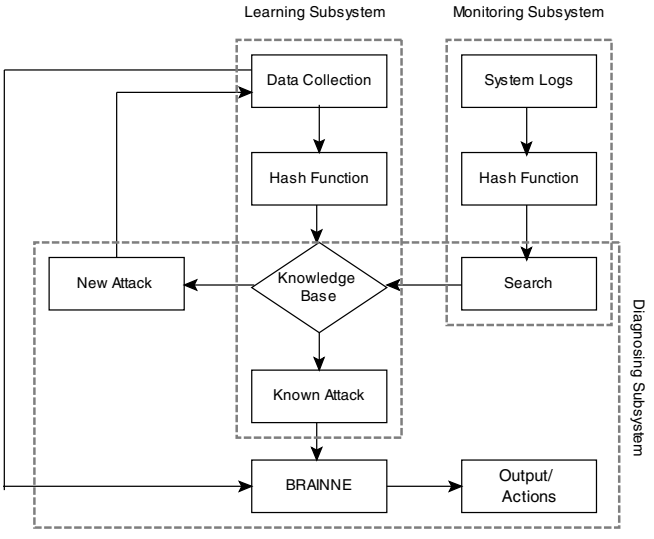


Fig. 4. Working of the Hybrid- IDS

5.2 Hash Function

Once the required data is collected in the data collection process, a hash function is applied to the error messages to get hash values corresponding to each error message obtained from the system log file. The following hash function is used in the hybrid-IDS: (i) return key.charAt(0)%contents.length; (ii) return key.charAt(0)% contents1. length

Hash function takes input as a variable length string (error message) and produces a fixed length numeric output which is called the hash value. To compute hash value, the module of ASCII value of the first letter of the error message is calculated with ASCII value of A which is 65.

An array is defined in the hash table and each error message from the log file is stored at the hash value position of the array. For example, if the hash value returned by a function is 10, so the error message is stored at array [10]. In case of a collision i.e. if the hash value computed for a message matches a hash value calculated earlier then it stores this value in the next available position to avoid collision. In this way, it is relatively easy to store, retrieve and search from the hash table as compared to other data structures. As the error messages are only added and not deleted or replaced in the knowledge base, hash table provides best search time for hybrid- IDS to search for a particular malicious traffic.

5.3 Monitoring Subsystem

This subsystem is responsible for monitoring the incoming traffic and extracts the error messages from the log file on Server2 in attack mode. The same hash function as used in learning subsystem is used to compute the hash values here and search for these values in the knowledge base. System monitoring is more concerned about the catching threats in real time considering the production Server2. It makes use of the

earlier study done in the learning subsystem to base its decisions. Here as the events and errors are recorded in the log file from the incoming traffic on Server2. The hash values are searched in the knowledge base to find if the current error is already there or not. In this way, catching threats is much faster depending on the efforts put to collect data in the learning subsystem.

5.4 Diagnosing Subsystem

Diagnosing subsystem covers the search part of monitoring subsystem, knowledge base and known attack parts of the learning subsystem. In addition to that, it contains new or unknown attacks, BRAINNE and the output /actions.

5.4.1 Known Attacks

In this part, hash values of the error messages obtained from system log file using the hash function are searched in the knowledge base. If the hash value matches with already existing hash value in the knowledge base then the outcome is the known attack and BRAINNE is applied to generate the IF-THEN rule. Already known error messages serve as IF part and if all the conditions are met on the left side a hypothesis becomes true which is followed by THEN part.

5.4.2 Unknown /Suspicious Attacks

If the hash value from the monitoring subsystem does not match with the already existing hash values in the knowledge base, the output comes as an unknown attack. Unknown attack within the context in this paper refers to the attacks which have not been studied before and have never occurred on the honeypot as well, while suspicious attacks are the new mutations of the existing attacks. So suspicious attacks are considered as unknown attacks because they have not been studied before and never occurred in the system.

Once an attack is classified as unknown attack, it is added to the learning subsystem for further analysis and studied to determine the cause corresponding to it. Once the cause is known for the new error, a rule is created using BRAINNE. Now when next time this attack will appear from monitoring subsystem it will be detected as known attack and the appropriate action will be taken to prevent its outcome.

5.4.3 BRAINNE

The hash value from the monitoring system is searched in the knowledge base; if there is a match it is called a known attack. The hash function in the knowledge base points to a hash table in which all the error messages are stored. This hash tables provide a constant lookup time, o(1), on average regardless of the size of the hash table.

For every match of the hash value, associated error messages is pointed and from the data collection process the corresponding cause is fed as input to the BRAINNE to generate the rule.

5.4.4 Output /Actions

Output from BRAINNE is the IF-THEN rules. From this output the experts at the network management and control computer know the required actions that need to be taken. The output of the hybrid- IDS is then used to conFig. the firewall and other

network configurations to avoid the occurrence of these attacks in future on the production server.

6 Results and Discussions

Once the system is set up with all the honeypot and network configurations as explained in Fig.s 1 and 2 within an organization it can be efficiently employed to detect known and unknown attacks. The results of the new hybrid- IDS show that it can take the required action in real time once it notices the malicious traffic that has been studied before. If the malicious traffic is not studied before i.e. it is new to the hybrid- IDS it outputs as unknown attack which is then studied by the administrator. Once the cause of the malicious traffic is known it can be easily added to the knowledge base. In this way, it is classified as a known attack if same malicious traffic appears again on the network.

The hybrid- IDS provides following 4 options when it is executed: *a: Read the error messages from a text file; b: Add a word from the keyboard; c: Search for the error messages*; *d: Write the knowledge base to a new file*; *q: Quit.*

The grand total of the CPU time for the hybrid- IDS once a search is made for the particular hash value in the knowledge base and the total run time is calculated using the following command: *Time java hybridIDS.*

The above command is executed before running the program to calculate the time hybrid-IDS takes once the malicious traffic appears and then to classify it as known or unknown attack. The results are presented at the end of the execution of the program as follows: *0.150u 0.030s 0:38.40 0.4% 0+0k 0+0io 1899pf+0w.*

The hybrid-IDS used 0.150 seconds of user time, 0.030 seconds of system time, and 38.40 seconds of real time. The sum of the user time and system time is the total CPU time of the program. 0.4% indicates the percentage of the CPU's time that the program used while it ran. The source code for the hybrid- IDS and the classes used, IDSmanager and hashTable, are not included in this paper because of space constraints.

7 Conclusion and Future Work

This paper presented the development of a new hybrid- IDS using data mining module BRAINNE in supervised learning mode to detect known and unknown attacks. The hybrid- IDS is implemented in an architecture that includes the use of fault tolerance and honeypot technology to provide fault tolerance to avoid single point of failure. The working of hybrid- IDS is explained in three parts: learning subsystem, monitoring subsystem and diagnosing subsystem. Further the concepts of hash functions and hash tables are employed for the efficient storage and searching operations to locate particular traffic in the knowledge base. The results indicate that the proposed model can be implemented on a network of an organization where security and reliability are crucial.

Future focus is to use the upcoming technologies such as data mining and intelligent agents in detection of attacks. These technologies can provide the alerts on malicious system behavior, making patterns of known attacks and making the traffic analysation task easier for a system administrator.

References

1. Carnigie Mellon, Software Engineering Institute, CERT® Coordination Center (CERT/CC): CERT/CC Statistics 1988-2005 http://www.cert.org/stats/cert_stats.html, (2005)
2. http://www.honeypots.ne/: Intrusion Detection, Honeypots and Incident Handling Resources (2005)
3. Lunt, T. F., Tamaru, A., Gilham, F., Jagannathan, R., Neumann, P.G., Javitz, H., Valdes, A., Garvey, T.D.: A Real-Time Intrusion Detection Expert System (IDES)- Final Technical Report, Computer Science Laboratory, SRI International, Menlo Park, California (1992)
4. Heady, R., Luger, G., Maccabe, A., Servilla, M.: The Architecture of A Network Level Intrusion Detection System, Technical report, Department of Computer Science, University of New Mexico (1990)
5. Pearl, J.: Probabilistic Reasoning in Expert Systems, Morgan Kaufman, San Mateo, California, (1988)
6. Oka, M., Oyama, Y., Abe, H., Kato, K.: Anomaly Detection Using Layered Networks Based on Eigen Co-occurrence Matrix, 7th International Symposium, RAID 2004, Sophia Antipolis, France, (2004) 10-15
7. Kruegel, C., Vigna, G.,Robertson, W.: A multi-model approach to the detection of web-based attacks, Computer Networks, Vol. 48, Elsevier, Amsterdam (2005) 717-738
8. Bolsky, M.I., Korn, D.G.: The KornShell Command and Programming Language, Prentice Hall, Englewood Cliffs, New Jersey (1989)
9. Debar, H., Becker, M., Siboni, D.: A Neural Network Component for an Intrusion Detection System, Proceedings, 1992 IEEE Computer Society Symposium, (1992) 240 – 250
10. Nuansri, N., Dillon, T.S., Singh, S.: An Application of Neural Network and Rule-Based System for Network Management: Application Level Problems, 30th Hawaii International Conference on System Sciences (HICSS) Vol. 5, Advanced Technology Track (1997) 474-479
11. http://www.ironcomet.com/sniffer.html: Receive Only Sniffing Cable (2005)
12. Sestito, S. Dillon, T.S.: Automated Knowledge Acquisition, Prentice Hall, New York (1994)
13. Nuansri, N., Dillon, T.S., Singh, S.: An application of neural network and rule- based system for network management: application level problems, Proceedings of the Thirtieth Hawaii International Conference (1997) 474 - 483

A Further Approach on Hypercube-Based Pairwise Key Establishment in Sensor Networks

Ping Li[1,2] and Yaping Lin[1]

[1] College of Computer and Communications, Hunan University, Changsha, 410082, China
lping9188@163.com

[2] School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, 410076, China
yplin@hnu.cn

Abstract. Security schemes of pairwise key establishment play a fundamental role in research on security issue in wireless sensor networks. The properties of weak connectivity in subsection based on hypercube model are addressed, for purpose of achieving inner-area pairwise key establishment. Also a hybrid hypercube model based on location-aware deployment is proposed, focusing on dramatic varieties of network size. Corresponding algorithms on inter-area pairwise key establishment are presented, as well as security and performance analysis.

1 Introduction

The security issue in wireless sensor networks has become research focus because of their tremendous applications available in military as well as civilian areas[1]. Security schemes of pairwise key establishment, which enable sensors to communicate with each other securely, play a fundamental role in research on security issue in wireless sensor networks[2].

Generally, pairwise key establishment includes the following four parts: 1). Key pre-distribution phase. Before node's deployment, the key setup server pre-loads a sum of keys (or polynomial shares) in every node. 2). Direct key establishment phase. For a given node deployed in the field, if its direct neighbors have a common key (or share) with that node, they then use it for secure communications directly. 3). Indirect key establishment phase. Once the phase of direct key establishment failed, intermediate nodes are necessarily introduced to form a key path, through which sensible information is transmitted to establish an indirect key. 4). Pairwise key establishment phase on remote nodes. When a node has ability to communicate its neighbors securely□it can then share secrets with remote nodes by means of multi-hop routing mechanisms[3], thus achieving key establishment through the entire network.

The key issue on pairwise key establishment is key pre-distribution phase. Two kinds of pre-distribution schemes are available, centralized and localized schemes according to information pre-loaded in each sensor. With respect to the former, Eeschnaure et al[4] presented a probabilistic key pre-distribution scheme. Based on the

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 835 – 844, 2006.

contributions made by [5], a lot of attention [6] has focused on polynomial based key pre-distribution. That is, the key setup server randomly generates a t-degree bivariate polynomial $f(x,y)$ over a finite field F_q. Notes that $f(x,y)= f(y,x)$ is always held. For nodes i and j, the server computes two shares of $f(x,y)$, denoted as $f(i,y)$ and $f(j,y)$, for the two nodes respectively. Thus they can compute the common key $f(i,j)$ directly.

With regard to the latter, Liu et al[7] developed a hypercube-based assignment scheme. Liu et al[8] also presented a location-aware deployment model, and developed corresponding scheme, using location information.

The contribution of this paper is two-fold. First, we model a local network with densely distributed nodes as a hypercube, inspect properties of k-dimensional weak-connectivity in subsection of hypercube model, and develop relative scheme on pairwise key establishment for inner-area nodes. Second, we develop a hybrid hypercube model with location-aware deployment knowledge, focusing on networks of large scale with huge number of nodes in a wide deployment area. And we also present inter-area pairwise key pre-distribution scheme. Our analysis indicates that this scheme provides stable performance on probability to establish pairwise key between sensors in the network.

The rest of this paper is organized as follows. Section 2 inspects properties of k-dimensional weak-connectivity in subsection of hypercube model, and presents inner-area scheme on pairwise key establishment. Section 3 describes a hybrid hypercube model, addressing key pre-distribution issue with deployment knowledge. Section 4 presents inter-area scheme. Section 5 analyzes performance of presented schemes before Section 6 concludes this paper.

2 Inner-area Pairwise Key Establishment

In this Section we consider a simpler situation: Assume that a fairly large number of sensor nodes are densely distributed in a small area. We believe that only in that kind of situation does it make sense to apply the properties of basic hypercube model on pairwise key establishment.

2.1 Hypercube-Based Pairwise Key Pre-distribution

Given a total of N sensor nodes in the network, this scheme constructs an n-dimensional hypercube. A node's coordinate is encoded into n l-bit binary strings (one for each dimension) where $l = \lceil \log_2 m \rceil$ and $m = \left| \sqrt[n]{N} \right|$. Each ID j is expressed as $<j_1,j_2,\ldots j_n>$, where j_i is called the sub-index of ID j.

For each node such as j, the setup server then distributes the following polynomial shares: $\{ f^1_{<j_2,\ldots,j_n>}(x,y), f^2_{<j_1,j_3,\ldots,j_n>}(x,y), \ldots, f^n_{<j_1,j_2,\ldots,j_{n-1}>}(x,y) \}$ to this sensor node.

To establish a pairwise key with node j, node i checks whether they have the same sub-index in n-1 dimensions. That is, if the hamming distance of the two nodes are $d_h(i,j)=1$, they share a common polynomial, and thus they can establish a direct key.

Otherwise, they need to go through path discovery to establish indirect key. Please refer to [7] for details.

2.2 Weak Connectivity Model in Subsection

Consider an *n*-dimensional hypercube with a total of *N* sensor nodes, and each node in the network is assigned to a unique coordinate $j_1 j_2 ... j_n$, where $0 \le j_1, ... j_n < v$ and $v = \lceil \sqrt[n]{N} \rceil$. For simplicity the hypercube can be expressed as *H*(*v*,*n*). In addition, every valid coordinate can be divided into *r* subsections in sequence, each of which has no more than *k* characters, where $r = \lceil n/k \rceil$.

Definition 1: The nodes *A* and *B* in *H*(*v*,*n*) are called logic neighbors, iif that only one character is different in their coordinates. Both the two nodes are called physical neighbors, iif that they are within each other's signal range. There exists a secure link between A and B if they are neighboring both logically and physically.

Definition 2: The node *A* in *H*(*v*,*n*) is isolated, iif that all the secure links between *A* and the other nodes are failed. Otherwise, the node *A* is reachable, if that *A* is faultless and has at least one active secure link.

Definition 3: The nodes *A* and *B* in *H*(*v*,*n*) are called logic neighbors in subsection, iif that only one section has different characters in their coordinates.

Definition 4: For a given character string with the length of *n*-*k*, $b_1\, b_2 \ldots b_{n-k}$, the corresponding *k*-dimensional hypercube *H*(*k*) contains v^k nodes and can be expressed as $b_1\, b_2 \ldots b_{n-k} * \ldots *$, where * denotes a character within 0,…,*v*-1.

Definition 5: (*k*-dimensional weak-connectivity in subsection): The hypercube *H*(*v*,*n*) is *k*-dimensional weak-connected in subsection, if the number of all reachable nodes in each section is larger than $v^k / 2$.

Lemma 1: Let an *n*-dimensional Hypercube *H*(*v*,*n*) satisfies the conditions of *k*-dimensional weak-connectivity in subsection. Then all the reachable nodes form a connected graph in any two *k*-dimensional sub-hypercube neighboring in subsection.

Proof. Assume that $r = \lceil n/k \rceil$, a coordinate of the length *n* is then divided into *r* sub-strings with the length of no more than *k*. Let H_k and H'_k are two subsection neighboring *k*-dimensional sub-hypercubes, expressed as $H_1(k) = b_1 b_2 ... b_{(r'-1)k'} \alpha_1 .. \alpha_k b_{(r'+1)k} ... b_{n-k} * ... *$ and $H_2(k) = b_1 b_2 ... b_{(r'-1)k} \beta_1 .. \beta_k b_{(r'+1)k} ... b_{n-k} * ... *$. Assume that the nodes *u* and *v* are reachable nodes, which belong to $H_1(k)$ and $H_2(k)$ respectively. Let $u = b_1 b_2 .. \alpha_1 ... \alpha_k ... b_{n-k} x_1 x_2 ... x_k$ and $v = b_1 b_2 .. \beta_1 ... \beta_k ... b_{n-k} y_1 y_2 ... y_k$. According to the property of *k*-dimensional weak-connectivity in subsection, there exist a reachable u_1 node in $H_1(k)$ and a reachable v_1 node in $H_2(k)$, expressed as $u_1 = b_1 b_2 .. \alpha_1 ... \alpha_k ... b_{n-k} c_1 c_2 ... c_k$, and $v_1 = b_1 b_2 .. \beta_1 ... \beta_k ... b_{n-k} c_1 c_2 ... c_k$. Note that

the two nodes u_1 and v_1 are reachable and they both belong the sub-hypercube $H_c(k)$, denoted as $H_c(k) = b_1 b_2 .. b_{(r'-1)k} * ... * b_{(r'+1)k} ... b_{n-k} c_1 .. c_k$. As the sub-hypercube $H_c(k)$ satisfies the conditions of *k*-dimensional weak-connectivity in subsection, there exists a reachable node *c*, expressed as $c = b_1 b_2 ... b_{(r'-1)k} d_1 ... d_k b_{(r'+1)k} ... b_{n-k} c_1 .. c_k$. The nodes u_1, v_1 and *c* are connected in $H_c(k)$. Thus the nodes *u* and *v* are connected.

Lemma 2: All of the reachable nodes in *n*-dimensional Hypercube *H*(*v*,*n*), which satisfies the conditions of *k*-dimensional local-weak-connectivity in subsection, form a connected graph.

Proof. For simplicity, we express a valid coordinate of a node with the length of *n* as a string containing *t* characters. Assume that the nodes $u = a_1 a_2 ... a_r$ and $v = b_1 b_2 .. b_r$ where $a_i, b_i \in [0, kv-1]$. With regard to a subsection $a_i, b_i, i \in [0, r]$, there exists a subsection c_i, which enables the nodes *u* and u_1 connected in $H_i(k)$ where $u_1 = a_1 .. a_{i-1} c_i a_{i+1} ... a_r$ and $H_i(k) = a_1 a_{i-1} * a_{i+1} ... a_r$. Also it makes the nodes *v* and $v_1 = b_1 .. b_{i-1} c_i b_{i+1} ... b_r$ connected in $H_i'(k) = b_1 b_{i-1} * b_{i+1} ... b_r$. Thus along with no more than 2*r*-1 intermediate nodes, such as $u_1, u_2 ... u_{r-1}, c = c_1 c_2 ... c_r, v_1, v_2 ... v_{r-1}$, the nodes *u* and *v* are connected.

2.3 Indirect Key Establishment

Assume that the two physically neighboring nodes A $(i_1, i_2, ..., i_n)$ and B $(j_1, j_2, ..., j_n)$ want to establish pairwise key between them. In case that $d_h((i_1, i_2, ..., i_n), (j_1, j_2, ..., j_n)) = k > 1$, the nodes perform the algorithm on indirect key establishment called ***Inter-Area(S,D)*** illustrated as follows.

The algorithm assumes that during the deployment phase nodes are required to exchange their connectivity information in subsection. That is, every node maintains a table *T* to record reachable nodes in each subsection. Let $S(a_1 \ldots a_n)$ and $D(b_1 \ldots b_n)$ be the two physically neighboring nodes. As the assumption of the algorithm has addressed, they exchange with each other their connection information in each subsection. Assume that the source node *S* initiates the key establishment phase. It thus creates a temporary table called T_D to record reachable nodes in common subsection code with those of node *D*. Node *S* then performs the following procedures:

P1: The subsection process on node's coordinate, such as $a_1 \ldots a_n \longrightarrow a_1', a_2' ... a_r'$, $b_1 \ldots b_n \longrightarrow b_1', b_2' ... b_r'$ where $a'_j, b'_j \in [0, kv-1]$.

P2: Node *S* checks T_D to find if there exist available nodes in each different subsection. If no available nodes in one of the subsections, the algorithm terminates. Otherwise it goes on the next procedure.

P3: Initialize the path *P*: $P \leftarrow S$ and the temporary node *C* ($c_1 c_2 \ldots c_r$): $C = \leftarrow S$.

P4: FOR (i=1; $i \le w$; i++){
IF ($c_i \neq b'_i$) {

Search a reachable node in the i^{th} subsection: $C_i = c_1 \ldots c_{i-1} x_i\, c_{j+1} \ldots c_r$ in the table T_D;
Add the intermediate nodes along with the path between $C(c_1 c_2 \ldots c_r)$ and $C_i(c_1 \ldots c_{i-1} x_i\, c_{j+1} \ldots c_r)$ to P in sequence;
}
$C(c_1 c_2 \ldots c_r)$: $C = \leftarrow C_i$; P: $P \leftarrow C$;
}

It's comparatively reasonable to model a sensor network in a small area as a hypercube with the properties of k-dimensional weak-connectivity in subsection. As one of necessary requirements, at least $(2^{k-1}+1)^r$ nodes are to be reachable for a network containing 2^n nodes. Let us consider a network with N=500 and k=r=3. If the deployment area can be divided 8 sub-areas, each of which owns 8 groups, a reachable nodes is required to be connected among at least 5 areas and 25 groups. As sensors are assumed densely distributed in a small area, we believe that such requirements can be satisfied normally.

3 Hybrid Model on Key Pre-distribution

In case of a sensor network in a large deployment field with a huge number of sensors, it is not reasonable to apply the properties to construct a key path throughout the network. For example, consider N=32000, let v=2, k=3 and r=5. Impractical situations occur such as minority nodes (at least 9%) should be connected, while those nodes are required to be scattered in main area of the field. In this Section we propose a hybrid hypercube model with location-aware deployment knowledge[8], focusing on inter-area pairwise key establishment throughout the entire network.

3.1 Location-Aware Deployment Model

Sensors are assumed to be deployed in a two-dimension area called the target field, and two sensors can communicate with each other if they are within each other's signal range. The target field is partitioned in equal sized (R×C) squares, each of which is called an area with the coordinate (L_x, L_y), denoting row L_y and column L_x. Each sensor is expected to be located in its home area before deployment phase. In some cases, the sensor may appear somewhere near its expected location. We assume that a node is distributed in its home area may appear at a particular point (within the range of maximum deployment area e) with certain probability, which is calculated by the integration of probability density function ε over this range. For a given node u, which may randomly appear anywhere at a distance of no more than e, the deployment error can be expressed as $\varepsilon_{L_x, L_y}(x, y) = 1/\pi e^2$, where $(x - L_x)^2 + (y - L_y)^2 \le e^2$.

3.2 Key Pre-distribution Issue

Assume that there exist $2^{l-1} m^n < N \le 2^l m^n$ nodes in the whole network. A node is coded with the following two parts (u_l, u_n) in the l+n dimensions: One is called location

code denoted as u_l, a binary string with the length of *l*, which is used to identify the expected area of a node. The other is inner code denoted as u_n, a character string with the dimension of *n*, which is used to distinguish the other nodes in the same expected area. The two parts can also be expressed as $\underbrace{00...0}_{l} \le u_l \le \underbrace{11...1}_{l}$, $\underbrace{00...0}_{n} \le u_n \le \underbrace{mm...m}_{n}$.

With regard to the location code, since there are N_A areas in the target field, the coordinate of each area can be encoded into a binary string with the length of *l*, where $l = \lceil \log_2^{N_A} \rceil$. For a given node *u*, the setup server first determines its home area, in which the sensor is expected to locate. And the server generates the corresponding location code u_l for the node. The setup server then discovers four areas adjacent to the sensor's home area, and encodes the coordinates of the two neighboring areas in row into location codes, expressed as u_l^{R+1} and u_l^{R-1} respectively. Also u_l^{C+1} and u_l^{C-1} are used to identify the two neighboring area in column of the node *u*'s expected area. Let S_u denotes the set of location codes of sensor *u*'s home and adjacent areas, expressed as $S_u = \{u_l^{R+1}, u_l^{R-1}, u_l^{C+1}, u_l^{C-1}, u_l\}$. We define $S_{u,v}$ as the set of common location codes shared by node *u* and *v*, expressed as $S_{u,v} = S_u \cap S_v$.

In order to make it easier to illustrate, the "-" operator is introduced to describe a string in *r*-1 dimension, such as $u_k - i_k =< i_1, i_2, ...i_{k-1}, i_{k+1}, ..i_r >$. We define the following string sequences:

$k_{1,1} = u_l^{R+1}, (u_n - j_1)$, $k_{1,2} = u_l^{R-1}, (u_n - j_1)$, $k_{1,3} = u_l^{C+1}, (u_n - j_1)$, $k_{1,4} = u_l^{C-1}, (u_n - j_1)$,
$k_{1,5} = u_l, (u_n - j_1)$, ... $k_{n,1} = u_l^{R+1}, (u_n - j_n)$, $k_{n,2} = u_l^{R-1}, (u_n - j_n)$, $k_{n,3} = u_l^{C+1}, (u_n - j_n)$,
$k_{n,4} = u_l^{C-1}, (u_n - j_n)$, $k_{n,5} = u_l, (u_n - j_n)$

Thus we get a key pre-distribution matrix with 5 columns and *n* rows:

$$K = \begin{bmatrix} k_{1,1} & k_{1,2} & ... & ... & k_{1,5} \\ ... & ... & ... & ... & ... \\ k_{n,1} & k_{n,2} & ... & ... & k_{n,5} \end{bmatrix} \quad (1)$$

The node *u* is then pre-loaded the key polynomial shares expressed as $F_u = \{f_{k_{i,j}}^{i,j}(u_{ID}, y)\}_{i=1,2,..n, j=1,2,.5}$.

4 Inter-area Pairwise Key Establishment

As addressed in Section 3, sensors may fall into neighboring areas near its expected location, which is determined by the maximum deployment error. In terms of a given local node, challenges arise as its direct neighbors could contain several nodes of other areas. Based on the construction of key pre-distribution matrix, in this section we present inter-area scheme to achieve pairwise key establishment for those "distant" nodes.

4.1 Direct Key Establishment

To establish a pairwise key with node *v*, node *u* checks whether they can share a common polynomial share by performing the following verifications:

$$S_{u,v} \neq \phi \tag{2}$$

$$d_h(u_n, v_n) \leq 1 \tag{3}$$

If $v_l = u_l$, that means both the two nodes are expected in the same area. Assume that they are different in the $j_n{}^{th}$ dimension, they share in the n^{th} row of polynomial shares in the matrix, denoted as $\{f_{k_{n,j}}^{n,j}(x,y)\}_{j=1,..5}$.

If $v_l \neq u_l$ and $v_l \in S_u$, that means the expected area of node *v* is neighboring to that of node *u* along with the row or the column. For example, $v_l = u_l^{R+1}$. If $d_h(u_n, v_n) = 0$, that means they have the same inner code while they are expected in different areas, and they have common polynomial shares, located in the first column of polynomial shares in the matrix of node *u* and the fifth column of polynomial shares node *v*, denoted as $F(u) = \{f_{k_{j,1}}^{j,1}(u,y)\}_{j=1,2,..n} = F(v) = \{f_{k_{j,5}}^{j,5}(v,y)\}_{j=1,2,..n}$. In case that $d_h(u_n, v_n) = 1$, that means the inner codes of the two nodes are different in one dimension, such as in the d^{th} dimension. Then they have the common polynomial share $F(u) = f_{k_{d,1}}^{d,1}(u,y) = F(v) = f_{k_{d,5}}^{d,5}(v,y)$ indicated as $k_{d,1}$ in node *u*'s *K* matrix and $k_{d,5}$ in node *v*'s matrix. In addition, consider $v_l \notin S_u$ but $S_{u,v} \neq \phi$. For example $v_l^{C-1} = u_l^{R+1}$, that means the two nodes are expected in the corresponding diagonal areas. They share $F(u) = f_{k_{d,1}}^{d,1}(u,y) = F(v) = f_{k_{d,4}}^{d,4}(v,y)$.

4.2 Indirect Key Establishment

Consider node *D* has fallen into the area of node *S*. In case that the two nodes *S* and *D* only satisfy the requirements (2), they can not establish a direct key. Intermediate nodes are required to form a key path. We assume that node *S* initiates key establishment request. It performs the following procedures:

P1: Node *S* maintains a list of key path *P*, which is used to record intermediate nodes; Initialize *P*: $P \leftarrow S$.

P2: Node *S* constructs a set of possible intermediate nodes denoted as

$$C = \{c \,|\, S_l = c_l \;\&\; d_h(c_n, D_n) \leq 1\}.$$

P3: Node *S* chooses a node *c* from the set and performs ***Inner-Area(S, c)***.

P4: Node *c* then checks if the destination node *D* is actually in its signal range. If so the algorithm terminates, otherwise go to **P3**.

P5: Recode the path from *S* to *c*.

On the premise of direct and indirect key establishment, a node has ability to communicate its neighbors securely. The node can then share secrets with remote nodes by means of various routing mechanisms such as multi-hop routing. We do not further address this issue as it is not the main idea of this paper.

5 Performance

Probability to establish direct key

Assume that the conditional probability of being in each other's signal range can be expressed as $p(v_l, u_l)$, where u_l and v_l denote location codes of the two sensors respectively. And there are $N_{cell} = N / N_A$ nodes in each area, the average number of sensors that u can directly communicate with can be estimated by $n_u = N_{cell} \cdot \sum_{\forall v_l} p(v_l, u_l)$.

If $v_l = u_l$, there are up to $n(m-1)$ nodes share a common polynomial with node u. Otherwise, if $v_l \neq u_l$ and $S_{u,v} \neq \phi$, there are up to $n \cdot m$ nodes can establish a direct key with node u based on hybrid hypercube model. Overall, the probability to establish a direct key is estimated by

$$p = p(u) = \frac{n_u^s + n_u^c}{n_u} = \frac{n(m-1) \cdot p(u_l, u_l) + nm \cdot \sum_{S_{u,v} \neq \phi} p(v_l, u_l)}{N_{cell} \cdot \sum_{\forall v_l} p(v_l, u_l)} \tag{4}$$

Figure 1a shows the probability to establish direct key given different maximum deployment error e with network scale of N =65000. In general, the smaller e is, the higher and more smooth the probability of establishing a direct key between two physically neighboring nodes. However, the larger e also leads to more sensors of neighboring areas to establish direct keys. Thus the overall probability maintains stable to a certain extent as the figure shows. Figure 1b shows the probability to establish direct keys given different network sizes.

Thus, the average key path length ignoring the factor of signal range can be estimated by $L_l = \sum_{i=1}^{r} (2i-1)p[i]$. As we only concern about $v^k / 2$ connected nodes in each subsection, the approximate average number of hops is about $v^k / 4 + 1$. Then the average number of hops can be expressed as $L_h = (v^k / 4 + 1) \cdot \sum_{i=1}^{r} (2i-1)p[i]$.

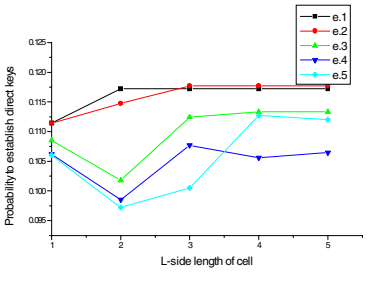


Fig. 1. a) Probability to establish direct key different deployment error

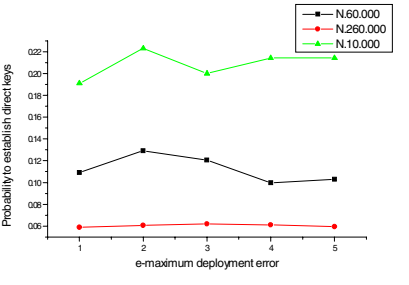


Fig. 1. b) Probability to establish direct key given different network size

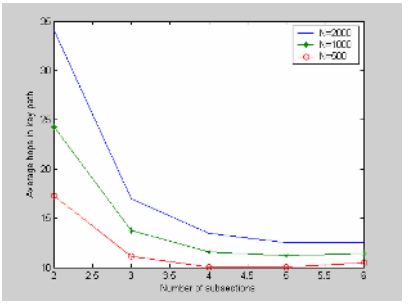


Fig. 2. Average number of hops of a key inner-area scheme

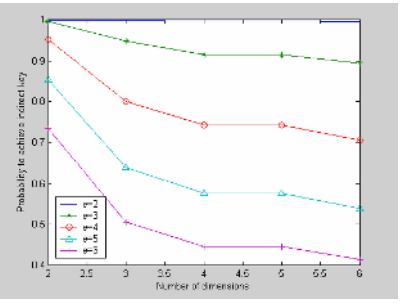


Fig. 3. Probability to achieve indirect key in path in inter-area scheme

Figure2 shows the relationship between the average number of hops and the number of subsections given different local network sizes. The number of hops drops dramatically as the number of subsections grows.

Probability to achieve indirect key in inter-area scheme

In order to satisfy the requirement $C = \{c \mid S_l = c_l \ \& \ d_h(c_n, D_n) \le 1\}$ addressed in Section 4.2, there are up to nm appropriate nodes of the local area able to establish direct key with the destination node. As a distant node called D drops into the local area, the probability that a local node is within the signal range of D is $1/e^2$ (assume that d_r=1). Then the probability that none of those nm nodes drops into the range can be estimated by $p_{lf} = (1-1/e^2)^{nm}$. Then the probability of having at least one node in the range is $p_{ls} = 1 - p_{lf}$. Figure3 shows the relationship between the probability to achieve indirect key and the number of dimensions in inter-area scheme.

6 Conclusion and Future Work

Our hypercube-based approach takes two steps to achieve pairwise key establishment. Firstly, we consider a simple situation such as a local network with densely distributed nodes. It is more reasonable to model part-area network as a hypercube, and we inspect the connectivity issue in subsection to deal with the situation of a number of nodes are out of communication. Secondly, by applying location-aware deployment knowledge, we present a hybrid hypercube model, and put emphasis on pairwise key establishment of inter-area sensor nodes. In our future work, we will study key path establishment in a small area on a Mote-based sensor network platform and evaluates its performance through real experiments.

References

1. Chee-yee Chong, Srikanta,P.Kumar. Sensor Networks: Evolution, Opportunities, and Challenges. Proceeding of the IEEE, 2003, 91(8):1247-1256.
2. Du,W., Deng,J., Han,Y.S. et.al. A Pairwise Key Predistribution Scheme for Wireless Sensor Networks, In Proceedings of 10th ACM Conference on Computer and Communication Security. 2003, 42-51.

3. Estrin,D., Govindan,R., Heideman,J., Kumar,S. Next century challenges: Scalable Coordination in Sensor Networks. Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking, Pages 263-270, 1999.
4. Eeschnaure,L., Gligor,V.D. A Key-management Scheme for Distributed Sensor Networks, In proceedings of the 9th ACM Coference on Computer and Communication Security. 2002, 41-47.
5. Blundo,C.., Desantis,A., Kutten,S., et.al. Perfectly Secure Key Distribution for Dynamic Conferences. In Advances in Cryptology-CRYPTO'92, 1993, LNCS, 740, 471-486.
6. Chan,H., Oerrig,A., Song,D. Random Key Predistribution Schemes for Sensor Networks, In IEEE Syposium on Research in Security and Privacy. 2003, 197-213.
7. Donggang Liu, Peng Ning, Rongfang Li, Establishing Pairwise Keys in Distributed Sensor Networks. ACM Transactions on Information and System Security. 2005,8(1): 41-77.
8. Donggang Liu, Peng Ning. Location-Based Pairwise Key Establishments for Static Sensor Networks. Available from http://discovery.csc.ncsu.edu/~pning/pubs/sasn03.pdf.

Key Predistribution in Sensor Networks

Guorui Li[1], Jingsha He[2], and Yingfang Fu[1]

[1] College of Computer Science and Technology, Beijing University of Technology, Beijing 100022, China
{liguorui, fmsik}@emails.bjut.edu.cn

[2] School of Software Engineering, Beijing University of Technology, Beijing 100022, China
jhe@bjut.edu.cn

Abstract. Sensor networks are widely used in a variety of commercial and military applications due to their self-organization characteristics and distributed nature. As a fundamental requirement for providing security functionality in sensor networks, key predistribution plays a central role in authentication and encryption. In this paper, we describe the hexagon-based key predistribution scheme and show how it can improve the performance of key predistribution in sensor network through the use of bivariate polynomials in a hexagonal coordinate system based on the deployment information about the expected locations of the sensor nodes. More specifically, we show that the hexagon-based key scheme can improve the probability of establishing pairwise keys between sensor nodes by more than 40% over previous schemes.

1 Introduction

With the development of wireless and microelectronics technologies, it has now become feasible to deploy a large number of low-cost, high-performance and low-power sensor nodes in a wireless sensor network. In such a network, security plays an essential role because the confidentiality, integrity and availability of the transmitted data between sensor nodes must be preserved in a hostile environment in which sensor networks are commonly used. As the basic requirement for providing security functionality, key management plays a central role in data encryption and in authentication. However, due to energy and resource constraints in sensor nodes, many ordinary security mechanisms are deemed impractical, and sometimes infeasible in sensor networks.

There are currently three types of key management schemes that are commonly used in sensor networks: trusted server scheme, self-enforcing scheme, and key pre-distribution scheme. The first type of key management scheme, i.e., the trusted server scheme, relies on a trusted server for key distribution and management, e.g., the Kerberos. This type of scheme is not very suitable for sensor networks because there is usually a lack of a trusted infrastructure in the application environments in which senor networks are used. The second type of key management scheme, i.e., the self-enforcing scheme, on the other hand, relies on asymmetric cryptography, e.g., key distribution

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 845 – 853, 2006.

and management using public key certificates. However, limited computation and energy resources in sensor nodes usually make it undesirable to use public key algorithms, such as RSA, for the sake of energy conservation. The third type of key management scheme, i.e., the key predistribution scheme, is such a scheme in which cryptographic keys are predistributed among all sensor nodes prior to deployment [1]. There have already been several key predistribution schemes in existence and we will discuss them in more details in the following discussion.

Eschenauer and Gligor proposed the basic probabilistic key predistribution scheme in which each sensor node is assigned a random subset of keys from a key pool before the network is deployed so that any two sensor nodes will have a certain probability to share at least one key [2]. Chan et al. improved the above scheme and proposed the q-composite key predistribution scheme and the random pairwise key scheme [3]. The q-composite key predistribution scheme is based on the basic probabilistic key predistribution scheme, but it requires that two sensor nodes share at least q predistributed keys as the basis for the establishment of a pairwise key between the two nodes. In the random pairwise key scheme, random pairwise keys are predistributed between a specific sensor node and a random subset of other sensor nodes. Such a scheme has the property that security compromise to a sensor node doesn't automatically lead to compromise to pairwise keys that are shared between uncompromised sensor nodes. Liu and Ning proposed a framework in which pairwise keys are predistributed by using bivariate polynomials [4]. The two also proposed two efficient instantiations, i.e., a random subset assignment scheme and a grid-based key predistribution scheme, for the establishment of pairwise keys in a sensor network. In addition, they proposed the closest pairwise key predistribution scheme and the closest polynomials predistribution scheme, which take advantage of sensor nodes' expected locations to predistribute appropriate keys to the sensor nodes and thus can improve the performance of key establishment [5]. However, all the schemes described above failed to take into consideration the information on deployment locations and signal propagation. Therefore, they lowered the probability of successful key establishment with an increase in the cost. Recently, we proposed a hexagon-based key predistribution scheme in which we took advantage of the broadcast nature of wireless communication and showed that this new scheme could improve the probability of establishing pairwise keys between sensor nodes by more than 40% over previous schemes [6].

In this paper, we first describe the hexagon-based key predistribution scheme in sensor networks in which we use the hexagon to simulate signal propagation. We then analyze the connectivity and security aspects of key predistribution schemes and show that the hexagon-based key predistribution scheme can greatly improve the probability of successful key establishment as well as the threshold security.

The rest of the paper is organized as follows. In the next section, we describe the hexagon-based key predistribution scheme. In Section 3, we analyze the connectivity and security aspects of the key predistribution system. In Section 4, we identify some related work in sensor network security. Finally, in Section 5, we conclude this paper and discuss some future research directions.

2 Hexagon-Based Key Predistribution Scheme

The hexagon-based key predistribution scheme involves the following four phases:

(1) Key predistribution phase

A key setup server would partition the target deployment field into *m* equal sized hexagons according to the hexagonal coordinate system [6] as shown in Fig. 1. Then, it builds *m* different bivariate polynomials of degree *t* over a finite field F_q and assigns these polynomials to hexagonal coordinate system randomly in order to make sure that each hexagon has a unique bivariate polynomial. For convenience, the key management server assigns a unique ID to each polynomial.

For each sensor node *i*, the key setup server first determines its home hexagon H_i where the sensor node is expected to be located and discovers the six hexagons $\{H_{ij} \mid j = 1,...,6\}$ that are adjacent to the sensor node's home hexagon. Then, the server computes $P_i(ID_i, y)$ and $\{P_{ij}(ID_i, y) \mid j = 1,...,6\}$ by evaluating hexagon H_i and $\{H_{ij} \mid j = 1,...,6\}$'s corresponding polynomial P_i and $\{P_{ij} \mid j = 1,...,6\}$ at sensor node *i*'s ID ID_i. Finally, the key setup server assigns $P_i(ID_i, y)$, $\{P_{ij}(ID_i, y) \mid j = 1,...,6\}$ and their corresponding IDs to sensor node *i* and store them into the node in order to build the pairwise keys.

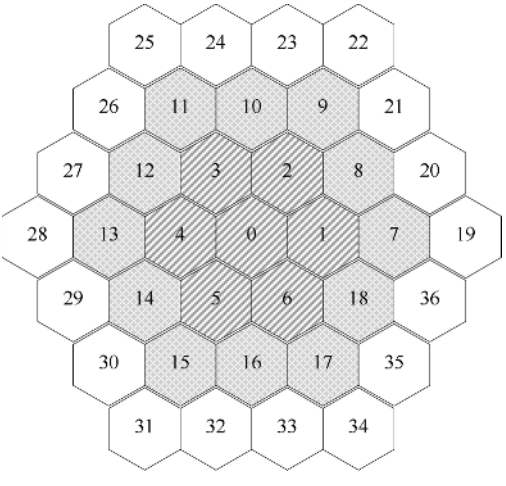


Fig. 1. The adjacent hexagons in the hexagon-based key predistribution scheme

(2) Direct key establishment phase

After deployment, if two sensor nodes wish to establish a pairwise key, they first need to identify a shared bivariate polynomial. If they can find at least one such polynomial, a common pairwise key can be established directly using the polynomial-based key establishment scheme presented in [6]. In order to find out whether they hold a shared polynomial, the two sensor nodes should exchange their polynomials' IDs. To protect

information associated with their polynomials' IDs, the two nodes may challenge each other for authentication. For example, sensor node i may broadcast an encryption list, α, $E_{ID_1}(\alpha)$, $E_{ID_2}(\alpha)$, …, $E_{ID_7}(\alpha)$ where $ID_i, i = 1,...,7$, is the IDs of the polynomials that sensor node i holds. If the other sensor node can correctly decrypts one of the $E_{ID_i}(\alpha)$ using one of its own polynomial ID_i, then they share the same polynomial ID_i and can proceed to establish a direct pairwise key using this shared polynomial.

(3) Path key establishment phase

If direct key establishment failed, the two sensor nodes can try to establish a pairwise key in the path key establishment phase. When a source sensor node broadcasts the ID of a destination sensor node, an intermediate sensor node can establish a path key for the two sensor nodes if it holds the pairwise keys with the source and with the destination sensor nodes, respectively. Otherwise, the intermediate sensor node would broadcast this message continuously until it discovers a sensor node that shares pairwise keys with the previous sensor node and with the destination sensor node, respectively. Then the path key can be established along the message broadcast path in the reverse direction.

(4) Sensor node addition and revocation phase

Some sensor nodes may be destroyed or compromised after a period of time. Then, they can no longer work properly or even lower the security of key management by disclosing the shared key polynomial information. This problem can be dealt with by adding new sensor nodes and predistributing these nodes with their own IDs along with the corresponding bivariate polynomial coefficients based on their deployed locations. Two sensor nodes can establish a direct key as long as they share at least one common bivariate polynomial. Therefore, the newly added sensor nodes can affiliate themselves into the existing sensor network seamlessly. On the other hand, each sensor node can record the compromised sensor nodes that share at least one common bivariate polynomial with itself along with their corresponding IDs, for the disclosure of bivariate polynomials will compromise the security of key management. If more than t sensor nodes that share the same bivariate polynomial are compromised, this polynomial is no longer considered to be secure. Consequently, we should remove this polynomial as well as the IDs of all sensor nodes that share the same polynomial to save memory resources.

3 Analysis

3.1 Connectivity Analysis

(1) The probability of direct key establishment

Similar to the analysis in [6], the probability of direct key establishment for any sensor node u in the hexagon-based key predistribution scheme is:

$$p_u = \frac{n_u^s}{n_u} = \frac{\sum_{C_j \in S_i} p(C_j, C_i)}{\sum_{\forall j} p(C_j, C_i)}$$

where n_u^s is the average number of sensor nodes that can establish a pairwise key with u directly, n_u is the average number of sensor nodes with which u can directly communicate, and S_i is the set of hexagons of the sensor nodes that share at least one common polynomial with sensor node u.

In the hexagon-based key predistribution scheme, each sensor node takes its deployment hexagon as the center and can share polynomials with sensor nodes deployed in its 19 adjacent hexagons. Let's assume that the sensor deployment density in hexagon is ϖ and signal propagation distance is d_r, then the probability of direct key establishment in the hexagon-based key predistribution scheme is:

$$p_u = \frac{n_u^s}{n_u} = \frac{19 \cdot \varpi \cdot \frac{3\sqrt{3}}{2} \cdot R^2}{\pi \cdot d_r^2 \cdot \varpi} = \frac{57\sqrt{3}R^2}{2\pi d_r^2}$$

where R is the diameter of the hexagon.

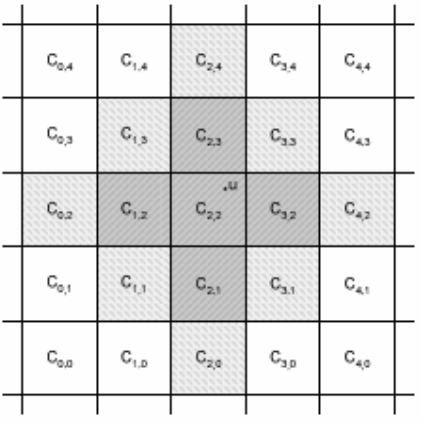


Fig. 2. The adjacent squares in a closest polynomial key predistribution scheme

In contrast, the probability of direct key establishment in a closest polynomial key predistribution scheme described in [5] is:

$$p_u' = \frac{n_u^s}{n_u} = \frac{13 \cdot \varpi \cdot L^2}{\pi \cdot d_r^2 \cdot \varpi} = \frac{13L^2}{\pi d_r^2}$$

where L is the side length of a square in a common rectangular coordinate system. As shown in Fig. 2, each sensor node can only communicate with the sensor nodes deployed in 13 adjacent squares in the common rectangular coordinate system.

To simplify our analysis, we assume that the signal propagation distance in both of these two schemes is the minimal distance between a sensor node and those that are within the signal range of the sensor node that can establish a direct pairwise key with it. Consequently, in the hexagon-based key predistribution scheme, $d_r = 3\sqrt{3}R$, whereas in the closest polynomial predistribution scheme, $d_r = \sqrt{10}L$. Therefore, the ratio between the probabilities of the two direct key establishment schemes is

$$\frac{p_u}{p_u'} = \frac{57\sqrt{3}}{26} \cdot (\frac{R}{L})^2 = \frac{57\sqrt{3}}{26} \cdot (\frac{\sqrt{10}}{3\sqrt{3}})^2 \approx 1.406 .$$

That is, the probability of successful direct key establishment in the hexagon-based key predistribution scheme is approximately 40% higher than that in the closest polynomial predistribution scheme presented in [5].

(2) The probability of two-hop path key establishment

Let's discuss the probability of path key establishment between two sensor nodes of two hops away in which it requires only one intermediate node to help establish the path key between the source and the destination sensor nodes. Similar to the analysis above, each sensor node can establish two-hop path key with sensor nodes deployed in its 61 adjacent hexagons in the hexagon-based key predistribution scheme and 41 adjacent squares in the closest polynomial predistribution scheme. So the ratio between the probabilities of two-hop path key establishment based on the two direct key establishment scheme is:

$$\frac{p_u}{p_u'} = \frac{\frac{3\sqrt{3}}{2}R^2 \cdot 61}{41 \cdot L^2} = \frac{183\sqrt{3}}{82} \cdot (\frac{\sqrt{10}}{3\sqrt{3}})^2 \approx 1.432 .$$

That is, the probability of establishing a two-hop path key in the hexagon-based key predistribution scheme is approximately 43% higher than that in the closest polynomial predistribution scheme presented in [5].

(3) The probability of multi-hop path key establishment

In general, the number of hexagons covered in the *i*-hop path key establishment based on the hexagon-based key predistribution scheme and the number of squares covered in the *i*-hop path key establishment based on the closest polynomial key predistribution scheme can be calculated using the following formulas, respectively:

$$\begin{cases} x_0 = 1 \\ x_1 = 19 \\ x_{i+1} = 2x_i - x_{i-1} + 24 \end{cases} \quad \begin{cases} y_0 = 1 \\ y_1 = 13 \\ y_{i+1} = 2y_i - y_{i-1} + 16 \end{cases} .$$

So the ratio between the probabilities of these two path key establishment scheme is:

$$\frac{p_u}{p_u'} = \frac{\frac{3\sqrt{3}}{2}R^2 \cdot x_{i+1}}{41 \cdot L^2 \cdot y_{i+1}} = \frac{5\sqrt{3}x_{i+1}}{9y_{i+1}} .$$

From Fig. 3, we can see that the trend in the case of hexagons goes up much faster than that in the case of squares. With an increase in the hop count, the probability of key establishment in the hexagon-based key predistribution scheme is therefore much higher than that in the closest polynomial predistribution scheme.

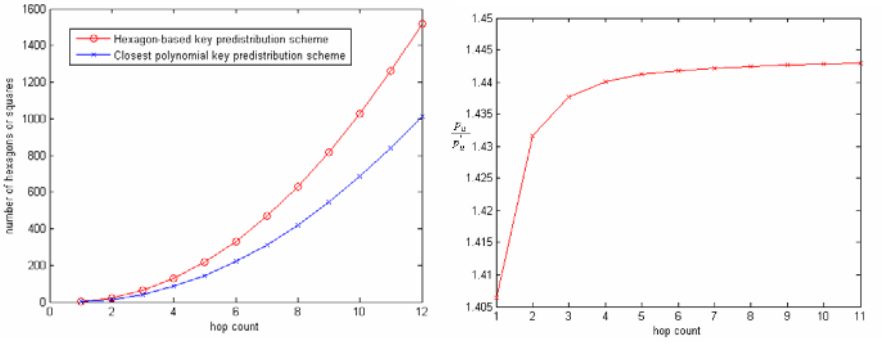


Fig. 3. (a) Comparison between the number of hexagons and that of squares in the two key predistribution schemes assuming the same hop count; (b) Relationship between the ratio of the key establishment probability in the two key predistribution schemes and the hop count

We can further see from Fig. 4 that the key establishment coverage ratio in the three-hop case will increase over that in the one-hop case. Actually, when the one-hop coverage ratio is up to 10%, the three-hop key establishment can cover the entire sensor deployment area. So in general, there should be no more than three hops for indirect key establishment between any two sensor nodes.

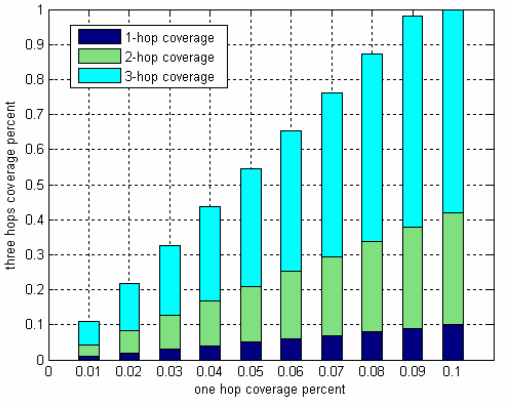


Fig. 4. Relationship between one-hop coverage ratio and three-hop coverage ratio in the hexagon-based key predistribution scheme

3.2 Security Analysis

According to the analysis result of the polynomial-based key predistribution scheme, if no more than t shares of a bivariate polynomial are disclosed, an attacker would not know the uncompromised pairwise keys established using this polynomial. Thus, the security of the hexagon-based key predistribution scheme depends on the average number of sensor nodes that share the same polynomial. Assume that there are m nodes on average in the signal range of each sensor node, the density of the sensor can be estimated to be $\varpi = \frac{m}{\pi d_r^2}$. Thus, the number of sensor nodes that share at least one common polynomial in the hexagon-based key predistribution scheme is $N_s = \frac{m}{\pi d_r^2} \cdot \frac{3\sqrt{3}}{2} R^2 \cdot 7 = \frac{21\sqrt{3} m R^2}{2\pi d_r^2}$. As long as $N_s \le t$, the hexagon-based key predistribution scheme is compromise-resistant.

We assume that a fraction p_c of sensor nodes in the sensor network have been compromised. Thus, among N_s sensor nodes that hold the same polynomial shares, the probability that i sensor nodes have been compromised can be estimated to be $P_c(i) = \frac{N_s!}{(N_s - i)! i!} p_c^i (1 - p_c)^{N_s - i}$. Thus, the probability that the bivariate polynomial is compromised is $P_c = 1 - \sum_{i=0}^{t} P_c(i)$. For any pairwise key established between uncompromised sensor nodes, the probability that it could be compromised is the same as P_c.

4 Related Work

Nowadays, there are many studies in sensor network security, which are mostly on key management, authentication, and vulnerability analysis. Other than the key predistribution scheme presented in [2-6], Perrig et al. developed a security architecture for sensor networks, which includes SNEP, a security primitive building block, and a broadcast authentication technique μTESLA [7]. Liu and Ning extended this technique to a multilevel key chain method to prolong the time period covered by a μTESLA instance [8]. Wood and Stankovic identified a number of DoS attacks in sensor networks [9].

5 Conclusion and Future Work

In this paper, we described the hexagon-based key predistribution scheme and showed that this scheme could increase the probability of key establishment by over 40% over

previous schemes. Our future work would focus on the development of methods and schemes that can be used to adjust the polynomial distribution by taking into consideration the difference between expected deployment locations and the actual deployment locations for the sensor nodes.

References

[1] Du, W., Deng, J., Han, Y., Chen, S., Varshney, P.: A key management scheme for wireless sensor networks using deployment knowledge. In *Proc. IEEE Infocom*, March 2004, 586-597

[2] Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In *Proc. 9th ACM Conference on Computer and Communications Security*, November 2002, 41-47

[3] Chan, H., Perring, A., Song, D.: Random key predistribution schemes for sensor networks. In *Proc. IEEE Symposium on Research in Security and Privacy*, 2003.

[4] Liu, D., Ning, P.: Establishing pairwise keys in distributed sensor networks. In *Proc. 10th ACM Conference on Computer and Communications Security*, October 2003, 52-61

[5] Liu, D., Ning, P.: Location-based pairwise key establishments for static sensor networks. In *Proc. 2003 ACM Workshop on Security in Ad Hoc and Sensor Networks*, 2003, 72-82

[6] Li, G., He, J., Fu, Y.: A Hexagon-based key predistribution scheme in sensor networks. In *Proc. International Workshop on Wireless and Sensor Networks*, August 2006.

[7] Perrig, A., Szwczyk, R., Wen, V., Culler, D., Tygar, D.: Spins: Security protocols for sensor networks. In *Proc. 7th Annual International Conference on Mobile Computing and Networks*, July 2001.

[8] Liu, D., Ning, P.: Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In *Proc. 10th Annual Network and Distributed System Security Symposium*, February 2003, 263-276

[9] Wood, D., Stankovic, J.A.: Denial of service in sensor networks. In *IEEE Computer*, Vol. 35, No. 10, October 2002, 54-62

A Strong Key Pre-distribution Scheme for Wireless Sensor Networks

Taeyeon Kim[1] and Gicheol Wang[2]

[1] Dept. of Computer Science and Information Communications,
Seonam University, Namwon, Jeonbuk, 590-711, Republic of Korea
kcopper7@hanmail.net
[2] CAIIT, Chonbuk National University, Jeonju,
Jeonbuk, 561-756, Republic of Korea
gcwang@dcs.chonbuk.ac.kr

Abstract. A prerequisite for secure communication between two sensor nodes is that the nodes should share a session key to bootstrap their trust relationship. The open problems are how to set up a session key between communicating nodes, and how to minimize any information about the keys disclosed to the other side during key agreement. Almost schemes one of the existing protocols cannot perfectly solve above problems due to some drawbacks. Accordingly, we propose a strong key pre-distribution scheme having the following merits. First, it supports authentication services. Second, as each node can only find w ($=\lceil W/2 \rceil$) keys of secret keys (i.e. W) that shared with other side, without revealing the unshared key information, it substantially improves resilience of network against node capture. Our scheme is based on the MRS scheme. Performance and security analyses have proven that our scheme is suitable for sensor networks in terms of performance and security aspects.

1 Introduction

Ubiquitous computing environment provides users with information access anytime and anywhere. In particular, sensor networks will be broadly deployed in real world and widely utilized for various applications, including armaments defense, environmental observation, health care, and so on. Usually Wireless Sensor Network (WSN) requires no centralized center or fixed network infrastructure, and can be deployed quickly and inexpensively as needed. The sensor nodes collaborate to collect, process, analysis, and disseminate the sensed data in hostile environments. However, the individual sensor node suffers from limited resources, battery, memory, processor, network bandwidth, and so on [1, 2].

A prerequisite for secure communication between two sensor nodes is that the nodes share a session key to bootstrap their trust relationship. It is natural that common key management techniques using asymmetric cryptographic algorithms are not appropriate for WSN due to limited resources; it is natural to use symmetric cryptographic algorithms in WSNs because they are relatively fast and induce low cost for cryptographic processes.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 854 – 863, 2006.

Almost all of key pre-distribution schemes [2-9] assumed that a random graph *G(n, P)* is a graph of *n* nodes for which the probability that a link exists between two nodes is *P*. And each one fully trusts the other side during the key pre-distribution. Without a doubt, nodes disclose their key information to neighboring nodes which held a random subset of keys of the key pool. At any rate, the existing protocols, which are based on random key pre-distribution, cannot perfectly satisfy requirement of the key management due to some drawbacks. The open problems are how to authenticate the key information of communicating nodes, how to securely set up a session key between communicating nodes, and how to minimize the amount of disclosed information about the keys to the other side.

To solve the first problem, almost all the schemes so far rely on three phases as followings: key pre-distribution, shared-key discovery and path-key establishment. But the second and the third problems are big challenges that have not yet solved.

In the existing schemes, if two nodes share at least one key, they consider each other to be worthy of confidence and generate a session key for further communication between them. But if an adversary accidentally generates the same key(s) that any key(s) in the key pool to impersonate one of legitimate nodes, he can get a session key after completing shared-key discovery phase.

Recently, Chan [5] proposed a key agreement scheme where each node can find keys shared with a communicating node without revealing the unshared keys. However, in this scheme, some security problems are discovered. This paper proposes a strong key pre-distribution scheme which resolves the second and third problem caused during the key discovery phase in WSNs.

The rest of the paper is organized as follows. In section 2, some related works on key pre-distribution scheme in WSNs are covered. In section 3 and 4, we review the modified Rivest's scheme and present a new secure key pre-distribution scheme for WSNs. In section 5, the performance results and security analyses are presented. Finally, section 6 concludes this paper.

2 Related Works

In this section, we review some of key pre-distribution schemes for ad hoc networks or WSNs. Eschenauer et al. first proposed a key pre-distribution scheme for WSNs [6]. Each node randomly picks a set of keys from a key pool before the node's deployment. After all of the nodes are deployed, each pair of its neighbors attempts to find a common key. The path-key establishment phase is used for assigning a path-key to selected node pairs in wireless communication range that do not share a common key.

Du et al. proposed a new key pre-distribution scheme using Blom's scheme and basic key pre-distribution scheme [7]. Their concerns were to propose a scheme which substantially improves the resilience of the network as compared with the existing schemes.

Chan et al. proposed three random key pre-distribution schemes for solving the security bootstrapping problem in WSNs [8]. Their basic idea is to require any two nodes to share at least q common keys for establishing a secure link and to perform node-to-node authentication by storing the identity (ID) of the nodes which hold the

same keys. And they described multipath key reinforcement to strengthen the security of an established link key.

Du et al. described a random key pre-distribution scheme that uses deployment knowledge [9]. It takes advantage of the location information to improve the key connectivity and avoids unnecessary key assignments. But it is not usually easy to guarantee the knowledge of nodes' expected location.

Chan [5] proposed a new random key pre-distribution scheme using privacy homomorphism and Rivest's scheme [10]. His basic idea is that in the key discovery phase, a node avoids leaking any information about the keys that the other side does not have.

Most of the above schemes cannot perfectly satisfy the requirement of the key management due to some drawbacks. In security aspects, a disclosure of secret key itself and various attacks by illegitimate nodes cause a lot of damage to secure communication among the nodes.

3 Overview of Modified Rivest's Scheme

Before we describe our scheme, we review the modified Rivest's scheme which is based on a scheme in [10]. A detailed scheme is described in [5]. The key pre-distribution phase ensures that each node is assigned a random subset of keys, *m,* from a key pool before deployment. And in shared-key discovery phase, each node finds the keys shared with the other side node. It does not disclose any information about the keys that the other side does not have. MRS ensures which the componentwise addition and multiplication (mod (p×q)) of the ciphertexts are the same as the encrypted values of the addition and multiplication of the corresponding plaintexts, and a value should have a number of different possible representations in the ciphertext domain. The algorithm for the shared-key discovery phase is performed as follows.

When MRS scheme is used in the shared-key discovery phase, each node makes use of a polynomial expression. For example, suppose that Alice wants to find out the common keys with Bob, and Alice has the key set $A = \{a_1, a_2, ..., a_m\}$ and Bob has the key set $B = \{b_1, b_2, ..., b_m\}$. Alice forms a polynomial expression:

$$f_A(x) = (x - a_1)(x - a_2)...(x - a_m) = x^m + A_{m-1}x^{m-1} + ... + A_1x + A_0 \quad (1)$$

Then she sends the encrypted coefficients of $f_A(x)$ to Bob.

Alice $\rightarrow$ Bob : $E^K(A_0), E^K(A_1), ..., E^K(A_{m-1})$, where K is a secret key which she has only.

Bob forms the following polynomial expression using the received message:

$$f_A^{'}(x) = x^m + E^K(A_{m-1})x^{m-1} + ... + E^K(A_1)x + E^K(A_0) \quad (2)$$

To generate a list of encrypted values (i.e., $rE^K(B_0), rE^K(B_1), ..., rE^K(B_{m-1})$), he applies his keys to the expression (2). Here, $E^K(B_i)$ is $f_A^{'}(B_i)$ and r is a random number.

Bob → Alice : $rE^K(B_0), rE^K(B_1), ..., rE^K(B_{m-1})$

She applies $D(rE^K(B_i))$ and can get $rf_A(b_i))$ for $0 \le i \le m-1$. Since she has no knowledge about r, she does not know what b_i is. But, if anything in $rf_A(b_i))$ is zero, she knows that two nodes share at least one key. Otherwise, she knows that they share no keys with each other.

3.1 Notation

To describe the proposed scheme and cryptographic algorithm, we use the following notations.

- n, C : size of network and the number of shared keys between any two node.
- z, s, m : size of the key space, the key pool and the key chain respectively
- SK : session key generated between authorization nodes
- KA, KB : secret key of node A and B
- $E^K(M)$, $D^K(M)$: message M encrypted and decrypted with key K
- $h()$: one-way hash function
- p, q : prime numbers
- r (or r_i), R_{i1}, R_{2i}, $R_{i1}^{'}$, $R_{2i}^{'}$: random numbers($\neq 0$), for $1 \le i \le m$.

4 Proposed Key Pre-distribution Scheme

4.1 Key Pre-distribution Phase

In the initialization phase, the base station picks a random key pool out of the total possible key space. Also, the key information in the key pool is combined a secret key K_i in the key space with a one-way hash function h_i, (K_i, h_i), $0 \le i \le z-1$. Each node randomly picks a key chain (i.e., $(K_i, h_i), i = 1, ..., m$) the key pool before deployment. The key chain is utilized to generate a session key between two nodes during the key discovery phase. And the hash function is utilized to authenticate the secret keys K_i. It is for the sake of decreasing the possibility that malicious nodes intentionally generate a random key chain.

4.2 Shared-Key Discovery Phase

During shared-key discovery, each node needs to find whether it shares any key with its neighbors. To do this, each node generally generates m non-linear equations with the secret keys it carries as expression (1) and broadcasts the message containing the encrypted coefficients of $f_A(x)$.

But in this paper, we make use of negotiatory keys so as to enhance security. The generation of negotiatory keys is as follows. For example, let Alice has the key set

$A = \{a_1 (= a_{11} \parallel a_{12}), a_2 (= a_{21} \parallel a_{22}), ..., a_m (= a_{m1} \parallel a_{m2})\}$ and Bob has the key set $B = \{b_1 (= b_{11} \parallel b_{12}), b_2 (= b_{21} \parallel b_{22}), ..., b_m (= b_{m1} \parallel b_{m2})\}$. We yield a negotiatory key by concatenating the first half of the key and a random bit-string and in reverse (see Fig. 1). That is, the half set of Alice's negotiatory keys is as the following (i.e. $\{s_{11} (= a_{11} \parallel R_{11})), s_{21} (= a_{21} \parallel R_{21})), ..., s_{m1} (= a_{m1} \parallel R_{m1}))\}$). The other half set is $\{s_{12} (= R_{12} \parallel a_{12}), s_{22} (= R_{22} \parallel a_{22}), ..., s_{m1} (= R_{m2} \parallel a_{m2})\}$. Similarly, the first half set of Bob's negotiatory keys is as $\{t_{11} (= b_{11} \parallel R_{11}^{'}), t_{21} (= b_{21} \parallel R_{21}^{'}), ..., t_{m1} (= b_{m1} \parallel R_{m1}^{'})\}$. Also, the second half set of Bob's keys is consists of the following elements (i.e. $\{t_{12} (= R_{12}^{'} \parallel b_{12}), t_{22} (= R_{22}^{'} \parallel b_{22}), ..., t_{m2} (= R_{m2}^{'} \parallel b_{m2})\}$).

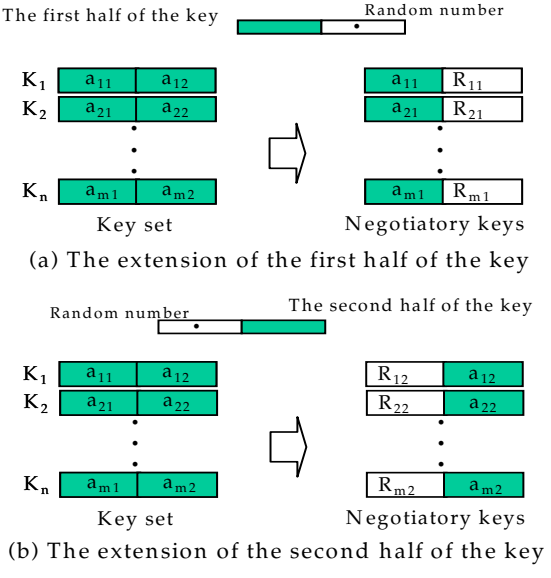


Fig. 1. The generation of negotiatory key

Next we describe the way how two nodes calculate their session key (see Fig. 2). The detailed description is as follows.

$$f_A(x) = (x - s_{11})(x - s_{21})...(x - s_{m1}) = x^m + A_{m-1}x^{m-1} + ... + A_1 x + A_0 \quad (3)$$

$$f_A^{'}(x) = x^m + E^{KA}(A_{m-1})x^{m-1} + ... + E^{KA}(A_1)x + E^{KA}(A_0) \quad (4)$$

$$f_B(x) = (x - t_{12})(x - t_{22})...(x - t_{m2}) = x^m + B_{m-1}x^{m-1} + ... + B_1 x + B_0 \quad (5)$$

$$f_B^{'}(x) = x^m + E^{KB}(B_{m-1})x^{m-1} + ... + E^{KB}(B_1)x + E^{KB}(B_0) \quad (6)$$

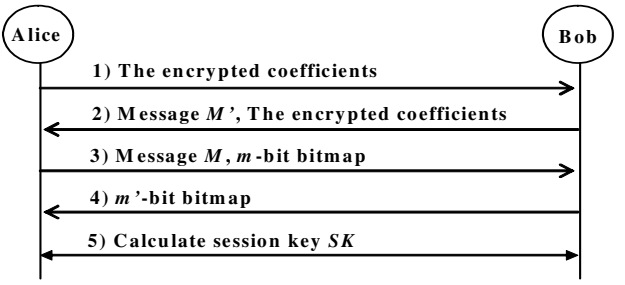


Fig. 2. The generation of session key

1) Alice calculates encrypted coefficients of $f_A(x)$ in expression (3), $E^{KA}(A_0), E^{KA}(A_1), ..., E^{KA}(A_{m-1})$ and sends them to Bob.
2) ⓐ Bob applies them to expression (4) and gets $f_A^{'}(t_{i1})$, for $1 \le i \le m$. To strengthen security, Bob chooses random numbers $r_i^{'}$ and calculates $M^{'} = r_1^{'} f_A^{'}(t_{11}), r_2^{'} f_A^{'}(t_{21}), ..., r_m^{'} f_A^{'}(t_{m1})$.
   ⓑ As the above, he calculates encrypted coefficients of $f_B(x)$ in expression (5), $E^{KB}(B_0), E^{KB}(B_1), ..., E^{KB}(B_{m-1})$.
   ⓒ He sends $M^{'}$ and the encrypted coefficients to Alice.
3) ⓐ Alice decrypts $M^{'}$, $D^{KA}(r_i^{'} f_A^{'}(t_{i1}))$, and calculates an *m*-bit bitmap with 1 at bits where $D^{KA}(r_i^{'} f_A^{'}(t_{i1}))$ is 0 and 0 elsewhere. A 1 at the *i*-th bit indicates to Bob that she also has t_{i1}. To enhance security, if the number of bits with 1 in *m*-bit bitmap (i.e. *W*) is more than 1, she randomly adjusts it to *w* ($= \left\lceil \frac{W}{2} \right\rceil < m$).
   ⓑ She applies the other side's encrypted coefficients to expression (6) and gets $f_B^{'}(s_{i2})$, for $1 \le i \le m$. Alice chooses random numbers r_i and calculates $M = r_1 f_B^{'}(s_{12}), r_2 f_B^{'}(s_{22}), ..., r_m f_B^{'}(s_{m2})$.
   ⓒ She sends M and an *m*-bit bitmap to Bob.
4) ⓐ Bob decrypts M, $D^{KB}(r_i f_B^{'}(s_{i2}))$ and calculates an *m'*-bit bitmap with 1 at bits where $D^{KB}(r_i f_B^{'}(s_{i2}))$ is 0 and 0 elsewhere. To enhance security, if the number of bits with 1 in *m'*-bit bitmap (i.e. *W'*) is more than 1, he randomly adjusts it to *w'* ($= \left\lceil \frac{W^{'}}{2} \right\rceil < m$).
   ⓑ He sends the *m'*-bit bitmap to Alice.
5) Each node generates a session key using both *m*-bit and *m'*-bit bitmap. That is, a new session key *SK* is generated as the hashed value of the concatenation of

shared keys (i.e. $SK = h(K_1 \| K_2 \| ... \| K_{w'})$). Where hash function $h()$ is that corresponds to secret key K_1.

The path-key establishment phase is omitted since it is assumed that our scheme employs the same protocol as Eschenauer and Gligor proposed.

5 Performance and Security Analyses

5.1 Performance Analyses

Our approach is compared with MRS proposed by Chan [5]. In our simulations, we induced two metrics to evaluate the availability and the security of the proposed scheme. One metric is the actual probability that any two neighboring nodes share at least one key during a key agreement phase. The other metric is the rate that all session keys are exposed to an attacker under the existence of one compromised node. For the sake of presentation, our scheme is hereafter referred to as SKS (Secure Key agreement Scheme).

The network model for our simulation is assumed as follows; (a) 200 nodes were randomly placed in a 100m × 100m area, (b) the length (r) of the key chain varied in 2, 6, and 10, (c) the number of cases that, in key pool, the first half of key is same to others is varied in 0% and 30%.

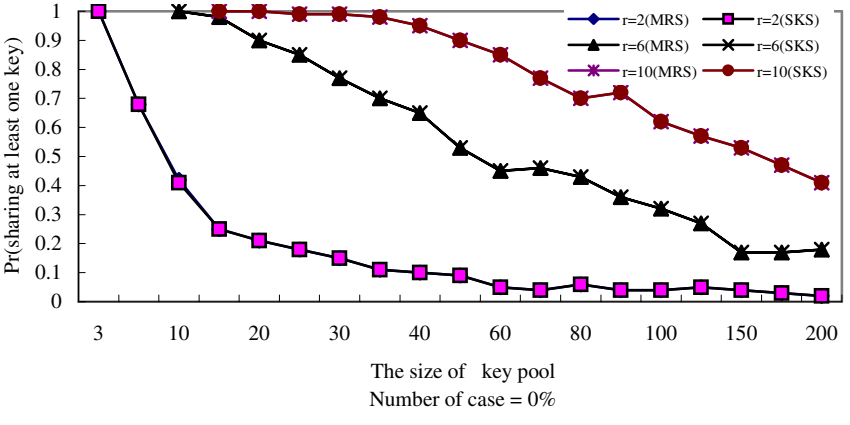


Fig. 3. Key sharing probability vs. key pool size (zero match case)

As shown Fig. 3 through 4, as the size of key pool increases, the probability that any two neighboring nodes share at least one key also decreases. In both schemes, if the first half of the keys is not the same to others', the key sharing probability is identical (see Fig. 3). However, as the cases that the first half of keys are same to others increased to 30% (See Fig. 4), the key sharing probability makes a little difference between two schemes.

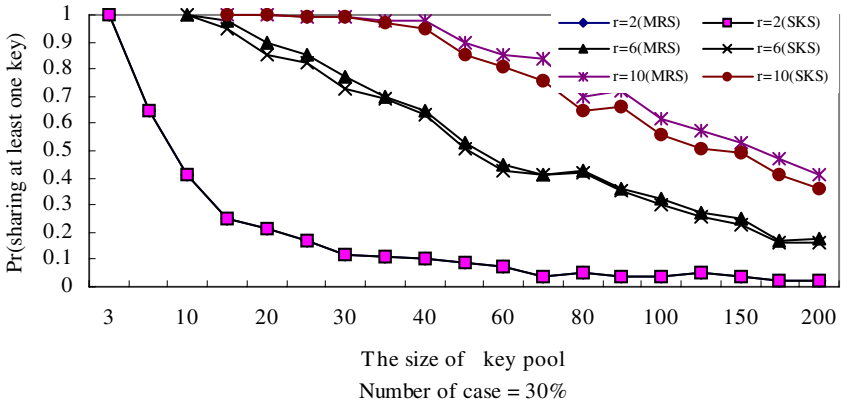


Fig. 4. Key sharing probability vs. key pool size (30% match case)

5.2 Security Analyses

5.2.1 Reducing the Number of Disclosing Shared Keys

Before completing the MRS protocol, malicious node can know how many keys of the key chain are in common with the other side without the other side's response. In the worst case, it is not difficult to guess any shared keys while the ratio of the shared keys to unshared keys is by far higher than the reverse of it. Eventually, other side may suffer from potential attacks even if the shared-key discovery is completed before finishing an attack.

Also, after finishing the shared-key discovery phase in MRS, each pair of nodes can know all the shared keys but they cannot know unshared keys. However, if one node of them is compromised by adversaries later, the other side also is easy to suffer from attacks by malicious nodes. To surmount these problems, we made use of negotiatory keys instead the secret keys that are generally used by the existing schemes and mechanism which restricts the number of disclosed shared keys. Consequently, if some nodes are captured, the probability that a session key between any two nodes is affected by malicious nodes is decreased considerably.

5.2.2 Key Authentication

Upon and after network initialization, in order to increase the communication and computation overhead of networks, a malicious node can broadcasts a random key chain falsified by itself to neighboring nodes. If any keys in the falsified key chain are in common with the other side, the attacker can establish a secure link with the legitimate node. However, since our scheme makes use of negotiatory key to provide the authentication of key information, it guarantees the key authentication. Even if an attacker luckily generates a key shared with a legitimate node, it cannot generate a session key for further communication between two nodes. This is because it has no corresponding one-way hash function.

5.2.3 Resilience Against Node Capture

We calculated the exposure rate of secret keys after a key agreement under the existence of one compromised node by an attacker. When the first half of the keys is not the same to others' or the first half of keys are same to others increased to 30%, Pr_{MRS}(any two neighboring nodes share at least one key) and Pr_{SKS}(any two neighboring nodes share at least one key) is identical or makes a little difference respectively (see Fig. 3,4). But when any node is compromised, SKS is fewer than MRS in the aspect of the number of the exposed secret keys. This is because his neighboring nodes do not disclose any information about the keys that the node does not have and do disclose at most $c(=\left\lceil \frac{C}{2} \right\rceil < m)$ keys of C shared keys.

And let be the session key *SK* for secure communication between two sensor nodes that are not compromised. When any node other than these two nodes is compromised, the probability that in SKS and MRS, *SK* is not among those keys carried by this compromised node are $(1-\frac{c}{w})$ and $(1-\frac{C}{w})$ respectively. When x nodes are compromised, the probability that in SKS and MRS, *SK* is exposed is $1-(1-\frac{c}{w})^x$ and $1-(1-\frac{C}{w})^x$ respectively. Therefore, this indicates that our scheme is more secure than the MRS scheme because $1-(1-\frac{c}{w})^x \le 1-(1-\frac{C}{w})^x$.

6 Conclusions

In this paper, we proposed a strong shared key pre-distribution scheme having the following merits. First, it supports authentication services. Second, as each node can only find w $(=\lceil W/2 \rceil)$ keys of secret keys (i.e. W) that shared with other side, without revealing the unshared key information, it substantially improves resilience of network against node capture. We made use of different random numbers instead of a random number. Consequently, the proposed scheme guarantees that two nodes agree a session key in a secure method and provides the robustness against the compromise of nodes. Simulation results have proven that it does not reduce the key sharing probability between any two nodes, providing the high robustness against a node compromise. Also, security analyses indicate that it is more secure than the basic MRS scheme. Judging from the security and availability of the proposed scheme, our protocol is extremely suitable for WSNs. In our future work, we will study the communication and computational overhead caused by the proposed scheme and devise an improved scheme for reducing the overheads.

References

1. Cam, H., Ozdemir, S., Muthuavinashiappan, D., Nair, P.: Energy-Efficient Security Protocol for Wireless Sensor Networks. Proc. of IEEE VTC Fall 2003Conference, Oct. 4-9, Orlando, (2003) 2981-2984

2. Zhu, S., Xu, S., Setia, S. and Jajodia S., Establishing Pairwise Keys for Secure Communication in Ad Hoc Networks: A probabilistic Approach. Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP'03), (2003) 1-10
3. Hwang, J. and Kim, Y., Revisiting Random key Pre-distribution Schemes for Wireless Sensor Networks. ACM Workshop on Security of Ad Hoc and Sensor Networks SASN'04, (2004) 43-52
4. Liu, D., Ning, P., Du, W.: Group-Based Key Pre-distribution in Wireless Sensor Networks. Proc. of 10th ACM Conference on Computer and Communications Security (CCS'03), (2003) 11-20
5. Chan, A.C-F.: Distributed Symmetric Key Management for Mobile Ad hoc Networks. IEEE INFOCOM (2004) 2414-2424
6. Eschenauer, L., Gligor, V.D.: A Key-management Scheme for DistributedSensor Networks. Proc. of the 9th ACM Conference on Computer and Communications Security, (2002) 41-47
7. Du, W., Deng, J., Han, Y.S., Varshney, P.K.: A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks. Proc. of ACM Conference on Computer and Communications Security (CCS'03) (2003) 42-51
8. Chan, H., Perrig, A., Song, D.: Random Key Predistribution Schemes for Sensor Networks. IEEE Symposium on Research in Security and Privacy, May 11-14 (2003) 197-213
9. Du, W., Deng, J., Han, Y.S., Varshney, P.K.: A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge. IEEE INFOCOM (2004) 586-597
10. Rivest, R.L., Adleman, L., Dertouzos, M.L.: On Data Banks and Privacy. In Foundations of Secure Computation, eds. R. A. DeMillo et al., Academic Press, (1978) 169-179.

Cooperative Public Key Authentication Protocol in Wireless Sensor Network

DaeHun Nyang and Abedelaziz Mohaisen*

Information Security Research Laboratory *of* InHa University
Inofmration Technology & Telecommunications Graduate School
253 YongHyun-dong, Nam-gu, Incheon 402-751, Korea
nyang@inha.ac.kr, asm@seclab.inha.ac.kr
http://seclab.inha.ac.kr

Abstract. Recent measurements for Public Key Cryptography (PKC) protocols on 8-bit wireless sensor nodes showed optimistic results. It has been shown that Elliptic Curve Cryptography (ECC) is quite applicable to WSN. Still, PKC is much expensive in terms of computation and memory compared by the Symmetric Key Cryptography (SKC). In addition, in PKC, each public key needs to be authenticated before it's used. We believe that sooner or later, PKC will be widely deployed in WSN. Therefore, we present a cooperative distributed public key authentication scheme that does not require any cryptographic overhead. In our scheme, each node is let to store a few number of hashed keys for other nodes. When a public key authentication is required, nodes who store this key help in authenticating it in a distributed and cooperative way. We consider the constrained resources of the sensor node. Additionally, we extend our scheme to fit with small range of authentication error.

Keywords: Public Key Authentication, Cooperative Protocol, Voting.

1 Introduction

Wireless Sensor Network (WSN) is a resulting successful mergence of different technologies. Advancements in different fields including microelectronics, semiconductors, networking, signal processing and many others led to this invention. The WSN consists of large number of inexpensive and resources constrained sensor nodes which work in a cooperative data-forwarding method to perform some sensing tasks. Sensors communicate in peer-to-peer mechanism in an open air environments that enables any man-in-the-middle (MITM), Sybil or even node replication attacks [7].

The growth of WSN applications which involved many ranging sensitive environments brought the necessity to provide security rules to guard the communication traffic between the sensor nodes. For more than five years, security research in WSN was limited to Symmetric Key Cryptography (SKC) protocols that require a key distribution. Notwithstanding of SKC limitations on the

* This work was supported by INHA University Research Grant.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 864–873, 2006.

side of the connectivity and resiliency, it showed a fastness and resources usage feasibility that is suitable for resources constrained sensor nodes. For the same reason, PKC wasn't used or even researched in WSN. Recently, some works on the PKC protocols (e.g. ECC, RSA) evaluation and efficiency measurements on sensor node platform showed optimistic results[10,11,12,13]. Based on recent researches, we believe that the advancement of processors equipped to the sensor nodes and the improvement of the PKC protocols will make it possible to take the advantage of the PKC into the WSN security. Once PKC is used in WSN, the resiliency and connectivity problems will not exist anymore since the PKC behaves perfectly in both sides (i.e. connectivity and resiliency equal 100%). [1] Recalling that, sooner or later PKC will be deployed in WSN. Therefore, Public Key Authentication problem should be solved. In this paper, we state the problem, briefly present the related work, and based on this we present our solution.

1.1 Problem Statment

PKC (*e.g.* RSA[9], ECC[16]) operates in a way that no need for a node to know other node's private key when encrypting a message for it. However, before the encryption, a node i requires node's j public key to encrypt a message for it. The most significant problem therefore is to determine whether a received key is really belonging to node j or not. One possible naive solution is to store all of the network public keys in each node that requires $(N-1) \times P$ bits/node for a network and keys of size N, P respectively, when two key authentication is required, the received key is simply searched in the keys set of the receiver node. Another solution is to store the keys corresponding hashed values which cost $(N-1) \times L$ bits/node where L is the length of the hashed key. It's clear that both of the solutions are inefficient in both memory and computation.

1.2 Related Work: Symmetric Key Cryptography (SKC)

There are intensive studies on SKC in WSN. Many schemes to secure WSN were developed. Perrig *et al.* developed SPINS that uses μTesla as a building block [8]. Additionally, for efficient SK management, many techniques and mechanisms were developed. Eschenauer-Gligor[5] proposed a key distribution scheme based on the random graph theory. In this scheme, a pair of node can establish a key when they have a shared key in their key ring. Chan et al. proposed q-composite scheme [3] based on [5]. In q-composite, a pair of nodes can communicate only they share q number of common keys in their key's ring. Du *et al.* also developed several schemes based on [1]. In [4], a similar results as in [6] was independently achieved to distribute pairwise keys for WSN. Liu *et al* [6] developed several schemes based on the Symmetric Bivariate Polynomial Protocol [2]. Moreover, different schemes based on [4,6,1,2] was developed using the deployment knowledge to reduce the used resources.

[1] Currently, BTnode[17] is equipped by ATMega128L which operates at 8 MHz and Crossbow mote is equipped by ATmega128 which dually operates at 8,16 MHz[18].

1.3 Related Work: Public Key Cryptography (PKC)

The recent results of the PKC protocols on sensor nodes showed relevant acceptable efficiency. In Gura *et al* work [10], practical measurements for ECC[16] and RSA[9] signatures verification was obtained. It was shown that ECC signature verification consumes 1.62 *ms* on the 8-bit ATmega128 processor which operates at 8 MHz. An extension of [10] on PKC protocols' enery consumption was developed in [11]. Watro *et al.* developed another limited PKC architecture with a practical evaluation of consumed resources per sensor node TinyPK [12]. Key distribution in TinyOS based on Elliptic Curve Cryptography (ECC) with real measurement and evaluation was also considered in Malan's *et al* work [13].

To the best of our knowledge, the public key authentication in WSN has been studied in a unique scheme by Du *et al*[15]. In this scheme, Merkle hash tree is used. Merkle hash tree is a binary tree of N leaves that represent the different node's hashed keys. Each internal parent till the root stores a hashed value of its corresponding children data block (*i.e.* using SHA1 of 160-bit). In [15], each node stores $log_2(N) + 1$ hashed values (which are selected from the node to the root of the tree). Once Bob's key authentication is required by Alice, Alice receives Bob's hashed values and public key. Locally, Alice perform SHA1 hashing for the received value and compares the resulting root with his own root. Depending on the equality of the resulting hashed value and the store root, Alice can determine whether Bob's key is real or not. Using the deployment knowledge, Merkle tree is split into sub trees (*i.e.* Merkle forest) to reduce the used memory per node. Therefore, each split reduces the used memory by one key.

1.4 Our Contribution

In this paper, we present a novel scheme for authenticating the public key in WSN. Our scheme uses distributed and cooperative mechanism to perform such a need. Our contribution relies in that we don't use any cryptographic operations to authenticate a key. In addition, each node stores a limited number of hashed keys for a set of different nodes that limits the used memory.

2 Network Model: Assumptions

A.1 Our network model assumes that there are hundreds of sensor nodes within the same radio range. Note that, the network size is dependent on the MAC layer and not Physical layer, which rationalizes the assumption.

A.2 The different nodes in the network can overhear every traffic between any pair of nodes. In the normal case, this overhearing is necessary to make the nodes decide whether they have to rely a frame or not even when it is not for them.

A.3 Static data is deliverable from one node to another while attackers reside in the same geographical area or the same radio coverage range.

A.4 Interception and modification of frame(s) are possible during one hope transmission only when the attacker knows who is about to send.

A.5 Attackers have the ability to inject forged frames.

3 Basic Protocol

To reduce the used resources in the sensor node in our protocol and to resist the flooding attack as well, some of the authentication decision is held in MAC layer. To enable such a decision, the following is a list of our modification on the MAC frame:

- Since we use specific frames for authentication, we added one bit flag **AC** to discriminate the authentication frames from the normal frames. Additionally, this bit will be used as a check bit for discarding any extra frames more than the required for the authentication (*i.e.* $> k$).
- The authentication frame can be authentication request frame, authentication response frame from the concerned node, or authentication response frame from an assisting node. Therefore, we added two bits (Authentication Request Reponse bits **ARR**) to discriminate those different frames. This addition is required for the flooding attacks resistance in the MAC layer.

Table 1 shows the corresponding meaning of initially assigned bits for **AC** and **ARR** bits. Making this clear, our protocol then consists of two phases: initialization and online procedure. In the following, two sections we will present it.

3.1 Protocol Initialization

In this protocol, we assume that: Each sensor node has the ability to do a random coin tossing with probability for head p. Probability p determines whether a node will assist in a key authentication or not.

Installation of PK information in the sensor nodes: For each sensor node i, the following procedure is performed for all the nodes in the network.

1. Security Authority (SA) randomly selects k number of nodes.
2. SA installs the public key information of a node i (K_i) to the k different sensors; where $K_i = hash$(Node i public key|Node i ID).

3.2 Authentication Protocol's Online Procedure

Eventually, a node j wants to get K_i which is the public key information of node i. At that time, the following procedure will be performed:

1. Sensor node j sends a request frame to sensor i informing that it needs K_i. Note that all the nodes in the network can overhear this request under the second assumption of section 2.
2. Not only sensor i, but also every sensor node that has the key K_i sends it to j. Different from the other nodes, sensor node i actually sends both K_i and its own public key.
3. As soon as node j receives the first response, it begins to count the number of the received response frames up to the threshold number of response. After this threshold, node j discards every incoming frame for the same key authentication.

4. Let k' be the number of incoming responses to node j. Defining e as the error bound (*i.e.* deviation from original k), the following will be performed:
   (a) If $|k' - k| \leq e$ then:
      i. Node j performs a majority voting to decide K_i.
      ii. Every node that has K_i must delete it from its memory.
   (b) If $|k' - k| > e$, that indicates a probability of an attack, and sensor node will discard the received frames to perform the request again.
5. Every other sensor node must perform the step 4 parallely, which is possible under the consumption **A.2**. Only if in case 3.(a), do the following:
   (a) Tosses a coin.
   (b) Only if the result is head, then store K_i. Note that, if we consider p as the probability of head (which is the same probability of keeping the K_i), then, $p = \frac{k}{N}$, where N is the number of nodes in the network and k is the number of nodes that will keep K_i on average.

3.3 Hurdle: Tossing Deviation Control

Even if we assign the tossing probability $p = k/N$, we can only guarantee that the number of the nodes that hold the public key information K_i of a node i is k "at average". Recall that, the probability of such process has the binomial distribution. Given the network size N and k at average, we can obtain the standard deviation of tossing probability is $(\sigma) = \sqrt{k(1 - (k/N))}$. An illustration of the deviation is in Fig 1(a). Even if we reduce the average of the authenticating nodes k, the standard deviation is little bit large for a small sized network. So, the receiver might be confused whether the difference is from attacker or from the large tossing deviation and communication noise. In the case of a large network size (say 1000 nodes), as in Figure 1(b), the authentication is held successfully since the deviation is bounded by small value compared to the size authentication assisting group k.

4 Solution: Efficient Protocol

To overcome the high deviation in a small size networks, we propose two modified versions of our basic protocol. The first version guarantees optimal security for limited number of authentication rounds (This security is guaranteed under the condition that no key is authenticated by the same assisting node twice. That means even if an attacker could monitor the network traffic and know the authentication set for the current key authentication, this type of information is not valid any more after the current authentication round. The second modified version is long-living with limited security feature and rounds traceability scheme (reversed situation of the first protocol, since with some probability for success, given enough information about c authentication rounds for a given key, it's more possible for the attacker to have successful chances for alterning the groups authentication replies.

4.1 *c*-Rounds Protocol: Optimal Security with Limited Life

The sensor network is very static. Thus, we can pre-compute the exact public key information distribution before deployment. This protocol has two phases of initialization and online protocol as follows:

Initialization Phase: In the initialization phase, for each sensor node with public key information K_i, the following is performed:

1. Security authority randomly picks c groups of node of size k from WSN.
2. The key information is paired as $\langle r, P_i \rangle$, where r is a serial and $0 < r \leq c$. The different pairs are loaded to the corresponding group of nodes.

Online Authentication Phase: In the online authentication phase, the following is performed:

1. A node j requests public key information from node i. Initially, j sends the request including a serial that express the authentication round (*i.e.* $\langle 1, req \rangle$ at the first time K_i authentication is required).
2. Not only the node i, but also all of the other node that contains the pair $\langle r, K_i \rangle$ respond the request. All of the nodes participating in the authentication have listeners on the traffic so that they can receive all copies of K_i.
3. As soon as node j receives the first authentication response from a node, it starts counting the received key information frames till a threshold number. After that, node j begins discarding any incoming extra frames.
4. The majority voting is performed at node j and all of the other authentication group to decide the acceptance of K_i.
5. If the majority voting admits the received key K_i, all copies of K_i in the authentication group with the current authentication round are removed from each sensor node memory.
6. The authentication round counter is increased by one so that the next authentication group assist in authenticating K_i in the next round.

Note that, the modified protocol does not require any coin tossing to predict the number of the next authentication round, but it relies on the predistribution of the public key iformation which guarantees a *zero deviation*. In addition, this protocol is limited to r authentication round for any key K_i. Also, when the key authentication proccess is performed, each node has the ability to recognize every incoming frame based on the **AC** and **ARR** bits values. Based on the current status of the authentication proccess, each node also can determine whether to pass the received authentication frame to the upper layer or not.

4.2 Long Living Protocol

The noticeable problem of the the above protocol is the limited life. In some intelligent attacks, it's possible to inject many faked keys for just limiting the life of the protocol. In the following, we extend the protocol life giving up a small fraction of the overall security. Note that, the initialization phase is typically the same like the initialization of the c-rounds protocol explained in 4.1.

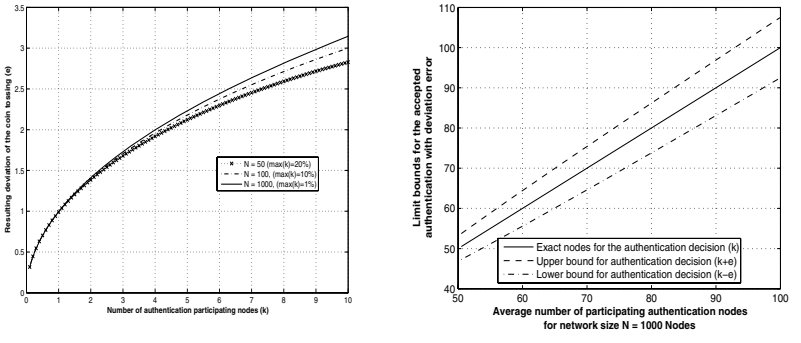
Online Authentication Phase

1. Once node K_i authentication is required by node j, node j firstly sends a request as $\langle r, req \rangle$ where r is a random integer as $0 < r \leq c$.
2. Not only node i, but all nodes holding the K_i and the random round number as a pair answer the request. Exceptionally, node i sends its own public key as well.
3. As in the early protocols, node i starts couting the number of the received K_i copies till a threshold number of frames. After the threshold value, j begins rejecting any addition frames for K_i authentication.
4. After the majority voting is performed, none of the round's nodes contents is deleted.
5. Again, an authentication is required for the same key, r is picked randomly and the request is performed.
6. The attackers has the ability to perform an attack, if and only if, he knows r and the authentication group in advance (say $\tau-$time before) that enables him to alter the authentication contents.

5 Evaluation

5.1 Authentication Decision and Tradeoffs

In our basic protocol, the voting decision can be performed as long as there are limited number of the forged messages (*i.e.* The attacker has a chance to deliver forged message with the timer of the authentication). Figure 1(a) shows the deviation for different network size when using limited number of authenticating nodes (*i.e.* up to 10 nodes per group). Figure 1(b) shows the upper and lower bounds for successful authentication when using a group of 5% to 10% of the overall network size.



(a) Tossing Deviation (b) Authentication Bounds

Fig. 1. The tossing deviation and the authentication upper and lower bounds for the actual required number of trusted nodes to authenticate a key

In the other hand, the $c-$ROUNDS and the LONG LIVING protocols don't generate any type of deviation, because the group of the different authentication proccesses is determined in advance before the installation of the protocol.

5.2 Overhead Evaluation and Comparison with Other Works

The resources overhead generated by our protocol is analyzed in terms of the memory, communication and computation. In the following we detail each of the required overhead.

- **Memory Overhead:** recall that each node can assist in k nodes' authentication. In addition, each node has r authentication rounds. Therefore, the required memory per node is $k \times r \times L$ bit, where L is the size of the hashed key in bit (*e.g.* SHA1 generates 160 bit hashes).
- **Communication Overhead** CT_{OH}: cosidering the distributed performance to perform a majority voting for a key, the required commuication is to send/receive k keys' hashes.
- **Computation Overhead:** using 8-bit processor (*i.e.* ATmega128L, $f = 8$ MHz) and 160 bit hashed keys, the required computation per authenticating one key is $(2.5k)\mu sec$. In general, assuming that W is the proccessor word size, L is the hashed key size and f is the frequency of the processor in Hz, the required computation in seconds is $CM_{OH} = \frac{L}{W} \times \frac{k}{f}$.

A detailed comparison of the consumed resources for the authentication is shown in Table 2.

5.3 Security Analysis

The security in our protocol relies on performing the authentication successfully without making the atackers affect the decision of the voting. In our first, second

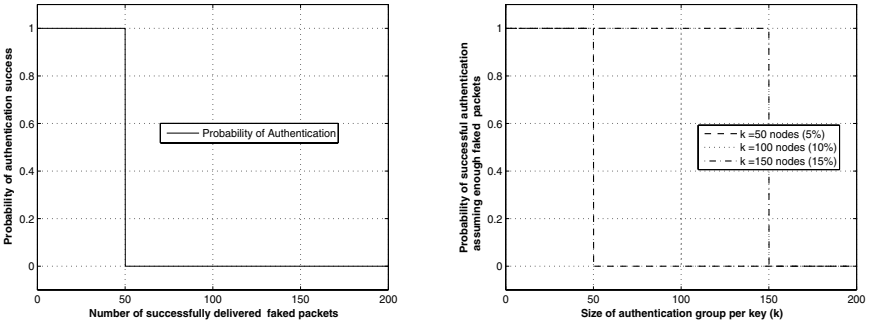
Table 1. Authentication (ARR) and (AC) flags indication

Feild value	**Stands for**
ARR(00)	Authentication request frame
ARR(01)	Authentication response frame from concerned node
ARR(10)	Authentication response frame from an assisting node
AC$(0, 1)$	Normal, Authentication frame respectively

Table 2. Resources Comparison between our scheme, RSA[9], ECC[16] and Du *et al* Scheme [15]. CT_{OH}, CM_{OH} stand for communication and computation overhead.

	Key/Hash size (bit)	CT_{OH} (bit)	CM_{OH} (ms)
RSA[9]	1024	1024	430
ECC[16]	160	320	1620
Du *et al.*[15]	160	$160k$	$7.2k$
Our scheme	160	$160k$	$2.5 \times 10^{-3}k$

and third protocols, this goes fine for a threshold number of injected faked keys for one key authentication. Assuming that the number of the nodes that assists in performing the authentication voting is k with an error range at e, each node requires $k - e/2$ number of faked keys to redirect the result of authentication. Even though, using our modification for the MAC frame and the threshold for accepting the required $k \pm e$ keys for authenticating a concerned node's key K_j will make it hard to deliver more than the k attacker's faked keys. For illustration, Figure 2(a) shows the behavior of the authentication chances per node for single attacker. Figure 2(b) shows the behavior when different k's are used.



(a) Single Node/Attacker (b) Different k's

Fig. 2. For 1000 nodes, (a) Single attacker behavior is simliar to the multiple when the number of attackers and nodes is equal. (b) different thresholds to authenticate a key.

6 Conclusion and Further Work

We introduced a novel distributed and cooperative protocol to authenticate public keys in WSN. Our protocol does not require any computational cryptographic overhead. In addition, our protocol considers the different constrained-resources of the sensor node. Even in the our basic protocol, the authentication can be held successfully with limited deviation. Our protocol is designed for one hop authentication. In the further work, we will investigate its extenssion to work for multi-hop. In addition, we will study the usage of parameters except of the deviation to perform authentication voting. More detailed mathematical evaluation for the effect of the attackers will be studied.

References

1. Blom, R.: An optimal class of symmetric key generation systems, Advances in Cryptography, Proc. EUROCRYPT 84 , LNCS-209, pp: 335-338, 1985.
2. Blundo, C., DE Santis, A., Herzberg, A., Kutten, S., Vaccaro, U., and Yung, M.: Perfectly secure key distribution for dynamic conferences, CRYPTO '92, LNCS-740, pp: 471-486, 1993.

3. Chan, H., Perrig, A., Song, D.: Random key predistribution schemes for sensor networks, IEEE SS&P, pp. 197-213, May 2003.
4. Du, W., Deng, J., Han, Y. S., and Varshney, P.: A pairwise key pre-distribution scheme for wireless sensor networks, ACM CCS'03, pp. 42-51, 2003.
5. Eschenauer, L., Gligor, V. D.: A key management scheme for distributed sensor networks, ACM CCS'02, pp. 41-47, 2002
6. Liu, D., Ning, P.: Establishing Pairwise keys in distributed sensor networks, ACM CCS'03, pp. 52-61, 2003.
7. Parno, B., Perrig, A., and Gligor V.: Distributed Detection of Node Replication Attacks in Sensor Networks, IEEE SS&P'05, May 2005.
8. Perrig, A., Szewczyk, R., Wen, V., Culler, D. E., Tygar, J. D.: SPINS: security protocols for sensor networks, MOBICOM'01, pp. 189-199, 2001.
9. Rivest, R. L., Shamir, A., Adleman, L. M.: A method for obtaining digital signatures and PK cryptosystems, Comm. of the ACM, 21(2): pp. 120-126, 1978.
10. Gura N., Patel A., Wander A., Eberle A., Shantz S. C.: Comparing Elliptic Curve Cryptography and RSA on 8-bit CPUs, CHES 2004: 119-132
11. Wander A., Gura N., Eberle H., Gupta V., Shantz S.C.: Energy Analysis of Public-Key Cryptography for Wireless Sensor Networks, PerCom'05, pp. 324-328.
12. Watro R.J., Kong D., Cuti S.F., Gardiner Ch., Lynn Ch., Kruus P.: TinyPK: securing sensor networks with public key technology, SASN'04, 59-64, 10-2004.
13. Malan D.J., Welsh A., Smith M.D.: A Public-Key Infrastructure for Key Distribution in TinyOS Based on ECC, IEEE SECON'04, pp. 71-80.
14. Pietro R.D., Law Y.W., Etalle S., Hartel P.H., Havinga P.: State of the Art in Security of Wireless Sensor Networks, IEEE Computer, 35(10): pp. 1-10.
15. Du W., Wang R., and Ning P.: An Efficient Scheme for Authenticating Public Keys in Sensor Networks. Sixth ACM MobiHoc, pp: 58-67.
16. Koblitz N., Menezes A., Vanstone S.: The State of Elliptic Curve Cryptography, Designs, Codes and Cryptography, 19, 173-193 (2000).
17. BTnode Project - ETH-Zurich: `http://www.btnode.ethz.ch/`
18. Crossbow Tech. Inc. Wireless Sensor Networks: `http://www.xbow.com/`

Restricted Universal Designated Verifier Signature

Xinyi Huang[1], Willy Susilo[1], Yi Mu[1], and Futai Zhang[2,⋆]

[1]Centre for Information Security Research,
School of Information Technology and Computer Science,
University of Wollongong, Australia
{xh068, wsusilo, ymu}@uow.edu.au
[2]School of Mathematics and Computer Science,
Nanjing Normal University, P.R. China
zhangfutai@njnu.edu.cn

Abstract. Similar to hand-written signatures, digital signatures are designed to provide authenticity, integrity and non-repudiation. In the arena of ubiquitous computing, the ability of convincing *any* third party should be restricted. In a service such as an Internet trial-browsing service, a user is allowed to access the service of t times without any charge, but will be charged on the $t+1$ count of access. In this paper, we introduce the notion of restricted universal designated verifier signature. In this notion, a signature holder can convince up to t verifiers, and the convincing statement is designated to these verifiers. However, when the signature holder uses the signature for $t+1$ times, then the signature will become publicly available. We note that this type of signature schemes has many applications in practice.

1 Introduction

Ubiquitous computing plays more important role in many aspects in life, such as human factors, computer science, engineering, and social sciences. However, how to implement security and trust among the users that connected to a network is an essential problem in such systems. Therefore, a necessary and practical authentication scheme must be deployed to satisfy different environments. The notion of a digital signature is one of the most fundamental and useful inventions of modern cryptography. Since a public key cryptosystem based on a trapdoor function model was introduced in the Diffie-Hellman paper [4], various signature schemes have been introduced to meet various needs in practical circumstances. Motivated by privacy issues associated with dissemination of digital signatures, the notion of Universal Designated Verifier Signature (UDVS) was introduced in [10].

⋆ Supported by Xidian University's Open Grant of Key Laboratory on Computer Network and Information Security of Ministry of Education of China (Grant Number: 20040105) and Partially supported by Ministry of Education of Jiangsu Province Grant 03KJA520066.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 874–882, 2006.

Essentially, UDVS schemes are digital signature schemes with additional functionality which allows *any* signature holder to *designate* the signature to any desired *designated verifier* such that the designated verifier can verify that the message was signed by the signer, but is unable to convince anyone else of this fact. To date, there exist two constructions of UDVS schemes [10, 11], which are based on bilinear pairing and standard Schnorr/RSA signature, respectively. UDVS schemes reduce to standard signatures when no verifier designation is performed. In the case where the signer and the signature holder are the same entity, then we achieve the notion of designated signature, as introduced in [5].

In some scenario, we would like to *limit* (or *restrict*) the use of a signature to t times. In our scenario, we would like to achieve restricted UDVS schemes. In this new paradigm, the signature holder is allowed to convince up to t designated verifiers. However if the signature holder tries to make more than t designation, then the signature becomes publicly available and can be identified. In practice, for instance voters should be granted maximum privacy and security on their first vote. This is achieved by allowing the voters to convince a voting ballot once. However, if the voter tries to make a second vote, the action should be identified as illegal. Another interesting application is an Internet trial browsing services. In this service, the user may be allowed to browse a maximum of three times without having to pay, but will be charged on the fourth count of access.

Our Contribution
In this paper, we firstly define a formal model of restricted UDVS schemes. In the new notion, a signature holder can convince up to t designated verifiers. However, if he/she tries to convince more than t designated verifiers, then the signature is exposed to the public and hence, it is publicly available and verifiable.

Roadmap
The rest of this paper is organized as follows. In Section 2, we review the preliminaries and background required throughout this paper. In Section 3, we define a formal model of restricted UDVS scheme together with its security requirements. In Section 4, we present our restricted UDVS scheme based on bilinear pairing, and show that it satisfies all the security requirements mentioned in Section 3. Section 5 concludes the paper.

2 Preliminaries

2.1 Previous Work

In [8], the notion of *ring signatures* was formalized and an efficient scheme based on RSA was proposed. A ring signature scheme allows a signer who knows at least one secret information (or trapdoor information) to produce a sequence of n random permutation and form them into a ring. This signature can be used to convince any third party that one of the people in the group (who knows the trapdoor information) has authenticated the message on behalf of the group. The authentication provides *signer ambiguity*, in the sense that no one can identify who has actually signed the message. In [1], a method to construct

a ring signature from different types of public keys, such as those for integer factoring based schemes and discrete log based schemes, was proposed. The proposed scheme is more efficient than [8]. The formal security definition of a ring signature is also given in [1].

Designated Verifier Proofs were proposed in [5]. The idea is to allow signatures to convince only the intended recipient, who is assumed to have a public-key. As noted in [8], ring signature schemes can be used to provide this mechanism by joining the verifier in the ring. However, it might not be practical in the real life since the verifier might not have any public key setup. In [3], Desmedt raised the problem of generalizing the designated verifier signature concept to a multi designated verifier scheme. This question was answered in [6], where a construction of multi designated verifiers signature scheme was proposed. The main idea of this scheme is to use a ring signature scheme to convince a group of verifiers on the authenticity of a signed message.

Group signatures were introduced by Chaum and van Heyst in [2]. Group signature scheme allows a group member to sign a message on behalf of the group such that everybody can verify the signature but *no one* can identify which group member provided it. However, there is a designated group manager who can reveal the identity of the originator of a signature in the case of later dispute. This act is referred to as "opening" a signature or also as revocation of a signer's anonymity.

In [9], a new concept called k-times anonymous authentication is introduced. In this paradigm, users are granted maximum privacy when they access services up to a limited number of times, but will be identified once they exceed the permitted maximum number of access times. The construction in [9] is based on group signature schemes.

2.2 Basic Concepts on Bilinear Pairings

Let $\mathbb{G}_1, \mathbb{G}_2$ be cyclic additive groups generated by P_1, P_2, respectively, whose order are a prime q. Let $\mathbb{G}_T$ be a cyclic multiplicative group with the same order q. We assume there is an isomorphism $\psi : \mathbb{G}_2 \to \mathbb{G}_1$ such that $\psi(P_2) = P_1$. Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a bilinear mapping with the following properties:

1. *Bilinearity*: $e(aP, bQ) = e(P, Q)^{ab}$ for all $P \in \mathbb{G}_1, Q \in \mathbb{G}_2, a, b \in \mathbb{Z}_q$.
2. *Non-degeneracy*: There exists $P \in \mathbb{G}_1, Q \in \mathbb{G}_2$ such that $e(P, Q) \neq 1_{\mathbb{G}_T}$.
3. *Computability*: There exists an efficient algorithm to compute $e(P, Q)$ for all $P \in \mathbb{G}_1, Q \in \mathbb{G}_2$.

For simplicity, hereafter, we set $\mathbb{G}_1 = \mathbb{G}_2$ and $P_1 = P_2 = P$. We note that our scheme can be easily modified for a general case, when $\mathbb{G}_1 \neq \mathbb{G}_2$.

Bilinear pairing instance generator is defined as a probabilistic polynomial time algorithm $\mathcal{IG}$ that takes as input a security parameter ℓ and returns a uniformly random tuple $param = (q, \mathbb{G}_1, \mathbb{G}_T, e, P)$ of bilinear parameters, including a prime number q of size ℓ, a cyclic additive group $\mathbb{G}_1$ of order q, a multiplicative group $\mathbb{G}_T$ of order q, a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ and a generator P of $\mathbb{G}_1$.

For a group $\mathbb{G}$ of prime order, we denote the set $\mathbb{G}^* = \mathbb{G} \setminus \{\mathcal{O}\}$ where $\mathcal{O}$ is the identity element of the group.

Complexity Assumptions

Definition 1. Bilinear Diffie-Hellman (BDH) Problem.
Given a randomly chosen $P \in \mathbb{G}_1$, as well as aP, bP and cP (for unknown randomly chosen $a, b, c \in \mathbb{Z}_q$), compute $e(P,P)^{abc}$.

For the BDH problem to be hard, $\mathbb{G}_1$ and $\mathbb{G}_T$ must be chosen so that there is no known algorithm for efficiently solving the Diffie-Hellman problem in either $\mathbb{G}_1$ or $\mathbb{G}_T$. We note that if the BDH problem is hard for a pairing e, then it follows that e is non-degenerate.

Definition 2. Bilinear Diffie-Hellman (BDH) Assumption.
If $\mathcal{IG}$ is a BDH parameter generator, the advantage $\mathtt{Adv}_{\mathcal{IG}}(\mathcal{A})$ that an algorithm $\mathcal{A}$ has in solving the BDH problem is defined to be the probability that the algorithm $\mathcal{A}$ outputs $e(P,P)^{abc}$ on inputs $\mathbb{G}_1, \mathbb{G}_T, e, P, aP, bP, cP$, where $(\mathbb{G}_1, \mathbb{G}_T, e)$ is the output of $\mathcal{IG}$ for sufficiently large security parameter ℓ, P is a random generator of $\mathbb{G}_1$ and a, b, c are random elements of $\mathbb{Z}_q$. The BDH assumption is that $\mathtt{Adv}_{\mathcal{IG}}(\mathcal{A})$ is negligible for all efficient algorithms $\mathcal{A}$.

3 Restricted UDVS Schemes

3.1 Model

A restricted UDVS scheme is comprised of three procedures namely sign-up, designation and open. In the sign-up procedure, a user obtains a signature σ from the signer (or CA, for instance). This signature is publicly verifiable, but this is kept by the user (and hence, the signature holder) to be used in the designation procedure. We note that although the signature is publicly verifiable, in some scenario, it is assumed that the signature is transmitted via an authenticated and secure channel (for instance, in the case of electronic voting). In the designation procedure, the signature holder can *designate* the signature to up to t verifiers. The verifiers will be convinced with the authenticity of the signature, but they cannot convince any other third party about this fact. When the signature holder uses the signature to convince the $t+1$ verifier, the *open* procedure can be invoked to reveal the signature from the CA (that was supposed to be owned by the signature holder *only*).

A restricted UDVS scheme consists of the following algorithms.

- Common Parameter Generation `GC`: on input a security parameter ℓ, outputs a string consisting of common scheme parameters cp (publicly shared by all users in the system).
- Signer Key Generation `GKS`: on input a common parameter string cp, outputs a secret-public key pair (sk_S, pk_S) for signer.
- Verifier Key Generation `GKV`: on input a common parameter string cp, outputs the secret/public key pair (sk_i, pk_i) for each verifier V_i.

- Signing $\texttt{Sign}$: on input signing secret key sk_S, message m, output signer's publicly-verifiable signature σ. That is: $\sigma \leftarrow \texttt{Sign}(m, sk_S)$.
- Public Verification $\texttt{Verify}$: on input signer's public key pk_S, a message/signature pair (m, σ), outputs verification decision $True$ or $\perp$. That is: $\{True, \perp\} \leftarrow \texttt{Verify}(m, \sigma, pk_S)$.
- Restricted Designation Signing $\texttt{RDesSign}$: on input a signer's public key pk_S, a verifier $V_i's$ public key pk_i and a message/signature pair (m, σ), outputs a restricted designated verifier signature $\hat{\sigma}$. That is: $\hat{\sigma}_i \leftarrow \texttt{RDesSign}(m, \sigma, pk_S, pk_i)$.
- Designated Verification $\texttt{DesVerify}$: on input a signer's public key pk_S, verifier $V_i's$ secret key sk_i, and a message/signature pair $(m, \hat{\sigma})$, outputs verification decision $True$ or $\perp$. That is:

$$\{True, \perp\} \leftarrow \texttt{DesVerify}(m, \hat{\sigma}, pk_S, sk_i).$$

- Opening a signature $\texttt{Open}$: on input $k \geq t{+}1$ designated verifiers $V_1, V_2, \cdots, V_k$ where

$$\{\forall t+1 : True \leftarrow \texttt{DesVerify}(m, \hat{\sigma}_i \leftarrow \texttt{RDesSign}(m, \sigma, pk_S, pk_i), pk_S, sk_i)\}$$

outputs the publicly verifiable signature σ. That is

$$\begin{aligned}\sigma \leftarrow\ & \texttt{Open}\{\forall t+1 : True \leftarrow \\ & \texttt{DesVerify}(m, \hat{\sigma}_i \leftarrow \texttt{RDesSign}(m, \sigma, pk_S, pk_i), pk_S, sk_i)\}.\end{aligned}$$

3.2 Security Properties

Completeness.
Completeness of restricted UDVS schemes is guaranteed by the following.

$$\begin{aligned}&Pr\ \{\ True \leftarrow \texttt{Verify}(m, \sigma \leftarrow \texttt{Sign}(m, sk_S))\} = 1\\ &Pr\ \{\ True \leftarrow \texttt{DesVerify}(m, \hat{\sigma}_i \leftarrow \texttt{RDesSign}(m, \sigma, pk_S, pk_i), pk_S, sk_i)\}\} = 1\\ &Pr\ \{\ \sigma \leftarrow \texttt{Open}\{\forall t+1 : True \leftarrow \texttt{DesVerify}(m,\\ &\qquad \hat{\sigma}_i \leftarrow \texttt{RDesSign}(m, \sigma, pk_S, pk_i), pk_S, sk_i)\}\} = 1\end{aligned}$$

Unforgeability.
We provide a formal definition of existential unforgeability of a restricted UDVS scheme(RUDVS) under a chosen message attack. It is defined using the following game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

- Setup: $\mathcal{C}$ runs the algorithm to generate the the system parameters and secret-public key of the original signer.
- Public Key Queries: $\mathcal{A}$ can request the public key of the verifier V_i. In response, $\mathcal{C}$ generates the secret-public key (sk_i, pk_i) of V_i and outputs pk_i to $\mathcal{A}$.
- RDesSign Queries: $\mathcal{A}$ can request a designated signature on a message m for the i^{th} designated verifier V_i. In response, $\mathcal{C}$ outputs a signature σ for the query of (m, V_i).

- DesVerify Queries: $\mathcal{A}$ can request a signature verification on a pair (m, σ) for the i^{th} designated verifier V_i. In response, $\mathcal{C}$ outputs True if it is correct, or $\perp$ otherwise.
- Output: Finally, $\mathcal{A}$ outputs a new pair (m^*, i^*, σ^*), where σ^* is a valid signature under V_{i^*} is the designated verifier, (m^*, V_{i^*}) has never been queried during the RDesSign Queries and m^* can be requested at most t times to the RDesSign algorithm.

The success probability of an adversary to win the game is defined by $Succ^{EF-CMA}_{RUDVS,\mathcal{A}}(\ell)$.

Definition 3. *We say that a restricted UDVS scheme(RUDVS) is existentially unforgeable under a chosen message attack if the probability of success of any polynomially bounded adversary in the above game is negligible. In other words,* $Succ^{EF-CMA}_{RUDVS,\mathcal{A}}(\ell) \leq \epsilon$.

Non-Transferability Privacy.

Definition 4. *We say that a restricted UDVS scheme (RUDVS) provides* non-transferable privacy *iff the receiver can generate a valid signature which is indistinguishable from the original one. That means, the designated verifier can always produce identically distributed transcripts that are indistinguishable from the one that was generated by the signature holder.*

4 A Restricted UDVS Scheme from Bilinear Pairing

In this section, we present a restricted UDVS scheme based on bilinear pairing. The scheme is as follows.

- `GC`: Choose a bilinear group-pair $(\mathbb{G}_1, \mathbb{G}_T)$ of the same prime order q where $q \geq 2^\ell$, ℓ is the security number. P is the random generator of the group $\mathbb{G}_1$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$ is a bilinear mapping and a cryptographic hash function $H_0 : \{0,1\}^* \rightarrow \mathbb{G}_1$.
- `GKS`: Pick a random $x_s \in \mathbb{Z}_q^*$ and compute $Y_s = x_s P$. The signer's public key pk_S is Y_s and the secret key sk_S is x_s.
- `GKV`: Pick a random $x_i \in \mathbb{Z}_q^*$ and compute $Y_i = x_i P$. The i^{th} verifier's public key pk_i is Y_i and the secret key sk_i is x_i.
- `Sign`: Given the signer's secret key x_s and the message m, compute $\sigma = x_s H$, where $H = H_0(m)$. The signature is σ.
- `Verify`: Given the signer's public key y_s and a message-signature pair (m, σ), accept if and only if $e(\sigma, P) \stackrel{?}{=} e(H_0(m), Y_s)$ holds.
- `RDesSign`: Choose $R_1, R_2, \cdots, R_t \in_R \mathbb{G}_1$ and set the function $F(u) = \sigma + \Sigma_{l=1}^{t} u^l R_l$ with a message-signature pair (m, σ). Given the i^{th} designated verifier $V_i's$ public key Y_i, computes $f_i = e(\Sigma_{l=1}^{t} u^l R_l, Y_i)$ and $F_i = F(i)$. The designated signature is (i, f_i, F_i).
- `DesVerify`: Given a signer's public key Y_s, the i^{th} designated verifier $V_i's$ secret key x_i and the message/designated signature pair(m, i, f_i, F_i), accept if and only if $e(F_i, Y_i) \stackrel{?}{=} e(x_i H, Y_s) \cdot f_i$ holds with equality, where $H = H(m)$.

– `Open`: Given $k \geq t+1$ signatures, anyone can reconstruct the function $F(u)$ by computing: $F(u) = \Sigma_{j \in \Phi} c_{uj}^{\Phi} F_j$, where $c_{uj}^{\Phi} = \Pi_{\iota \in \Phi,\, \iota \neq j} \frac{u-\iota}{j-\iota} \in \mathbb{Z}_q$ is the Lagrange coefficient for a set $\Phi \subset \{1, \cdots, n\}$ such that $|\Phi| \geq t+1$, and compute $F(0) = \sigma$. The result of the `Open` algorithm is the publicly verifiable signature σ.

4.1 Security Proofs

Theorem 1. *Our scheme provides completeness.*

Proof. To show this argument, we need to show the following.

1. Signature generated by the `Sign` algorithm will pass the `Verify` algorithm. $e(\sigma, P) = e(x_s H_0(m), P) = e(H_0(m), Y_s)$.
2. Signature generated by the `RDesSign` algorithm will pass the `DesVerify` algorithm.

$$\begin{aligned} & e(F_i, Y_i) = e(\sigma + \Sigma_{l=1}^{t} u^l R_l, Y_i) \\ = & e(\sigma, Y_i) e(\Sigma_{l=1}^{t} u^l R_l, Y_i) = e(x_s H_0(m), x_i P) \cdot f_i \\ = & e(x_i H_0(m), x_s P) \cdot f_i = e(x_i H, Y_s) \cdot f_i. \end{aligned}$$

3. Anyone can correctly recover the signature from the `Open` algorithm.

   Since $F(u) = \sigma + \Sigma_{l=1}^{t} u^l R_l$, the following equation holds: $\sigma = F(0) = F(u)_{u=0} = (\Sigma_{j \in \Phi} c_{uj}^{\Phi} F_j)_{u=0} = \Sigma_{j \in \Phi} c_{0j}^{\Phi} F_j$. where $c_{0j}^{\Phi} = \Pi_{\iota \in \Phi,\, \iota \neq j} \frac{0-\iota}{j-\iota} \in \mathbb{Z}_q$ is the Lagrange coefficient for a set $\Phi \subset \{1, \cdots, n\}$ such that $|\Phi| \geq t+1$.

Theorem 2. (Non-Transferability Privacy) *The designated verifier can produce a signature which is indistinguishable from the original signature.*

Proof. Given a message m, the i^{th} designated verifier V_i can always use his secret key x_i to generate an indistinguishable signature by computing

$$f_i = e(F_i, Y_i) / e(x_i H_0(m), Y_s)$$

for $F_i \in_R \mathbb{G}_1$ which is chosen randomly by V_i. Note that (i, f_i, F_i) is a valid signature of the message $\hat{m}$. We need to show that the transcripts (i, f_i, F_i) simulated by the receiver V_i are indistinguishable from those (i, f_i', F_i') that he receives from the signature holder.

Let $(i, \hat{f}_i, \hat{F}_i)$ be a randomly chosen signature in the set of all possible valid restricted universal designated verifier signatures from the signature holder to the designated verifier V_i. Then we have

$$Pr[(i, f_i, F_i) = (i, \hat{f}_i, \hat{F}_i)] = Pr[f_i = \hat{f}_i] \cdot Pr[F_i = \hat{F}_i] = \frac{1}{q^2}$$

and

$$Pr[(i, f_i', F_i') = (i, \hat{f}_i, \hat{F}_i)] = Pr[f_i' = \hat{f}_i] \cdot Pr[F_i' = \hat{F}_i] = \frac{1}{q^2}$$

which have the same probability. This means that both distribution of probabilities are the same. □

Theorem 3. (Unforgeability) *Let $\mathcal{A}$ be an* EF-CMA-adversary *who can successfully get a valid forgery of our restricted UDVS scheme with success probability $Succ^{EF-CMA}_{RUDVS,\mathcal{A}}$ after q_H queries to the hash function, q_S queries to the* `RDesSign` *algorithm and q_V queries to the* `DesVerify` *algorithm in polynomial time t. Then there exists an algorithm $\mathcal{B}$ who can use $\mathcal{A}$ to solve an instance of BDH problem with probability*

$$Succ^{BDH}_{\mathcal{B}} \geq \frac{1}{q_S + q_V} \cdot (1 - \frac{1}{q_S + q_V + 1})^{q_S + q_V + 1} \cdot Succ^{EF-CMA}_{RUDVS,\mathcal{A}}$$

Proof. For the proof of this theorem, we refer the reader to the full version [7].

5 Conclusion

In this paper, we presented a novel notion called *Restricted Universal Designated Verifier Signature* scheme. Our notion allows a signature holder to designate the signer's signature to up to t designated verifiers. However, after being used t times, the signature cannot be designated to any other third party. We noted that this type of signature schemes has many applications, such as trial Internet browsing or electronic voting. We presented a concrete scheme based on bilinear pairing and showed that our scheme is secure in the random oracle model.

References

1. M. Abe, M. Ohkubo, and K. Suzuki. 1-out-of-n Signatures from a Variety of Keys. Advances in Cryptology - Asiacrypt 2002, Lecture Notes in Computer Science 2501, pages 415 – 432, 2002.
2. D. Chaum and E. van Heyst. Group signatures. Advances in Cryptology - Eurocrypt '91, Lecture Notes in Computer Science 547, pages 257 – 265, 1991.
3. Y. Desmedt. Verifier-Designated Signatures. Rump Session, Crypto 2003, 2003.
4. W. Diffie and M. Hellman. New directions in cryptography. IEEE IT, 22:644 – 654, 1976.
5. M. Jakobsson, K. Sako, and R. Impagliazzo. Designated Verifier Proofs and Their Applications. Advances in Cryptology - Eurocrypt '96, Lecture Notes in Computer Science 1070, pages 143 – 154, 1996.
6. F. Laguillaumie and D. Vergnaud. Multi-Designated Verifiers Signatures. Sixth International Conference on Information and Communications Security (ICICS 2004), Lecture Notes in Computer Science, pages 495 – 507, 2004.
7. X. Huang, Y. Mu, W. Susilo, and F. Zhang. Restricted Universal Designated Verifier Signature. http://www.uow.edu.au/ wsusilo/publications.html
8. R. L. Rivest, A. Shamir, and Y. Tauman. How to Leak a Secret Advances in Cryptology - Asiacrypt 2001, Lecture Notes in Computer Science 2248, pages 552 – 565, 2001.
9. K. Sako, S. Yonezawa, and I. Teranishi. Anonymous Authentication: For Privacy and Security. NEC Journal of Advanced Technology, Special Issue on Security for Network Society, Vol. 2 No. 1, pages 79 – 83, 2005.

10. R. Steinfeld, L. Bull, H. Wang, and J. Pieprzyk. Universal Designated-Verifier Signatures. Proceedings of Asiacrypt 2003, Lecture Notes in Computer Science 2894, pages 523 – 543, 2003.
11. R. Steinfeld, H. Wang, and J. Pieprzyk. Efficient Extension of Standard Schnorr/RSA signatures into Universal Designated-Verifier Signatures. Proceedings of 7th International Workshop on Theory and Practice in Public Key Cryptography (PKC 2004), Lecture Notes in Computer Science 2947, pages 86 – 100, 2004.

Research on Pairwise Key Establishment Model and Algorithm for Sensor Networks

Lei Wang[1,3], Yaping Lin[1], Minsheng Tan[2], and Chunyi Shi[3]

[1] College of Software, Hunan University, Changsha, 410082, P.R. China
{Wanglei, Yplin}@hnu.cn
http://ss.hnu.cn/newweb/teachers/zhuanzhijiaoshi/wanglei/
[2] School of Computer Science&Technology, Nanhua University, 421001, P.R. China
Mstan@hnu.cn
[3] Dept of Computer Sci & tech, Tsinghua University, Beijing, 100084, P.R. China
Scy@hnu.cn

Abstract. Security schemes of pairwise key establishment, which enable sensors to communicate with each other securely, play a fundamental role in research on security issue in wireless sensor networks. A new kind of pairwise key pre-distribution model for sensor networks is proposed. And in addition, based on which, an efficient dynamic key path establishment algorithm is designed. Theoretic analysis and experimental figures show that the new algorithm has better performance than those previous related works.

1 Introduction

Security communication is an important requirement in many sensor network applications. As one of the most fundamental security services, pairwise key establishment enables the sensor nodes to communicate securely with each other using cryptographic techniques, so it is important to research on the pairwise key establishment algorithm for sensor networks[1-4].

Eschenauer and Gligor proposed a basic probabilistic key pre-distribution scheme for pairwise key establishment [1] in 2002. One year later, Chan et al. extended their ideas and presented two key pre-distribution schemes: a q-composite key pre-distribution scheme and a random pairwise keys scheme [2]. Inspired by the polynomial-based key pre-distribution protocol [3], Liu et al. further developed the idea addressed in the previous works and proposed a general framework of polynomial pool-based key pre-distribution [4]. Based on such a framework, they presented two pairwise key pre-distribution schemes: a random subset assignment scheme and a grid-based scheme. A similar approach to those schemes described by Liu et al was independently developed by Du et al. [5]. All of those schemes above improve the security over the basic probabilistic key pre-distribution scheme. However, the pairwise key establishment problem in sensor networks is still not well solved. In 2004, Liu proposed a new hypercube-based pairwise key predistribution scheme [6], [1]which

[1] Wang Lei (1973-), male, associate professor of Hunan University, P.R.China. His major research interests include networks and DNA computation.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 883 – 892, 2006.

extends the grid-based scheme from a two dimensional grid to a multi-dimensional hypercube. The analysis shows that hypercube-based scheme has many attractive properties, but it requires any two nodes in sensor networks can communication directly with each other. This strong assumption is impractical in most of the actual applications of the sensor networks.

In this paper, we present a kind of new cluster-based distribution model of sensor networks, and for which, we propose a new pairwise key pre-distribution scheme. The main contributions of this paper are as follows: Combining the deployment knowledge of sensor networks and the polynomial pool-based key pre-distribution, we setup a cluster-based topology that is practical with the real deployment of sensor networks. Based on the topology, we propose a novel cluster distribution based hierarchical hypercube model to establish the pairwise key. The key contribution is that our scheme does not require the assumption of all nodes can directly communicate with each other as the previous schemes do, and it still maintains high probability of key establishment, low memory overhead and good security performance. In addition, based on the newly proposed model, a dynamic key path establishment algorithm is designed. Theoretic analysis and experimental figures show that the new algorithm has better performance than those previous related works.

2 Preliminaries

2.1 Model of Clusters Deployed Sensor Networks

In some actual applications of sensor networks, sensors can be deployed through airplanes. Supposing that the deployment rounds of sensors are k and the communication radius of any sensors is r, and then the sensors deployed in the same round can be regarded as belonging to a same *Cluster*. We assign a unique *cluster number* l ($1 \le l \le k$) for each cluster. Suppose that the sensors form a connected graph in any cluster after deployment through airplanes.

2.2 Hierarchical Hypercube Model

Let there are N nodes totally, then a k-levels Hierarchical Hypercube named $H(k,u,m,v,n)$ can be constructed as follows:

The N nodes are divided into k clusters averagely, and the $[N/k]$ nodes in any cluster are connected into an n-dimensional Hypercube: In the n-dimensional Hypercube, any node is encoded as $i_1 i_2 \ldots i_n$, which are called *In-Cluster-Hypercube-Node-Codes*, where $0 \le i_1, i_2, \ldots i_n \le v\text{-}1, v = [\sqrt[n]{N/k}\,]$, $[j]$ equals to an integer not less than j. So we can obtain k such kind of different hyper cubes.

The k different hypercubes obtained above are encoded as $j_1 j_2 \ldots j_m$, which are called *Out-Cluster-Hypercube-Node-Codes*, where $0 \le j_1, j_2, \ldots j_m \le u\text{-}1, u = [\sqrt[m]{k}\,]$. And the nodes in the k different hypercubes are connected into m-dimensional hypercubes according to the following rules: The nodes with same *In-Cluster-Hypercube-Node-Codes* and different *Out-Cluster-Hypercube-Node-Codes* are connected into an m-dimensional hypercube. (The graph constructed through above steps is called a k-levels Hierarchical Hypercube abbreviated as $H(k,u,m,v,n)$.)

Any node A in $H(k,u,m,v,n)$ can be encoded as (i, j), where $i(i=i_1i_2\ldots i_n, 0 \le i_1,i_2,\ldots i_n \le v\text{-}1)$ is the *In-Cluster-Hypercube-Node-Code* of node A, and $j(j=j_1j_2\ldots j_m, 0 \le j_1,j_2,\ldots j_m \le u\text{-}1)$ is the *Out-Cluster-Hypercube-Node-Code* of node A.

Obviously, the clusters deployed sensor network can be mapped into a k-levels- hierarchical hypercube model as follows:

At first, the k clusters in the sensor network can be mapped into k different levels (or hypercubes) in the k-levels- hierarchical hypercube model. Then, the sensor nodes in each cluster can be encoded with the *In-Cluster-Hypercube-Node-Codes*, and the sensor nodes in the k different clusters with the same *In-Cluster-Hypercube-Node-Codes* can be encoded with the *Out-Cluster-Hypercube-Node-Codes* according to the above steps respectively. Consequently, the whole sensor network has been mapped into a k-levels- hierarchical hypercube model.

3 Dynamic Key Path Establishment for Sensor Networks

3.1 Generation of Polynomials Pool and Key Pre-distribution

Supposing that, the sensor network includes N nodes, and is deployed through k different rounds. Then we can predistribute keys for each sensor node on the basis of the $H(k,u,m,v,n)$ model as follows:

Step 1: Key setup server randomly generates a bivariate polynomials pool $F=\{ f^{i}_{l,<i_1,i_2,\ldots,i_{n-1}>}(x,y),\ f^{j}_{<i_1,i_2,\ldots,i_n,j_1,j_2,\ldots,j_{m-1}>}(x,y) \mid 0 \le i_1 \le i_2 \le \ldots \le i_{n-1} \le v\text{-}1,\ 1 \le i \le n, 1 \le l \le k, 0 \le j_1 \le j_2 \le \ldots \le j_{m-1} \le u\text{-}1,\ 1 \le j \le m\}$ with $v^n * m * u^{m-1} + [N/v^n] * n * v^{n-1}$ different t-degree bivariate polynomials over a finite field F_q, and then assigns a unique polynomial ID to each bivariate polynomial in F.

Step 2: In each round, key setup server assigns a unique node ID: $(i_1i_2\ldots i_n,j_1j_2\ldots j_m)$ to each sensor node from small to big, where $0 \le i_1,i_2,\ldots i_n \le v\text{-}1$, $0 \le j_1,j_2,\ldots j_m \le u\text{-}1$.

Step 3: key setup server assigns a unique cluster ID: l to all the sensor nodes deployed in the same round, where $1 \le l \le k$.

Step 4: key setup server predistributes $m+n$ bivariate polynomials $\{ f^{1}_{l,<i_2,i_3,\ldots,i_n>}(i_1,y),\ldots, f^{n}_{l,<i_1,i_2,\ldots,i_{n-1}>}(i_n,y); f^{1}_{<i_1,i_2,\ldots,i_n,j_2,\ldots,j_m>}(j_1,y),\ldots, f^{m}_{<i_1,i_2,\ldots,i_n,j_1,\ldots,j_{m-1}>}(j_m,y) \}$ and the corresponding polynomial IDs to the sensor node deployed in the lth round and with ID $(i_1i_2\ldots i_n, j_1j_2\ldots j_m)$.

3.2 Dynamic Key Path Establishment

From the following example we can know that there are many parallel paths in the $H(k,u,m,v,n)$ model for any two given source and destination nodes, since the $H(k,u,m,v,n)$ model is high fault-tolerant[9,10].

For example: Considering the key path establishment example based on Fig.2: $A((012),(1234)) \rightarrow C((112),(1234)) \rightarrow D((122),(1234)) \rightarrow E((121),(1234)) \rightarrow F((121),(2234)) \rightarrow B((121),(2334))$, supposing that node $F((121),(2234))$ has compromised, then from Fig.2, we can know that there exists another alternative key path as $A((012),(1234)) \rightarrow C((112),(1234)) \rightarrow D((122),(1234)) \rightarrow E((121),(1234)) \rightarrow M((121),(1334)) \rightarrow B((121),(2334))$, which can be used to establish the indirect

pairwise key between node *A* and *B*, where node *E* shall route through nodes *D* and *K* to establish direct pairwise key with node *M*, and node *M* shall route through nodes *N*, *O*, *G*, *H*, *I*, *J* to establish direct pairwise key with node *B*.

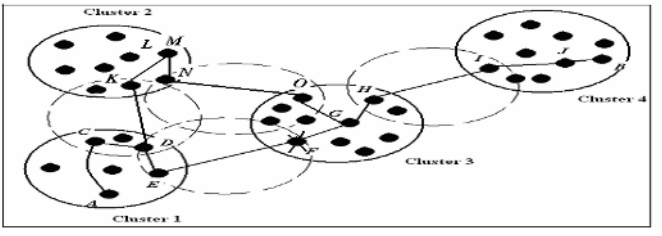


Fig. 1. Alternative key path establishment example

Since the sensors are source limited, so they are easy to die or out of the communication radius, therefore the algorithm proposed in the above section cannot guarantee to establish correct key path efficiently. In this section, we will propose a dynamic key path establishment algorithm, which can improve the probability of key path effectively. Before introducing the detailed dynamic key path establishment algorithm, we analysis the high fault-tolerance of $H(k,u,m,v,n)$ model firstly.

3.3 Local Weak-Connectivity of $H(k,u,m,v,n)$ Model

Definition 1: A node *A* in *n*-dimensional hypercube/sub-cube $H(v,n)$ is called *reachable*, iff node *A* is not fault and not *n-alone*. A node *A* in *n*-dimensional hypercube/sub-cube $H(v,n)$ is called *n-alone*, iff all the neighboring nodes of *A* or all the links between node *A* and any one of the non-fault neighboring nodes in $H(v,n)$ are fault. A node *A* in *n*-dimensional hypercube/sub-cube $H(v,n)$ is called *q-reachable*, iff all the *u* neighboring nodes in the *q*-th dimension of node *A* and all the links between node *A* and any one of those neighboring nodes are not fault.

Definition 2 (*k*-dimensional local weak-connectivity): If all the reachable nodes in any *k*-dimensional sub-cube $H(v,k)$ ($k \geq 1$) of the *n*-dimensional hypercube $H(v,n)$ form a connected graph, and there exist a node *A* in $H(v,k)$, which is *q-reachable*, then $H(v,n)$ is called *k-dimensional local weak-connected*.

Definition 3: Since any string $b_1\ b_2 \ldots b_{n-k}$ with *n-k* length corresponds to a *k*-dimensional sub-cube $H(v,k)$ with u^k nodes, so we can use the string $b_1\ b_2 \ldots b_{n-k}*\ldots*$ to represent $H(v,k)$, where $* \in \{0,\ldots,u\text{-}1\}$.

From the construction of $H(v,k)$, it is easy to know that all the *k*-dimensional sub-cubes in $H(v,n)$ are isomorphic , and $H(v,n)$ includes all of the *k*-dimensional sub-cubes isomorphic with $H(v,k)$. I t is easy to prove that there are v^{n-k}-1 different *k*-dimensional sub-cubes, which are isomorphic with $H(v,k)$, and have no interconnected nodes with $H(v,k)$ at the same time.

Then, we can prove that the $H(v,n)$ model that satisfies the two kinds of proposed local weak-connectivity conditions has the following good properties:

Property 1: All the reachable nodes in the n-dimensional hypercube $H(v,n)$ that satisfies the k-dimensional local weak-connectivity condition form a connected graph.

Proof: We use induction to prove that all the reachable nodes in $H(v,n)$ form a connected graph for any $n \geq k$.

When k=1, obviously the theorem's conclusion stands, so we shall consider $k \geq 2$.

1) If $n=k$, then it is easy to know that the theorem's conclusion stands from the definition 2.

2) Supposing that the theorem's conclusion stands when $n=t$ ($n>k$). It means that all the reachable nodes in $H(v,t)$ form a connected graph.

3) When $n=t+1$: since the $(t+1)$-dimensional hypercube $H(v,t+1)$ is k-dimensional local weak-connected $\Rightarrow$ all the reachable nodes in each k-dimensional sub-cube of $H(v,t+1)$ form a connected graph, and there exists at least one q-reachable node in each k-dimensional sub-cube of $H(v,t+1)$.

For any two reachable nodes A and B in $H(v,t+1)$, we can prove that nodes A and B are connected.According to the isomorphism defined in definition 10, we can suppose that $H(v,t+1)=b_1b_2\ldots b_{n-t-1}*\ldots*$, where $b_1b_2\ldots b_{n-t-1}$ is a fixed string with length $n-t-1$. So the $(t+1)$-dimensional hypercube $H(v,t+1)$ can be divided into v different t-dimensional cub-cubes $H_1(v,t)\ldots H_v(v,t)$, where $H_1(v,t)= b_1b_2\ldots b_{n-t-1}0*\ldots*,\ldots,H_v(v,t)= b_1b_2\ldots b_{n-t-1}(u-1)*\ldots*$.

If nodes A and B are in a same t-dimensional sub-cube: $H_1(v,t)\ldots H_v(v,t)$, then according to the inductive hypothesis, it is easy to know that nodes A and B are connected. $\Rightarrow$ All reachable nodes in $H(v,t+1)$ form a connected graph.

If nodes A and B belong to two different t-dimensional cub-cubes of $H_1(v,t)\ldots H_v(v,t)$, supposing that $A=b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots a_t \in H_i(v,t)$, $B=b_1b_2\ldots b_{n-t-1}jb\ d_2\ldots d_t \in H_j(v,t)$, where $i \neq j$. Since $H_i(v,t)$ is k-dimensional local weak-connected $\Rightarrow$ There exists a node C, which is q-reachable $\Rightarrow$ All nodes in the nodes set $\{b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots a_q\ldots a_t|q=0,\ldots,v-1\}$ are reachable, and form a connected graph with node C. Obviously, node $b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots d_q\ldots a_t \in \{b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots a_q\ldots a_t|q=0,\ldots,v-1\} \Rightarrow$ Node A and node $b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots d_q\ldots a_t$ are connected. In addition, since node $b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots d_q\ldots a_t$ and node $B= b_1b_2\ldots b_{n-t-1}jb\ d_2\ldots d_t$ belong to the same t-dimensional hypercube $b_1b_2\ldots b_{n-t-1}*\ldots*\ d_q*\ldots*$, so according to the inductive hypothesis, it is easy to know that node $b_1b_2\ldots b_{n-t-1}ia\ a_2\ldots d_q\ldots a_t$ and node B are connected. $\Rightarrow$ Node A and node B are connected. $\Rightarrow$ All the reachable nodes in $H(v,t+1)$ form a connected graph.

From above proofs, we can know that All the reachable nodes in the n-dimensional hypercube $H(v,n)$ that satisfies the k-dimensional local weak-connectivity condition form a connected graph. □

3.4 Local Weak-Connectivity Based Dynamic Key-Path Establishing Algorithm

According to the actual deployment of sensor network, in the same cluster, since the sensors are deployed at the same time, so they are densely distributed, and most of then are in the communication radius, therefore, it is easy to know that the sensor nodes in the same cluster can keep the structure of $H(k,u,m,v,n)$ model preformed at the key predistribution stage. Since we have proved that the sub-cube $H(v,n)$ in

$H(k,u,m,v,n)$ model has high fault-tolerant ability, then the sub- sensor network $H(v,n)$ also has high fault-tolerant ability. It means that the whole sub-sensor network can keep global connectivity even if there are a lot of compromised or fault sensors.

Mapping the non-fault sensors in the sub-sensor network into the reachable nodes in $H(v,n)$ model, and compromised or fault sensors into the fault nodes in $H(v,n)$ model, then it is obvious that the sub-sensor network $H(v,n)$ will satisfy those two kinds of proposed local weak-connectivity conditions.

So, during the path key establishment between the sensor nodes in the same cluster, the new proposed algorithm will establish actual key path on the basis of the above two kinds of local weak-connectivity concepts.

But for sensors belonging to different clusters, there are only a small number of them that can communicate with each other directly. And in addition, after deployment of the sensors, the two nodes neighboring to each other in the logical space $H(k,u,m,v,n)$ are perhaps not neighboring in the actually deployed sensor network environment, so the the $H(k,u,m,v,n)$ model preformed at the key predistribution stage will be destroyed. Then it will lead to high routing costs if the routes are still established according to the properties of the logical space $H(k,u,m,v,n)$ only. So, during the path key establishment between the sensor nodes in different clusters, the new proposed algorithm will establish actual key path on the basis of the location information [11] of the sensors.

On the basis of the above description, we have mapped the actual cluster deployed sensor network into a $H(k,u,m,v,n)$ model, where each sub-sensor network in the same cluster has been mapped into a $H(v,n)$ model. So, next we will not distinct sensor network with $H(k,u,m,v,n)$ model, and sub-sensor network, cluster with $H(v,n)$ model any longer. The main ideas of the pairwise key path establishment algorithm (*Algorithm I*) based on the k-dimensional local weak-connected n-dimensional hypercube model $H(v,n)$ are as follows:

The destination of *Algorithm I* is to establish a correct key path from the source sensor to the destination sensor, when there are some compromised or fault sensors or links in the whole sensor network. At first, algorithm will compare each position of the codes of nodes A and B from left to right, if the same then pass, otherwise transfer the value of node A in the current position into the corresponding value of node B in the current position. Then after n-k times transform, the former n-k positions of node A and B are the same. Finally, algorithm will find a route to node B in the k-dimensional sub-cube including node B. Therefore, we have obtained a correct key path from node A to B. If there is not a correct key path from node A to B found, then according to the properties 1, it is easy to know that the sub-sensor network $H(v,n)$ is not k-dimensional local weak-connected. From the former description, we can know that $H(v,n)$ will be local weak-connected in most of applications, since the sensors are deployed densely in the same cluster.

***Algorithm I*:** The first category of path pairwise key establishment algorithm for sub-sensor network on the basis of the k-dimensional local weak-connected n-dimensional hypercube model $H(v,n)$.

Input: Sub-sensor network $H(v,n)$, which has some compromised /fault sensors and fault links, And two reachable nodes $A(a_1 \ldots a_n, a'_1 \ldots a'_m)$ and $B(b_1 \ldots b_n, b'_1 \ldots b'_m)$ in $H(v,n)$.

Output: A correct key path from node A to B in $H(v,n)$.

1) Obtain the former half part of the code strings of node A to B: $A \leftarrow a_1 \ldots a_n$, $B \leftarrow b_1 \ldots b_n$, where a_j, $b_j \in [0,v\text{-}1]$. /* From the isomorphism defined in definition 3, we can know that the k-dimensional sub-cube including node B can be expressed as $b_1 \ldots b_{n-k} * \ldots *$. */

2) Initial the path: P: $P \leftarrow A$.

3) Initial the temp string C: $C = c_1 \ldots c_n \leftarrow A$.

4) FOR (j=1, $j \leq n$, j++){ IF($a_j \neq b_j$){

4.1) According to the property 1, a pair of connected reachable nodes can be found through neighboring k-dimensional sub-cubes: $C = c_1 \ldots c_{j-1}\ c_j\ c_{j+1} \ldots c_{n-k} \ldots c_n, b_1 \ldots b_{j-1}\ b_j\ x_{j+1} \ldots x_{n-k} \ldots x_n$, where $c_t = b_t$ ($t \in [1,j]$).

4.2) Expand the path P: Add the route from node $b_1 \ldots b_{j-1}\ a_j\ a_{j+1} \ldots a_{n-k}\ a_{n-k+1} \ldots a_n$ to node $b_1 \ldots b_{j-1}\ b_j\ x_{j+1} \ldots x_{n-k} \ldots x_n$ into P.

4.3) $C \leftarrow b_1 \ldots b_{j-1}\ b_j\ x_{j+1} \ldots x_{n-k} \ldots x_n$}}

5) Expand the path P: Add the route in the k-dimensional sub-cube $b_1 \ldots b_{n-k} * \ldots *$ from the node $b_1 \ldots b_{n-k}\ x'_{n-k+1} \ldots x'_n$ to the destination node $B = b_1 \ldots b_n$ into P. Algorithm exits, and then a correct key path from node A to B has been found finally.

From the above description of *Algorithm I*, it is easy to prove that the time complexity of *Algorithm I* is $O((n-k)v^k)+O(v^k)=O(nv^k)$.

On the basis of the *Algorithm I*, by combing the location information of sensors, we can present the whole dynamic key path establishment algorithm based on $H(k,u,m,v,n)$ model for cluster deployed sensor networks as follows:

***Algorithm II*:** Dynamic key path establishment algorithm based on $H(k,u,m,v,n)$ model for cluster deployed sensor networks.

Input: Sub-sensor network $H(k,u,m,v,n)$, which has some compromised /fault sensors and fault links, And two reachable nodes $A(a_1 \ldots a_n, a'_1 \ldots a'_m)$ and $B(b_1 \ldots b_n, b'_1 \ldots b'_m)$ in $H(k,u,m,v,n)$, where $a'_t \neq b'_t$, $t \in [1,s]$, $a'_t = b'_t$, $t > s$.

Output: A correct key path from node A to B in $H(k,u,m,v,n)$.

1) Obtain the code strings of node A and B: $A \leftarrow (a_1 \ldots a_n, a'_1 \ldots a'_m)$, $B \leftarrow (b_1 \ldots b_n, b'_1 \ldots b'_m)$, where a_j, $b_j \in [0,u\text{-}1]$, a'_j, $b'_j \in [0,v\text{-}1]$.

2) If $a'_1 \ldots a'_m = b'_1 \ldots b'_m$, then node A can find a route to B according to the *Algorithm I*.

3)Otherwise, let $I_0 = A(a_1 \ldots a_n, a'_1 \ldots a'_m)$, $I_1 = (b_1 \ldots b_n, b'_1\ a'_2 \ldots\ a'_m), \ldots, I_s = B(b_1 \ldots b_n, b'_1\ b'_2 \ldots b'_s\ a'_{s+1} \ldots a'_m)$. Since each pair of nodes I_t .and I_{t+1} .belong to two neighboring groups respectively, then I_t .and I_{t+1} .can establish a key path according to the *Algorithm I* also.

4) Algorithm exits. If such kind of a correct key path exists, then through which node A can establish an indirect pairwise key with node B. Otherwise, node A fails to establish an indirect pairwise key with node B. And node A will tries again to establish an indirect pairwise key with node B some time later.

4 Algorithm Analyses

Theorem 1: When there exist no fault and compromised nodes, by using new pairwise key predistribution scheme based on $H(k,u,m,v,n)$ model, the probability of direct or indirect pairwise key establishment between any two nodes is 100%.

Proof: According to the hypothesis that the whole sensor network is globally connected, it is obvious that there exists a direct or indirect pairwise key between any two sensor nodes. It means that the probability of direct or indirect pairwise key establishment between any two nodes is 100%. □

Theorem 2: Supposing that the total sensors is N in the sensor network, then when $u \geq v^2$, the probability of direct pairwise key establishment between any two nodes, while using the key distribution scheme based on the hypercube model $H(v,p)$, is smaller than that while using the key distribution scheme based on the $H(k,u,m,v,n)$ model.

Proof: Since $u \geq v$, then we can let $u=v^t$, where $t \geq 2$. Since the total number of nodes in $H(v,p)$ is $v^p=N$, the total number of nodes in $H(k,u,m,v,n)$ is $u^m * v^n=N$. Let $p=x+n$, then there is $u^m * v^n = v^x * v^n \Rightarrow u^m=v^x \Rightarrow x=tm$.

It is easy to know that the probability of direct pairwise key establishment between any two nodes can be estimated as $P=(m(u-1)+n(v-1))/(N-1)$. According to the description in [7], it is well know that the probability of direct pairwise key establishment between any two nodes can be estimated as $P'= p(v-1)/(N-1)= (x(v-1)+n(v-1))/(N-1)$.

Next, we will prove that $m(u-1) \geq x(v-1)$: $m(u-1)= m(v^t-1)$, $x(v-1)= tm(v-1)$. Construct a function as $f(t)= v^t -1- t(v-1)$, where $t \geq 2$. When $t=2$, it is obvious that there is $f(t)= v^t -2v+1=(v-1)^2 \geq 0$ and $f'(t)=t\ v^{t-1}- v+1 \geq 2v- v+1= v+1>0$. So, there is $f(t) \geq 0 \Rightarrow v^t -1 \geq t(v-1) \Rightarrow m(v^t-1) \geq tm(v-1) \Rightarrow m(u-1) \geq x(v-1)$.

Therefore, the conclusion of the theorem stands. □

Supposing that the total number of nodes in the sensor network is N, Fig.2 illustrates the comparison of the direct pairwise key establishment probability of $H(k,u,m,v,n)$ and $H(v,p)$, when $u=4$ and $v=2$. From Fig.2, it is easy to know that the theorem 2 stands.

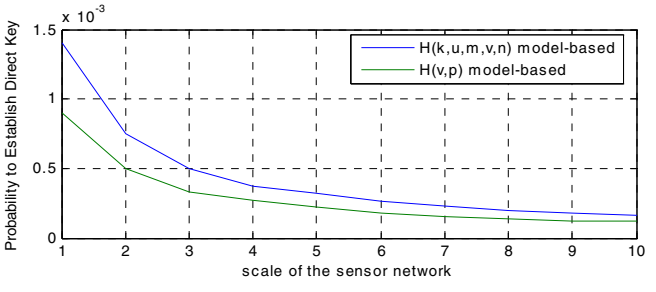


Fig. 2. Comparison of the direct pairwise key establishment probability of $H(v,n)$ and $H(k,u,m,v,n)$ models

Theorem 3: Supposing that the total sensors is N in the sensor network, and the fraction of compromised nodes is p_c, then when $u>v$, the number of affected nodes of the $H(v,p)$ model based key predistribution scheme, is bigger than that of the $H(k,u,m,v,n)$ model based key predistribution scheme.

Proof: Since the number of affected nodes of the $H(k,u,m,v,n)$ model based key predistribution scheme is $p_c \times N \times (m+n)$, and it is proved in [7] that the number of affected nodes of the $H(v,p)$ model based key predistribution scheme is $p_c \times N \times p$. Let $p=x+n$, then there is $u^m * v^n = v^x * v^n \Rightarrow u^m = v^x$. Since $u>v \Rightarrow x>m \Rightarrow p_c \times N \times (m+n) < p_c \times N \times (x+n) = p_c \times N \times p$. □

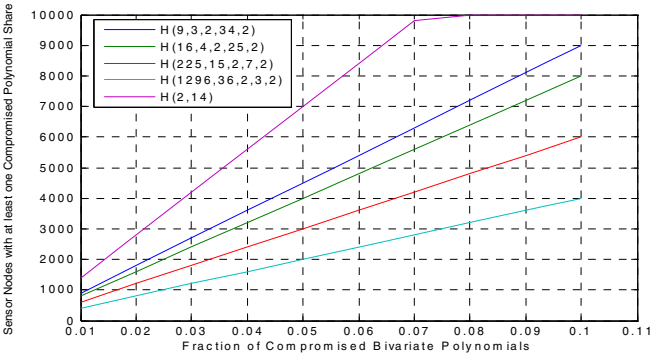


Fig. 3. The comparison between p_c and the number of sensor nodes with at least one compromised polynomial share in sensor networks based on $H(9,3,2,2,n)$ and $H(2,p)$ distribution models

Supposing that the scale of the sensor network is N=10000, Fig.3 presents the comparison between p_c and the number of sensor nodes with at least one compromised polynomial share in sensor networks based on $H(9,3,2,2,n)$ and $H(2,p)$ distribution models. From Fig.3, it is easy to know that the conclusion of theorem 3 is correct, and the number of the affected sensor nodes in the sensor network increases with the increasing of the number of compromised nodes, when the scale of the sensor network is fixed.

5 Conclusions

A new hierarchical hypercube model named $H(k,u,m,v,n)$ is proposed, which can be used for pairwise key predistribution for cluster deployed sensor networks. And based on which, an innovative dynamic pairwise key path algorithm is designed according to the good characteristics of node codes and high fault-tolerance of $H(k,u,m,v,n)$ model. In the newly proposed model, nodes are not needed to be able to communicate with each other directly such as the traditional pairwise key establishment algorithm shall need. So the newly proposed algorithm is an efficient and suitable pairwise key establishment algorithm for the cluster deployed sensor networks.

References

1. L. Eschenauer, V. Gligor: A key-management scheme for distribute sensor networks. In: Vijay Atluri(ed.): 9th ACM Conference on Computer and Communication Security. ACM Press, Washington DC, USA (2002) 41-47.

2. H. Chan, A. Perrig, D. Song: Random key predistribution schemes for sensor networks. In: U. Maurer (ed.): IEEE Symposium on Security and Privacy. IEEE Press, California, USA (2003) 197–213.
3. C. Blundo, A. D. Santis, A. Herzberg, et.al: Perfectly-secure key distribution for dynamic conferences. Lecture Notes in Computer Science. Vol.740. Springer-Verlag, Berlin Heidelberg New York (1993) 471–486.
4. D. Liu and P. Ning. Establishing pairwise keys in distributed sensor networks. In: Felten, Amit. Sahai (eds.): 10th ACM Conference on Computer and Communications Security. ACM Press, Washingtion, DC, USA (2003) 52-61.
5. W. Du, J. Deng, Y. Han, P. Varshney: A pairwise key pre-distribution scheme for wireless sensor networks. In: Felten, Amit. Sahai (eds.): 10th ACM Conference on Computer and Communications Security. ACM Press, Washingtion, DC, USA (2003) 42-51.
6. Donggang Liu, Peng Ning, Rongfang Li: Establishing Pairwise Keys in Distributed Sensor Networks. ACM Transactions on Information and System Security. 20(2005)41-77.
7. L. Fang, W. Du, N. Peng: A Beacon-Less Location Discovery Scheme for Wireless Sensor Networks. In: Znati T, Knightly E, Makki K (eds.): INFOCOM 2005. IEEE Press, Miami, FL, USA (2005)161-171.
8. Wang Lei, Lin Ya-ping: Maximum safety path matrix based fault-tolerant routing algorithm for hypercube interconnection network. Journal of software. 15(2004) 994-1004.
9. Wang Lei, Lin Ya-ping: Maximum safety path vector based fault-tolerant routing algorithm for hypercube interconnection network. Journal of China Institute of Communications. 16(2004)130-137.
10. Lin Ya-ping, Wang Lei: Location information based hierarchical data congregation routing algorithm for sensor networks. Chinese Journal of electronics. 32(2004)1801-1805.
11. W. Heinzelman, J. Kulik, H. Balakrishnan: Negotiation Based Protocols for Disseminating Information in Wireless Sensor Networks. ACM Wireless Networks. 8(2002)169-185.

A DRBAC Model Based on Context for Smart and Secure Services in Intelligent Ubiquitous Home*

Jong Hyuk Park[1], Ji-Sook Park[1], Sang-Jin Lee[1], and Byoung-Soo Koh[2]

[1] Center for Information Security Technologies, Korea University,
5-Ka, Anam-Dong, Sungbuk-Gu, Seoul, Korea
{hyuks00, mongseel, sangjin}@korea.ac.kr
[2] DigiCAPs Co., Ltd., Jinjoo Bldg. 938-26 Bangbae-Dong, Seocho-Gu, Seoul, Korea
bskoh@digicaps.com

Abstract. In intelligent ubiquitous home environments, access control should be able to accommodate dynamic mechanism that was considered various access control components including temporal and spatial information. In this paper, we propose dynamic role based access control model can support smart and secure services suitable for intelligent ubiquitous home. The proposed model is appropriate for user-oriented service based on ubiquitous computing. In the proposed model, a permission is assigned by context information. In addition, this paper observes detailed access control procedures for service in intelligent ubiquitous home, providing use case scenarios and prototype design.

1 Introduction

The evolution of computing technology and networking is bring an era of Ubiquitous Computing (UC). All objects and spaces in everyday life are being intellectualized, allowing users access them anytime, anywhere. Although users might not be aware of them, various interactions among computers are made, providing useful service [1, 2]. UC will initially be realized in services in Intelligent Ubiquitous Home (IUH), which can be divided into Intelligent Ubiquitous Home Entertainment Service (IUHES) - VOD, IP-TV, super-speed Internet/Data, Intelligent Ubiquitous Home Automation Service (IUHAS) - door lock, anti-theft, anti-fire, Intelligent Ubiquitous Home Health Care Service (IUHHCS) - remote diagnosis, remote prescription, self-diagnosis.

The basic concept of the UC is to be 'user-oriented.' To fulfill this concept, base technologies are needed, such as Zigbee, UWB, RFID, WSN, etc. Furthermore, for security of services in IUH, measures such as authentication and authorization (or access control) between user and components shall be considered.

* This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 893 – 901, 2006.

There are Discretionary Access (DAC) and MAC (Mandatory Access Control) as traditional access control methods. Because of the weakness of the models which were not treated in realistic methods, RBAC model was proposed by Sandhu, his colleagues and NIST [3~5]. In addition, it was recognized as the abstract concept for rights management in application and database management system in the 1980's and the beginning of 1990's. and Dobson and McDermid [6] use a terminology called "Functional Roles". Furthermore, Baldwin [7] introduced Least Privilege as a layered model which is related and organized each other in NPD (Named Protection Domain)s.

However, the access control methods are not appropriate for IUH because the models are based on components of access control in fixed environment - user, object, subject and permission (or role). In IUH, a dynamic access control method is necessary, where 5W and 1H (Why, What, Who, Where, When and How) [8, 9] are considered. In IUH, an overall consideration should be made for an environment where information of user and user context (location, time, preference) can be appropriately combined and used in service [10, 11].

That is to say, because the early RBAC had many conditional matters in modeling, extended RBAC model was appeared. Bertino proposes *TRBAC* (Temporal Role-Based Access Control) model [12]. The TRBAC is possible to active and inactive of periodic role, so it used time dependent relationship among events when a role was active or inactive. After that, *GTRBAC* (Generalized Temporal Role-Based Access Control) [13] was proposed through the complement and extension of this model. Giuri and Iglio [14] proposed a new mechanism which defined context based access control. It could decide the policy about rights according to content of object by extending the concept that is the existing permission.

Since 2000, the study about extended RBAC model [15~17] has been progressed. It is the access control scheme which is compatible with security in ubiquitous environment through engrafting context information on dynamic environment. But, so far the study has been remained in the level of conceptual model.

In this paper, the proposed IUHS-DRBAC (for Intelligent Ubiquitous Home Service - Dynamic Role Based Access Control) model is based on context including temporal and spatial information to consider the dynamic environments of IUH. Moreover, we design a prototype of the IUHS-DRBAC based on [3~5]. Finally, by providing use case scenarios of the proposed model, this paper observes detailed active procedures in smart and secure services in IUH.

The rest of this paper is organized as follows. In Section 2, we present a IUHS-DRBAC model for the secure service in IUH. Finally, the conclusion and future works are discussed in Section 3.

2 IUHS-DRBAC Model

In this section, we discuss design, service flow, use case scenarios, and prototype design and analysis.

2.1 IUHS-DRBAC Design

The IUHS-DRBAC model consists of components of the basic RBAC - users, roles, permissions, session, and context which can apply dynamic environments elements. Table 1 depicts a formal definition of the IUHS-DRBAC and Figure 1 presents graphically the model.

Table 1. Formal Definition of the IUHS-DRBAC Model

The proposed DRBAC model includes concepts of a basic RBAC and is formalized as follows:

- *U, R, P, S, C r*epresent the finite set of users, roles, permissions, sessions, and context information respectively within the system.
- PA $\subseteq$ P x R represents the finite set of permission to role assignments. This is a many-to-many relationship.
- UA $\subseteq$ U x R represents the finite set of user to role assignments. This is a many-to-many relationship.
- user: S $\rightarrow$ U, a function that maps a session s_i to a user.
- RH $\subseteq$ R x R is a partial order on R, called the role hierarchy or role dominance relation, also written as $\geq$, where $r_1 > r_2$, only if all permissions of r_2 are also permission of r_1 are user of r_2.
- For *RBAC 0:* $S \rightarrow 2^R$, a function that maps a session s_i to a set of roles, where $roles(s_i) \subseteq \{r \mid (user(s_i), r) \in UA\}$, and each session s_i has the permissions $\cup r \in roles(s_i) \{p \mid (p, r) \in PA\}$.
- For *RBAC1*, *roles:* $S \rightarrow 2^R$ is modified from *RBAC0* to require $roles(s_i) \subseteq \{r \mid (\exists r' \geq r)[(user(s_i), r') \in UA]\}$ and each session s_i has the permissions $\cup r \in roles(s_i) \{p \mid (\exists r' \leq r)[(p, r') \in PA]\}$.
- For *Context Constraints*, *DRBAC* adds context constraints in the form of restrictive functions that operate on *basic RBAC* components to meet the specific needs of the protection policies of an organization and factor of dynamic environments. Constraints in DRBAC include *SoD (Separation of Duties)* and C*ardinalities* including concepts of the *RBAC 2.* In addition, It includes *context* such as environments, time, location, etc.
- CCA $\subseteq$ CC x R represents the finite set of CC to role assignments. This is a many-to-many relationship.

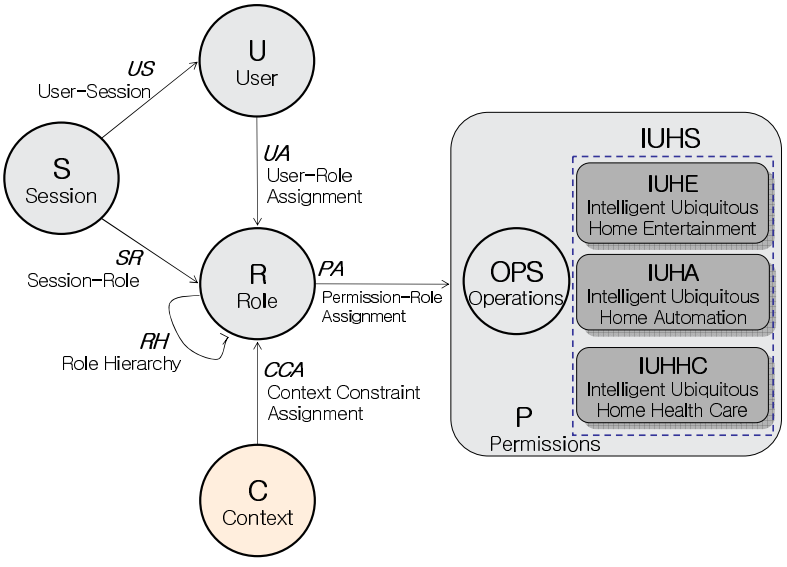


Fig. 1. IUHS-DRBAC Model

2.2 IUHS-DRBAC Service Flow and Use Case Scenarios

2.2.1 Intelligent Ubiquitous Home Service Flow

In this subsection, we construct the whole service flow of the IUHS-DRBAC model and the use case scenarios of the proposed model. In addition, we discuss specific working process among AC model components such as user, subject, role, permission, etc.

We assume the follows: IUH Gateway (IUHG) can collect context information through Infra-technologies such as Zigbee, UWB, RFID, WSN, etc. Furthermore, it can control and manage the whole intelligent home service through free wireless communication among sensor nodes.

When user enters his / her home through a home gate, user is authenticated through biometric information recognition or camera recognition. With regard to authentication simply, we consider not only user / device authentication but also domain authentication. The IUHG takes function of the domain controller and all home devices in domain which have the private domain key. In addition, one domain has one key at least and one device can be registered in several domains.

When the IUHS was requested by user, user is categorized as three classes (Family, Visitor, Temporal visitor) by authenticated information (User Classification). After that, User-Role is assigned and Context Constraints (CC) process is working to consider dynamic environment. The CC is proceeded in sequence such as UCS → CF → CC, and the User Context Searching (UCS) searches user context from context-DB. In addition, the Context Fusion (CF) combine context attributes of the user context from the context-DB with current context attributes of the

components which is formed User Location (UL), Scheduled Job (SJ), Current Environments (CE), etc. and then assigns CC.

2.2.2 IUHES Use Case Scenarios

We assumed that in IUH, user location is recognized by the image sensor, speech sensor, etc. The context information including user location transmits to centrol the IUHG through a WSN Module is embedded on watch, mobile phone, ring, etc. in realtime.

In this subsection, we discuss use case scenarios of the Intelligent Ubiquitous Home Entertainments Service (IUHES) and the Intelligent Ubiquitous Home Automation Service (IUHAS).

- **IUHES Scenario**

Table 2. Assumption for the IUHES Scenario

Context Information	Detail information
User (James) Classification	Family, Student, Man, 18 age
Scheduled Job in context-DB	20:00-23: 00 Studying 18:00-20: 00 Rest
Authentication	When all users enter home gate, user is authenticated. Its information is used to service in IUH.

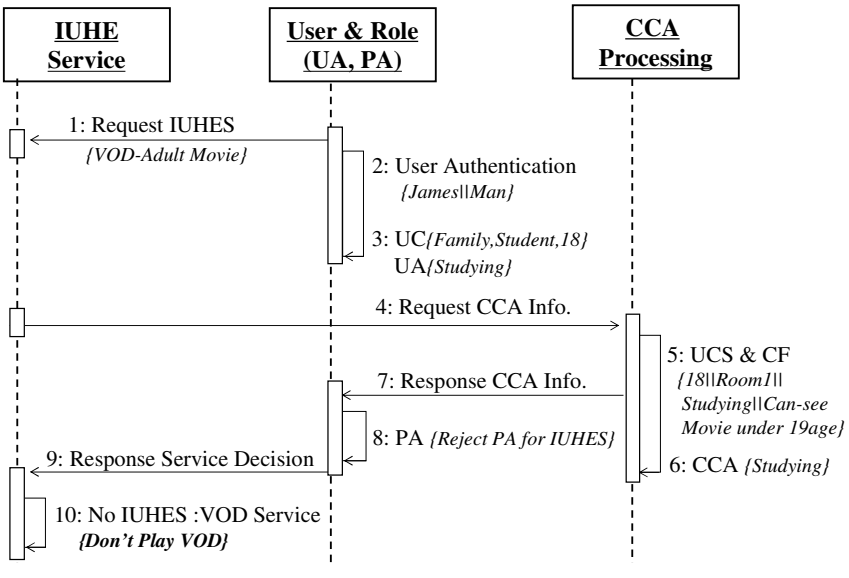


Fig. 2. IUHES Scenario and Context Information Flow

After having dinner in the kitchen, James goes to his room and tries to watch a IUHES (VOD) for adults over 20 years. At the moment, the time is 20:30. We discuss the scenario where the service is determined through CCA and PA procedure based on context information.

As he enters the home gate, user authentication is made. In IUH, context information such as user location, preference, status, etc. freely communicates with the central IUHG through the WSN module.

Through the authenticated information, James' user class data is set to {Family, Student, 18, Man}. In order to provide VOD service for adults over 20 years, the UA is executed, as well as the CCA processing for PA. During the CCA processing, context information saved in the context DB (User Class Information and User Scheduled Job Information) is compared to assign context constraints. In the case, the PA becomes 'No Permission', home entertainment (VOD) service for adults is finally rejected (Refer to Figure 2).

- **IUHAS Scenario**

Table 3. Assumption for the IUHAS Scenario

Context Information	Detail information
User (Tom) Classification	Family, Child, Man, 7 age
Scheduled Job in context-DB	- Washing Time: No limit - Temperature Limit : 15~40 - Turning off automatically if 30 minutes elapsed during one's absence since user turns on water in bathroom.
Authentication	When all users enter home gate, user is authenticated. Its information is used to service in IUH.

To prevent Tom, 7 years old, from getting burned while he bathes himself, water can be withheld from exceeding a certain temperature. Just like other cases, user authentication is made when he enters the home gate. As he takes a bath in the bathroom, The DRBAC model is applied to control water temperature automatically.

If the temperature requested by Tom exceeds the limit set by his parents or administrator (15~40C), the WSN module senses the request and transmits the information to the central IUHG, and the request was blocked. Furthermore, if he leaves a tap open for more than 30 minutes without being in the bathroom, tap is automatically closed. In short, when he requests for the change of temperature, the UA is executed, as well as the CCA processing for the PA. During the CCA processing, context information saved in the context DB is compared to assign context constraints. In the case, "No Permission," is assigned, the IUHA service doesn't be provided to Tom (Refer to Figure 3).

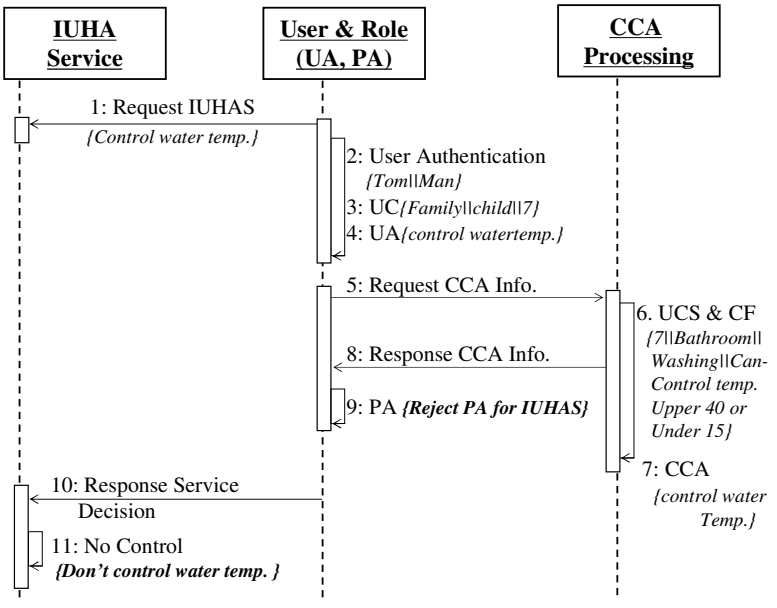


Fig. 3. IUHAS Scenario and Context Information Flow

2.3 IUHS-DRBAC Prototype Design and Analysis

In this subsection, we discuss prototype design of the IUHS-DRBAC model and compare it with the existing RBAC models to explain the features that have been improved.

With the consideration against context information flowing at IUHES scenario in subsection 2.2.2, we can find that the PA has been completed based on James' context information such as time, location, user schedule, and so on.

Table 4 shows some parts of the Context DB. Fields in Contex-DB constistes of User, Time(s), User Location, Home Services, User Scheduled Jobs information, PA for Service such as follows.

Table 4. Some parts of the Context DB about the IUHES Scenario

User	Time (s)	Location	Services in IUH		User Scheduled Information	PA for Service	Home Service Decision
			HE, HA, HHC	Etc			
. . .		. . .			. . .	. . .	. . .
James	71445	Kitchen	-	Dinner	James={ Family, Student, Man, 18, Rest}	-	-
. .		. .			. .	. .	. .
James	73832	James's Room	HE = VOD	-	James={Family, Student, Man, 18 ,Studying}	0	HS=NO
. .		. .			. .	. .	. .

Basically, access right is held by users. It is assumed that information granting or denying access of each user to certain services or objects (access possible or access impossible) are already saved in a Service Policy Server (SPS). The SPS holds access information of each user to certain objects, and according to user classification by user authentication, user context information is determined. In all attributes, however, only "AND" conditions should exist without any "OR" conditions, as shown below.

$$Context\ Attributes = James \cap HS \cap JS \cap CE$$

Table 5 shows comparison among the existing models and the proposed model.

With regard to considering spatial information including user location information, in the exiting models (RBAC 96, GTRBAC) not mention permission assignment about it. But, the proposed model provides service at only designated space applying spatial information received through the WSN module.

With regard to considering temporal information such as Time, Date, Year, in the exiting model, the RBAC96 model not mention and GTRBAC model mention partially (periodic, duration constraint). But, the proposed model combines and applies basic security policy and temporal information.

With regard to considering role inheritance among domains, in the exiting model not mention domain. But, in the proposed model, even if user moves among domains or user modifies resource related with corresponding service, it allows user to inherit role among domains by user location and user schedule.

Table 5. Comparison between proposed model and existing model

Considering Items	RBAC96	GTRBAC	Proposed model
Considering spatial information (User location information)	X	X	○
Considering temporal information (Time, Date, Year)	X	△	○
Considering role inheritance among domains	X	X	○

○: Satisfied, △: Average, X: Unsatisfied.

3 Conclusion

In this paper, we proposed dynamic role based access control model can support smart and secure services suitable for intelligent ubiquitous home. The proposed model was considered role and authority of the existing RBAC model and was supported role constraints according to user location information. In addition, permission was assigned based on context information such as user location, scheduled jobs, current environments, etc. Moreover, by providing use case scenarios and prototype design, this paper observes detailed access control procedures for service in intelligent ubiquitous home.

Future research should focus on methods to complement and improve DRBAC for smart and secure services proposed in this research by developing a conjunctive model to enhance protection of user privacy.

References

1. Mark Weiser: Hot topic: Ubiquitous Computing IEEE Computer (1993), 71-72
2. Jong-Hyuk Park, Jun-Choi, Sang-Jin Lee, Hye-Ung Park, and Deok-Gyu Lee, "User-oriented Multimedia Service using Smart Sensor Agent Module in the Intelligent Home", CIS 2005, Springer-LNAI, Vol. 3801 (2005), 313 – 320
3. David F. Ferraiolo and D. Richard Kuhn: Role-based access controls, 15th NIST-NCSC National Computer Security Conference, Baltimore (1992), 554-563
4. Gregory Tassey, Michael P. Gallaher, Alan C. O'Connor, Brian Kropp: The Economic Impact of Role-Based Access Control, NIST Planning Report 02-1 (2002)
5. Sandhu, R.S., Coyne, E.J., Feinstein, H.L. and Youman C.E.: Role Based Access Control Models, IEEE Computer, Vol. 29, No. 2 (1996), 38-47
6. Dobson, J.E., and J.A.McDermid: Security Models and Enterprise Models, in Database Security Ⅱ: Status Prospects, C. E. Landwehr(ed.), North Holland (1989), 1-39
7. Baldwin, R. W.: Naming and Grouping Privileges To Simpify Security Management in Large Database, in Proceeding IEEE Computer Society Symposium on Research in Security and Privacy (1990), 184-194
8. S. Jang, W. Woo: ubi-UCAM: A Unified Context-aware Application Model, HCI Korea 2003, Vol. 12, No. 2 (2003), 346-351
9. G.D. Abowd, E. D. Mynatt: Charting Past, Present, and Future Research in Ubiquitous Computing, ACM Trans. on Computer-Human Interaction, Vol. 7, No.1 (2000), 29-58.
10. Anind K. Dey, Daniel Salber and Gregory D. Abowd: A Context-based Infrastructure for Smart Environments, MANSE '99, Dublin, Ireland (1999)
11. Jong-Hyuk Park, Heung-Soo Park, Sangjin Lee, Jun Choi, Deok-Gyu Lee: Intelligent Multimedia Service System Based on Context Awareness in Smart Home, KES 2005, LNAI , Vol. 3681, Australia (2005), 146-1152
12. Bertino, E., Bonatti, P. A. and Ferrari E. TRBAC: A Temporal Role-Based Access Control Model, ACM Transaction on Information and System Security (TISSEC), Vol. 4, No. 3 (2001), 191-233
13. James B.D. Joshi, Elisa Bertino, Usman Latif, and Arif Ghafoor: A Generalized Temporal Role-Based Access Control Model, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, Vol. 17, No. 1 (2005)
14. L. Giuri and P. Iglio: Role templates for content-based access control, In Proceedings of the Second ACM Workshop on Role Based Access Control, Virginia, USA (1997)
15. R.J. Hulsebosch, A.H. Salden, M.S. Bargh, P.W.G. Ebben, J. Reitsma: Context Sensitive Access Control, SACMAT'05 (2005)
16. Antonio Corradi, Rebecca Montanari, Daniela Tibaldi: Context-based Access Control Management in Ubiquitous Environments, NCA'04 (2004)
17. S. Srinivasan A. Dey M. Ahamad M. J. Covington, W. Long and G. Abowd: Securing context-aware applications using environment roles (2001)

Investigating Authentication Mechanisms for Wireless Mobile Network

Binod Vaidya[1], YoungJin Kim[2], Eung-Kon Kim[3],, and SeungJo Han[4]

[1] Dept. of Electronics & Computer Eng., Tribhuvan Univ., Nepal
bvaidya@ioe.edu.np
[2] DACOM Corporation, Korea
kyj2000@chal.net
[3] Dept. of Computer Science, Sunchon National Univ., Korea
kek@sunchon.ac.kr
[4] Dept. of Information & Communication Eng., Chosun Univ., Korea
sjbhan@chosun.ac.kr

Abstract. As the portable devices are becoming popular, the demand for ubiquitous Internet access services has grown extremely. However the increasing demand for ubiquitous services imposes more security threats to communications due to open mediums in wireless networks. The further widespread deployment of Wireless IP networks, however, depends on whether secure networking can be achieved. We propose an efficient authentication framework for wireless mobile network, which is based on one-time password (OTP) mechanism. Further we have simulated the implementation of proposed scheme and EAP-TLS and analyzed the performance in terms of various performance metrics.

1 Introduction

As the portable devices are becoming popular, the demand for ubiquitous Internet access services has grown extremely. With public wireless local area network (WLAN) system, it can provide fast Internet access at higher speeds using portable devices such as laptop computers and Personal Digital Assistances (PDA). The IEEE 802.11 [1] WLAN, popularly known as Wireless Fidelity (Wi-Fi) has grown steadily in popularity since its inception and is now well positioned to complement much more complex and costly other technologies such as the Third Generation (3G).

WLANs provide unprecedented flexibility in that an area not originally intended as a collaborative workspace can accommodate a large number of wireless clients. However, WLANs present some serious security challenges such as authentication.

In this paper, we propose an efficient authentication framework for wireless mobile network, which is based on one-time password (OTP) mechanism.

2 Authentication Mechanism Using EAP-TLS

In IEEE 802.1x [2], a network-to-client authentication mechanism utilizing EAP (Extensible Authentication Protocol) [3] is used as the encapsulation protocol

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 902–911, 2006.

for upper-layer authentication information. Thus the IEEE 802.11 document for improved WLAN security [4] specifies the usage of IEEE 802.1x for access control for mobile clients, and for dynamic key distribution to be used between the mobile station and its AP.

EAP-Transport Layer Security (EAP-TLS) [5] is a Point-to-Point Protocol (PPP) extension supporting additional authentication methods. EAP-TLS is based on a certificate approach, and requires trusted Certificate Authorities (CAs).

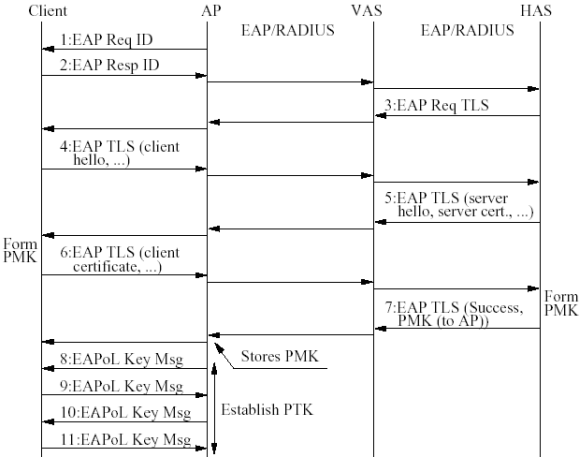


Fig. 1. Authentication Mechanism using EAP-TLS

Authentication mechanism using EAP-TLS is shown in Fig. 1. In 802.1x as used by 802.11i, neither the AP nor Client will allow any data packets to be exchanged before the EAP message exchange completes successfully. [6][7][8] Suppose the client is attempting to acquire access in a remote domain run by a roaming services. The AP forwards the client's identity to a Visiting authentication server (VAS). Based on the domain part of the client's identity VAS acts as proxy and forwards the message to HAS.

EAP uses a Send/ACK mechanism to provide reliable transport. It is important that each of the TLS messages can be carried inside a single packet. TLS session resumption has the advantage of avoiding any TLS related public key operations during handover. It is reasonable to assume that they both have their certificates issued by the same CA and that they both have the certificate of the CA stored locally.

As a side-effect of the TLS handshake, client and HAS each constructs a pre-master key (PMK), which HAS forwards to the AP (via LAS) along with the EAP-TLS success message. The final messages are EAP over LAN (EAPoL) key messages that use the PMK to mutually authenticate client and the AP, and to establish the dynamic keys to use for unicast (pairwise transient key (PTK)).[9]

3 OTP Based Authentication Mechanism

There is a desire to increase the security and performance of authentication mechanisms due to strong user authentication, protection of user credentials, mutual authentication and generation of session keys. As demand for strong user authentication grows, OTP-based authentications tend to become more common.

We put forward a scheme for implementing authentication mechanism based on the hash function. Initially Lamport [10] proposed a One-time password /hash-chaining technique, which has been used in many applications. S/KEY is an authentication system that uses one-time passwords. The S/KEY system [11] is designed to provide users with one-time passwords which can be used to control user access to remote hosts. One Time Password (OTP) system [12] designed to provide protections against passive eavesdropping and replay attacks.

The IETF RFC 3748 [3] has defined that with EAP, OTP method can be used. Further the Internet draft [13] defines the one-time password (OTP) method, which provide one-way authentication but not key generation. As a result, the OTP method is only appropriate for use on networks where physical security can be assumed. This method is not suitable for using in wireless IP networks, or over the Internet, unless the EAP conversation is protected.

3.1 Overviews

In this scheme, the AP receives the password which is processed from the client and the server's original password exclusive-OR (XOR) the stream bits, in order to defend malice attack. We devise concept of authentication scheme using one-time password for IEEE 802.11 network. The client needs remote authentication, so the AP not only receive the password from the client but fetch client's basic data from Home Network Server (HNS) by roaming, then the AP computes and compares the hash value to accomplish mutual authentication. The key exchange between the client and the AP generates a session key and responses new stream bits to HNS to update client's database. Finally, the client sends one-time password (OTP) to the AP, and the client refreshes secret key of the WEP in IEEE 802.11, to defend replay attack.

In order to prevent client and AP in this protocol from several attack methods. The techniques include per-packet authentication and encryption, etc are adopted. This scheme can be used for defending malice attack by verification of the client's password and defending replay attack by refreshing secret key.

3.2 Operations

There are four phases in proposed authentication scheme, which are as follows:

1. Phase 0: Discovery phase;
2. Phase 1: OTP Authentication phase;
3. Phase 2: Secure key exchange phase; and
4. Phase 3: Refreshing OTP phase.

The conversation phases and relationship between the parties is shown in Fig 2.

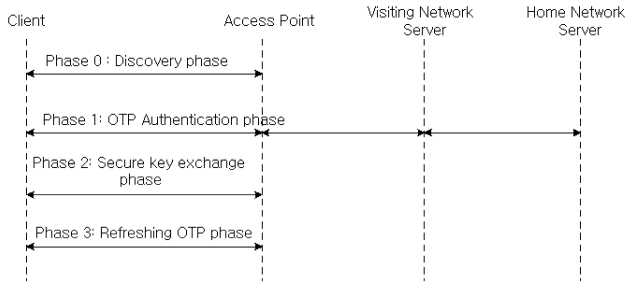


Fig. 2. Conversation Overviews

Phase 0: Discovery phase

In the discovery phase (phase 0), the client or the station (STA) and Access Point (AP) locate each other and discover each other's capabilities. IEEE 802.11 provides integrated discovery support utilizing Beacon frames, allowing the STA to determine the capabilities of AP. When the client receives Beacon frame, it knows itself is in Visiting Network, and requires authentication from Home Network.

Phase 1: OTP authentication phase

The authentication phase (phase 1) begins once the client and AP discover each other. This phase always includes OTP authentication. In this phase, the client authenticates itself as a legal member for a particular AP and needs remote authentication. The steps are shown as the follows:

Step 1: The AP send Request message.

Step 2: The client will then send $\{ID_C \| ID_{HNS} \| Y_C \| H(ID_C, A)\}$. Where ID_C is client's identity; ID_{HNS} is identity of Home Network Server (HNS); $Y_C = a^{R_C} \bmod q$; R_C is random number produced by a client; $H()$ is one way hash function; $A = PW_C \oplus SB$; PW_C is client's password; and SB is stream bits generated by the server to share with the client.

Step 3: After receiving $\{ID_C \| ID_{HNS} \| Y_C \| H(ID_C, A)\}$ from the client, the AP sends $\{ID_{HNS} \| ID_{AP} \| ID_C\}$ to the Visiting Network Server (VNS) . Where ID_{AP} is AP's Identity.

Step 4: After receiving ID_{HNS} from the roaming client, the VNS will encrypt ID_C and ID_{AP} with K_{VH} which is a secret shared key between VNS and HNS. Then it sends $\{ID_{HNS} \| E_{VH}(ID_C \| ID_{AP})\}$ to the HNS.

Step 5: Depending on ID_C, the HNS examines PW_C, SB, and client's privacy key (K_C) from user's database, and generates a new stream bits (SB_N) and A. The ID_C and SB_N are encrypted with K_C producing $E_{K_C}(ID_C \| SB_N)$. Then $E_{K_C}(ID_C \| SB_N)$, ID_C, and A are encrypted by HNS with symmetric cryptosystem to get $E_{VH}(ID_C \| ID_{AP} \| A \| E_{K_C}(ID_C \| SB_N))$. Then HNS will send $\{ID_{VHS} \| E_{VH}(ID_C \| ID_{AP} \| A \| E_{K_C}(ID_C \| SB_N))\}$ is send to the VNS.

Step 6: Upon receiving $\{ID_{VHS} \| E_{VH}(ID_C \| ID_{AP} \| A \| E_{K_C}(ID_C \| SB_N))\}$ VNS will decrypt it using K_{VH}, then $E_{K_C}(ID_C \| SB_N), ID_C$ and A are encrypted by VNS with symmetric cryptosystem using K_{AS}, which is shared key between AP and VHS. Then, it sends $\{ID_{AP} \| E_{AS}(ID_C \| A \| E_{K_C}(ID_C \| SB_N))\}$ to AP.

Step 7: Upon receiving $\{ID_{AP} \| E_{AS}(ID_C \| A \| E_{K_C}(ID_C \| SB_N))\}$ from the VNS, then the AP computes and verifies $H(ID_C, A)$ between the client and the server. If true, it will send authentication success frame $\{ID_C \| Y_A \| E_{K_C}(ID_C \| SB_N)\}$ to the client. Where $Y_A = a^{R_A} \bmod q$; R_A is random number produced by AP. Y_A will be used for key exchange in later steps.

Step 8: The client accomplishes authentication procedure, then it sends $\{ID_C \| IPA_{HNS} \| H(SB_N)\}$ to AP. Where IPA_{HNS} is IP address of HNS.

Step 9: AP will send $\{ID_C \| H(SB_N)\}$ to the HNS to update client's database in the HNS.

Step 10: The client and the AP apply Y_A or Y_C to generate a session key (K), then the client sends $\{ID_C \| E_K(otp \| ctr \| ID_C)\}$ to the AP. Where *otp* is the client's one-time password shared with the AP; and *ctr* is the counter of otp.

Step 11: The AP responds with $\{ID_C \| H(otp, ctr)\}$ to the client.

It should be noted the counter is a positive integer and will decrease by one once the otp is changed. When it decreases to zero, the client should reinitialize his otp.

Phase 2: Secure key exchange phase

The Secure key exchange phase (phase 2), begins after the completion of OTP authentication. In practice, the phase 2 is used to negotiate a new secret key between the client and the AP. The steps of the protocol are as shown in the following:

Step 1: The client sends $\{ID_C \| E_K(otp_{I+1} \| H(otp_I, ctr, ID_C))\}$ to AP. It should be noted that $H(opt_{I+1})$ is equal to otp_I and K is session key by OTP authentication phase (phase 1).

Step 2: The AP checks the MAC and ID_C by checking if $H(opt_{I+1})$ is equal to otp_I. If it is true, the client is a legal member. It will replace otp_I by opt_{I+1} and decrease counter by 1. And AP transmits $H(otp_I, ctr)$ to client.

Step 3: The client checks hash value and decreases the counter by 1.

By these steps, the client and the AP can achieve the goal of mutual authentication. In this phase, the client and the AP have the WEP key just for this session.

Phase 3: Refreshing OTP phases

When the counter decreases to zero, the client should change its one-time password, otherwise the AP shall prohibit the client from using its services. The steps for refreshing *otp* are shown in the following:

Step 1: The client sends its $\{ID_C \| E_K(otp_{I+1} \| otp_N \| ctr_N \| H(otp_{I+1}, otp_N, ctr_N, ID_C))\}$ to AP, where $H(otp_{I+1})$ is equal to otp_I, otp_N is its new one-time

password, ctr_N is a new counter for (otp_N) and K is session key by authentication phase.

Step 2: The AP verifies MAC and otp_{I+1} (check $H(otp_{I+1})$ is equal to otp_I). If this is true, the client is a legal member, and AP replaces the otp_I by otp_N and resets ctr to ctr_N. Then the AP sends $\{ID_C \| H(otp_I, ctr)\}$ to the client.

Step 3: The client checks his ID and the hash value.

3.3 Security Analysis

The proposed authentication mechanism is analyzed:

Malice attack Protection - In this scheme, the AP receives the password which is processed from the client and the server's original password exclusive-OR (XOR) the stream bits, in order to defend malice attack.

Mutual Authentication - This scheme provides mutual authentication. Strong mutual authentication can prevent user from being impersonated.

Replay Protection - This scheme is inherently designed to avoid replay attacks by choosing one-time password is that it refreshes secret key of the WEP in IEEE 802.11 anytime. Because of refresh OTP, the lifetime of the share key can be extended.

Confidentiality - Each the element of the key exchange is protected by a shared secret that provides high confidentiality.

4 Performance Evaluations

It has been conceived a system architectural model in order to analyze the authentication mechanisms in wireless network as shown in Fig. 3. In order to evaluate the performance of the above authentication scheme in Wireless IP network, we have designed experimental model and simulated using OPNET Modeler [14].

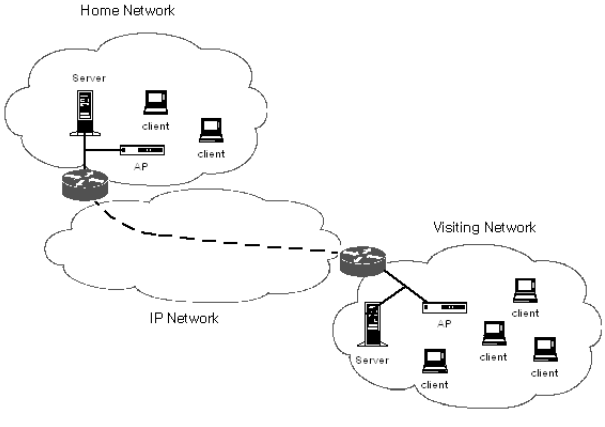


Fig. 3. System Architeture Model

In wireless networks, such as Mobile IP networks, the challenges of authentication lie in obtaining the credentials, such as keys, for mobile node (MN) authentication when MNs are roaming among wireless networks. To provide efficient and secure authentication in wireless IP networks, authentication architecture and authentication scheme are two main issues. The objective of authentication architectures is to provide secure interconnection between wireless networks. The authentication scheme is designed to verify the user and generate credentials with mutual trust. Since the mutual trust is to protect the communication between networks and MNs, authentication process is necessary to provide security. We have considered EAP-TLS and proposed authentication scheme based on OTP protocol as authentication schemes. The roaming user can be authenticated in the Visiting Network by Home Network Server (HNS).

In wireless IP networks, the challenges of authentication lie in obtaining the credentials, such as keys, for client authentication when clients are roaming among wireless networks. However, the efficiency of authentication with respect to authentication delay and bandwidth efficiency is significant, because an authentication process introduces an overhead of communications. Thus the authentication protocol influences QoS metrics, such as authentication delay, throughput and response time due to authentication overhead.

4.1 Assumptions

For the experimental purpose, two scenarios have been designed. First one with the implementation of OTP-based authentication scheme whereas second with implementation of EAP-TLS in wireless network.

For the experimental evaluation, we have assumed the various performance metrics in order to analyze the effect of proposed authentication scheme and the EAP-TLS. We have considered following performance metrics:

- Response Time: the total time required for traffic to travel between two points. It includes connection establishment, security negotiation time as well as the actual data transfer time.
- Authentication Delay: the time involved in an authentication phase of a given protocol.

We have implemented the OTP-based authentication mechanism and the EAP-TLS in our experiment. For Voice over IP (VoIP) services, a base voice codec scheme is considered to be G.729.

For the wireless users, we have considered IEEE 802.11b as our WLAN protocol. And the transmission speed is 11 Mbps between the AP and the clients. We have considered the scenario that a mobile client is roaming into visiting network domains.

4.2 Results and Analysis

In order to investigate the performance of OTP-based authentication scheme and EAP-TLS authentication scheme in wireless 802.11 network, we have analyzed

different aspects of experimental results obtained. Particularly we have investigated the impact of proposed authentication scheme and EAP-TLS in Wireless IP network on the response time and authentication delay with number of concurrent VoIP calls.

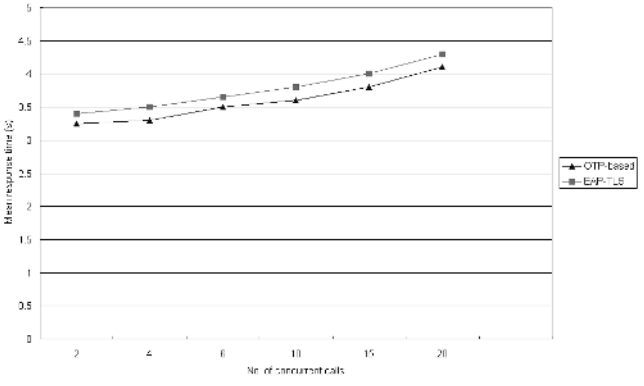


Fig. 4. Mean Response Time

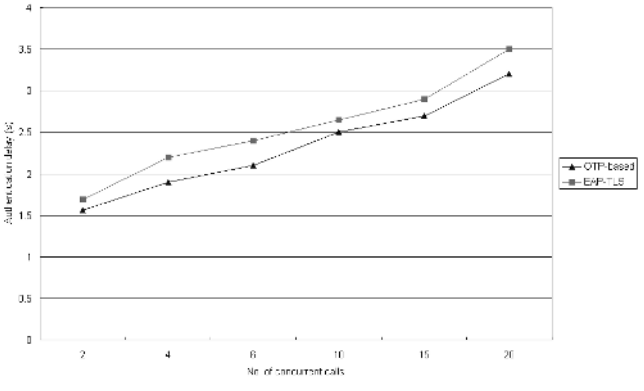


Fig. 5. Authentication Delay

Fig 4 illustrates the mean response time for different number of concurrent VoIP calls with implementation of OTP-based and EAP-TLS authentication schemes. The result shows that with increase in no of concurrent VoIP calls, the mean response time increases. It can be seen that the response time in second scenarios has slightly higher than that of first one.

The Authentication delays for different numbers of concurrent VoIP calls with implementation of OTP-based and EAP-TLS authentication schemes is depicted in Fig. 5. On analyzing two scenarios of the experiment, it can be seen that the Authentication delay for OTP-based is reduced than that for EAP-TLS.

5 Conclusions and Future Work

In this paper we have shown the major security measures in wireless 802.11 LAN. We put forward OTP-based authentication scheme for Wireless IP network. Proposed authentication scheme has advantage that it makes the simpler computational complexity with sufficient security. The concept of refresh password is proposed to renew the secret key of the WEP in IEEE 802.11. Moreover, it will be convenient for a user to register once to a server for roaming every AP.

We have simulated the experimental scenarios with implementation of OTP-based scheme and EAP-TLS in Wireless IP network. Performance analysis of two different authentication methods over wireless IP network was investigated in order to view its impact on performance measurements such as response time and authentication delay. Both mean response time and authentication delay is reduced in case of OTP-based authentication scheme.

In the future work, we will further investigate tradeoffs involved in the different approaches. Furthermore we will formulate a formal analysis of authentication mechanism using one-time password protocol for wireless IP network using BAN logic, which is the one of known logics.[15] [16]

Acknowledgement

This study was supported (in part) by research funds from Chosun University 2006.

References

1. Institute of Electrical and Electronics Engineers: Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific Requirements Part 11 - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802.11-2003, 2003
2. IEEE Standards for Local and Metropolitan Area Networks: Port based Network Access Control, IEEE Std 802.1X-2004, Dec 2004
3. B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz: Extensible Authentication Protocol (EAP), IETF RFC 3748, Jun 2004
4. IEEE P802.11i/D10.0. Medium Access Control (MAC) Security Enhancements, Amendment 6 to IEEE Standard for Information technology - Telecommunications and information ex-change between systems - Local and metropolitan area networks - Specific requirements Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications. April 2004.
5. B. Aboba, D. Simon: PPP EAP TLS Authentication Protocol, IETF RFC 2716, Oct 1999
6. M. Gast: 802.11 Wireless Networks - The Definitive Guide, O'Reilly, Dec. 2002
7. K.H Baek, S. W. Smith, and D. Kotz: A Survey of WPA and 802.11i RSN Authentication Protocols, Dartmouth College Computer Science, Technical Report TR2004-524, Nov 2004

8. B. Potter, and B. Fleck: 802.11 Security, O'Reilly, Dec. 2002
9. J. Edney, and W. A. Arbaugh: Real 802.11 Security - Wi-Fi Protected Access and 802.11i, Addison Wesley, Jul 2003
10. L. Lamport: Password Authentication with insecure communication, Communications of the ACM, Vol. 24 No. 11, Nov. 1981, pp 770-722
11. N. Haller: The S/KEY One-Time Password System, Proc. of the Symposium on Network and Distributed Systems Security, Internet Society, CA, USA, Feb. 1994
12. N. Haller, C.Metz, P. Nesser, and M. Straw; A One-Time Password System, IETF RFC 2289, Feb 1998
13. L. Blunk, J. Vollbrecht, and B. Aboba: The One Time Password (OTP) and Generic Token Card Authentication Protocols, Internet draft <draft-ietf-eap-otp-00.txt>
14. OPNET Modeler Simulation Software, http://www.opnet.com.
15. M. Burrows, M. Abadi and R. Needham: A logic of authentication, ACM Transactions on Computer Systems, 8(1), Feb. 1990, pp 18-36.
16. M. Abadi, and M. Tuttle: A semantics for a logic of authentication, Proc. of the Tenth Annual ACM Symposium on Principles of Distributed Computing, Aug. 1991, pp. 201-216.

M²AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags

Pedro Peris-Lopez, Julio Cesar Hernandez-Castro, Juan M. Estevez-Tapiador, and Arturo Ribagorda

Computer Science Department, Carlos III University of Madrid
{pperis, jcesar, jestevez, arturo}@inf.uc3m.es

Abstract. Low-cost Radio Frequency Identification (RFID) tags affixed to consumer items as smart labels are emerging as one of the most pervasive computing technologies in history. This presents a number of advantages, but also opens a huge number of security problems that need to be addressed before its successful deployment. Many proposals have recently appeared, but all of them are based on RFID tags using classical cryptographic primitives such as Pseudorandom Number Generators (PRNGs), hash functions, or block ciphers. We believe this assumption to be fairly unrealistic, as classical cryptographic constructions lie well beyond the computational reach of very low-cost RFID tags. A new approach is necessary to tackle the problem, so we propose a minimalist lightweight mutual authentication protocol for low-cost RFID tags that offers an adequate security level for certain applications, which could be implemented even in the most limited low-cost tags as it only needs around 300 gates.

Keywords: Ubiquitous Computing, RFID, Tag, Reader, Pseudonym, Privacy, Mutual-Authentication.

1 Introduction

At the moment, the most extended identification systems are barcodes. Recently, the mass deployment of Radio Frequency Identification (RFID) systems has taken place [8]. Around 5 billion barcodes are read daily, so efficiency gains from using RFID tags could substantially lower the cost of tagged items [9,12]. The penetration of RFID systems is nowadays mainly limited by privacy concerns and by their cost, which must be between 0.05 and 0.1 € to be considered affordable.

The low cost demanded for RFID tags forces them to be very resource limited. Typically, they can only store hundreds of bits, have 5-10K logic gates, and a maximum communication range of a few meters. Within this gate counting, only between 250 and 3000 gates can be devoted to security functions. It is interesting to recall that for a standard implementation of the Advanced Encryption Standard (AES), 20-30K gates are needed. Additionally, power restrictions

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 912–923, 2006.

should be taken into account, since most RFID tags in use are passive. Furthermore, these systems are unable to store passwords securely because they are not tamper-resistant at all.

The remainder of the paper is organized as follows. A short review of the main problems associated with RFID systems is outlined in section 2. In section 3, we prose a minimalist lightweight mutual authentication protocol (M^2AP) for low-cost RFID tags. A security evaluation and performance analysis of this new protocol is presented in section 4. In section 5, the proposed architecture for implementing our protocol is explained in detail . Finally, the last section is devoted to some conclusions summarizing this work.

2 Risks and Threats

Although RFID systems may emerge as one of the most pervasive computing technologies in history, there are still a number of problems that need to be solved before their massive deployment. One of the fundamental issues still to be addressed is privacy. Products labeled with tags reveal sensitive information when queried by readers, and they do it indiscriminately.

A problem closely related to privacy is tracking, or violation of location privacy. This happens because the answers provided by tags are usually predictable: in fact, most of the times, tags provide the same identifier, allowing a third party to easily establish a link between a given tag and its holder or owner. Even in the case in which individual tags try not to reveal any kind of valuable information, this tracking can still be possible by using an assembly of tags (constellation), so non-trivial solutions must be applied in order to address these tracking problems.

Although the two aforementioned problems are the most important security questions that arise from RFID technology, there are some others worth mentioning: physical attacks, denial of service, counterfeiting, spoofing, eavesdropping, traffic analysis, etc.

3 Lightweight Protocol

The major challenge of providing security for low-cost RFID tags is that these devices are very limited computationally, even unable to perform the most basic cryptographic operations. Surprisingly, most of the proposed solutions are based on the use of hash functions. Since the work of Ohkubo [7] in 2003, there has been a huge number of solutions based on this idea [2,3,5]. Although this apparently constitutes a good and secure approach, engineers face the nontrivial problem of implementing cryptographic hash functions with only 250-3000 gates. In most of the proposals, no explicit algorithms are suggested and finding one is not an easy issue, since traditional hash functions (MD5, SHA-1, SHA-2) can not be used [10]. In [14], we can find a recent work on the implementation of a new hash function with a reduced number of gates, but although this proposal seems to be light enough to fit in a low-cost RFID tag, the security of this hash scheme remains an open question.

In this paper, we propose a lightweight mutual authentication protocol between RFID readers and tags. In the following, we consider that low-cost RFID tags are devices with a very small amount of memory and very constrained computationally (< 1000 gates).

3.1 Suppositions of the Model

Our protocol is based on the use of pseudonyms, concretely on *index-pseudonym* (*IDS*). The *index-pseudonym* (96-bit length) is the index of a table (a row) where all the information about a tag is stored. Each tag has an associated key which is divided in four parts of 96 bits ($K = K1 \parallel K2 \parallel K3 \parallel K4$). As the *IDS* and the key (K) need to be actualized each time the tag is read, we need in total 480 bits of rewritable memory (EEPROM or FRAM). We also need a ROM memory to store the static tag-identification number (ID) of 96 bits, which univocally identifies the tag.

For the implementation of our protocol, all the costly computing operations are done by the reader. We suppose that readers are devices with enough computing power to generate random numbers and, in general, to perform any kind of cryptographic operations. On the contrary, tags are very limited devices that only have around 1000 logical gates for security functions. Our proposal is based on the use of simple operations: $\oplus$, $\wedge$, $\vee$, and *sum mod* 2^m.

The communication must be initiated by readers due to the fact that low-cost tags are passive. We also suppose that both the backward and the forward channels can be listened by an attacker, despite their asymmetry. Finally, we consider that the communication channel between the reader and the database is secure.

3.2 The Protocol

We can divide the protocol in four main stages: tag singulation, mutual authentication, index-pseudonym updating and key updating. In this section, we outline how the protocol works, while in the next one a security evaluation and a performance analysis are presented.

1. ***Tag Singulation***
   Before starting the protocol for mutual authentication, the reader should identify the tag. The reader will send a *"hello"* message to the tag, who will answer the reader by sending its current *index-pseudonym* (*IDS*). By means of the *IDS*, the reader will be able to access the tag secret key ($K = K1 \parallel K2 \parallel K3 \parallel K4$), which is necessary to carry out the next authentication stage. Only an authorized reader can access this information.
2. ***Mutual Authentication***
   Our protocol consists on the exchange of two messages between the reader and the tag. An scheme of the protocol is illustrated in *Figure 1.*

Reader → Tag:
$IDS^{(n)}_{tag(i)} \oplus K1^{(n)}_{tag(i)} \oplus n1 \parallel (IDS^{(n)}_{tag(i)} \wedge K2^{(n)}_{tag(i)}) \vee n1 \parallel IDS^{(n)}_{tag(i)} + K3^{(n)}_{tag(i)} + n2$
Tag → Reader:
$(IDS^{(n)}_{tag(i)} \vee K4^{(n)}_{tag(i)}) \wedge n2 \parallel (IDS^{(n)}_{tag(i)} + ID_{tag(i)}) \oplus n1$

Fig. 1. M^{2}AP Protocol

- Reader Authentication
  The reader will generate two random numbers, $n1$ and $n2$. With $n1$ and the subkeys $K1$ and $K2$ associated to the tag, the reader will generate $A \parallel B$:

$$A \parallel B = IDS^{(n)}_{tag(i)} \oplus K1^{(n)}_{tag(i)} \oplus n1 \parallel (IDS^{(n)}_{tag(i)} \wedge K2^{(n)}_{tag(i)}) \vee n1 \quad (1)$$

  which is the part of the message that allows the tag authentication. With $n2$ and $K3$, the reader will generate the submessage C:

$$C = IDS^{(n)}_{tag(i)} + K3^{(n)}_{tag(i)} + n2 \quad (2)$$

  that will be used for updating the *index-pseudonym* (IDS) and the key (K). Once the three parts of the message are generated ($A \parallel B \parallel C$), they are concatenated and sent to the tag.
- Tag Authentication
  With submessages A and B, the tag will authenticate the reader. From submessage C, the tag will obtain the random number $n2$, that will allow it to update the *index-pseudonym* (IDS) and the key (K). Once these verifications are performed, the tag will generate the answer message. This message will be composed of two parts $D \parallel E$. Part D,

$$D = (IDS^{(n)}_{tag(i)} \vee K4^{(n)}_{tag(i)}) \wedge n2 \quad (3)$$

  will allow the reader to authenticate the tag. By means of part E,

$$E = (IDS^{(n)}_{tag(i)} + ID_{tag(i)}) \oplus n1 \quad (4)$$

  the tag is able to transmit its static identifier in a secure form.

3. ***Index-Pseudonym Updating***
   Once the tag and the reader have been mutually authenticated, the *index-pseudonym* must be updated as follows:

$$IDS^{(n+1)}_{tag(i)} = (IDS^{(n)}_{tag(i)} + (n2 \oplus n1)) \oplus ID_{tag(i)} \quad (5)$$

4. ***Key Updating***
   Another important security issue is key updating. After the reader and the tag have been authenticated, the key updating stage must be carried out. As

tags are very computationally contrained devices, this task can be made only by using efficient operations ($\oplus$, $\wedge$, $\vee$, and *sum mod* 2^m). These operations have to be already implemented in the tag for the normal protocol running, so its use will not imply an increase in the gate counting. Nevertheless, we can not either forget the temporary requirements of the tag which must be able to answer 100 times/sec (see section 5) at least. These speed requirements put a limitation on the number of possible operations that can be performed with each component of the key (Ki). Taking, all these considerations into account, the proposed equations for key updating are the following ones:

$$K1_{tag(i)}^{(n+1)} = K1_{tag(i)}^{(n)} \oplus n2 \oplus (K3_{tag(i)}^{(n)} + ID_{tag(i)}) \quad (6)$$

$$K2_{tag(i)}^{(n+1)} = K2_{tag(i)}^{(n)} \oplus n2 \oplus (K4_{tag(i)}^{(n)} + ID_{tag(i)} \quad (7)$$

$$K3_{tag(i)}^{(n+1)} = (K3_{tag(i)}^{(n)} \oplus n1) + (K1_{tag(i)}^{(n)} \oplus ID_{tag(i)}) \quad (8)$$

$$K4_{tag(i)}^{(n+1)} = (K4_{tag(i)}^{(n)} \oplus n1) + (K2_{tag(i)}^{(n)} \oplus ID_{tag(i)}) \quad (9)$$

If we analyze the previous equations, we will obtain that the probability of zeros and ones for every Ki is approximately 0.5 and the Hamming distance between $K_{tag(i)}^{n}$ and $K_{tag(i)}^{n+1}$ is 47.5 (on average). According to the temporary requirements, for the worst case, which is the architecture of 8 bits, we are well into the limit of 100 answers/sec, so we successfully fulfill the temporary requirements in all the cases.

4 Evaluation

4.1 Security Analysis

Once we have presented the proposed mutual-authentication protocol, we will evaluate its security, studying the same properties that Yang analyzes in [13], in order to be able to compare their characteristics.

1. ***User Data Confidentiality***
   Tag *ID* must be kept secure to guarantee user privacy. The tag sends it in the message E ($E = (IDS_{tag(i)}^{(n)} + ID_{tag(i)}) \oplus n1$) where the *ID* is added with the *index-pseudonym*, then the result xored with the nonce $n1$. This hides tag *ID* to a nearby eavesdropper equipped with an RFID reader.
2. ***Tag Anonymity***
   As tag *ID* is static, we should send it, and all other interchanged messages in seemingly random wraps (i.e. to an eavesdropper, only random numbers are sent). As we have seen, readers generate the message $A \parallel B \parallel C$. This message will serve, as well as to authenticate the reader, to transmit in a secure form the random numbers $n1$ and $n2$ to the tag. The first random number ($n1$) will be used to hide the tag *ID* and the combination $n1 \oplus n2$ will serve to update the index-pseudonym. By means of this mechanism, we are able to make almost all the computational load to fall on the side of

RFID readers, since one of our hypothesys is that very low-cost tags can not generate random numbers. Thus, tag anonymity is guaranteed, and the location privacy of a tag owner is not compromised either. There is one interesting scenario that we will explain in more detail in the following, as one could think that in this case, the tracking of a tag owner is possible. In this scenario, the attacker sends *"hello"* messages to the tag and receives as answer the *IDS* from it. Then, he stops the authentication step. A little time later he repeats the process, hoping that the *IDS* has not changed yet. We know that, if the authentication process fail the *IDS* can not be updated. The attacker can not generally track the owner tag because it is very probable that between two successive requests of the attacker, the tag is read by one or several legitimate readers, who will update the *IDS*. If an intruder wants to guarantee that the *IDS* has not changed, it needs to send more than 100 answers/sec in order to saturate the tag, so not allowing a legitimate reader to access it. In this case, this attack would be considered a DoS attack, which is an inherent problem in RFID technology as it happens in other technologies that use the radio channel to which, for the moment, there is no known solution (apart from spread spectrum).

3. ***Data Integrity***
   A part of the tag memory is rewritable, so modifications are possible. In this part of the memory, the tag stores the *index-pseudonym* (*IDS*) and the key (K) associated with itself. If an attacker does succeed in modifying this part of the memory, then the reader would not recognize the tag and should implement the updating protocol of the database.
4. ***Mutual Authentication***
   We have designed the protocol with both reader-to-tag authentication, which is achieved by message $A \parallel B \parallel C$, and tag-to-reader authentication, obtained with message $D \parallel E$.
5. ***Forward Security***
   Forward security is the property that security of messages sent today will be valid tomorrow [7]. Since key updating is fulfilled after the mutual authentication, a future security compromise on an RFID tag will not reveal any previously transmitted data.
6. ***Man-in-the-middle Attack Prevention***
   A man-in-the-middle attack is not possible because our proposal is based on a mutual authentication, in which two random numbers $n1$ and $n2$, refresh in each iteration of the protocol, are used.
7. ***Replay Attack Prevention***
   An eavesdropper could store all the messages interchanged between the reader and the tag (different protocol runs). Then, he could try to impersonate a reader, re-sending the message $A \parallel B \parallel C$ seen in any of the protocol runs. It seems that this could cause the losing of synchronization between the database and the tag, but this is not the case because after the mutual authentication, the *index-pseudonym* (*IDS*) and the key K ($K = K1 \parallel K2 \parallel K3 \parallel K4$) were updated.

Table 1. Comparison between Protocols

Protocol	HLS [11]	EHLS [11]	HBVI [5]	MAP [13]	M^2AP
User Data Confidentiality	×	△	△	◯	◯
Tag Anonymity	×	△	△	◯	◯
Data Integrity	△	△	◯	◯	△
Mutual Authentication	△	△	△	◯	◯
Forward Security	△	△	◯	◯	◯
Man-in-the-middle Attack Prevention	△	△	×	◯	◯
Replay Attack Prevention	△	△	◯	◯	◯
Forgery Resistance	×	×	×	◯	◯
Data Recovery	×	×	◯	◯	×

†† Notation: ◯ Satisfied △ Partially Satisfied × No Satisfied

8. ***Forgery Resistance***
   The information stored in the tag is sent operated ($\oplus$, $\wedge$, $\vee$, or *sum mod* 2^m) with random numbers ($n1$, $n2$). Therefore, the simple copy of information of the tag by eavesdropping is not possible.
9. ***Data Recovery***
   Intercepting or blocking messages is a DoS attack that prevents tag identification. As we do not consider that these attacks can be a serious problem for very low-cost RFID tags, our protocol does not provide data recovery. In those scenarios in which this problem is considered important, an extended version of the protocol is possible and straighforward. In this implementation, each tag will have $l + 1$ database records, the first one associated with the actual *index-pseudonym* (n) and the others associated with the potential next index-pseudonyms ($n+1, n+2, ..., n+l$). Moreover, each tag will need k additional bits of ROM memory to store the associated data-base entry like in [5]. As before, the reader will use the *IDS* to access to all the information associated with the tag. The reader will store a potential *index-pseudonym* each time the answer of the tag is blocked. Once the tag and the reader had mutually authenticated, the potencial *index-pseudonyms* could be deleted. The storage of the potencial *index-pseudonyms* will allow to easily recover from the lose or interception of messages.

Table 1 shows a comparison made by Yang [13] of the security requirements of different proposals, our proposal (M^2AP) is added in the last column.

4.2 Performance Analysis

It is important to carefully analyze the performance of the proposed scheme, to show that it can safely be implemented even in low-cost tags. As mentioned in section 3.1, we have assumed that the connection between the reader and the database is secure. Also, readers and databases are devices with non-limited computing and storing capabilities. Due to these reasons we can collapse the notion of the reader and the back-end database into a single entity ($R + B$). So,

in the performance analysis of our protocol, we will consider that reader and database form a single entity. As in the previous section, we will consider the same overheads (computation, storage and communication) used by Yang [13].

1. ***Computation Overhead***
   Low-cost RFID tags are very limited devices,with only a small amounts of memory, and very constrained computationally (only between 250 and 3000 logics gates can be devoted to security-related tasks). Additionally, one of the main drawbacks that hash-based solutions have is that the load on the server side $(R + B)$ is proportional to the number of tags. As we can see in *Table 2* this problem is also present in Yang's solution [13]. On the other hand, in our proposal, we have completely solved this problem by using an *index-pseudonym* that allows a tag to be univocally identified.
2. ***Storage Overhead***
   As Yang does, we assume that the sizes of all components are L bits, that the PRNG and the hash function are $h, h_k : \{0,1\}^* \rightarrow \{0,1\}^{\frac{1}{2}L}$ and $r \in_U \{0,1\}^L$. Our protocol is based on pseudonyms, concretely on an L-bit *index-pseudonym* (*IDS*), so each tag has to store it. For the implementation of our protocol, each tag should have an associated key of length $4L$, which is used for mutual authentication between the reader and the tag. Moreover, the tag has to store an unique identification number of length L. The reader has to store the same information, so it requires a memory of $6L$ bits.
3. ***Communication Overhead***
   The proposed protocol accomplishes mutual authentication between tag (T) and reader $(R + B)$, requiring only four rounds. As we can see in *Table 2*, other protocols require, at least, one or two additional messages to be exchanged. Taking into account that low cost tags are passive, and that the communication can only be initiated by a reader, four rounds may be considered as a reasonable number of rounds for mutual authentication in RFID environments.

5 Implementation

In this section, we will explain in detail the proposed architecture for implementing our protocol: the reader sends the message $A \parallel B \parallel C$, which is received by the tag. The tag will check the authenticity of this message for authenticating the reader. Once the tag has authenticated the reader, it will send the message $D \parallel E$ to authenticate itself.

One of the first and more relevant subjects to consider is whether to choose a serial or a parallel implementation. Serial means processing the bitstream bit by bit, and parallel means processing the whole message, for example, $A \parallel B \parallel C$ at the same time. It is a common assumption that a minimum of 100 tags should be authenticated per second. As in [4], due to the low-power restrictions of RFID tags, the clock frequency must be set to 100 KHz. So, a tag may use up to 1000

Table 2. Computational Loads and Required Memory

Protocol	Entity	HLS [11]	EHLS [11]	HBVI [5]	MAP [13]	M^2AP
No. of Hash Operations	T	1	2	3	2	¬
	B	¬	Nt	3	$2Nt$	¬
No. of Keyed Hash Operations	R	¬	¬	¬	1	¬
	B	¬	¬	¬	1	¬
No. of PRNG Operations	T	¬	1	¬	¬	¬
	R	¬	¬	¬	1	¬
	B	¬	¬	1	¬	¬
No. of Basic Operations[1]	T	¬	¬	¬	4	19
	R+B	¬	¬	¬	$2(Nt+1)$	21
Number of Authentication Steps		6	5	5	5	4
No. of Encryptions	B	¬	¬	¬	1	¬
No. of Decryptions	R	¬	¬	¬	1	¬
Required Memory Size	T	$1\frac{1}{2}L$	$1L$	$3L$	$2\frac{1}{2}L$	$6L$
	R+B	$2\frac{1}{2}L$	$1\frac{1}{2}L$	$9L$	$9\frac{9}{2}L$	$6L$

†† Notation:
¬ : Not Required Nt: Number of Tags L: Size of Required Memory
[1] Basic Operations: $\oplus$, $\wedge$, $\vee$, and $+$

clock cycles to answer a reader. Due to these characteristics, it is not necessary to resort to a parallel implementation. As we can see in *Figure 2*, we have decided not to process at the same time all the message, but to do it in blocks of m bits.

The proposed architecture is independent of the word length used. We have analyzed the features of five different word lengths ($m = 8, 16, 32, 64, 96$). In *Figure 2*, we can see a scheme of the proposed architecture. On the left of the figure, we have the memory, which is filled with the *index-pseudonym* (IDS), the key K ($K1 \parallel K2 \parallel K3 \parallel K4$) and the ID. The access to the memory is controlled by a sequencier. Due to the fact that the messages are build up of three components, we will need a m-bit register to store intermediate results. In the middle of the figure we have the Arithmetic Logic Unit (ALU). This unit will made the following operations of size m bits (word length): $\oplus$, $\wedge$, $\vee$, and *sum mod* 2^m. The ALU has two inputs, one of these values is stored in the memory, and another which is selected (C_1) between the bitstream and the value stored in the register. The control signal C_2 will select the operation that will be used in the ALU.

In the worst case of our protocol ($m = 8$), we need 1000 clock cycles for implementing the mutual authentication. So, if we consider that the clock frequency is set to 100KHz, this means that the tag answers in 10 millisecond. A tag can authenticate 100 times per second, so the temporary requirements are fulfilled in all the cases.

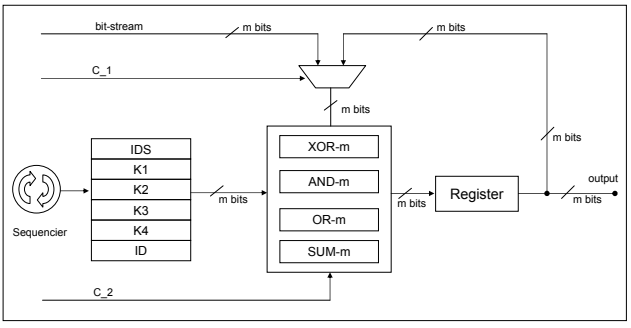


Fig. 2. Logic Scheme

Another important aspect to study is the number of logical gates necessary for implementing our protocol. The functions $\oplus$, $\wedge$, and $\vee$ will be implemented with the same number of logic gates like the word length (m). For the implementation of the adder circuit with carry, a parallel architecture is proposed ($S = A \oplus [B \oplus C_{ENT}]$; $C_{SAL} = BC_{ENT} + AC_{ENT} + AB$). Six logic gates are needed for each bit that is added in parallel. Additionally, a 20% of logic gates are considered for control functions. The following table summarizes the features of the proposed architecture:

Table 3. Features

Word Length (m)		**8-bit**	**16-bit**	**32-bit**	**64-bit**	**96-bit**
Number of Gates	ALU	72	144	288	576	864
	Control	14	29	58	115	173
	Total	86	173	346	691	1037
Number of Clock Cycles		960	480	240	120	80
Answer/sc		104	208	416	833	1250

As we can see in the previous table, in the best case ($m = 8$), our protocol needs around 100 gates. In *Table 4*, we show also the number of logical gates needed for implementing various hash functions and AES encryption. A traditional hash function such as MD5 or SHA needs more than 16K gates, which is by far higher than the capabilities of low-cost RFID tags [10]. An efficient implementation of AES encryption has been recently published [6], which does not need many logical gates (only 3595), but it needs a coprocessor. Unfortunately, this significantly increases the price of RFID tags, and is not attainable in low-cost RFIDs. Additionally, there is also a proposal of an implementation of a new universal hash function for ultra low-power cryptographic hardware applications. Although this solution only needs around 1.7K gates, a deeper security analysis of the hash function is needed, and has not been accomplished yet.

Table 4. Core Comparison

Solutions	Implementation	Gate Counting
Hash	Universal Hash Yksel [14]	1.7K Gates
	MD5 Helion [10]	16K Gates
	Fast SHA-1 Helion [10]	20K Gates
	Fast SHA-256 Helion [10]	23K Gates
AES Unit[2]	JungFL [6] E/D/RK[2]	3089 Gates +/+/+
	Feldhofer [4] E/D/RK[1]	3595 Gates +/+/+
	Amphion CS5265 [1] E/D/RK[1]	25000 Gates +/+/+

†† Notation:
[1]AES with Coprocessor.
[2]E= Encryption, D=Decryption, RK= Round Key Generation.

Finally, although we have not implemented the circuit physically, due to the known fact that power consumption and circuit area are proportional to the number of logical gates, it seems that our implementation will be suitable even for very low-cost RFID tags.

6 Conclusions

RFIDs tags are devices with very limited computational capabilities, which only have between 250 and 3000 logics gates that can be devoted to security-related tasks. Cryptographic primitives such as PRNGs, block ciphers and hash functions lie well beyond the computational capabilities of very low-cost RFID tags, but until now, most of the security solutions for RFID are based on them.

A new approach must be taken to tackle the problem, at least for low-cost RFID tags. For this reason, we propose a very lightweight mutual authentication protocol that could be implemented in low-cost tags (<1000 logic gates). In order to be able to use our proposal, tags should be fitted with a small portion of rewritable memory (EEPROM or FRAM) and another read-only memory (ROM). The assumption of having access to rewritable memory is also implicitly made in all the existing solutions based on hash functions.

In spite of being very limited in resources, the main security aspects of RFID systems (privacy, tracking, etc.) have been consider in this article, and solved efficiently (less than 1000 gates are needed, even in the worst implementation, in our case $m = 96$ bits). As shown in *Table 2*, our protocol displays superior benefits to many of the solutions based on hash functions. So, not only we have

been able to avoid the privacy and tracking problems, but also many other attacks such as man-in-the-middle attack, forwarding replay, etc.

Finally, another paramount characteristic of our scheme is its efficiency: tag identification by a valid reader does not require exhaustive search in the back-end database. Furthermore, only two messages need to be exchanged in the singulation stage, and another two in the mutual-authentication stage.

References

1. Amphion: CS5265/75 AES Simplex encryption/decryption. *http://www. amphion.com*, 2005.
2. E.Y. Choi, S.M. Lee, and D.H. Lee. Efficient RFID authentication protocol for ubiquitous computing environment. In *Proc. of SECUBIQ'05*, LNCS, 2005.
3. T. Dimitriou. A lightweight RFID protocol to protect against traceability and cloning attacks. In *Proc. of SECURECOMM'05*, 2005.
4. M. Feldhofer, S. Dominikus, and J. Wolkerstorfer. Strong authentication for RFID systems using the AES algorithm. In *Proc. of CHES'04*, volume 3156 of *LNCS*, pages 357–370, 2004.
5. D. Henrici and P. Müller. Hash-based enhancement of location privacy for radio-frequency identification devices using varying identifiers. In *Proc. of PERSEC'04*, pages 149–153. IEEE Computer Society, 2004.
6. M. Jung, H. Fiedler, and R. Lerch. 8-bit microcontroller system with area efficient AES coprocessor for transponder applications. Ecrypt Workshop on RFID and Lightweight Crypto, 2005.
7. M. Ohkubo, K. Suzuki, and S. Kinoshita. Cryptographic approach to "privacy-friendly" tags. In *RFID Privacy Workshop*, 2003.
8. C.M. Roberts. Radio frequency identification (RFID). *Computers and Security*, 25(1):18–26, 2006.
9. W. Sean and L. Thomas. Automatic identification and data collection technologies in the transportation industry: BarCode and RFID. Technical report, 2001.
10. Datasheet Helion Technology. High Performance MD5. Fast SHA-1. Fast SHA-256. hash core for ASIC, 2005.
11. S.A. Weis, S.E. Sarma, R.L. Rivest, and D.W. Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In *Security in Pervasive Comp.*, volume 2802 of *LNCS*, pages 201–212, 2004.
12. Kirk H.M. Wong, Patrick C.L. Hui, and Allan C.K. Chan. Cryptography and authentication on RFIDnext term passive tags for apparel products. *Computers in Industry*, 57(4):342–349, 2006.
13. J. Yang, J. Park, H. Lee, K. Ren, and K. Kim. Mutual authentication protocol for low-cost RFID. Ecrypt Workshop on RFID and Lightweight Crypto, 2005.
14. K. Yksel, J.P. Kaps, and B. Sunar. Universal hash functions for emerging ultra-low-power networks. In *Proc. of CNDS'04*, 2004.

Context-Enhanced Authentication for Infrastructureless Network Environments

Ryan Wishart[1,⋆], Jadwiga Indulska[1,2], Marius Portmann[1,2], and Peter Sutton[1]

[1] School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, Australia
{wishart, jaga, marius, p.sutton}@itee.uq.edu.au
[2] National ICT Australia ⋆⋆

Abstract. Infrastructureless networks are becoming more popular with the increased prevalence of wireless networking technology. A significant challenge faced by these infrastructureless networks is that of providing security. In this paper we examine the issue of authentication, a fundamental component of most security approaches, and show how it can be performed despite an absence of trusted infrastructure and limited or no existing trust relationship between network nodes. Our approach enables nodes to authenticate using a combination of contextual information, harvested from the environment, and traditional authentication factors (such as public key cryptography). Underlying our solution is a generic threshold signature scheme that enables distributed generation of digital certificates.

1 Introduction

Infrastructureless network environments, including both Mobile Ad hoc Networks (MANET) and many pervasive computing environments, have enjoyed increased attention of late. While the lack of fixed infrastructure in these networks makes them quick to deploy, it also presents a problem from a security perspective.

Within this paper we focus on one particular aspect of security, authentication, and provide a solution that overcomes many of the problems of the infrastructureless environment that have hampered previous approaches. These previous authentication approaches have typically assumed that the network is a region under the control of a centralised authority. This centralised authority shares secret knowledge with all of the nodes that are permitted entry to the network, and can use that knowledge to authenticate the nodes. This secret knowledge may

⋆ The work reported in this paper has been funded in part by the Co-operative Research Centre for Enterprise Distributed Systems Technology (DSTC) through the Australian Federal Government's CRC Programme (Department of Industry, Science & Resources).

⋆⋆ National ICT Australia is funded by the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 924–935, 2006.

take the form of a secret password, as in Kerberos [1], knowledge of specialised hardware possessed by the node (such as a smart card) or defining biometric characteristics (when authenticating people).

Problems arise when these traditional authentication approaches are applied to infrastructureless networks as (1) there is a lack of trusted infrastructure, (2) the network size is often highly dynamic, and (3) the network may consist of nodes that have never encountered one another before and thus have no knowledge of one another to base the authentication procedure on.

In this paper we present a multi-factor, context-enhanced authentication mechanism for infrastructureless network environments. The mechanism consists of a distributed collection of mutually authenticated nodes that can check particular authentication factors and are trusted to do so. These factors can be either traditional factors such as digital certificates, or contextual factors (such as location). By supporting contextual factors, our mechanism can perform authentication of new nodes in situations where the new node and the authentication mechanisms nodes have no shared knowledge. To generate certificates in the absence of a central Certification Authority, our approach uses a threshold signature generation scheme.

The remainder of this paper is organised as follows. Section 2 provides background information on the threshold signature generation technique underlying our approach. Section 3 surveys related work in the field of authentication with particular attention given to mechanisms intended for Mobile Ad hoc Networking (MANET) and pervasive computing environments. The requirements for an underlying generic threshold signature scheme (TSS) are outlined in Section 4. In Section 5 the types of context information that can be included in the authentication process are discussed. The operation of our authentication mechanism is discussed in Section 6. Extensions to the authentication mechanism to permit grouping of nodes based on the factors they used during authentication are then covered in Section 7. An evaluation of our approach is discussed in Section 8 before concluding remarks are given in Section 9.

2 Threshold Signature Generation

Threshold signature generation techniques permit a group of entities (which we shall refer to as secret share holders) to distributedly generate a digital signature on a certificate without the need for a central Certification Authority. To achieve this, the private key to be used for the signature generation is split into a number of "secret shares" by a trusted dealer. One secret share is then given to each secret share holder [2].

To generate a digital certificate it is necessary for unsigned copies of the certificate to be distributed to a threshold number of the secret share holders. Secret share holders that receive a certificate then apply their secret share to their copy of the certificate. This generates a partial signature on the copy of the certificate. These partially signed copies of the certificate can then be combined together to generate a valid digital certificate. The task of combining the partially

signed certificates into a valid digital certificate can be performed either by a dedicated "combiner" node, or by the node requesting the certificate (as in [3]).

Several developments can be applied to threshold signature schemes to increase their usefulness. These developments include eliminating the need for a trusted dealer through Joint Secret Sharing [4], and a means to check if partial signatures are valid (referred to as Verifiable Secret Sharing [5]). A third development, Proactive Secret Sharing [6], enables distributed refreshing of secret shares. Refreshed secret shares are incompatible with previously used secret shares.

3 Related Work

Authentication mechanisms for networks with fixed infrastructure are not easily migrated to infrastructureless network environments as they often depend on the availability of trusted infrastructure, as is the case with the widely used Kerberos [1] protocol.

Pirzada and McDonald [7] developed a distributed implementation of Kerberos for MANET environments that replicates the Kerberos Authentication Server across a group of highly trusted nodes. In an infrastructureless environment, where all nodes are potentially strangers, locating highly trusted nodes is likely to be problematic.

A different approach is taken by Zhou and Haas [8] who suggest using a distributed Certification Authority (based on Shamir's secret sharing [2]). However, they consider only traditional authentication mechanisms, and their approach cannot cope with nodes previously unknown to the system. Distributed signature generation schemes are used by Saxena et al. [9] and Luo et al. [10] for admission control to MANET networks. Neither of these two papers discuss their supporting authentication in detail. However, they do show that a signature generation scheme can be used in the MANET environment for security purposes.

Glynos et al. [11] present a mechanism for authentication in MANET environments that combines traditional authentication with a limited set of context information used to identify MANET nodes. The context information is compared against profiled information to establish MANET node identity.

A similar approach is taken by Covington [12], although Covington does not support traditional authentication mechanisms. A major failing with both Glynos et al. and Covington's solutions is that they cannot easily be used in an infrastructureless network as they require the creation of context profiles for each node that may join the network.

4 Requirements for the Underlying Threshold Signature Scheme

Our approach to context-enhanced authentication in infrastructureless environments has been designed to operate on top of a generic threshold signature scheme (TSS). In this section we discuss the requirements our design places on this underlying TSS. The specific requirements are that the TSS should:

1. be cryptographically secure
2. handle fluctuating network size
3. support Joint Secret Sharing so that a trusted dealer is not required
4. support Verifiable Secret Sharing techniques to identify invalid partial signatures
5. support Pro-active Secret Sharing techniques which can be used to refresh the secret shares
6. have a low computation cost associated with generating new secret shares, and refreshing existing secret shares

The primary requirement for the underlying TSS is that it be cryptographically secure within the time period that the authentication mechanism is likely to be deployed.

The second of the requirements is that the TSS algorithm support a fluctuating network size. This is particularly important as the infrastructureless network environment is likely to vary in size considerably during its lifetime. A TSS algorithm that requires all secret shares to be recomputed every time the size of the network changes would be impractical.

The third of these requirements states that the TSS algorithm should be compatible with the Joint Secret Sharing techniques that permit distributed generation of secret shares. This is required as in an infrastructureless network environment the existence of trusted nodes cannot be assumed. Generation of the authentication mechanism's public and private keys as well as the secret shares must be shared amongst the nodes of the authentication mechanism.

The fourth requirement is for the TSS to support Verifiable Secret Sharing so that partial signatures on certificates can be validated. This is required to prevent malicious nodes submitting invalid partial signatures in an attempt to thwart the signature generation scheme.

Fifth in the list of requirements is that the TSS algorithm should use Proactive Secret Sharing (PSS) to periodically refresh the secret shares of valid secret share holders. This can be used to prevent a malicious party from slowly acquiring a threshold number of the secret shares.

The algorithm will likely be run on computationally limited information devices like handheld computers or mobile telephones. With this in mind, the final requirement is that the TSS algorithm be computationally efficient when initialising, generating new secret shares or refreshing of secret shares. An example of an algorithm that fulfils these requirements is that of Luo et al. [10].

5 Context Information and Authentication

In this section we provide a characterisation of security-relevant context information that can be included in the authentication process. This context must:

- be verifiable by a reliable, preferably authenticated, node
- describe some contextual attribute of the node, (e.g., location, activity)
- be sufficiently accurate, and of sufficient granularity, to have a bearing on the authentication decision

The types of context information that meet these requirements will depend on the circumstances in which the authentication mechanism is deployed. Consider the example of an emergency response team that arrives at a disaster site and seeks to setup a MANET for communication and data exchange. If the devices that will form the network are all from the same response team it is likely that they will have traditional means of authentication, such as public key certificates. To strengthen the authentication provided by these traditional authentication mechanisms, and to cope with situations where the public key used by one device cannot be verified by another device (such as when non emergency workers, who bring their own devices, arrive on site to advise the emergency response team), context authentication can be used.

Each context authentication factor supported by the authentication mechanism will be agreed upon when the authentication mechanism is started. Each of these factors will also require a value, quoted in partial signatures. The same applies to traditional authentication factors. This value is based on the increase in confidence, or certainty, the authentication mechanism has in the identity of the factor user. Factors that deliver a large increase in certainty, as might be the case with public key authentication, have a higher value than factors that deliver a marginal increase in identity certainty for a new node (e.g., proximity-based authentication).

Table 1 contains context factors that might be used in the emergency response scenario. Example values are also provided for the context factors. An assumption is made that the signature threshold for the TSS is 15 partial signatures.

Table 1. Example context authentication factors used by devices in an emergency scenario

Context Types	Value in Partial Signatures
device proximity	5
verifiable location history	10
verifiable interaction history	10

6 Context-Enhanced Authentication for Infrastructureless Environments

In this section we present our multi-factor, context-enhanced authentication mechanism for infrastructureless environments. Our mechanism requires a new node to locate authentication mechanism nodes and use factors with those authentication mechanism nodes to authenticate. For each successful use of a factor, the new node receives a number of partial signatures towards its authentication certificate. The number of partial signatures received depends on the value of the factor.

We assume that the infrastructureless network is comprised of mobile devices under the control of human users. These devices have long-range wireless communication abilities, and possibly support range-limited communication such as infra-red or physical contact-based communication. These devices are also able to obtain trusted context information either by sensing it themselves or from sources that they trust. Each of the devices is computationally capable of performing public key cryptography, and is able to establish a secure communication channel to any of the other devices in the network.

The operation of our authentication mechanism progresses in two distinct phases: the initialisation phase where the authentication mechanism is set-up, and the post-initialisation phase which begins immediately after the initialisation and continues until the infrastructureless network ceases to exist. For the purposes of this paper we concentrate on the creation and operation of the authentication mechanism only. Revocation of certificates is considered outside the scope of this work.

6.1 Initialisation Phase

The initialisation phase sees a group of nodes come together and agree to form the authentication mechanism. As it is likely these nodes are mutual strangers we make the assumption that an external channel is used to bootstrap a trust relationship between this initial group of nodes. This bootstrapping process might take the form of node users sharing secret information verbally, or possibly through physical contact of the nodes. Once a trust relationship has been established between the nodes, the operational parameters of the authentication mechanism can be decided. These parameters are recorded in the authentication mechanism policy, and include:

- the authentication factors to be supported
- the threshold signature scheme to use
- the signature threshold value, t, for the TSS algorithm
- the number of secret shares to allocate to each node in the authentication mechanism, (referred to as k)
- the value of the authentication factors

Once the operational parameters for the authentication mechanism have been decided, the initialisation phase proceeds as follows:

1. the nodes distributively generate a public and a private key for the authentication mechanism
2. each node in the initial group receives k secret shares in the authentication mechanism private key

To generate an authentication certificate, t duplicates of an unsigned version of the certificate must be partially signed by other authentication mechanism nodes. This proceeds as follows:

1. node M generates t unsigned duplicates of an authentication certificate for itself
2. M then constructs an USE($factor$) message and sends it to the other authentication mechanism nodes
3. node L, capable of checking authentication factor $factor$, responds with WILL_CHECK($factor$)
4. M sends L copies of its unsigned certificates, with the number sent equal to the value of $factor$
5. L then checks that M can use $factor$
6. if the check succeeds, L partially signs the unsigned certificates provided by M using a different one of its secret shares on each of the unsigned certificates. L then returns the now partially signed certificates to M.
7. M repeats steps 2 to 6 using different factors until it has t partially signed certificates. M can then combine the partially signed certificates to produce a valid authentication certificate for itself.

In step 3 it is possible that multiple nodes reply to M's request. The choice of which respondent to use the factor with will depend on the type of $factor$. For example if $factor$ is distance dependent, then M will likely choose the closest respondent. If there are no respondents in step 3, M must repeat step 2 with a different factor. If M cannot do this, then M cannot be authenticated and the protocol terminates.

Assuming the completion of the initialisation phase, each node in the initial group is now part of the authentication mechanism and in possession of:

- the authentication mechanism policy
- the authentication mechanism public key
- k secret shares in the private key
- an authentication certificate for itself

As M is required to interact with different authentication mechanism nodes as part of the authentication process it may be tempted to engage in "double spending" whereby it uses the same factor multiple times. Our approach to deal with this is discussed in Section 7.

6.2 Post-initialisation Phase

After the initialisation phase is completed all future authentications of new nodes proceed according to the steps below:

1. node M locates an authentication mechanism node and requests the authentication mechanism policy
2. M then generates t unsigned duplicates of an authentication certificate for itself
3. M constructs an USE($factor$) message and sends it to the authentication mechanism nodes, specifying the factor it wants to use as the term $factor$
4. one of the authentication mechanism nodes, L, who can check authentication factor $factor$, responds with WILL_CHECK($factor$)

5. M then sends L a number of duplicates of its unsigned authentication certificate, with the number sent equal to the value of $factor$
6. L then checks that M can use $factor$
7. if the check succeeds, L partially signs the unsigned certificates provided by M using a different one of its secret shares on each of the certificates M provided, before returning the partially signed certificates to M
8. M must repeat steps 3 to 7 using different factors to acquire t partially signed certificates. Once this is done, M can combine the partially signed certificates to produce a valid authentication certificate for itself

M may receive duplicates of the WILL_CHECK message in step 4. These duplicates are handled as per the method described in the initialisation phase.

If M reaches a stage where it has no more factors it can use, and has not acquired the threshold number of partial signatures, then the protocol terminates and M is not authenticated.

Provided that M is able to complete step 8, it will be in possession of the following:

- the authentication mechanism public key
- the authentication mechanism policy
- a signed authentication certificate for itself

If M is permitted to become one of the authentication mechanism nodes, it requests an allocation of secret shares according to the process defined below:

1. M sends a REQUEST_JOIN message to the nearest node in the authentication mechanism along with its authentication certificate
2. the authentication mechanism nodes verify M's authentication certificate using the authentication mechanism public key
3. provided M is permitted to join the authentication mechanism, k secret shares are generated for M and securely communicated to it

7 An Extended Version of the Authentication Scheme to Support Authentication Levels

Our approach enables new nodes to authenticate using a combination of different factors. These could be a small number of high-value factors, or a large number of low-value factors. To differentiate between these two cases, we extend the authentication mechanism to support *Authentication Levels*. To achieve a higher Authentication Level, nodes must use higher-valued factors. The number of these Authentication Levels and the associated entry requirements will be implementation specific, and thus decided during the initialisation phase of the authentication mechanism.

At this stage we have identified two possible methods for implementing Authentication Levels. In the first method, each Authentication Level is represented

by a separate TSS, with all the schemes run concurrently. In this arrangement each of the TSS could use different thresholds. The lower Authentication Level would require fewer partial signatures to achieve than do subsequent higher Authentication Levels.

The second method of implementing the Authentication Levels uses only a single TSS and thus has lower overhead than the first approach. As there is only one TSS there can be only one signature threshold and so to authenticate all nodes must acquire the same number of partial signatures. The assignment of the Authentication Levels needs to be done after the authentication certificate is generated so that the authentication factors used can be checked. This checking requires a record of the factors used to be attached to the authentication certificate. This record can also be used to detect illegal "double spending" where new nodes use the same authentication factor multiple times to gain the threshold number of partial signatures.

When examined, the computational cost of the second method is significantly less than that of the multi-TSS implementation where the costs increase linearly with the number of threshold signature schemes operated. The two methods offer a trade-off between increased computation in the case of the first method, and increased communication with the added difficulty of establishing lists of credentials, in the case of the second method. The decision as to which of the methods to use should be made during the initialisation phase and need to take into account the computational capacity of the network.

8 Evaluation

In this section we present the results of two sets of simulations we performed to examine the operation of our authentication mechanism. The simulations assumed a worst case scenario where factors, and the abilities to check factors, were spread randomly throughout the infrastructureless network. The authentication rate of new nodes that applied to join the network was measured when altering: the number of neighbouring authentication mechanism nodes, the chance of particular factors occurring in the network, and the value of factors supported by the authentication mechanism nodes.

In both sets of simulations each new node could have a maximum of 10 authentication factors (no distinction was made between context and traditional authentication factors). Each factor was assigned to a new node with probability p. Authentication mechanism nodes could check a maximum of 10 different kinds of authentication factors. The probability of an authentication mechanism node being able to check a particular factor was also defined as p.

The simulations were conducted under the assumption that the authentication of a new node failed if the new node was unable to acquire the threshold number of partial signatures. This occurred if the new node did not have enough factors to authenticate, or when the new node could not locate authentication mechanism nodes capable of checking its factors.

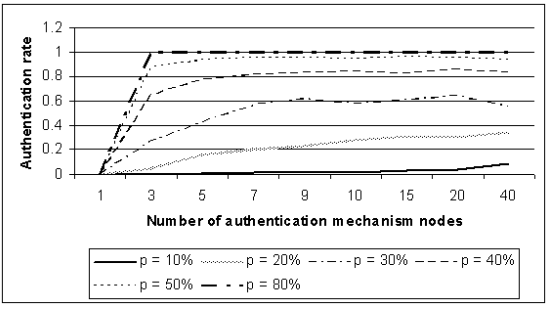


Fig. 1. Graph of authentication rate for varying numbers of authentication mechanism nodes

8.1 Simulation Set One

In the first set of simulations the effect on authentication rate of altering the number of authenticatication mechanism nodes was examined. Authentication mechanism node populations of 1, 3, 5, 7, 9, 10, 15, 20 and 40 were used. The chance of a new node having any one of the 10 supported factors was calculated for p = 10% , 20%, 30%, 40%, 50% and 80%. All factors were given a value of 5, and the signature threshold was set at 15.

The results, plotted in Figure 1, suggest that high authentication rates can be achieved, even with small numbers of authentication mechanism nodes. This is provided that the authentication factors support by the authentication mechanism nodes occur frequently in the network (this corresponds to p being greater than 40% in our simulations).

8.2 Simulation Set 2

This set of simulations examined the relationship between the authentication rate and authentication factor value. Factor values of 1, 2, 4, 5, 6, 7, 8 and 10 were used during the testing. In addition, different simulations were performed with $n = 5$, 10 and 15, where n refers to the number of authentication mechanism nodes the new node could contact. The value of p was held constant at 50% (i.e. $p = 50\%$) over all the tests. The signature generation threshold was set at 20.

From Figure 2 it can be seen that, for the particular parameters used in this simulation, increasing the number of authentication mechanism nodes beyond 5 had little effect on authentication rate. Far more important were the value of the factors used by new nodes. Extremely low factor values (less than 3) prohibited the authentication of new nodes, while high values (approaching 10) resulted in very high authentication rates. This can be explained as follows. For a factor value of 1, authentication was impossible. For a factor value of 2, the new nodes needed to have (and use) all 10 possible factors to reach the signature threshold of 20. Factor values of 7 and above required at most 3 factors, making authentication much more likely. As can be seen on Figure 2, the general increase in authentication rate was interrupted when the factor value equalled 6. This was

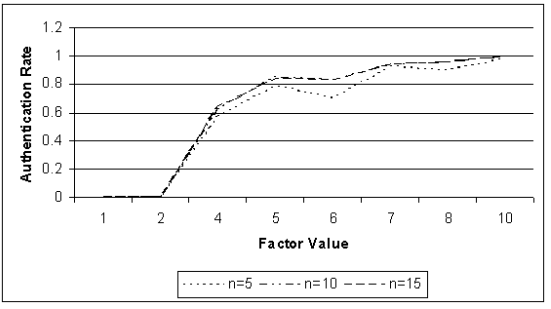


Fig. 2. Authentication rate of new nodes when varying factor value

because when the factor value was set to 5, each new node had to use 4 factors to gain the 20 partial signatures needed to authenticate. These 20 partial signatures had to come from at least 2 different authentication mechanism nodes (each authentication mechanism node had 10 secret shares). With a factor value of 6, each new node still had to use 4 factors to gain the required minimum 20 partial signatures. However, the partial signatures had to come from 4 authentication mechanism nodes (as each authentication mechanism node only had 10 secret shares, it could only check and partially sign for 1 factor). This had the effect of halving the population of authentication mechanism nodes available to the new node. As would be expected in this case, the anomaly was strongest for the plot of $n = 5$, and weakest for the plot of $n = 15$. The authentication rate recovered again for factor values of 7, as only three factors had to be used by a new node to obtain the threshold 20 partial signatures.

9 Conclusions

In this paper we presented a multi-factor, context-enhanced authentication mechanism capable of operating effectively in an infrastructureless network environment. The main contributions of the paper are that (1) it presented an authentication mechanism that makes use of both traditional and contextual information factors, (2) provides a characterisation of security-relevant contextual information, (3) describes the operation of the authentication mechanism, and (4) provides simulations to determine the effect of varying operational parameters for the authentication mechanism. These simulations were conducted for a worst case scenario where high node mobility resulted in the presence of particular factors, and the ability to check them, being randomly distributed throughout the infrastructureless network.

Based on the results of the simulations, increasing the value of factors was found to increase the authentication rate of new nodes significantly. Most importantly for infrastructureless network environments, our simulations suggest a small number of authentication mechanism nodes can authenticate new nodes with a very high success rate, provided that the authentication factors supported

by the authentication mechanism are carefully chosen to have a high probability of occurring in nodes within the infrastructureless network.

References

1. Neuman, B., Ts'o, T.: Kerberos: An Authentication Service for Computer Networks. IEEE Communications **32**(9) (1994) 33–38
2. Shamir, A.: How to Share a Secret. Communications of the ACM **22**(11) (1989) 612–613
3. Luo, H., Kong, J., Zerfos, P., Lu, S., Zhang, L.: Self-securing Ad Hoc Wireless Networks. In: Proceedings of the Seventh International Symposium on Computers and Communications, ISCC 2002. (2002) 567–574
4. Ingemarsson, I., Simmons, G.: A Protocol to Set Up Shared Secret Schemes Without the Assitance of a Mutually Trusted Third Party. In: Advances in Cryptology - EUROCRYPT'91. Lecture Notes in Computer Science, Springer-Verlag (1991) 266–282
5. Feldman, P.: A Practical Scheme for Non-Interactive Verifiable Secret Sharing. In: Proceedings of the 28th Annual Symposium on the Foundations of Computer Science, IEEE (1987) 427–437
6. Hertzberg, A., Jarecki, S., Krawczyk, H., Yung, M.: Proactive secret sharing or: How to cope with perpetual leakage. In: Proceedings of CRYPTO'1995. (1995) 339–352
7. Pirzada, A., McDonald, C.: Kerberos Assisted Authentication in Mobile Ad Hoc Networks. In: 27th Australiasian Computer Science Conference. (2004)
8. Zhou, L., Haas, Z.: Securing ad hoc networks. IEEE Networks **13**(6) (1999) 24–30
9. Saxena, N., Tsudik, G., Yi, J.: Efficient Node Admission for Short-lived Mobile Ad Hoc Networks. In: IEEE Conference on Networking Protocols (ICNP). (2005)
10. Luo, H., Kong, J., Zerfos, P., Lu, S., Zhang, L.: URSA: ubiquitous and robust access control for mobile ad hoc networks. IEEE/ACM Transactions on Networking **12**(6) (2004) 1049–1063
11. Glynos, D., Kotzanikolaou, P., Douligeris, C.: Preventing Impersonation Attacks in MANET with Multi-Factor Authentication. In: Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'05). (2005) 59–64
12. Covington, M.: A Flexible Security Architecture for Pervasive Computing Environments. PhD thesis, College of Computing, Georgia Institute of Technology (2004)

Location Privacy in Mobile Computing Environments

John P. Baugh and Jinhua Guo

Department of Computer and Information Science
University of Michigan – Dearborn
4901 Evergreen Rd, Dearborn, Michigan 48128
{jpbaugh, jinhua}@umich.edu

Abstract. In general, privacy can be viewed as the right to be left alone when desired (solitude), the right to remain anonymous (anonymity), and the right to confidentiality (secrecy of information). More specifically, location privacy is "the ability to prevent other parties from learning one's current or past locations". In this paper, we focus on two primitives that make up location privacy: *identity information* and *location information.* Identity information has to do with the static attributes and characteristics that uniquely identify a person. Information about an individual's identity can also be inferred based upon their location at various times (in other words, their *activities* can give away identity information). The other type of information upon which we will focus, *location information*, deals specifically with the whereabouts of an individual or group. We will also describe *location-aware applications and services* and their relationship with location privacy.

1 Introduction

Location-based services answer three questions: Where am I? What is around me? How do I get there? They determine the location of the user by using one of several technologies for determining position, and then use the location and other information to provide personalized applications and services.

One notable example of a system in which location is required to provide a service is OnStar [1], a commercially deployed location-based service. OnStar allows an OnStar assistant to locate an individual in need of help. It does this utilizing both GPS and cellular technology. In the event of an accident, the OnStar module located in the vehicle will transmit information to the OnStar Call Center, where an assistant may provide the appropriate help.

Location-aware applications can be incredibly helpful and convenient. However, with the divulgence of location information, serious privacy issues arise. As we have seen, location-aware applications allow for the provision of a multitude of services (such as OnStar, E911, etc.) But if misused, they can allow for an individual to be tracked throughout his/her day. Since this misuse could lead to blackmail, torture, or even death (in covert situations), it is crucial that only authorized individuals and services use location information in authorized ways. Not surprisingly, we refer to the privacy of an individual's whereabouts as *location privacy.*

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 936 – 945, 2006.

In mobile environments, security has taken the spotlight as a primary concern, leaving location privacy largely ignored. A fast growing topic within the context of mobile environments is that of vehicle safety and communication (see [2]). One application of vehicle communication is that of EZ-Pass [3]. Users of EZ-Pass have a prepaid account from which funds are taken when a special tag is scanned whenever the user passes through a special EZ-Pass lane at toll booths.

However, in addition to the amazing benefits and conveniences offered by location-aware applications, there are also major concerns that arise from them. Although services such as EZ-Pass provide convenience, they could be used by law enforcement to issue automatic speeding tickets, or by malicious entities to track and stalk individuals.

What we ultimately seek to protect is location privacy, which is best described in terms of its two major characteristics: *protection of location information,* and *protection of identity information.* Location information is the relevant data about an individual's current and previous locations. This may include coordinates, landmarks, times, or other information that would help to locate an individual. Identity is the unique information about an individual that would allow that individual to be recognized as a separate entity from a set of individuals.

This paper seeks to enhance understanding of current issues and goals of location privacy by means of reviewing the current literature and offering insight into these issues, and to convince the reader of the importance of location privacy.

The remainder of this paper is organized as follows. In Section 2, we describe location-based services and technologies, and location information (how it is obtained, used, etc.) We present the protection of identity information in Section 3. Section 4 describes laws and policies that affect the privacy of individuals. In Section 5, we discuss some of the many open issues and future directions for research. We conclude this paper in Section 6.

2 Location Information and Location-Based Technologies

As mentioned earlier, location information is the relevant data about an individual's location. In this section, we discuss various technologies that allow the obtainment of location information, including Active Badge [4], GPS [5], Active Bat [6], the Cricket Location System [7, 8, 9], and Cellular Phones [10].

2.1 Active Badges and Active Bats

Active Badges are devices that transmit their unique, global identifiers, which are then picked up by a network of infrared sensors (receivers) in the environment. They work by actively transmitting their location information, thus making it very easy to locate an individual or a piece of equipment equipped an Active Badge. They are generally used in a closed, office environment to allow individuals to obtain location information on their coworkers, generally for location and coordination purposes.

An extension of the ability of the Active Badge system is the Active Bat system [6]. Active Bats are worn in a similar fashion to the Active Badges, but are found using ultrasonic technology instead of infrared technology. The Active Bat system

utilizes triangulation by gathering information on the worn transmitter from three locations in the ceiling and utilizes triangulation.

The usage of Active Bat or Active Badge raises serious privacy concerns. One remedy proposed in [4] is to simply take the device off. However, this obviously affects the usefulness of the system.

2.2 Global Positioning System (GPS)

The United States Department of Defense developed the GPS as a means by which to target enemy units and infrastructure, and for other military purposes. However, GPS has become commonplace in civilian applications, ranging from emergency location and assistance (such as OnStar [1], mentioned earlier) to simply finding directions to a restaurant or hotel (i.e., using GPS for navigation).

GPS has a constellation of satellites orbiting the Earth and several ground stations, allowing users of the system to triangulate (to be more precise, trilaterate) their position using three (and sometimes four) of these satellites. The satellites actively transmit *their* location information, allowing users with the appropriate equipment to calculate their distance from these satellites. GPS is accurate within several meters, and the improved Differentiated GPS is accurate within one or two meters.

Location privacy is preserved through utilization of GPS due to the fact that the individual applications and hardware perform the calculations themselves using the satellites' locations, rather than depending on the satellites to locate the individual. However, it is important to note that although GPS does not actively transmit a user's location, it is often built into devices that have the capability to do so.

2.3 The Cricket Location-Support System

GPS is not very useful in indoor environments. Active Badge and Active Bat (also described earlier) are useful in indoor environments such as offices and hospitals, but do not provide location privacy short of removing the transmitters.

The Cricket Location-Support system is an inexpensive location-support service that seeks to take advantage of location privacy, such as that provided by GPS, but to work properly in an indoor environment such as the case with Active Badges or Active Bats. The user carries a *listener,* which gathers location information from *beacons* located throughout a building. Thus, unlike the Active Badge and Active Bat systems, the environment component of the system is the component that advertises its location, rather than the component carried by the user. In a similar fashion to GPS, the environmental beacons forfeit their own location privacy to allow the user to determine his/her location. This is why we refer to the system as having *location support* rather than *location tracking.*

2.4 Cellular Phones

Cellular phones transmit data using radio signals. These phones are called cellular because cellular towers create *cells* in a certain area in which the devices can communicate. When switching to different cells, the process is generally transparent to the user.

Although they provide many benefits, these phones have serious location privacy considerations. If necessary, cellular towers can sometimes be used to triangulate an individual's position, thus determining that individual's location. Additionally, GPS-enabled phones could actively transmit information about an individual's location. Similar transmission is done legitimately in technologies such as OnStar [1], described earlier. However, it is obvious that this technology could be abused.

Some authors have noted specific instances in which companies and governments have utilized the cellular networks to gather data on individuals [11, 12]. According to these authors, cellular phones are often used for tracking individuals and collecting information about them.

3 Anonymity and Identity Protection

In the previous section, we described one of the characteristics of location privacy: protection of location information. In this section, we describe the other characteristic: protection of *identity* information.

Many applications are not useful unless they have the ability to access a user's location information. In these situations, complete protection of location information is simply not desirable. A prime example of this would be OnStar (described earlier). Since location data is key to such a system, protection of an individual's identity information becomes the primary means by which location privacy is attained.

We shall focus on the primary means by which identity information is protected: *anonymity.* Anonymity is the quality of being indistinguishable from other members of a set of subjects. This set of subjects is called the *anonymity set.*

3.1 Mix Nodes and Mix Zones

Mix Nodes are special nodes in a network. When they receive packets of information, they reorder them, making the incoming packets unlinkable with outgoing packets. That is, any packet coming in to a mix node cannot be associated with any packet exiting the node.

Therefore, an attacker cannot follow a message from its source to its destination without eliciting help from mix nodes in the network. Mix nodes throughout the network would have to collude in order to provide information regarding the various packets they have reordered.

A specific identity-preservation technique important to mobile computing environments is that of the *mix zone*, which is based upon mix nodes (described above). This technique was developed by Beresford and Stajano of the University of Cambridge, and is described in [13]. In [13, 14], the authors define two types of zones: *application zones* and *mix zones.* An application zone is a spatial region in which users may send and receive messages and access various location-based services. However, a mix zone is a spatial region in which users do not communicate, and do not request callbacks.

The effectiveness of mix zones depends on *pseudonyms.* Pseudonyms are false names (false identities). A user must have a pseudonym for each application with which he/she interacts. If a user were to only have a single pseudonym used for all

applications, then the applications could collude and provide one another with information. Additionally, a user must change his/her pseudonyms frequently in order to avoid being tracked. If a user maintained a pseudonym for an application over an extended period of time, then the purpose of the pseudonym would be defeated: the pseudonym would act as a long-term identity. Thus, simply having a single, fixed pseudonym is not sufficient.

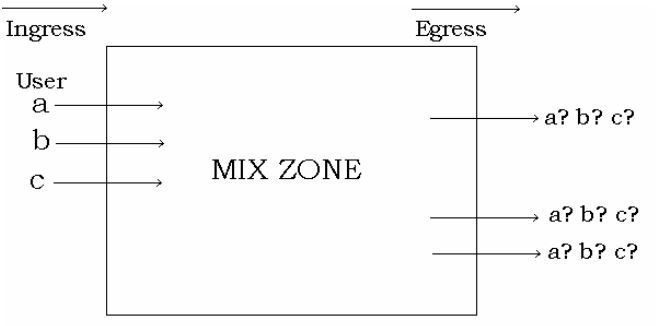


Fig. 1. An example of mix zone

As we can see from Fig. 1, users enter the mix zone where they do not request callbacks and do not communicate. They must then change their pseudonyms before exiting the mix zone. Therefore, when they leave the mix zone and enter an application zone, it becomes more difficult for an application to correlate one pseudonym entering with another pseudonym exiting. However, Beresford and Stajano point out that this is not foolproof. There are certain movement patterns that are more likely than others. For example, a user is not as likely to make a U-turn, but rather more likely to continue in a straight line.

3.2 Onion Routing and Tor

The Onion Routing [15, 16] architecture attempts to prevent the usage of traffic analysis to acquire information about individuals. It attempts to provide anonymity for both the sender and the receiver by utilizing anonymous socket connection via proxy servers.

We identify an *initiator* as a node that sends information to another node initially, and a *responder* who receives the initial communication from the initiator, and responds. Additionally, *routing nodes* know only information about the node that sent them the information and the node to which they must forward the information.

To hide the routing information, the information (data stream) must travel through various nodes on its way to the intended responder. When a particular node receives data, it doesn't know whether the node that just sent it the "onion" (multi-layer encrypted message with padding) is the initial sender (initiator) or just another intermediate node (routing node) along the onion's path. Likewise, when the node prepares to forward the onion, it does not know whether or not the node it must

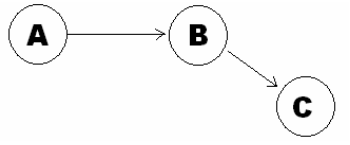


Fig. 2. Onion routing

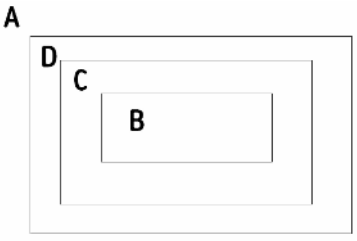


Fig. 3. The structure of the onion routing Message

forward the onion to is the intended recipient (responder) or, again, just another routing node along the onion's route.

Consider Fig. 2. Node B receives a message from node A, and this message is addressed to node C. However, because of the way Onion Routing works, node B does not no if node A was the origin of the message (the initiator), or if node C is the final recipient of the message.

The structure of the message is similar to what is shown in Fig. 3. The namesake of Onion Routing is due to the resemblance of the message to an onion, with various layers. In the above figure, Node A is the original sender (initiator), and node B is intended as the recipient (responder node) of the message (payload).

Node A first encrypts the message with the public key or shared secret key of the intended recipient, B. Then, A encrypts the result with C's key, then finally with D's key. Node A then sends the onion to D. D does not know if A is just forwarding an onion that A received, or if A is the origin of the onion. After Node D decrypts the onion, it knows that the node that it must forward the onion to is C, so D forwards the onion to C. This continues until B receives the onion, decrypts it and receives the message.

Tor [17] is the most recent development in Onion Routing technology, and seeks to improve on the original Onion Routing scheme. The details may be found in [17].

In summary, Onion Routing is important to our discussion of location privacy due to its application of protecting sender anonymity and receiver anonymity from intermediate nodes in the network.

3.3 Group Signatures and Ring Signatures

Some new encryption algorithms are privacy-aware. An example of such a privacy-aware encryption algorithm is that of *group signatures*. Group signatures were originally introduced by Chaum and van Heijst in [18]. Groups consist of several members, and one manage for the group, the group manager (GM). Instead of multiple public keys for each member of the group, there exists a single group public key (*gpk*). Each member can produce a signature using his/her own secret signing key and the gpk [18, 19].

The foundation of group signature schemes is in the ability of any member of the group to digitally sign a message on behalf of the group. This signature is verifiable,

in that it can be verified that it came from a particular group. However, the individual within the group that signed the message is not identifiable, except by the group manager. The GM uses his/her own secret key (the group manager secret key, or *gmsk*) along with a given signature, *s* to determine the identity of the member of the group that generated the signature, which is called *traceability.*

An entity who does not possess the gmsk on the other hand, should not be able to determine the identity of a group member who signed a message *s*. This quality is known as *anonymity.* In other words, the members of the group are anonymous within the group (essentially, their anonymity set), and are indistinguishable from other group members.

Some have suggested [19] that many of the other qualities that many group signature schemes possess are derived from the *traceability* and *anonymity* qualities. For example, *unlinkability* means that when a member signs multiple messages, the resulting signatures should not be linkable, that is, display characteristics that expose that they came from the same signer. The *exculpability* quality of a group signature scheme means that no one should be able to sign a message and make it appear as if it came from a different member of the group.

Another quality pertaining to forgery is *unforgeability.* This quality is similar to exculpability, but pertains to forgery produced from outside the group. No one except members of the group should be able to forge signatures and make them appear as if they came from the group. Thus, the distinction between exculpability and unforgeability is that exculpability can be seen as prevention of internal (insider) threats, while unforgeability can be seen as a prevention of external (outsider) threats.

Coalition-resistance is yet another desirable property of group signatures. If some subset of the group (proper subset or even the entire group) colludes, they cannot create a valid group signature that the GM cannot attribute to one of the members in the colluding subset.

Group signatures schemes must consist of at least five algorithms in order to be effective. These algorithms are `Setup`, `Join`, `Verify`, `Sign`, and `Open`. The `Setup` procedure, as the name suggests, initializes the group public key, the group manager secret key, and other basic data about the group. `Join` allows new members to join the group. `Verify` utilizes the group public key and a message, and determines if the signature of the message is valid, i.e. came from a particular group. `Sign` uses an individual member's private key to sign a message. Finally, `Open` is used when the GM needs to determine the identity of a member who signed a particular message (traceability).

One of the serious problems with the original group signature scheme is that the signatures grow linearly with respect to the size of the group. This is obviously unacceptable for very large groups (such as groups that may be found in a vehicular environment). Thus, in [20], an efficient group signature scheme for larger groups is discussed, in which the key sizes are independent of the group size.

In [21], a very appealing short group signature scheme is introduced. It is based upon Strong Diffie-Hellman (SDH) and Linear assumptions. The size of the signatures used in this scheme is under 200 bytes and is comparable in security to regular RSA of similar length.

An interesting potential application of group signatures in vehicular environments is presented in [2]. Various group divisions exist. Specifically, an Emergency Vehicle

Group would consist of police cars, fire trucks, ambulances, etc. In this sort of situation, if an emergency vehicle approaches another group of vehicles (not necessarily a formal group, but rather a collection of vehicles), the vehicles will be alerted to the approach of the vehicle.

However, the individual identity of the emergency vehicle does not need to be divulged. Thus, they can use group signatures, and the vehicles in the vicinity can authenticate and verify that the approaching vehicle is indeed an emergency vehicle.

4 Laws and Policies

In some circumstances, we must divulge our location information. For example, in emergency situations, emergency vehicles will need to locate individuals in need of help. Identity information may also be required in order to determine allergies, and other possible medical issues that could arise. Thus, laws and policies may provide the final safeguard against abuses of information obtainment.

The US Privacy Act of 1974 [23] was amended in order to protect individuals from unscrupulous use of their private information. The US Electronic Communications Privacy Act of 1986 [24] specifically addresses the use of wiretapping. The US Wireless Communications and Public Safety Act of 1999 [25] establishes some basic laws and regulations regarding E-911 and other location-based services. Similar laws and policies to those in the United States have been enacted in Europe and Japan (see [26]).

In general, we believe that existing privacy protections laws lag behind the technology advances and are inadequate to safeguard citizens from possible government intrusions and corporation maltreatment of personal privacy made possible by new information technologies.

5 Open Issues and Directions for Further Research

Although many problems regarding location privacy have been solved, several important issues still exist.

The issue of scalability is a key issue. In regards to large-scale mobile environments (especially vehicular environments), research must be performed in order to gather data and to develop new approaches and models to better implement commercial and governmental structures in these environments.

In situations where a silent period may be employed, there is definite interest in determining the optimal silent period length. Although a random silent period attempts to ensure an acceptable degree of privacy, the quality of service must not be ignored either.

If a scheme uses a central authority, we must decide how much information and power this authority may have. If a central authority exists in a model that is intended to provide location privacy, we should put limits on the central authority's power. We could limit the central authority's powers by means of checks-and-balances, much the way a government does. Thus, accountability is an important issue.

What if a user uses a location-aware application and can benefit from having a specific set of settings for that application? Customization is another very important issue in regard to location-based and location-aware services.

6 Summary and Conclusions

Location privacy is an issue that is unfortunately often left unaddressed. This special type of privacy consists primarily of protection of identity, and protection of location information. It is not always possible to protect both identity and location information, but often it is possible to protect one or the other. When an application requires an individual's location information, it is often necessary to provide protection for the individual's personally identifiable information (such as by using pseudonyms), and vice versa.

Location-based services and location-aware applications are becoming more and more commonplace. However, these services and applications raise serious privacy concerns. Although some have suggested that many people are willing to give up some privacy for the sake of convenience, people in general are still concerned about their privacy [27]. Research is being done to integrate location-awareness into traffic management and cooperative driving using group signatures to aid in protection of location privacy [2]. Thus, there exist the seemingly dichotomous desires of convenience (through access to location-based services and applications) and that of privacy. We must take care in developing a framework that avoids a "Big Brother" scenario, but still provides convenience and a degree of transparency.

Acknowledgements

This work is supported in part by the US NSF Grant CNS-0521142.

References

1. Onstar. http://www.onstar.com/us_english/jsp/explore/onstar_basics/technology.jsp
2. J. Guo, J.P. Baugh, Security and Privacy in Vehicle Safety Communication Applications. SAE International 2005
3. EZPass, http://www.ezpass.com
4. R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. ACM Transactions on Information Systems, volume 10, no. 1 : pp.91-102, 1992
5. I.A. Getting, The Global Positioning System. IEEE Spectrum, volume 30, no. 12 : pp.36-47, December 1993
6. A. Ward, A. Jones, and A. Hopper. A New Location Technique for the Active Office, IEEE Pers. Communication , volume 4, no. 5 : pp. 42-47, October 1997
7. Nissanka B. Priyantha et al., The Cricket Compass for Context-Aware Mobile Applications
8. Nissanka B. Priyantha, Anit Chakraborty, Hari Balakrishnan, The Cricket Location-Support System, 6th ACM National Conference on Mobile Computing and Networking (ACM MOBICOM), Boston, MA, August 2000

9. Adam Smith et al., Tracking Moving Devices with the Cricket Location System, Published in ACM MobiSys, 2004
10. Julia Layton, Marshall Brain, and Jeff Tyson, How Cell Phones Work http://www.how stuffworks.com
11. Schneier On Security. http://www.schneier.com/blog/archives/2005/07/automatic_surve.html
12. http://www.thenewspaper.com/news/06/696.asp
13. A. Beresford, F. Stajano, Location Privacy in Pervasive Computing, Published by the IEEE CS and IEEE Communications Society, Pervasive Computing pg. 46, January-March 2003
14. A. Beresford, Location Privacy in Ubiquitous Computing, Published in University of Cambridge Technical Report, UCAM-CL-TR-612, January 2005
15. David M. Goldschlag, Michael G. Reed, Paul F. Syverson, Hiding Routing Information, Workshop on Information Hiding, Cambridge, UK, May 1996
16. Paul F. Syverson, Michael G. Reed, David M. Goldschlag, Onion Routing Access Configuration, Published in DISCEX 2000: Proceedings of the DARPA Information Survivability Conference and Exposition Vol. 1, IEEE CS Press, pp. 34-40 Naval Research Laboratory, Hilton Head, SC, January 2000
17. Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-Generation Onion Router. Proceedings of the 13th USENIX Security Symposium, August 20004
18. D. Chaum, E. van Heijst, Group signatures. Advances in Cryptography – Eurocrypt '91 Springer-Verlag (1991), pp. 257-265
19. M. Bellare, Daniele Micciancio, Bogdan Warinschi, Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions. Advances in Cryptography Eurocrypt '93 Springer-Verlag (2003)
20. J. Camenisch, M. Stadler, Efficient Group Signature Schemes for Large Groups. Avances in Cryptography, CRYPTO '97, 1997.
21. D. Boneh, Xavier Boyen, Hovav Shacham, Short Group Signatures. Advances in Cryptography – CRYPTO '04 Springer-Verlag (2004)
22. Ronald Rivest, Adi Shamir, Yael Tauman, How to Leak a Secret: Theory and Applications of Ring Signatures, 2004
23. United States Privacy Act, 5 U.S.C. § 552a http://www.usdoj.gov/04foia/privstat.htm, 1974
24. 24.United States Electronic Communications Privacy Act, http://straylight.law.cornell.edu/uscode/html/uscode18/usc_sup_01_18_10_I_20_119.html, 1986
25. United States Wireless Communications and Public Safety Act. http://thomas.loc.gov/cgi-bin/query/z?c106:H.R.438.IH:, 1999
26. Wireless Location Privacy: Law and Policy in the U.S., EU and Japan, http:// www isoc.org/ briefings/015/
27. Roy Campbell et al. Towards Security and Privacy for Pervasive Computing.

Utilizing Secure Three Hop Links to Agree Pairwise Keys in Wireless Sensor Networks

Gicheol Wang[1] and Dongsun Park[2]

[1] CAIIT, Chonbuk National University,
Jeonju, Jeonbuk 561-756, Republic of Korea
gcwang@dcs.chonbuk.ac.kr
[2] Division of Electronics and Information Engineering, Chonbuk National University,
Jeonju, Jeonbuk 561-756, Republic of Korea
dspark@chonbuk.ac.kr

Abstract. Because sensor networks consist of devices with weak physical security, they are likely to be compromised by an attacker. So, it is very important to establish a pairwise key securely between the communicating nodes. However, utilizing one hop local keys are known to be very vulnerable to threats caused by compromised nodes. This paper proposes a scheme where each node establishes three hop local keys at network boot-up time and employs them for a later pairwise key establishment. When any two nodes agree a pairwise key, all nodes on the route between two nodes contribute to the agreement of the pairwise key. Here, the initial three hop local keys are employed for encrypting a secret key delivered from a node to other nodes. Therefore, the proposed scheme bothers attackers to compromise much more nodes than the scheme using one hop local keys only. The simulation results showed that the proposed scheme provides better performance and higher security than the one hop local key based scheme.

1 Introduction

To the wide employment of sensor networks, they should guarantee a secure communication to each node in the network. This is because sensor nodes are deployed in unattended and often adversarial environments. That is, under the harsh conditions, an attacker can easily overhear or modify the traffic between sensor nodes and disguise as a legal sensor node. For this reason, communication between sensor nodes should assure of confidentiality and authentication to each node. Namely, in sensor networks, it is very important to establish a secure key between any two nodes and distribute it to both nodes in a secure manner. This unique key, which is distinct from other pairs of sensor nodes, is referred to as pairwise key hereafter.

Because the assumption of having TTPs (Trusted Third Parties) in sensor networks is unrealistic, all nodes should be responsible of the cooperative key distribution and management duty. As a result, the pairwise key establishment scheme using a key shared with base station [1] is not suitable for sensor networks. Also, due to the limited computation and energy resources of sensor nodes, it is undesirable to use public

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 946 – 955, 2006.

key based approach [2] employing public key cryptography, such as Diffie-Hellman protocol [3] and RSA [4].

For such reasons, most of key management schemes for sensor networks force each node to obtain some keys from a key server in advance and make a key agreement using the keys. However, they cause a serious security problem owing to the duplication of keys. Namely, pre-obtained keys in a node also exist in other nodes with a predefined probability in these schemes. In these schemes, if a node is compromised by an attacker, it has an impact on other nodes having the same keys. Furthermore, as an attacker compromises more nodes in the elapse of time, key agreement between any two nodes is exposed to the attacker more frequently. Another approach forces each node to establish local keys with its neighbors and use them for later pairwise key establishment [5-6]. The local key establishment procedure is called secure link setup. Using local keys, each source of a pairwise key establishment splits its secret key into n shares and delivers one share to a neighbor of the destination. The neighbor also employs local keys to deliver the share to the destination. Then, each entity of key establishment makes a key agreement using the same shares. However, this approach causes heavy communication and computation overhead. Moreover, this approach is also vulnerable to the increase of compromised nodes.

The proposed scheme also forces each node to establish local keys at network boot-up time and employ the local keys for later pairwise key establishment. However, basic principle for key agreement, range of local key establishment, and usage of local keys are significantly different from the previous approach. First, the previous approach allows each source of a key establishment to lead the key agreement but the proposed scheme forces all nodes on the route of the key establishment to contribute the agreement. Second, in the proposed scheme, the number of messages exchanged is proportional to the path length between entities of a key establishment and it is reduced significantly as compared with the previous approach, whose message exchange is proportional to node density. Lastly, the proposed scheme mitigates the threat resulted from the node compromise by extending the range of local key establishment.

The organization of this paper is as follows. Section 2 describes the existing approaches for secure key establishment in sensor networks. In section 3, range in local key establishment is selected via a simulation and the detailed description of the proposed scheme follows. Section 4 analyzes the efficiency and the security of the proposed scheme through the comparison with the previous scheme using one hop local keys. Section 5 concludes the paper.

2 Related Work

Escenauer and Gilgor proposed a pairwise key establishment scheme where each node obtains some keys from a key pool and establishes some keys with neighbors using them [7]. Then each node establishes pairwise keys with the nodes which are further than one hop, using the established keys. Du et al. proposed another pairwise key establishment scheme based on node deployment knowledge [8]. With the deployment knowledge, the key server can distribute only a small number of keys to each node. However, because each node shares a sufficient number of keys with the nodes

which are in close vicinity to it, it can establish a pairwise key with them securely. Above two schemes are also based on the key pre-distribution. These key pre-distribution based schemes are very vulnerable to the increase of compromised nodes. This is because a lot of pre-distributed keys are shared with other nodes according to the probability of key pre-distribution. Liu and Ning proposed a general framework for pairwise key establishment based on pre-distribution of secret information [9].

Zhu et al. proposed a pairwise key establishment scheme which can be applied to the mobile environment without any modification [10]. In this scheme, a key server pre-distributes m distinct keys to each node from a key pool of l keys. Then the source node determines the number of secret shares (e.g. n) according to a pre-defined security level and looks for proxies as the number of secret shares. Here, the proxy nodes are the nodes which share the same keys with both the source and the destination. The source node generates a pairwise key and splits it into n secret shares. Then, the source node sends the shares to the destination node through the proxies, and the destination reconstructs the pairwise key by performing an XOR operation with the shares. Therefore, an adversary should achieve all local keys used for encryption of n secret shares to find out a pairwise key between the source node and destination node. Security of this scheme depends on the probability that each key is distributed from the key pool to a node (i.e. m/l). That is, the smaller the probability is, the higher the network security level is. Also, the more the number of secret shares becomes, the higher the network security level becomes. The most serious problem is that the number of compromised pairwise keys significantly increases as the number of compromised nodes increases.

A pairwise key establishment scheme without key pre-distribution, called LEAP (Localized Encryption and Authentication Protocol), was proposed in [5]. In this scheme, each node establishes one hop pairwise keys with its neighbors using the initial network–wide key. When a source of key establishment wants to establish a pairwise key with the destination at a distance of two hops, it broadcasts a message searching neighbors of the destination. A neighbor of the destination can be a role of proxy, because it shares two pre-established pairwise keys with the source and the destination. After generating a pairwise key, the source splits it into n shares and forwards the n shares to the destination through the proxies. Notice that all shares are encrypted with one hop pairwise keys before the transmission. The destination reconstructs the original pairwise key after receiving all shares. This scheme can be employed on the fly for any two nodes which are further than two hops away from each other, by extending the search range of proxies. As long as not all neighbors of the destination are compromised by attackers, this scheme is still secure. Therefore, to achieve a higher security, this scheme requires a lot of nodes connected to the destination. However, this leads to the increase of communication and computation overhead. Moreover, even if small number of nodes is compromised, many secret shares are exposed to attacker and this in turn poses a serious threat to a pairwise key establishment. Dutertre et al. proposed a similar scheme where each node establishes pairwise keys with neighbors using a key shared among all nodes [6]. However, this scheme does not deal with the pairwise key establishment between any two nodes which are further than one hop away from each other.

3 Pairwise Key Establishments Using Initially Established Three Hop Local Keys

3.1 Range Determination of Initial Local Key Establishment

In the proposed scheme, each node establishes local keys with its neighboring nodes which are at most d (>1) hops away at network boot-up time and saves them in its memory. Notice that each node considers the nodes sharing the same local key as friends. When any two nodes want to establish a pairwise key, the nodes on the route between them send their secret shares to both nodes. Two nodes construct the pairwise key by performing an XOR operation with the secret shares. Here, the nodes on the route employ the local keys to protect the secret shares. Notice that the exchange of secret shares can be assisted by the friends which share the same local key with the source or the destination of the key establishment. To raise the security of secret shares transmitted, it is needed to employ a lot of local keys originated from the friends during the exchange. If each node on the route discovers such friends in a large range (e.g. r (>2) hops), they may acquire a sufficient number of local keys. However, this causes a significant communication overhead. Conversely, if each node established local keys with more neighbors (i.e. a large value of d) at network boot-up time, there will be no need for each node to discover such friends in a large range. Also, the range of local key establishment determines the secure transmission range within which any two nodes can securely exchange their secret information with each other. That is, if the range of local key establishment is extended to two hops, only communications within two hops can be secured by the local keys. To minimize the overhead resulted from the discovery of such friends, we fixed the r value to 1. Then, to determine the value of d, we conducted a simulation with the following parameters. Initially, 100 nodes were distributed in 500m * 500m plane and the transmission range was set to 30m. Each node established local keys with its neighboring nodes at network boot-up time. Then, when a node wants to deliver its secret information to its friends, it tries to send the secret information securely using one hop friends' local keys. Then each node measures how many times the transmission of secret information is succeeded under the existence of nodes with disable state. In sensor networks, nodes tend to be destroyed or to become in disable state due to energy exhaustion, theft, and so on. In this simulation, 10% of whole nodes were in disable state, and they were selected randomly. The measurements in Figure 1 show the rate at which a node can communicate with friends securely via help of one hop friends under the existence of disable state nodes. Next, when the range of local key establishment was extended, the variation of the rate was also measured.

As shown in Figure 1, it is natural for the rate to increase as node density increases. If each node establishes one hop local keys only, it is very difficult to communicate with friends via one hop friends. This is caused by the existence of disable state nodes. If the range of local key establishment increases further than one hop, the rate increases significantly. Another observation from Figure 1 is that four hop local key establishment cannot make sufficient improvement over the three hop local key establishment. Therefore, we assume that each node establishes three hop local keys using the initial network-wide key at the network boot-up time.

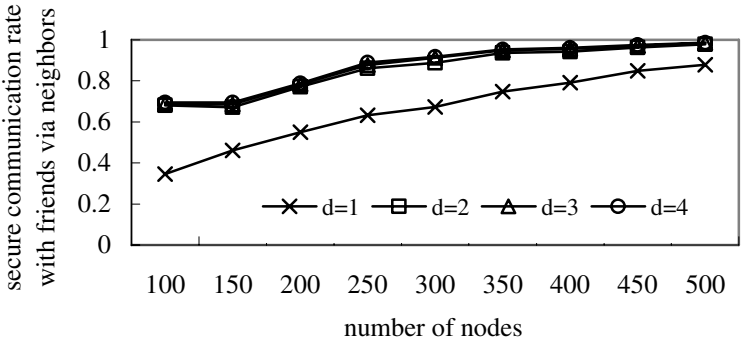


Fig. 1. Secure communication rate with friends via one hop friends vs. number of nodes

3.2 Three Hop Local Key Establishment

It is assumed that all nodes are initially trustworthy at network boot-up time, and an attacker cannot compromise a node during the three hop local key establishment. Because the local key establishment is taken place before deploying the nodes into the workplace, this assumption is feasible in practical circumstances. After generating three hop local keys, each node erases the initial network-wide key from its memory to prevent an attacker from joining a network. We employ the following notations in the rest of this paper.

- K_I: initial network-wide key
- F_K: pseudo random function
- $\{M\}_K$: encryption of message M with a symmetric key K
- $MAC(K,M)$: message authentication code of M with a symmetric key K
- $nonce_A$: a random number generated by node A

First of all, each node establishes one hop local keys with its neighbors through the following steps.

1. A node u generates a random number $nonce_u$ and broadcasts a hello message containing its identifier and the random number.
2. Node v receiving a hello message from a neighbor compares the neighbor's identifier with its identifier. If the neighbor's identifier is lower, it generates its master key ($K_v = F_{K_I}(v)$) and the one hop local key using the neighbor's identifier and its master key ($K_{uv} = F_{K_v}(u)$). Otherwise, it discards the hello message. Then it computes the MAC value for $nonce_u$ and its identifier using its master key, and responds to node u with the MAC value. If node v receives the hello message from a higher ID node, it does not respond.
3. Node u generates node v's master key ($K_v = F_{K_I}(v)$) and the one hop local key shared with node v ($K_{uv} = F_{K_v}(u)$) after verifying the node v's legality by computing the MAC value.

After establishing one hop local keys, each node broadcasts a list of nodes with which it established one hop local keys, and the list is encrypted with the initial network-wide key. Establishment of two hop local keys is performed via the comparison of the identifier as the step 2 in the one hop local key establishment. For instance, a node u receives a list of nodes from a neighbor v (e.g. $\{a, z\}$). Firstly, node u generates master keys of node a and z (i.e. $K_a = F_{K_1}(a)$, $K_z = F_{K_1}(z)$). Then, because node u's identifier is higher than that of node a's, it establishes the two hop local key with node a using its master key (i.e. $K_{au} = F_{K_u}(a)$). On the contrary, because node u has a lower identifier than z, it establishes the two hop local key with node z using z's master key (i.e. $K_{uz} = F_{K_z}(u)$). Three hop local key establishment follows the two hop local key establishment. Notice that each node can find the route path to any node within three hops during the three hop local key establishment.

When a new node joins a network, it tries to establish one hop local keys with the neighboring nodes. Then, the new node can establish one hop local keys with the nodes which have higher identifiers than the new node. On the contrary, the new node cannot establish the nodes which have lower identifiers than the new node, since they do not have the initial network-wide key. Then, the new node can establish two and three hop local keys with neighboring nodes having higher identifiers as described above. Notice that the exchange of node list between any two nodes is encrypted by the one hop local key established between them.

3.3 Pairwise Key Establishments Using Three Hop Local Keys

When a sensor node wants to establish a pairwise key with a friend, it needs not find a route to the friend. If a node wants to establish a pairwise key with a non-friend node, it should find the route to the node. For example, node u wants to find a route to node v which is a non-friend. Firstly, node u sends a route inquiry message including node v's identifier and the sequence number to three hop distance nodes. Here, the sequence number is employed for discard of redundant messages. The three hop distance nodes check whether v is a friend or not. If v is a friend, they send back a route reply message including the route to node v. Otherwise, they record the sender of the route inquiry message as the next intermediate node to u and send the route inquiry message to their three hop distance nodes. If a node has already received the same route inquiry message, it discards the message. Above procedure repeats until a route to node v is found. If a route to v is found, a route reply message is sent back via the reverse path to u. Notice that the messages exchanged between three hop distance nodes are protected by three hop local keys.

In Figure 2, node u, w, and x generate its secret key and send it to u and v. Each node on the route notifies the time when it can send its secret key by receiving a key setup message. Because the security of a pairwise key relies on the security of secret keys from nodes on the route, it is required to protect the secret keys from attacks. Therefore, each node on the route sends its secret key after encrypting with local keys. Here, the local keys can be originated from its storage or its neighbors. This selection is governed by dividing the secret key by 2. Namely, if

the remainder is 0, the node employs its local key. Otherwise, it obtains the local keys of neighbors via inquiry and reply. For instance, in Figure 2, nodes *w* and *x* obtain local keys from neighbor nodes prior to sending the secret keys. Of course, neighbor nodes provide their local keys to the inquiry node only if they share a local key with at least one node of source and destination. Then each node selects two keys from the received local keys and employs one key for the transmission to node *u* and the other key for the transmission to node *v* respectively. Destination *v* generates a key for encryption of its secret key (i.e. K_{part}) using received secret keys. Notice that source *u* can also construct the same key. Node *v* also constructs the pairwise key (i.e. K_{uv}) shared with *u* by performing an XOR operation with received secret keys and its own. Then *v* generates a response message and sends it to source *u*. The response message consists of the encrypted secret key of *v* and the MAC value of DONE message authenticated with the pairwise key. After decrypting the destination's secret key, source *u* constructs the pairwise key shared with destination *v* by performing an XOR operation with the *v*'s decrypted secret key and the pre-received secret keys. Lastly, *u* verifies the correctness of the pairwise key by confirming the DONE message. If the source *u* cannot receive the DONE message during a specific period of time, the key establishment falls into a failure. Then the source *u* retries the key establishment in a later time.

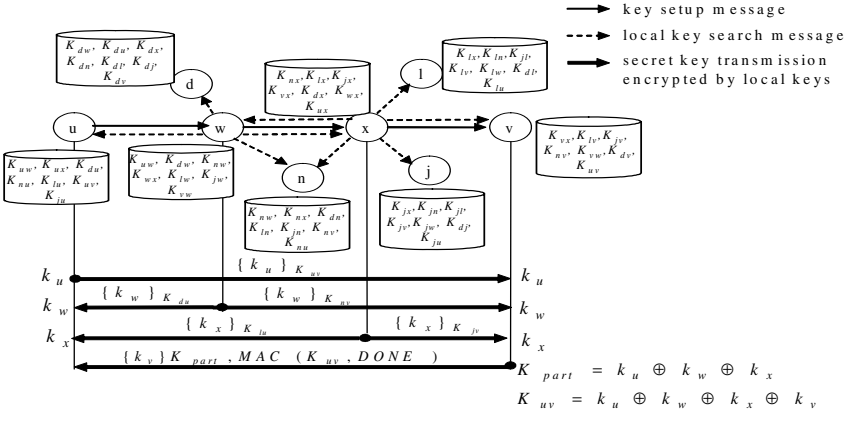


Fig. 2. Pairwise key establishment with at most 3 hop distance node

Unfortunately, above procedure for pairwise key establishment cannot be employed when a node on the route should transmit its secret key to a target node which is further than three hops away. In this case, the node requires the three hop distance node to relay its secret key on behalf of it.

4 Simulation Results

To evaluate efficiency and security of the proposed scheme, an event-driven simulator was developed. Because our scheme carries out a pairwise key establishment without

pre-distribution of keys, comparison with the schemes using pre-distributed secret information [7-10] seems to be unsuitable. Therefore, our scheme is compared with LEAP [5] where each node establishes one hop local keys with its neighbors and use them for pairwise key establishments. LEAP has already been described in section 2. Simulation parameters and their values are listed in Table 1.

Table 1. Simulation parameters

Parameter	Value
Number of nodes	200~500
Simulation area	500meter * 500meter
Transmission range	40meter
Simulation time	600 seconds
Rate of compromised nodes	10%~70%
Period of key establishment	2 seconds

The number of messages exchanged during a key establishment was selected as a metric for efficiency evaluation. Notice that the number of messages exchanged in the proposed scheme includes the number of messages exchanged during the local key establishment. If certain nodes are compromised by an attacker, the local keys from the compromised node are exposed to the attacker. Then, if the local keys obtained from compromised nodes include all local keys employed during a pairwise key establishment, the pairwise key is exposed to the attacker. Therefore, under the existence of compromised nodes, the exposure rate of local keys employed during a pairwise key establishment was selected as a metric for security evaluation. Here, compromised nodes were selected randomly. Our scheme in all simulation results is referred to as THLKA (Three Hop Local key based Key Agreement).

Figure 3 shows the variation of the number of messages exchanged as the number of nodes increases. As shown in Figure 3, LEAP is influenced by the increase of nodes. This is because node density becomes higher as the number of nodes increases under the fixed simulation boundary. That is, under a high node density, a source of key establishment should exchange a lot of messages to seek proxy nodes. Also, as the number of proxy nodes increases, communication overhead between the source and the proxy nodes increases accordingly. On the other hand, because our scheme does not need to seek any proxy nodes, it reduces the number of exchanged messages. Furthermore, it is much less influenced by node density. This is because main factor having an impact on message exchange is not the node density but the route length between the source and the destination.

Next, to grasp the effect on security caused by existence of compromised nodes, we estimated the exposure rate of local keys employed for pairwise key establishments, as the number of compromised nodes increases. As shown in Figure 4, when the rate of compromised nodes reaches to 20%, 80% of local keys employed for pairwise key establishment are exposed to the attacker in LEAP. This is because key shares are delivered to the destination of key establishment via proxy nodes in LEAP. If only one node between a source and a proxy node is compromised, the key share is

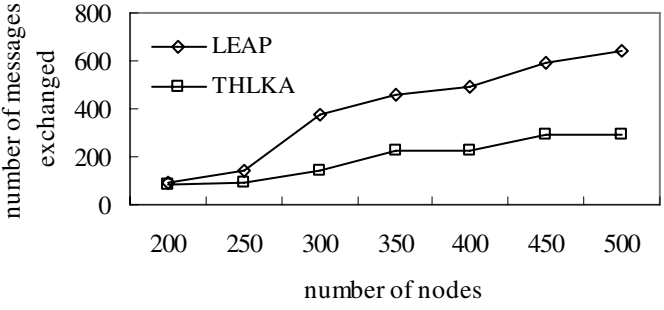


Fig. 3. Number of messages exchanged vs. number of nodes

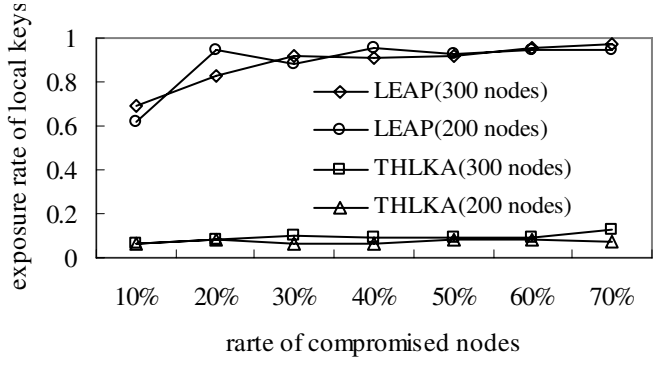


Fig. 4. Exposure rate of local keys vs. rate of compromised nodes

known to the attacker immediately. This fact gives us a meaningful intuition. If an attacker wants to know a pairwise key between any two nodes, he can easily achieve his aim by compromising only some nodes on the routes between the source and the proxy nodes. In contrast to LEAP, our scheme let neighbors of nodes on the route as well as all nodes on the route to take part in pairwise key establishments. This indicates that an attacker should compromise much more nodes to get a pairwise key than in LEAP. Therefore, our scheme is much more impervious to compromise of nodes. As shown in Figure 4, even though the rate of compromised nodes reaches to 70%, only 12% of local keys needed for getting a pairwise key are exposed to the attacker.

5 Conclusion

We proposed a pairwise key establishment scheme using initial three hop local keys in wireless sensor networks. The proposed scheme does not depend on the probability of key pre-distribution and keeps durability against the compromise of nodes, since a lot of nodes are involved in a pairwise key establishment. The simulation results have proven that the proposed scheme reduces messages exchanged during a key establishment as compared with LEAP using one hop local keys only. More importantly, it

was shown by simulations that the proposed scheme minimizes the effects resulted from the increase of compromised nodes.

Acknowledgement

This work was supported by the grant of Post-Doc. Program, Chonbuk National University (2005).

References

1. Perrig, A., Szewczyk, R., Wen, V., Culler, D., Tygar, J.D.: SPINS:Security Protocols for Sensor Networks. Proc. of the 7th ACM/IEEE Int'l Conf. on MbiCOM (2001) 189-199
2. Wang, G., Bang, S., Cho, G.: A Pair-wise Key Establishment Scheme without Predistributing Keys for Ad-hoc Networks. Proc. of IEEE ICC'05 (2005)
3. Diffie W., Hellman, M.E.: New Directions in Cryptography. IEEE Trans. on Information Theory, Vol. 22, No. 9 (1976) 33-38
4. Rivest, R.L., Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, Vol. 2, No. 2 (1978) 120-126
5. Zhu, S., Setia, S., Jajodia, S.: LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. Proc. of the 10th ACM Conference on CCS '03 (2003)
6. Dutertre, B., Cheung, S., Levy, J.: Lightweight Key Management in Wireless Sensor Networks by Leveraging Initial Trust. SRI International, Tech. Rep. SRI-SDL-04-02 (2004)
7. Eschenauer, L., Gilgor, V.D.: A Key-Management Scheme for Distributed Sensor Networks. Proc. 9th ACM Conf. on CCS '02 (2002) 41-47
8. Du, W., Deng, J., Han, Y.S., Chen, S., Varshney, P.K. : A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge. Proc.of IEEE INFOCOM 2004, Vol. 1 (2004) 586-597
9. Liu. D., Ning, P.: Establishing Pairwise Keys in Distributed Sensor Networks. Proc. of the 10th ACM CCS '03 (2003) 52-61
10. Zhu S. Xu, S., Setia, S., Jajodia, S.: Establishing Pair-wise Keys For Secure Communication in Ad Hoc Networks: A Probabilistic Approach. Proc. of the 11th Int'l Conf. on Network Protocols (2003) 326-335

ECGSC: Elliptic Curve Based Generalized Signcryption*

Yiliang Han[1,2], Xiaoyuan Yang[1], Ping Wei[1],
Yuming Wang[2], and Yupu Hu[2]

[1] Engineering College of Armed Police Force
Key Lab. of Computer Networks and Information Security
710086 Xi'an, China
[2] College of Communication Engineering, Xidian University
710069 Xi'an, China
yilianghan@hotmail.com

Abstract. Signcryption is a new cryptographic primitive that simultaneously fulfills both the functions of signature and encryption. The definition of *Generalized Signcryption* is proposed in the paper firstly. Generalized signcryption has a special feature that provides confidentiality or authenticity separately under the condition of specific inputs. Based on ECDSA, a signcryption scheme called ECGSC is designed. It will be equivalent to an AtE(OTP$_\$$, MAC) encryption scheme or ECDSA when one of party is absent. A third party can verify the signcryption text publicly in the method of ECDSA. Security properties are proven based on Random Oracle mode: confidentiality (CUF-CPA), unforgeability (UF-CMA) and non-repudiation. Compared with the others, ECGSC presents a 78% reduction in computational cost for typical security parameters for high level security applications.

1 Introduction

One of the essential problems for computer systems is the confidential and authenticated message delivery and storage. Traditionally, the composition of authentication and encryption is used to avoid the forgery and ensure confidentiality of the contents of a letter [1]. The method is used in some famous security protocols, such as SSL, IPSec. Unfortunately, the method is not practical for two reasons. Firstly, the cost is the sum of the authentication and encryption. Secondly, arbitrary schemes cannot guarantee the security. WEP(Wired Equivalent Privacy) is well known for its bad design.

Signcryption is a novel cryptographic primitive which achieves the combined functionality in a single step and keeps higher efficiency [2]. Most of recent researches about signcryption focus on two objects: (1) Trying to design practical signcryption schemes based on common cryptosystems [2, 3, 4, 5, 6, 7]; (2) Trying to prove the security of abstract structures that can be used in signcryption

* This work is supported by National Natural Science Foundation of China (64073037).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 956–965, 2006.

designing [1, 8, 9, 10]. Though many results were published, there is no scheme based on standard elliptic curve scheme until our work.

Not all messages require both confidentiality and integrity. Traditional signcryption will not be feasible in this case. The known solution is that signcryption is replaced by a signature or an encryption scheme. So, applications must contain at least three cryptographic primitives, which will be infeasible in some space restricted applications such as ubiquitous computing and embedded systems.

This paper is motivated by above results. Contributions lie in two aspects: (1) The definition of *Generalized Signcryption* which will provide three functions (signcryption, signature and encryption) is proposed. (2) A generalized signcryption scheme, ECGSC, which is based on standard signature scheme ECDSA, is proposed.The formal proof based on Random Oracle mode[11] for ECGSC and efficiency of ECGSC are shown also.

A signcryption scheme usually comes from a signature scheme. ECGSC is based on ECDSA (Elliptic Curve Digital Signature Algorithm). Brown has given a precise security proof [12]. It is secure against all of known attacks except for duplicate signature [13]. ECDSA is one of the most famous schemes because of its security and efficiency. It has been adapted to some signature standards: ISO 15946-2, ANSI X9.62, FIPS 186.2, IEEE1363-2000, SEC1 et al.

2 Signcryption and Generalized Signcryption

2.1 Definition of Signcryption

Signcryption is a two-party protocol like other public key schemes. Message m will be signcrypted by a sender S and sent to a specific recipient R who will designcrypt and verify. The trusted party has the right to settle sender's repudiation without recipient's private key.

Definition 1. *A signcryption scheme* $\mathbf{SC} = (Gen, SC, DSC)$ *consists of three algorithms: Gen generates a pair of keys for user* U, $(SDK_U, VEK_U) \leftarrow Gen(U, T)$, *where* T *is a security parameter,* SDK *is a secret key,* VEK *is a public key. SC is a probabilistic signcryption algorithm. For any* $m \in M, \nu \leftarrow SC(m, SDK_S, VEK_R)$ *is a signcryption text. DSC is a deterministic designcryption algorithm. For any signcryption text,* $m \cup \perp \leftarrow DSC(\nu, SDK_R, VEK_S)$, *where*$\perp$ *denotes invalid.*

Definition 2. *A signcryption scheme* $\mathbf{SC}$=(Gen, SC, DSC) *is correct only if:* $\forall S, R$ *and* $m \in M, \exists DSC(SC(m, SDK_S, VEK_R), SDK_R, VEK_S) = m$.

The first formal proof mode for signcryption was given by Baek, Steinfeld and Zheng[14]. A signcryption scheme is secure, if the following conditions are satisfied [2]: *Unforgeability*: it is computationally infeasible for an adaptive attacker, who is allowed to query Alice's signcryption algorithm, to masquerade Alice in creating an signcrypted text. *Non-repudiation*: it is computationally feasible for a third party to settle a dispute when Alice denies the fact that she is the originator of a signcrypted text with Bob as its recipient. *Confidentiality*: it is computationally infeasible for an adaptive attacker to gain any partial information on the contents of a signcrypted text.

2.2 Generalized Signcryption

In complex systems, some messages may need to be signed and encrypted, while others may need to be signed or encrypted only. Traditional signcrytion (in Definition 1) will halt because keys of one specific paty are absent. Zheng suggestted that applications may switch to other signature and encryption to solve this problem[2]. Namely, applications must implement at least three primitives: signature, encryption and signcryption. But the approach is impossible in some space restricted applications such as ubiquitous computing and embedded systems.

Generalized signcryption is a signcryption with more flexibility and practicability. So, it provides three modes: signcryption, signature-only and encryption-only. Signcryption provides double functions when confidentiality and authenticity are required simultaneously. Signature-only and encryption-only provides single signature or encryption function. Namely, a generalized signcryption scheme will be equivalent to a signature scheme or an encryption scheme in special cases without any additional computation.

We use the identity of operators to distinguish the three cases. In public key settings, performing the signature/encryption operation requires keys of specific sender/specific recipient. Performing signcryption operation requires keys of both parties. It is signcryption operation when both specific parties exist. It is signature/encryption-only when one of specific parties exists.

Definition 3. *A generalized signcryption scheme* $\mathbf{GSC}=(Gen, SC, DSC)$ *consists of three algorithms: Gen as above. SC is a probabilistic algorithm. For* $m \in M, \nu \leftarrow SC(m, SDK_S, VEK_R)$. *When* $R \in \Phi$, $SC(m, SDK_S, VEK_R) = Sig(m, SDK_S)$, $DSC(\nu, SDK_R, VEK_S) = Ver(\nu, VEK_S)$. *DSC is a deterministic algorithm. For any signcryption text,* $m \cup \perp \leftarrow DSC(\nu, SDK_R, VEK_S)$. *When* $S \in \Phi$, $SC(m, SDK_S, VEK_R) = Enc(m, VEK_R)$, $DSC(\nu, SDK_R, VEK_S) = Dec(c, SDK_R)$. *Where,* $\mathbf{ENC}=(Gen, Enc, Dec)$ *is an encryption scheme:* $c \leftarrow Enc(m, VEK_R)$, $m \leftarrow Dec(c, SDK_R)$. $\mathbf{SIG}=(Gen, Sig, Ver)$ *is a signature scheme:* $s \leftarrow Sig(m, SDK_S)$, $\{\top, \perp\} \leftarrow Ver(s, VEK_S)$. $\top$ *denotes valid.* $\perp$ *denotes invalid.*

3 ECGSC: Elliptic Curve Based Generalized Signcryption

3.1 Description of ECGSC

Message $m \in \{0,1\}^l$ will be signcrypted by a sender Alice and designcrypted by a recipient Bob. We denote Alice by A, and Bob by B.

Parameters. According to SEC1, an elliptic curve $E(Fp)$over finite field Fp is a sixtuple: $T = (p, a, b, G, n, h)$, where G is a base point, prime n is the order of G, O is the point at infinity, $[n]G = O. \forall P \in E(Fp), \exists P + O = P$. $Q = [x]G$ denotes multiple double additions on elliptic curve. $\in_R$ denotes choose an element randomly. Bin is some information about Alice and Bob. $\{0,1\}^l$ denotes a l bits binary string. Ke, Km and Ks are binary strings.

Some hash functions are used. $H : \{0,1\}^* \rightarrow Z_p^*$. $K : Z_p^* \rightarrow \{0,1\}^{Z+*}$. $LH(.) : \{0,1\}^* \rightarrow \{0,1\}^{l+z}$. $MAC_k : \{0,1\}^l \times \{0,1\}^t \rightarrow \{0,1\}^z$, is a keyed message authentication function with k, where $|k|$=t, $|m|$=l, l+$|MAC(.)|$=$|LH(x_2)|$, $|.|$indicates the number of bits in the binary representation of an integer. $H(0) \rightarrow 0$, $K(0) \rightarrow 0$, $LH(0) \rightarrow 0$, $MAC_0 \rightarrow 0$.

Keys Generation

$(d_A, Q_A) \leftarrow \text{Gen}(\text{Alice}, T)$
$d_A \in_R \{1, \cdots, n-1\}$; $Q_A = [d_A]G$;
$(0, O) \leftarrow \text{Gen}(U, T), U \in \Phi$.

$(d_A, Q_A) \leftarrow \text{Gen}(\text{Bob}, T)$
$d_B \in_R \{1, \cdots, n-1\}$; $Q_B = [d_B]G$;

Signcryption. $\text{SC}(m, d_A, Q_B)$

1. $k \in_R \{1, \cdots, n-1\}$;
2. $R \leftarrow [k]G = (x_1, y_1); r \leftarrow x_1 \bmod p$; $[k]P_B = (x_2, y_2)$;
3. $Ke \leftarrow LH(x_2); (Km, Ks) \leftarrow K(y_2)$;
4. If $d_A = 0, s \leftarrow \phi$; else $s = k^{-1}(H(m\|Bin\|Ks) + rd_A) \bmod n$;
5. $e \leftarrow MAC_{Km}(m\|s)$;
6. $c = (m\|e) \oplus Ke$; return $\nu = (c, R, s)$.

Designcryption. $\text{DSC}(\nu, d_B, Q_A)$

1. $r \leftarrow R; (x_2, y_2) = [d_B]R$;
2. $Ke \leftarrow LH(x_2)$; $(Km, Ks) \leftarrow K(y_2)$;
3. $(m\|e) \leftarrow c \oplus Ke$;
4. $e' \leftarrow MAC_{Km}(m\|s)$; If $e \neq e'$, return $\perp$; else if $s = \phi$, return m;
5. $u_1 \leftarrow s^{-1}H(m\|Bin\|Ks)$; $u_2 \leftarrow s^{-1}r$;
6. $R' = [u_1]G + [u+2]Q_A$; if $R \neq R'$, return $\perp$; else return$m$.

Verified Publicly. The trusted party will settle Alice's repudiation. The trusted party performs the following algorithm after $\nu' = (H(m\|Bin\|Ks), R, s)$ has been published by Bob.

$\text{VP}(\nu', Q_A)$

1. $u_1 \leftarrow s^{-1}H(m\|Bin\|Ks)$; $u_2 \leftarrow s^{-1}r$;
2. $R' \leftarrow [u1]G + [u2]Q_A$; if $R \neq R'$, return$\perp$; else return $\top$.

ECGSC is the unique standard verifiable scheme. Doublet $(R, s)$is an ECDSA signature on message $H(m\|Bin\|Ks)$ which can be verified in a standard mode. Bao&Deng's scheme[3] is the first publicly verifiable scheme which has two shortcomings: (1) It is based on a non-standard signature scheme; (2) Public $H(m)$ will release the partial information. Though SC-DSA[5] is based on DSA, its verification operation is not a standard one.

Signature-only Mode and Encryption-only Mode. Let the recipient $R = \phi$, ECGSC will become ECDSA scheme: $(m, R, s) \leftarrow \text{SC}(m, d_A, O)$; $\{\top, \perp\} \leftarrow \text{DSC}(\nu, 0, Q_A)$. Let the sender $S = \phi$, ECGSC will become an encryption scheme: $(c, R) \leftarrow \text{SC}(m, 0, Q_B)$; $m \cup \perp \leftarrow \text{DSC}(\nu, d_B, O)$.

3.2 Security of ECGSC

Except for security notions mentioned in section 2.1, it can define insider security and outsider security[10]. Obviously, insider security is stronger. We will give reduction proofs based on known results by using above notions.

Definition 4. (*Elliptic curve discrete logarithm problem*) *Compute* $x \leftarrow ECDLP(G, Y)$. *Where,* $Y = [x]G$, G *is a base point,* $x \in [1, \cdots, n-1]$ *and* $Y \in < G >$.

Unforgeability. In the sense of insider security, dishonest Bob is the most powerful attacker to forge a signcryption, because he has the private key d_B which is required to directly verify a signcryption from Alice. Then the problem will turn into the verification of the normal ECDSA. Brown has proved the security of ECDSA [12]: if hash function is idealized as a random oracle, then ECDSA has active existential unforgeability. Using the hypothesis that ECDSA is secure against UF-CMA, unforgeability of ECGSC will be proved based on Random Oracle mode[11].

Proposition 1. *Assume that there exists an adversary ASC that wins the UF-CMA game against ECGSC in time* t, *using* q_{sc} *queries to its signing oracle and* (q_h, q_m) *queries to its random oracles. Then there exists an algorithm AS that wins the UF-CMA game against the ECDSA signature scheme as follows*:

$$\mathbf{Adv}^{\mathbf{UF-CMA}}_{\mathbf{ASC}}(T, t, q_{sc}, q_h, q_m)$$
$$\leq \mathbf{Adv}^{\mathbf{UF-CMA}}_{\mathbf{AS}}(T, t', q_{sc}, q_h + q_{sc}) + 2q_h + q_{sc}(q_{sc} - 1)/2n$$

The algorithm AS asks q_{sc} *queries to its signing oracle and* $q_{sc} + q_h$ *queries to its random oracle. It runs in time* t'.

Proof. The following listing specifies the initial UF-CMA game against ECGSC in its entirety. Note the usage of random oracles in place of the cryptographic hash functions H and MAC. The oracles are simulated by lazy evaluation using lists L_H and L_{MAC} to maintain state between queries. A signing oracle $Oracle_{SC}$ provides the signcryption service for the input messages.

Game 0:
$(d_A, Q_A) \leftarrow \mathrm{Gen}(A, T)$; $(d_B, Q_B) \leftarrow \mathrm{Gen}(B, T)$; $Bin \leftarrow A \| B$;
$(m^*, \nu^*) \leftarrow \mathrm{ASC}(T, Q_A, Q_B, Oracle_{SC}, Oracle_H, Oracle_{MAC})$;
ASC wins, if $m^* \leftarrow \mathrm{DSC}(\nu^*, Q_A, d_B)$ and m^* is never a query to $Oracle_{SC}$.
$Oracle_{SC}(m)$
Return $\mathrm{SC}(m, d_A, Q_B)$.
$Oracle_H(m \| Bin \| Ks)$
If $(m \| Bin \| Ks, h)$ in L_H, return h; else $h \in_R \{0,1\}^{|p|}$, add $(m \| Bin \| Ks, h)$ to L_H; return h.
$Oracle_{MAC}(m)$
If (m, e) in L_{MAC}, return e; else $e \in_R \{0,1\}^z$, add (m, e) to L_{MAC}; return e.

Next, we consider the following algorithm AS, which plays the UF-CMA game against ECDSA and uses ASC as a subroutine.

Game 1: AS$(T, Q_A, Oracle_{Sign}, Oralce_H)$
$(d_B, Q_B) \leftarrow \text{Gen}(R, T)$; $Bin \leftarrow A\|B$;
$(m^*, \nu^*) \leftarrow \text{ASC}(T, Q_A, Q_B, Sim_{SC}, Sim_H, Sim_{MAC})$;
If $m^* \leftarrow \text{DSC}(Q_S, d_B, \nu^*)$ and m^* is never a query to $Oracle_{Sign}$, $(c^*, R^*, s^*) \leftarrow \nu^*$; return (m^*, R^*, s^*); else return $\perp$.

Signcryption will be forged by the assistance of signing oracle $Oracle_{Sign}$ which provides the signing service. A random oracle $Oracle_H$ with list L_H will be in place of hash function H.

$Sim_{SC}(m)$
$(r, s) \leftarrow Oracle_{Sign}(m)$; $R \leftarrow r$; $(x_2, y_2) = [d_B]R$;
$Ke \leftarrow LH(x_2)$; $(Km, Ks) \leftarrow K(y_2)$;
$e \leftarrow Sim_{MAC_{Km}}(m)$;
$c \leftarrow (m\|e) \oplus Ke$;
$h \leftarrow Oracle_H(m)$, add (m, h) to L_H;
$h' \leftarrow Oracle_H(m\|Bin\|Ks)$;
Add $(m\|Bin\|sig, h')$ to L_H;
$k \leftarrow \text{ECDLP}(T, R)$; $s' = k^{-1}(h' + ks - h) \bmod n$; $\nu \leftarrow (c, R, s')$;
Return ν.

$Sim_H(m\|Bin\|Ks)$
$m \leftarrow m\|Bin\|Ks$;
$h \leftarrow Oracle_H(m)$, add (m, h) to L_H;
$h' \leftarrow Oracle_H(m\|Bin\|Ks)$;
Add $(m\|Bin\|Ks, h')$ to L_H;
Return h'.

$Sim_{MAC}(m)$
If (m, e) in the list L_{MAC}, return e; else $e \in_R \{0,1\}^z$, add (m, e) to L_{MAC}, return e.

The only event that may cause Game 0 and Game 1 to differ, is that Sim_{SC} returns a value h whose preimage has already been assigned a value in L_H. To bound the probability of this occurring, we take into account the worst scenario that may occur: ASC asks q_h queries before querying Sim_{SC} with the same m, q_{sc} times. The total probability that no errors have occurred is $\mathbf{Prob}(\text{h}) = 2q_h + q_{sc}(q_{sc} - 1)/2n$. Where, h denotes the collision, $\neg$h denotes no collision. ASC denotes the event that ASC wins. AS denotes the event that AS wins. ECDLP denotes the event that ECDLP is solved. Thus, ECDLP implies AS. The following equation is satisfied when h does not happen:

$\mathbf{Prob}(\text{ASC} \mid \neg\text{h}) = \mathbf{Prob}(\text{ECDLP} \vee (\text{ASC} \mid \neg\text{h})) = \mathbf{Prob}(\text{AS} \mid \neg\text{h})$
So, the probability of AS wins:
$\mathbf{Prob}(\text{ASC})$
$= \mathbf{Prob}(\text{ASC} \wedge \neg\text{h}) + \mathbf{Prob}(\text{ASC} \wedge \text{h}) \leq \mathbf{Prob}(\text{ASC} \mid \neg\text{h})\mathbf{Prob}(\neg\text{h}) + \mathbf{Prob}(\text{h})$
$= \mathbf{Prob}(\text{AS} \mid \neg\text{h})\mathbf{Prob}(\neg\text{h}) + \mathbf{Prob}(\text{h}) = \mathbf{Prob}(\text{AS} \wedge \neg\text{h}) + \mathbf{Prob}(\text{h})$
$= \mathbf{Adv}_{\mathbf{AS}}^{\mathbf{UF-CMA}}(T, t', q_{sc}, q_h + q_{sc}) + 2q_h + q_{sc}(q_{sc} - 1)/2n.$

In the same time, ASC wins the UF-CMA game against ECGSC in time t, using q_{sc} queries to its signing oracle and (q_h, q_m) queries to its random oracles. The probability of ASC wins as follows:

$\mathbf{Adv}_{\mathbf{ASC}}^{\mathbf{UF-CMA}}(T, t, q_{sc}, q_h, q_m) = \mathbf{Adv}(\mathrm{ASC})$

ECGSC is secure against UF-CMA. □

Non-repudiation. As well as signature schemes, unforgeability implies non-repudiation if there is no duplication of the signcryption. If the signcryption scheme is malleable or forgeable, Alice will have an opportunity to deny. Non-repudiation of ECGSC can be achieved only if no duplicate signcryption exits because of its unforgeability. Stern, Pointcheval et al. have found that ECDSA is a duplicate signature, because the map $f : R \to r$ is not unique [21]. The two symmetrical points have the same x-coordinate: $R = (x_R, y_R), -R = (x_R, -y_R)$, so the same signature (r, s) can be got by (m_1, R, s) and $(m_2, -R, s)$. The flaw is fixed in ECGSC by using $f : R \to R$ instead of $f : R \to r$. Thus, ECGSC is not a duplicate signcryption because the map $f : R \to R$ is unique. Non-repudiation of ECGSC is achieved through verification of the triplet $(H(m\|Bin\|Ks), R, s)$. Thus, ECGSC is a non-repudiation scheme.

Confidentiality. ECGSC will be proved to be a provable secure encryption scheme. Krawczyk has proved the AtE(authenticated then encrypt) mode is CUF-CPA (chosen plaintext attacks) with the CBC (Cipher Block Chaining with a secure underlying block cipher) or OTP (One Time Padding, stream ciphers that xor data with a (pseudo) random pad)[1].

Definition 5. *An encryption scheme with ciphertext unforgeable is denoted by CUF-CPA. It is infeasible for any attacker F that has access to an encryption oracle $Oracle_E$ with key k to produce a valid ciphertext under k not generated by $Oracle_E$ as a response to one of the query by F. Namely, we quantify cipher unforgeability by function $E(q, Q, t)$ defined as the maximal probability of success for any cipher forger F that queries q plaintexts totaling Q bits and spends time t in the attack. $E(q, Q, t)$ is negligible.*

Definition 6. *The OTP encryption under $f \in F$ of plaintext x is performed by choosing $r \in_R \{0,1\}^l$ and computing $c = f(r) \oplus x$. Where $F = \{f \mid f : \{0,1\}^l \to \{0,1\}^{l'}\}$, $x \in M$. The ciphertext is the pair (r, c). If f is chosen at random and there are no repetitions in the value r, OTP schemes will be noted as $OTP_\$$. $AtE(OTP_\$, MAC)$ composition: (i) computes $t = MACk(x)$; (ii) appends t to x; (iii) output $c = f(r) \oplus (x\|t)$. Where, $MAC_k : \{0,1\}^* \times \{0,1\}^t \to \{0,1\}^n$, $|k|$=t.*

Lemma 1. *$AtE(OTP_\$, MAC)$ is secure against CUF-CPA, if message authentication function MAC is secure against IND-CMA (Indistinguishability - chosen message attacks).*

The proof of Lemma 1 can be found in [1].

Proposition 2. *ECGSC is secure against CUF-CPA.*

Proof. Defining two functions: (i) $x(R) = x_R$ denotes the operation of computing x-coordinate of a point R; (ii) $E(x) = R$ denotes the operation of embedding x into an elliptic curve as a point R.

Let $r = x(R) = x_1$ and $R = [k]G$. The value of r is random because of the same property of k. Let $f(.) = LH(x([d_B]E(.)))$. Function $f(.)$ is private and selected randomly, because d_B is private and selected randomly. While $f(r) = LH(x([d_B]E(r))) = LH(x([d_B]E(x_1))) = LH(x([d_B]R)) = LH(x_2) = Kenc$. Km is the authentic key that can be computed by both the sender and recipient. Hence, ECGSC is a composition in AtE(OTP$_\$$, MAC) manner. $H(.)$ is a secure hash function which achieves the IND-CMA security.

Then by Lemma 1, ECGSC is a CUF-CPA scheme and implements secure channels. □

3.3 Efficiency of ECGSC

In this section, ECGSC will be compared with other typical schemes which include SCS[2], Bao&Deng[3], KCDSA[4], SC-DSA[5], TBOS[6] and ECSCS[7].

Computation Cost. In public key cryptosystems, computing modular multiplication, modular exponential, modular inverse and multiples of points on elliptic curve consumes the most of computational resources, while the cost of addition, hash, encrypt or decrypt (symmetric cryptosystems) are negligible.

Remark 1. SCS is the fastest scheme in all of the four DLP based schemes(SCS, B&D, KCDSA and SC-DSA). The operation of multiple double additions on elliptic curve can be expected to be about 8 times faster than the operation of modular exponential[15]. By the results, the computation cost of keys generation operation in ECGSC is 1/8 of that in SCS; signcryption operation in ECGSC is 1/4 of that in SCS, and designcrption is 1/5 of that in SCS. ECGSC saves computational costs 78% over SCS in all.

TBOS is the only scheme based on RSA. By the result of [15], the computation cost of keys generation operation and signcryption operation in ECGSC are 1/8 of that in TBOS approximately; and designcrption is 1/5 of that in TBOS. ECGSC saves computational costs 82% in all.

ECSCS is the only known scheme based on ECDLP except for our ECGSC. The computation cost of ECGSC is slightly higher than that of ECSCS. The cost of signcrytion operation in ECGSC is 2 times of ECSCS. The cost of designcrption operation in ECGSC is 1.5 times of ECSCS.

To sum up, ECGSC has the highest speed in all of the verifiable schemes.

Communication Cost. Notes in notations: For DLP based schemes (SCS, B&D, KCDSA, SC-DSA), $|\alpha|$denotes the size of finite field, $|q|$denotes the order of base element. For RSA based schemes (TBOS), $|N|$denotes the size of public module, $|G|$denotes a hash function used in TBOS. For ECDLP based schemes (ECSCS, ECGSC), $|p|$denotes the size of finite field Fp, $|n|$denotes the order of

base point. $|D|$denotes the block length of the block cipher. $|h|$denotes the secure hash function outputs length. $|LH|$denotes the length of hash function with long message digest. $|KH|$denotes the length of keyed hash function. $|C_S|$denotes the length of all of the data that must be transfered.

Table 1. Comparison of communication cost

Schemes	$\|m\|$	$\|C_S\|$	$\|m\|/\|C_S\|1$	$\|m\|/\|C_S\|2$
SCS	$\|D\|$	$\|D\|+\|KH\|+\|q\|$	18%	26%
ECSCS	$\|D\|$	$\|D\|+\|h\|+\|n\|$	18%	26%
B&D	$\|D\|$	$\|D\|+\|h\|+\|q\|$	18%	26%
KCDSA	$\|D\|$	$\|D\|+\|h\|+\|q\|$	18%	26%
SC-DSA	$\|D\|$	$\|D\|+2\|q\|$	17%	25%
TBOS	$\|N\|+\|h$-$\|G\|$	$\|N\|$	50%	67%
ECGSC	l	$\|n\|+\|LH\|+2\|p\|$	32%	35%

Remark 2. The minimum security parameters recommended for the current practice as follows: $|\alpha|$=1024bits, $|q|$=160bits, $|N|$=1024bits, $|p|$=131bits (109 also may be chosen), $|n|$=160bits. The length of the block cipher is 64bits (e.g.IDEA). The security parameters recommended for long term security as follows: $|\alpha|$=2048bits, $|q|$=192bits, $|N|$=2048bits, $|p|$=191bits, $|n|$=192bits. The block length of block cipher is 128bits (e.g. AES). ECGSC has the highest communication cost in all of ELGamal type schemes except for TBOS.

4 Conclusion

The target of generalized signcryption is to fulfill multiple functions using a universal primitive. There are two problems concerned in generalized signcryption designing: (1) distinguishing among three cases: signcryption, signature-only and encryption-only; (2) selecting a special operation which will output specific value under specific inputs. We use ϕ to identify a nonsexist party because the algorithm will mask encryption/signature operation. The security must be investigated carefully when symmetric ciphers are used.

References

1. Krawczyk, H.: The order of encryption and authentication for protecting communications (or: How secure is SSL?). In: Kilian J. (ed.): Advances in Cryptoloty-CRYPTO'01. Lecture Notes in Computer Science, vol. 2139.pringer-Verlag, Berlin Heidelberg New York (2001) 310-331
2. Zheng,Y.: Digital signcryption or how to achieve cost (signature B.S. (eds.): Advances in Cryptoloty-CRYPTO'97, Lecture Notes in Computer Science, vol. 1294. springer-Verlag, Berlin Heidelberg New York (1997) 165-179

3. Bao, F.and Deng, R.H.: A signcryption scheme with signature directly verifiable by public key. In: Imai H., Zheng Y. (eds.): Public Key Cryptography-PKC'98, Lecture Notes in Computer Science vol.1431, springer-Verlag, Berlin Heidelberg New York (1998) 55-59
4. Yum, D.H. and Lee, P.J.: New Signcryption Schemes based on KCDSA. In: Proceedings of the 4th International Conference on Information Security and Cryptology, Seoul, South Korea, (2002) 305-317
5. Shin,J.B.,Lee,K.and Shim,K.: New DSA-Verifiable Signcryption Schemes. In: Proceedings of the 5th International Conference on Information Security and Cryptology, Seoul, South Korea, (2003) 35-47
6. Malone-Lee, J. and Mao, W.: Two birds one stone: Signcryption using RSA. In: Joye M. (ed.): Topics in Cryptology - Cryptographers'Track, RSA Conference 2003, Lecture Notes in Computer Science, vol. 2612, springer-Verlag, Berlin Heidelberg New York (2003) 210-224
7. Zheng, Y. and Imai, H.: How to construct efficient signcryption schemes on elliptic curves. Information Processing Letters, 1998, 68(5): 227-233
8. An, J.H., Dodis Y. and Rabin, T.: On the security of joint signature and encryption. In: Knudsen L. (ed.): Advances in Cryptology-EUROCRYPT'02, Lecture Notes in Computer Science, vol. 2332, springer-Verlag, Berlin Heidelberg New York (2002) 83-107
9. Dodis, Y., Rreedman, M., Jarecki, S., Jarecki, S. and Walfish S.: Versatile padding schemes for joint signature and encryption. In Pfitzmann B. (ed.): Proceedings of 11th ACM Conference on Computer and Communication Security (CCS2004), Washingtion DC, USA, (2004) 196-205
10. Dent, A. W.: Hybrid Signcryption Schemes With Insider Security. In: Proceedings of Information Security and Privacy - ACISP 2005, Brisbane, Australia, (2005) 253-266
11. Bellare, M. and Rogaway, P.: Random oracle are practical: a paradigm for designing efficient protocols. In: Proceeding of the First ACM Conference on Computer and Communication Security (CCS1993), Fairfax, Virginia, USA, (1993) 62-73
12. Brown, D.: Generic Groups, Collision Resistance, and ECDSA. Design, Codes Cryptography, 2005, 35(1): 119-152
13. Stern, J., Pointcheval, D., Malone-Lee, J. and Smart Nigel P.: Flaws in Applying Proof Methodologies to Signature Schemes. In: Yung Moti (ed.): Advances in Cryptology-Crypto'02, Lecture Notes in Computer Science, vol. 2442, Bspringer-Verlag, Berlin Heidelberg New York (2002), 93-110
14. Baek, J., Steinfeld, R. and Zheng, Y.: Formal Proofs for the Security of Signcryption. In: Naccache D. , Paillier P. (eds.): Public Key Cryptography'02, Lecture Notes in Computer Science, vol. 2274, springer-Verlag, Berlin Heidelberg New York (2002) 80-98
15. Koblitz, N., Menezes, A. and Vanstone S.: The state of elliptic curve cryptography. Designs, Codes and Cryptography, 2000, 30(19): 173-193

Effective Control of Abnormal Neighbor Discovery Congestion on IPv6 Local Area Network

Gaeil An and Jaehoon Nah

Network Security Research Division,
Electronics and Telecommunications Research Institute (ETRI),
161 Gajeong-dong, Yuseong-gu, Daejon, 305-350, Korea
{fogone, jhnah}@etri.re.kr

Abstract. Neighbor Discovery (ND) protocol is very important in ubiquitous networks because it can provide IP auto-configuration and address resolution. However, a malicious user can make access router of local area network (LAN) generate useless ND protocol messages by sending it abnormal data packets with fictitious destination IP address. If a malicious user sends the access router the enormous volume of abnormal traffic, this may result in network congestion and degrade quality of service (QoS) not only for ND-requested normal traffic, but also for ND-free normal traffic. In this paper, we propose a scheme that is able to effectively control ND congestion by rate-limiting ND protocol messages generated by abnormal data packet. In our scheme, when an access router receives a ND-requested packet, it checks if the destination IP address of the packet exists actually on the target LAN. If yes, it sends out the ND message for the packet using good QoS in packet forwarding service. Otherwise, it uses bad QoS. To learn topology of the target LAN, the router monitors all traffic from the target LAN. Through simulation, we show that our scheme can guarantee not only QoS of ND-requested data traffic, but also QoS of ND-free data traffic irrespectively of the degree of attack strength.

1 Introduction

One of the most important parts in IPv6 is ICMPv6 (Internet Control Message Protocol version 6) [1]. ICMPv6 combines functions previously subdivided among different protocols, such as ICMP (Internet Control Message Protocol version 4), IGMP (Internet Group Membership Protocol), and ARP (Address Resolution Protocol).

Neighbor Discovery (ND) protocol has been proposed as a part of ICMPv6 [2]. ND protocol is used by nodes on the same link to discover each other's presence and link-layer addresses, to find routers, and to maintain reachability information about the paths to active neighbors. ND protocol is very important in ubiquitous networks because ND protocol can provide IP auto-configuration that a node can configure its own IPv6 address automatically and also address resolution that translates IP address to link-layer address [3].

However, ND protocol has a problem that is vulnerable to attacks. For example, malicious host or router could attack any other host/routers by spoofing ND protocol messages. To protect ND protocol, Secure Neighbor Discovery (SEND) protocol has

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 966 – 976, 2006.

been proposed [4][5]. Its functions include address ownership proof mechanism, NDP message protection mechanism, replay attack prevention mechanism, and router authority certification mechanism.

Even if SEND protocol is able to protect ND protocol, it is still vulnerable to Denial of Service (DoS) attack [6] executed in terms of remotely exploitable attack [7]. An attacking node begins fabricating IP addresses with the subnet prefix of the target network and continuously sending packets to the target network. The last hop router is obligated to resolve these abnormal addresses by sending ND protocol message. The enormous number of ND protocol messages result in consuming network resources, thereby degrade the quality of service (QoS) for normal network traffic. In this paper, we call the attack Neighbor Discovery-Denial of Service (ND-DoS) attack.

In this paper, we propose a scheme that is able to effectively defeat ND-DoS attack by controlling ND protocol messages generated by abnormal data packet. In our scheme, when a router receives a ND-requested packet, it checks if the destination IP address of the packet exists actually on the target LAN. If yes, it sends out the ND message for the packet using good QoS in packet forwarding service. Otherwise, it uses bad QoS. To learn topology of the target LAN, the router monitors all traffic from the target LAN.

The rest of this paper is organized as follows. Section 2 introduces ND-DoS attack and the related study. Section 3 describes our architecture and algorithm for controlling ND-DoS attack in detail. The performance of our scheme is evaluated in section 4. Finally, conclusion is given in section 5.

2 ND-DoS Attack and Related Study

ND protocol supports neighbor discovery, router discovery, IP Auto-configuration, address resolution, neighbor unreachability detection, duplicate address detection, and redirection.

In ND protocol, there are five types of messages, Neighbor Solicitation (NS), Neighbor Advertisement (NA), Router Solicitation (RS), Router Advertisement (RA), and Redirect. NS message is used to request link-layer address of a neighbor node. The node that receives a NS message sends back a NA message giving its link-layer address. RS message is used to discover default router and to learn the network prefixes. The router that receives a RS message sends back a RA message. Redirect message is used by router to inform other nodes of a better first hop toward a destination.

The most critical problem in ND protocol is to be vulnerable to attack. To address the issue, SEND protocol is proposed. SEND protocol protects ND protocol by providing rigid security mechanisms such as IP/Router Authentication, ND message protection, and replay attack prevention. However, SEND protocol also is still vulnerable to DoS attack.

Fig. 1 shows an example of ND-DoS attack that can be performed in ND and SEND protocol. When an IPv6 router receives a data packet, if the link-layer address of the packet is founded in ARP (Address Resolution Protocol) table, then the router just forwards the packet to its destination node using its link-layer address. Otherwise,

the router stores the packet in ND buffer, sends out a NS message, and waits for a NA message. If the router receives a NA message, it looks for packets in ND buffer that corresponds to the link-layer address in the NA message, forwards them to their destination nodes. For example, in Fig. 1 when a IPv6 router receives a packet with destination IP address IP5, it stores the packet in ND buffer and performs ND protocol because it doesn't know the link-layer address of the packet. Subsequently, ND protocol sends out a NS message and after a while it receives a NA message from the IP5 node. The NA message includes the link-layer address of IP5 node. Finally, the router forwards the packet with destination IP address IP5 to IP5 node using the link-layer address of IP5 node.

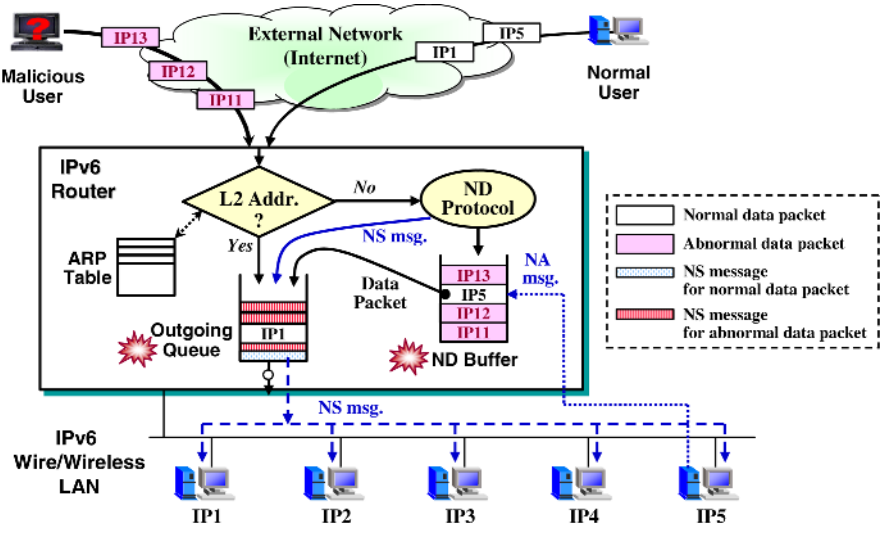


Fig. 1. ND-DoS attack on IPv6 wire/wireless LAN: IPv6 LAN in this paper may be wire or wireless. ND-DoS attack results in ND buffer congestion caused by abnormal data packets and outgoing queue congestion caused by NS messages for abnormal data packet.

The attacker can execute a ND-DoS attack on IPv6 wire/wireless LAN that bombards the router with packets with fictitious destination addresses, causing the router to busy itself by performing address resolution for non-existing destination IP addresses. For example, in Fig. 1 when the IPv6 router receives a packet with destination IP address IP11, it stores the packet in ND buffer and sends out a NS message to discovery the link-layer address of the packet. But, the router will not receive any NA message in response to the NS message because the destination IP address of the packet is a fictitious address that does not exist on the IPv6 LAN.

ND-DoS attack results in congestion at the following two points: ND Buffer and outgoing queue. Generally, an attacker brings about a great number of NS messages on the target IPv6 LAN by generating a large number of abnormal data packets with non-existing destination IP address and by sending them to the target IPv6 LAN.

A large number of abnormal data packets may make ND buffer of the router of the target network full, causing drop of normal data packets that waits for NA message. A great number of NS message may also result in network congestion on the target IPv6 LAN and cause normal data packet to experience congestion at outgoing queue. As a result, the QoS for normal data packet is degraded.

Therefore, it is required an excellent scheme that is able to protect QoS for normal packets from ND-DoS attack by managing ND buffer and outgoing queue effectively. Currently, there are several traditional algorithms for controlling buffer congestion. They can be classified according to the packet drop policy as follows [8]:

1) Tail-drop: discards the newly arrived packet.
2) Head-drop: discards the oldest packet in the buffer
3) Random-drop: discards a randomly selected packet from the buffer.

It is not likely that any of those algorithms can be used as a solution for handling ND-DoS attack effectively. The reason is that the more the number of the attack packets increases, the higher the drop probability of the normal packets goes.

There have been proposed a rate-limit scheme [9] and a compact neighbor discovery scheme [10], which are able to prevent outgoing queue congestion by reducing the volume of NS messages. Rate-limit is a scheme that limits the bandwidth for ND messages to a threshold. Rate-limit is likely to have a problem that normal packets are dropped because not only NS message for abnormal packet but also NS message for normal packet can be dropped under ND-DoS attack. This is because Rate-limit does not provide a mechanism that distinguishes normal packet from abnormal.

The compact neighbor discovery is a scheme that uses only a single NS message in stead of multiple NS messages in discovering the link-layer addresses of multiple packets. It is said that this scheme can achieve a bandwidth gain around 40 in the target IPv6 LAN. However, this scheme does not address the problem of protecting normal packets that are stored in ND buffer in case that ND buffer is full.

3 Prio-drop Scheme for Controlling Abnormal ND-Congestion

In this session, we propose prio-drop scheme, which is able to control the abnormal neighbor discovery congestion caused by ND-DoS attack. Our prio-dro scheme can protect normal traffic from abnormal ND congestion that occurs at both ND buffer and outgoing queue of router.

3.1 Packet Classification and Forwarding

In dealing with ND-DoS attack, one of the most important things is to draw a clear line between normal and abnormal packets. We pay attention to a fact that ND-DoS attacker generates abnormal packet with non-existing destination IP address. This is because ND-DoS attacker has no knowledge on topology of target IPv6 LAN.

Our strategy for defeating ND-DoS attack is very simple, but strong. Our idea is to check if the destination IP address of the incoming packet actually exists on the target LAN. If yes, the packet is regarded as normal packet. Otherwise, it is suspected as

abnormal packet. To learn topology information of the target LAN, the access router of the target LAN monitors all the packets from the target LAN.

In packet forwarding, we employ differentiated service that provides normal packet with high priority of service and abnormal packet with low priority of service. We provides abnormal packet with poor packet forwarding service instead of packet discarding. This is because there is a possibility that abnormal packet could be normal packet in reality. We believe that our strategy may weaken the false-positive problem that the packet of normal user is regarded as abnormal packet.

3.2 Architecture and Algorithms of Prio-drop Scheme

Fig. 2 shows our prio-drop architecture for controlling ND-DoS traffic, which is applied to IPv6 access router. Our architecture consists of an outgoing queue that has two kinds of priority queues, ND buffer that stores ND-requested data packet and its priority, and Animate-Node table that stores the source IP addresses of LAN nodes. In outgoing queue, high-priority queue provides much better packet forwarding service than low-priority queue. Animate-Node table can be built much faster than ARP table by monitoring and analyzing all the traffic from the target IPv6. The router stores only the 64-bit interface-ID of the source IP address of the collected packet in Animate-Host table.

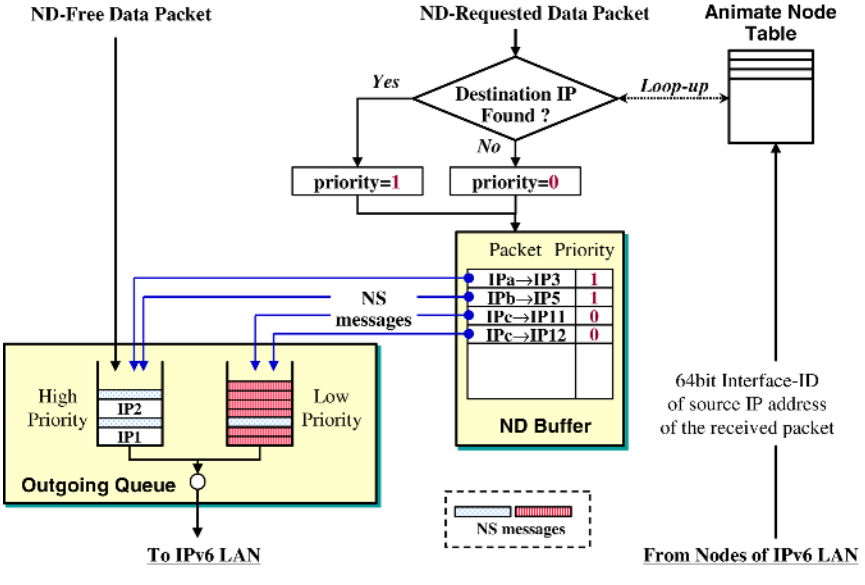


Fig. 2. Prio-drop Architecture for Controlling ND-DoS traffic in IPv6 access router: ND-free packet means that its link-layer address is found in ARP table

As shown in Fig. 2, if the access router receives a ND-free data packet of which link-layer address is found in ARP table, then it sends the packet to high-priority queue because the packet is normal. If the access router receives a ND-requested data

packet, first of all, it checks if the destination IP address of the packet is found in Animate-Node table. If yes, the packet is regarded as normal packet. Otherwise, it is regarded as abnormal packet. Normal and abnormal packets are stored in ND buffer and their priority is set to 1 and 0, respectively. Finally, NS massage for discovering the link-layer address of the incoming packet is created. NS massage for normal packet is transmitted using high-priority queue. On the other hand, NS massage for abnormal packet is transmitted using low-priority queue. Our algorithm for handling ND-requested data packet is shown in Fig. 3.

```
PROCEDURE Prio-Drop-based-Neighbor-Discovery (inPkt)
  // inPkt: the incoming packet that requests ND
  //        (Neighbor Discovery).
  create new NDentry in ND buffer;
  NDentry.pkt ← inPkt;
  IF (NDentry.pkt.dst is found in Animate Node Table)
    THEN  NDentry.pri ← HIGH;
    ELSE  NDentry.pri ← LOW;
  END IF
  // to check whether ND buffer is full
  IF (NDentry.count() > MAX_BUFFER_SIZE )
    select a victim_NDentry that includes the oldest
    packet among those with the least priority from
    ND buffer;
    IF (NDentry == victimNDentry)
      THEN  delete NDentry from ND buffer;
            return;
      ELSE  delete victimNDentry from ND buffer;
    END IF
  END IF
  create NS message for NDentry.pkt;
  IF (NDentry.pkt.pri == HIGH)
    THEN transmit NS message using high-priority queue;
    ELSE transmit NS message using low-priority queue;
  END IF
END Prio-Drop-based-Neighbor-Discovery
```

Fig. 3. Prio-Drop-based Neighbor-Discovery algorithm for handling ND-requested Packet

ND-DoS attack results in congestion in ND buffer. When ND buffer is full, our scheme discards the oldest packet among those with the least priority in ND buffer to protect normal packets from the attack, as shown in Fig. 3. For example, if ND buffer in Fig. 2 is full, then the packet with source IP address, IPc and destination IP, IP12 (IPc → IP12) is discarded because its priority is 0 and it is the oldest one in the ND buffer.

ND-DoS attack also results in congestion in the outgoing queue of the access router. In our scheme, ND-free data packet and NS message for normal packet is transmitted using high-priority queue. So, the normal packet is not influenced by network congestion that originates from ND-DoS attack because NS message generated by ND-DoS attack is transmitted using low-priority queue.

```
PROCEDURE Is-ND-DoS-Attacker (abnormalSrcIP)
  // abnormalSrcIP: the source IP address of the
  // incoming packet with priority 0.
  // Suspect List: store ND-DoS attack suspects
  // attackSuspect: an object that has three attriutes,
  // ip, stime(the starting time) and count.
  attackSuspect ← lookup_Suspect_List(abnormalSrcIP);
  IF (attackSuspect == NULL)
    THEN  create new attackSuspect;
          attackSuspect.ip    ← abnormalSrcIP;
          attackSuspect.stime ← nowtime;
          attackSuspect.count ← 1;
    ELSE
          attackSuspect.count++;
  END IF
  IF (attackSuspect.count/(nowtime-stime) > THRESHOLD)
    THEN
         // attackSuspect.ip is suspected of ND-DoS
         // attacker. All the incoming packets with
         // attackSuspect.ip may be discarded
         // for a given time.
         attackSuspect.stime ← nowtime;
         attackSuspect.count ← 1;
         return YES;
    ELSE return NO;
  END IF
END Is-ND-DoS-Attacker
```

Fig. 4. Is-ND-DoS-Attacker algorithm

Our scheme is capable of detecting the source IP address of ND-DoS attack. If a source IP address generates abnormal packets with priority 0 more than a threshold, we suspect the source IP address as ND-DoS attacker and may discard all the packets with the source IP address for a given time. For example, if the threshold is set to 1, IPc in ND buffer of Fig. 2 is suspected of ND-DoS attacker. The algorithm for checking if the source IP address of the incoming packet is ND-DoS attacker is shown in Fig.4.

4 Simulation and Performance Evaluation

In order to evaluate the performance of our scheme prio-drop, we implemented ND protocol, Animate-Node table, and ND buffer on ns-2 network simulator [11].

4.1 Simulation Environment

As the network topology for the simulation of ND-DoS attack, we use the network of Fig. 1. In Fig. 1, the malicious user and the normal user, each is connected to the external network using a link with the bandwidth of 2Mbps and the delay of 5ms. And, the IPv6 router is connected to the external network using a link with the bandwidth of 5Mbps and the delay of 2ms. The IPv6 LAN has the bandwidth of 3Mb and the delay of 20ms.

In the simulation scenario, the normal user generates ND-free UDP traffic at 0.5 Mbps and also ND-requested UDP packets at 0.2 Mbps and then sends them to the target IPv6 LAN. The malicious user generates ND-requested UDP traffic increasing by 15Kbit per second, and then sends them to the target IPv6 LAN to execute ND-DoS attack.

To evaluate the performance of our scheme, we install the existing scheme such as tail-drop, head-drop, and random-drop, and our scheme prio-drop on the IPv6 router.

4.2 Simulation Results

Fig. 5 shows the performance of ND protocol by schemes under ND-DoS attack. As shown in Fig. 5-(a), the existing schemes don't control the NS messages for abnormal data traffic. So they don't protect the NS messages for normal data traffic from ND-DoS attack as shown in Fig. 5-(b). However, our scheme, prio-drop protects the NS messages for normal data traffic by distinguishing normal packet from abnormal packet and providing the NS messages for abnormal packet with low-priority packet forwarding service, as shown in Fig. 5-(a) and 5-(b). Fig. 5-(c) shows the delay time of ND protocol for normal packet in our prio-drop. Fig. 5-(c) indicates that our scheme is not almost influenced to the degree of strength of ND-DoS attack.

Fig. 6 shows the performance of normal traffic by schemes under ND-DoS attack. Fig. 6 indicates that ND-DoS attack degrades not only the QoS of ND-requested normal traffic, but also that of ND-free normal traffic. In the simulation of ND-requested normal data traffic, the performance of the existing schemes is worse in proportion to the degree of the strength of ND-DoS attack. The tail-drop among the existing schemes is worst in the performance. Our scheme, prio-drop protects ND-requested normal data traffic almost perfectly and irrespectively of the degree of the strength of the ND-DOS attack as shown in Fig. 6-(a). Fig. 6-(b) and 6-(c) shows the performance of ND-free normal data traffic from the external network and from the IPv6 LAN, respectively. The existing schemes do not guarantee the bandwidth of the normal data traffic requested. However, our scheme, prio-drop almost perfectly guarantees the bandwidth of the normal data traffic requested.

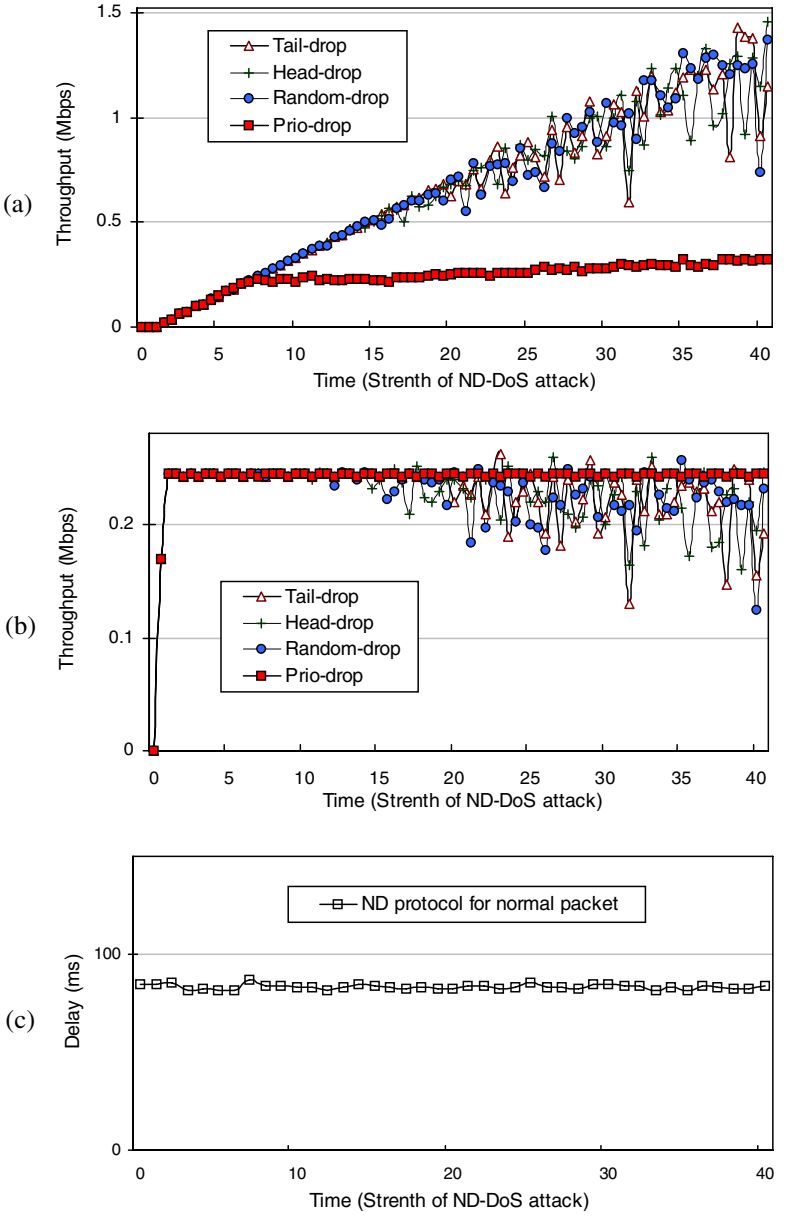


Fig. 5. Performance of ND protocol by schemes under ND-DoS attack: (a) Throughput of NS message for abnormal packet, (b) Throughput of NS message for normal packet, (c) Delay time of ND protocol for normal packet in our prio-drop

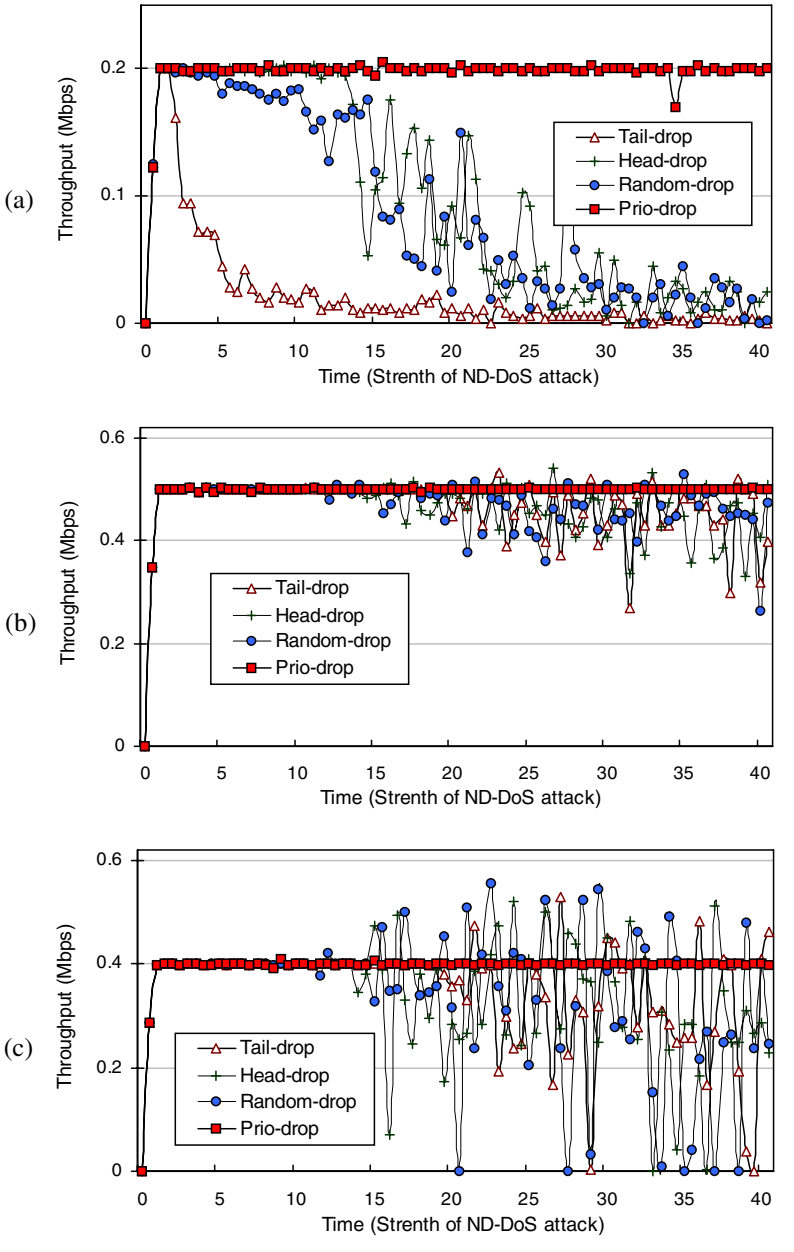


Fig. 6. Performance of normal traffic by schemes under ND-DoS attack: (a) ND-requested normal data traffic from external network, (b) ND-free normal data traffic from external network, (c) Normal data traffic from the target LAN

5 Conclusion

In this paper, we propose the prio-drop scheme that is able to effectively defeat ND-DoS attack by controlling ND protocol messages generated by abnormal data packet.

We simulated our prio-drop and the existing schemes to evaluate their performance. The simulation results show that ND-DoS attack is a very dangerous attack so as to degrade not only the QoS for ND-requested normal traffic, but also the QoS for ND-free normal traffic. And also, it shows that our scheme can protect the QoS for normal traffic from ND-DoS attack by effectively controlling NS messages for abnormal data traffic. Our future work is to implement our scheme on real IPv6 router.

References

1. A. Conta and S. Deering: Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. IETF, RFC 2463 (1998)
2. T. Narten, E. Nordmark, W. Simpson: Neighbor Discovery for IP Version 6 (IPv6). IETF, RFC 2461 (1998)
3. Yu-Chee Tseng, Jehn-Ruey Jiang, and Jih-Hsin Lee: Secure Bootstrapping and Routing in an IPv6-Based Ad Hoc Network. Proc. of ICPP Workshops (2003) 375-383
4. J. Arkko, J. Kempf, B. Zill, and P. Nikander: SEcure Neighbor Discovery (SEND). IETF RFC 3971 (2005)
5. J. Arkko, T. Aura, and et al.: Securing IPv6 Neighbor and Router Discovery. Proc. of the 3rd ACM workshop on Wireless security, (2002) 77-86
6. X. Geng and A. B. Whinston: Defeating Distributed Denial of Service Attacks. IT Pro (2000) 36-41
7. P. Nikander, J. Kempf, E. Nordmark: IPv6 Neighbor Discovery (ND) Trust Models and Threats. IETF RFC 3756 (2004)
8. S. Tanenbaum: Computer Networks, Fourth Edition. Prentice Hall (2002)
9. Cisco Systems: Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks. White paper, http://www.cisco.com/warp/…/newsflash.html, (2000)
10. Pars Mutaf and Claude Castelluccia: Compact Neighbor Discovery: a Bandwidth Defense through Bandwidth Optimization. Proc. of INFOCOM'05 (2005)
11. UCB/LBNL/VINT: ns Notes and Documentation. http://www.isi.edu/nsnam/ns

A Secure and Auto-configurable Environment for Mobile Agents in Ubiquitous Computing Scenarios*

Javier López, Antonio Maña, and Antonio Muñoz

GISUM group, Computer Science Department, E.T.S.I. Informática
University of Malaga, Spain
{jlm, amg, amunoz}@lcc.uma.es

Abstract. The increased heterogeneity and dynamism of new computing paradigms and especially of ubiquitous computing models is boosting the need for auto-configurable systems. In these new scenarios, heterogeneity and dynamism are inherent properties and applications are built by aggregating distributed information and services that are not under the control of a single entity. The current trend towards distributed computing poses important problems related to the transmission of large amounts of data between the nodes of the computing system; the control over the information; and the flexibility to adapt to heterogeneous client requirements. These characteristics, difficult to manage by traditional computing models, are making the mobile agent paradigm to gain momentum and increasing the interest of researchers and industry in this paradigm. In this paper we present a solution to provide a secure and auto-configurable environment for mobile agents in ubiquitous computing scenarios, based on two main building blocks: trusted platforms and profiles.

Keywords: Security, Agents, Profiles, Trusted Computing, Ubiquitous computing.

1 Introduction

Personalization and ubiquity are key properties for on-line services, but at the same time, they challenge the development of these systems because of the complexity of the required architectures and the security concerns they introduce. In particular, the current infrastructures for the development of personalized ubiquitous services are not flexible enough to accommodate the configuration requirements of the various application domains. To address such issues, highly configurable infrastructures are needed. An intimate relationship exists between auto-configurable systems and ubiquitous environments due to the nature of these environments, in which a device interacts with the context and adapts itself to it, performing auto-configuration.

In these new scenarios, heterogeneity and dynamism are inherent properties and applications are built by aggregating distributed information and services that are not under the control of a single entity. Furthermore, the current trend towards distributed

* Work partially supported by E.U. through projects Ubisec (IST-506926) and SERENITY (IST-027587) and by Spanish Ministry of Science and Education through research grant PR2005-0175.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 977 – 987, 2006.

computing poses important problems related to the need to transmit large amounts of data between the distributed nodes of the computing system; the control over the information; and the flexibility to adapt to heterogeneous client requirements. These characteristics are difficult to manage by traditional computing models. For these reasons, the mobile agent paradigm is gaining momentum and the interest of researchers and industry in this paradigm is increasing.

The mobile agent paradigm uses the network to carry software objects that are to be executed in the sites of service providers. A client orchestrates the work of a server by sending to the server an agent that is responsible for performing all of the required actions related to the services and data offered by the server. For instance, consider the case of a digital library server offering several Tb. of information. In order to allow the remote processing of this information we can find different schemes:

- *The server offers only basic services in order to access the data*. In this case clients need to download the data and to process it locally. This is the scheme of many traditional client-server systems. This scheme reduces the processing power required in the server, but has three main drawbacks: (i) the system is very inefficient due to the overhead caused by data transmission and redundant storage; (ii) the server looses control over the information; and (iii) in cases of very dynamic information sources, the transmission delays can cause problems of synchronization between the copies of the data in the client and the server. This is especially true when clients can modify the data because, in this case, there are important problems caused by concurrent access by many clients.
- *The server offers advanced information processing services*. This scheme solves the problem of data transmission and redundancy, as the data is processed locally by the server. However, the flexibility of the system is limited, because it is very difficult for the server to foresee all processed that may become necessary for the clients. The main problems arise when clients need to access many heterogeneous information servers, because it is difficult that all servers offer the same services with the same access mechanisms. This is the approach followed in service-oriented computing. Clients need to download only the partial results produced by the server, in order to compute locally the final results. For this reason, the control of the server has more over the information than the previous case, although not complete.
- *The server offers the possibility of sending agents to perform the data processing*. In this case, the server offers only basic access services, which are easier to manage from the points of view of efficiency and security. The flexibility is very good because agents are then responsible for implementing the complex data processing required by the client. Only results need to be sent to the client. The control of the server over the information is complete.

The agent paradigm can represent a valuable model for the interaction of applications and devices in ubiquitous environments. Each element in the ubiquitous scenario can act both as client (sending agents to other elements) and as server (allowing other elements to send agents to it).

The main motivation of our work is to define a secure and adaptable execution environment for mobile agents based on the use of profiles and taking advantage of the new Trusted Computing architectures. We define a profile as a repository

materialized as a structured data object, containing properties and features, as well as present and past status about an entity, in our case an agent. Profiles are normally used to convey the properties of an entity to other entities, not being intended as information storage for internal use by the entity itself. Therefore, profiles are conceived with the objective of being shared with others. Our approach includes mechanisms for the secure management of profiles, which are key elements for the auto-configuration of the hosts for the execution of the mobile agents.

We use mechanisms for guaranteeing that execution environments (a.k.a. agencies) provided by the hosts are trustworthy. This is achieved by using the mechanisms provided by hardware devices known as Trusted Platform Modules (TPMs). TPMs are the core elements of the secure computing model defined by the Trusted Computing Group (TCG) [1]. We also use the remote attestation capability provided by the TCG model. In the combination of these two central elements, we highlight the intrinsic auto-configurability of agent systems and we enhance it by using profiles.

The rest of the paper is structured as follows. Section 2 provides an overview of relevant related work. Section 3 describes the proposed solutions for achieving our objectives. Finally, section 4 presents conclusions and describes some ongoing work.

2 Background and Related Work

The growing interest in ubiquity, dynamic adaptability, profiling and auto-configuration is evident in the literature [2-5]. In the emerging computing paradigms, this interest is augmented by the need to provide dynamic autonomous responses to ever-changing contexts and unforeseen interactions.

Profiles have been used for capturing information about users and preferences. Some authors describe profiling as "the process of inferring a set of characteristics (typically behavioural) about an entity and then treating that entity (or other entities) in the light of these characteristics" [6]. Based on this definition Pearson describes a method based on the use of trusted agents for self-profiling in e-commerce scenarios by which customers can have greater control over their related profiles [7]. However, the objective of our profiles is to inform agencies about the needs of the agents and at the same time to provide tools for agencies to control and monitor the behaviour of agents. The real value of profiles depends on the accuracy of the information they contain. Therefore, the protection of these profiles is an important aspect to consider.

Regarding the processing and description of profiles, RDF [8] provides a way to define a generic data model so facilitating a multi purpose mechanism to describe resources. Another approach, CC/PP [9] proposes a framework for the management of information about devices capabilities and user preferences. This framework, based on the RDF approach, has proved useful for content customization. UAProf [10] provides a solution to define a specific vocabulary concerning device information. Finally, FIPA [11] defines device ontologies for the communication of devices.

Many proposals use profiles to characterize users' behaviour while they browse through the Internet [12]. The information obtained is used to construct a transient navigation profile, which is useful to anticipate future actions of the user. Reference [13] follows the same approach, using Bayesian networks as a tool for creating profiles of visitors in a museum in order to customize the information.

2.1 Security in Agent-Based Systems

The inherent complexity of information security is increased in agent-based ubiquitous systems. In fact, securing these systems requires protecting any element from every other. Some of the general software protection mechanisms can be applied to the protection of agents. However, the specific characteristics of agents mandate the use of tailored solutions. First, agents are most frequently executed in potentially malicious platforms. Then, from the point of view of platforms, agents are potentially malicious pieces of software. Therefore, we can not simplify the problem as is done in other scenarios by assuming that some elements of the system can be trusted.

Then, the security of an agent system can be defined in terms of many different properties such as confidentiality, non repudiation, etc. but it always depends on ensuring the correct execution of the agent on agent servers (a.k.a. agencies) within the context of the global environments provided by the servers [14].

Finally, conflict management, communication, intelligence and negotiation are important components of collaborative multi-agent activity. Thus, a collaborative agent must be able to handle situations in which conflicts arise and must be capable of negotiating with other agents in order to fulfil its goals. These capabilities are especially relevant for the security of the agent.

Several mechanisms for secure execution of agents have been proposed in the literature with the objective of securing the execution of agents. Most of these mechanisms are designed to provide some type of protection or some specific security property. In this section we will focus on solutions that are specifically tailored or especially well-suited for agent scenarios. More extensive reviews of the state of the art in general issues of software protection can be found in [15, 16].

Some protection mechanisms are oriented to the *protection of the host system against malicious agents*. Among these, SandBoxing is a popular technique that is based on the creation of a secure execution environment for non trusted software. In the agent world a sandbox is a container that limits, or reduces, the level of access its agents have and provides mechanisms to control the interaction among them.

Another technique, called proof-carrying code, is a general mechanism for verifying that the agent code can be executed in the host system in a secure way [17]. For this purpose, every code fragment includes a detailed proof that can be used to determine whether the security policy of the host is satisfied by the agent. Therefore, hosts just need to verify that the proof is correct (i.e. it corresponds to the code) and that it is compatible with the local security policy. In a variant of this technique, called proof-referencing code, the agents do not contain the proof, but just a reference to it [18]. These techniques share some similarities with the constraint programming technique; they are based on explicitly declaring what operations the software can or can not perform. One of the most important problems of these techniques is the difficulty of identifying which operations (or sequences of them) can be permitted without compromising the local security policy.

Other mechanisms are oriented towards *protecting agents against malicious servers*. Sanctuaries [19] are execution environments where a mobile agent can be securely executed. Most of these proposals are built with the assumption that the platform where the sanctuary is implemented is secure. Unfortunately, this assumption is not applicable in our scenario. Several techniques can be applied to an

agent in order to verify self-integrity in order to avoid that the code or the data of the agent is inadvertently manipulated. Anti-tamper techniques, such as encryption, checksumming, anti-debugging, anti-emulation and some others [20, 21] share the same goal, but they are also oriented towards the prevention of the analysis of the function that the agent implements. Additionally, some protection schemes are based on self-modifying code, and code obfuscation [22]. In agent systems, these techniques exploit the reduced execution time of the agent in each platform.

Software watermarking techniques [23, 16] are also interesting. In this case the purpose of protection is not to avoid the analysis or modification but to enable the detection of such modification. The relation between all these techniques is strong. In fact, it has been demonstrated that neither perfect obfuscation nor perfect watermark exists [24]. All of these techniques provide short-term protection; therefore, in general they are not applicable for our purposes. However, in some scenarios, they can represent a suitable solution, especially, when combined with other approaches.

Many proposals are based on checks. In these systems the software includes software and hardware-based "checks" to test whether certain conditions are met. However, because the validation function is included in the software, it can be discovered using reverse engineering and other techniques. This is particularly relevant in the case of agents. Theoretic approaches to the problem have demonstrated that self-protection of the software is unfeasible [25].

In some scenarios, the protection required is limited to some parts of the software (code or data). In this way, the function performed by the software, or the data processed, must be hidden from the host where the software is running. Some of these techniques require an external offline processing step in order to obtain the desired results. Among these schemes, function hiding techniques allow the evaluation of encrypted functions [26]. This technique protects the data processed and the function performed. For this reason it is an appropriate technique for protecting agents. However, it can only be applied to the protection of polynomial functions.

The case of online collaboration schemes is also interesting. In these schemes, part of the functionality of the software is executed in one or more external computers. The security of this approach depends on the impossibility for each part to identify the function performed by the others. This approach is very appropriate for distributed computing architectures such as agent-based systems or grid computing, but has the important disadvantage of the impossibility of its application to off-line environments

Finally there are techniques that create a *two-way protection*. Some of these are hardware-based, such as the Trusted Computing Platform. With the recent appearance of ubiquitous computing, the need for a secure platform has become more evident. Therefore, this approach adds a trusted component to the computing platform, usually built-in hardware used to create a foundation of trust for software processes [27].

3 Secure Execution of Agents in Ubiquitous Computing Scenarios

The main goal of this paper is to provide a secure and auto-configurable environment for mobile agents in ubiquitous computing scenarios. In order to achieve this goal we will base our approach on two main building blocks: trusted platforms and profiles.

On the one hand, in order to enhance the security of the execution environment our approach uses the concept of Trusted Platform. Because we are focusing on ubiquitous scenarios where we consider every device to be able to act as agency, and where the interaction with other previously unknown devices and applications will be the frequent, the security must be based on mechanisms that allow one party to verify the trustworthiness of the others. The idea behind the Trusted Computing paradigm was introduced in 1997 by Arbaugh, Farber and Smith [28]. The technology currently known as Trusted Computing has been developed by the Trusted Computing Group (TCG) on the basis of the specifications developed by the Trusted Computing Platform Alliance (TCPA). According to the TCG documentation, "the distinguishing feature of TCG technology is arguably the incorporation of 'roots of trust' into computer platforms." The TCG technology is not only for personal computers. In fact, it can be applied to most computing devices, such as PDAs, mobile phones, etc. The basic idea in our approach is to use the services provided by the TCG architecture in order to allow agents to verify that the agencies where they will be executed are trustworthy. In particular, we will check that the agencies run on top of trusted hardware and software configurations and that they have not been tampered with.

On the other hand, in order to facilitate auto configuration of the agencies, we will use secure profiles (i) for informing the agencies about the security requirements of the agents; (ii) for facilitating auto-configuration of the agency; and (iii) for supporting advanced monitoring of the behaviour of the agent.

3.1 Trusted Computing Support

The basic idea behind the concept of Trusted Computing is the creation of a chain of trust between all elements in the computing system, starting from the most basic ones. Therefore, the chain starts with a tamperproof hardware device, known as *Trusted Platform Module* (TPM), which analyses the BIOS of the computer and, in case it is recognized as trusted, passes control to it. This process is repeated for the master boot record, the OS loader, the OS, the hardware devices and finally the applications. This process can be tailored to suit the boot sequences of other devices such as routers, PDAs, mobile phones, etc. In Trusted Computing scenarios trusted applications run exclusively on top of protected and pre-approved supporting software and hardware.

One of the features of the Trusted Computing model is that it makes possible for both the platform owner and arbitrary third parties to obtain evidence about the integrity and configuration of a platform by measuring the platform components and comparing such measures to predefined values. The TPM component of the TCG architecture can also securely store secrets such as cryptographic keys and platform configuration measures in special shielded memory locations known as *Platform Configuration Registers* (PCRs). Additionally, TPMs can perform security relevant operations such as encryption and production of digital signatures. The process of obtaining metrics of those platform characteristics that affect its security and dependability; and storing and putting digests of those metrics in shielded locations is known as *integrity measurement*. *Integrity reporting* is the process of attesting to the contents of integrity storage. This is done by digitally signing specific internal TPM data using an *Attestation Identity Key* (AIK). These features are designed to allow platforms to enter any state, including those not identified as secure, but to prevent

that a platform can lie about states that is was or was not in. An independent process may evaluate the integrity state(s) and determine appropriate responses.

Remote attestation is another interesting feature. In the typical scenario where Alice and Bob are communicating, Alice can take advantage of the remote attestation feature in order to determine whether the current configuration of Bob's platform is safe. This is possible for Alice because the Trusted Computing technology provides mechanisms for her to measure (obtain a cryptographic hash) of the configuration of Bob's platform. If this configuration is altered or modified, a new hash value must be generated and sent to Alice in a certificate. These certificates attest the current state of Bob's platform and allow her to accept or reject the communication.

In particular, in our scenario we use this remote attestation mechanism between the TPMs of the platforms where each agency runs in order to verify that the agency software has not been tampered with and that it is running over appropriate software, firmware, and hardware configurations. The agency where the agent is currently running (source agency) is responsible for using remote attestation procedures in order to verify that the next agency in the agent itinerary (destination agency) is also trustworthy. In this way, assuming that the agent is started in a trusted agency (the home agency), we can be sure that the agent runs only on trusted agencies. Attestation is carried out by the TPM of the source agency in order to ensure that destination agencies provide secure and dependable execution environments for the agent.

Fig. 1 shows an interaction diagram that illustrates how the verification of the destination agency is done with the help of the TPMs of both agencies. For the sake of simplicity we include only two agencies. It is straightforward to extend this process for multiple agencies in the case of multi-hop agents. The process starts when the source agency (*A1*) receives from an agent (*ag1*) a request to verify the configuration of the destination agency (*A2*) according to some requirements (*A2req*). This explicit notification is necessary because the verification is done by the source agency with the help of the local TPM (*tpm1*). There are some cases in which agents can carry out this process without the intervention of the source agency. However, this second alternative requires the agents to be able to access the services of the local TPM directly, which is not likely to be allowed.

Then, the source agency TPM uses the remote attestation mechanism in order to obtain the configuration of the destination agency (*A2conf*). This process requires the collaboration of the destination agency TPM (*tpm2*). If the required configuration is successfully verified, the agent can safely migrate to the destination agency. In this scenario we assume the simplest case where the requirements from the agent can be directly compared by the TPM (e.g. requirements are expressed in the form of TPM measurements). In some cases, it is possible that the process of checking the conformance between the agent requirements and the destination agency configuration requires some more complex processing. In these cases, we foresee that the source agency will be responsible for carrying out such process. An example of such process is the so called *semantic attestation*, which provides enhanced flexibility to the attestation mechanism at the cost of more complex processing.

In summary this scheme provides guarantees of the security and dependability of the destination agency before the agent runs on it. The scheme is simple and efficient and can be successfully applied to multi-hop agents.

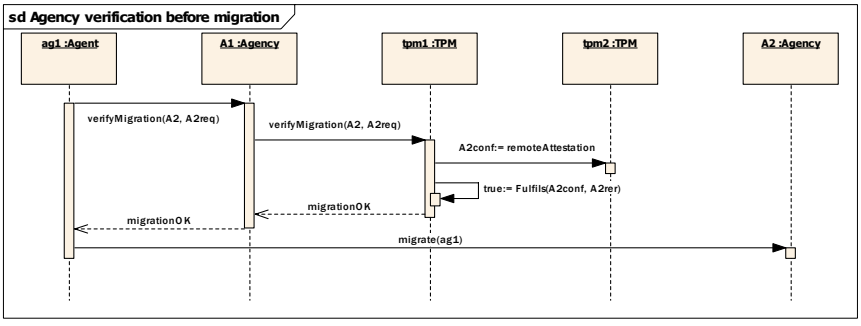


Fig. 1. Verification of the destination agency before migration

3.2 Profiles and Auto-configuration

In the approach proposed by Ghosh et al [29], profiling the behaviour of a program is performed through the study of the set of system calls of the program. We use the same idea, though with agents. The main idea is to make the profile contain a declaration of the task that the agent intends to perform, following an approach analogous to the proof-carrying code [17]. However, we propose that, additionally to this information, the profile contains information concerning the agencies visited together with the set of relevant operations executed in each agency. This information will be added in the profile dynamically by each agency. This is noteworthy for the case when an agent can contain malicious code, a code that can not be detected by agent execution in only one agency but a malicious code segmented and executed in the different agencies visited by the agent. Thus, detection needs a global supervision.

Fig. 2 shows how the complete agent migration process is performed, from a source agency (*A1*) to a destination agency (*A2*). In this case the interaction includes the transmission of the agent profile and the auto-configuration of the destination agency. In the first step, the agent (*ag1*) sends a request for migration to the source agency (*A1*). As in the previous example, it is necessary to specify the destination agency (*A2*) and the configuration requirements (*A2req*). Then, the source agency sends a message requesting the verification of the destination agency to the local TPM (*tpm1*). This one will carry out a remote attestation with the remote TPM (*tpm2*). In case this attestation is successful, and the destination host configuration fulfils the agent requirements, the local TPM requests the public key from *tpm2*. This public key is used by the encryption service of the local TPM to encrypt the agent profile (*ag1.profile*). Then, source agency sends the encrypted profile to destination agency. Finally, the encrypted profile is decrypted by the TPM of the remote agency.

At this point the destination agency verifies the agent profile with regards to its local policy. In case the profile is accepted, the destination agency auto-configures itself in order to receive and execute the agent and notifies this circumstance to the source agency. Finally, the agent can migrate to the destination agency.

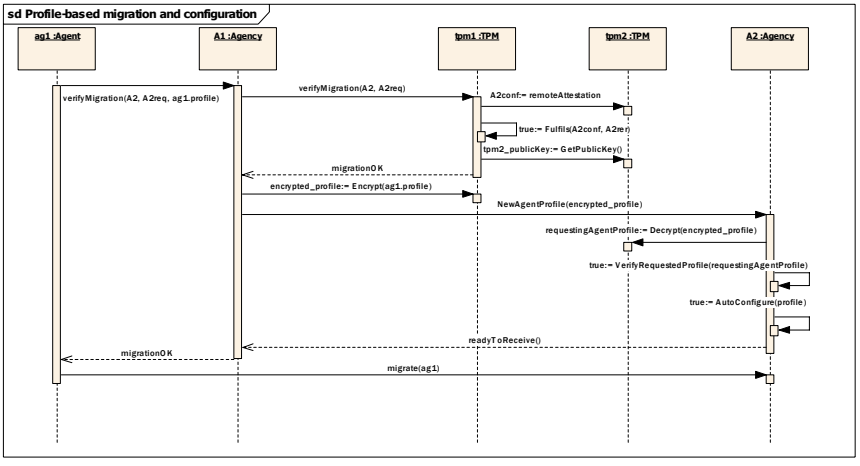


Fig. 2. Auto-configuration of the destination agency based on the agent profile

Once the fundamentals for providing a secure run environment for mobile agents has been described in detail, we follow with the description of their self-configurability. As mentioned, we will use the agent profile. To better understand what an agent means, Fig. 3 shows an example of an agent profile in XML format. In this example we have structured the information of the agent in five main blocks. Firstly, we have the information related to general requirements of the agents, like amount of memory required, the type of communication, etc. Secondly, we have established the security requirements for the execution of the agent. In the example we have information like the cryptographic mechanisms used, the access to TPM, and the certification of the platform. Then, we have established another section with the platform configuration where we can indicate if the agent requires the agency to run in trusted mode or not. The next section contains the dynamic part of the profile. Each agency visited will include new information on this section. We have highlighted the monitoring information, as it provides interesting features to our scheme. As it can be seen, the final section is left open for any extension required for specific cases.

Coming back to the monitoring, this scheme is interesting because it allows agencies to include information inside the agent profile, so that information could be used for later supervision tasks. This would be useful in the cases when an agent performs something suspicious but not detectable in the partial execution of an agent in an agency. This mechanism introduces the possibility of supervising the global behaviour of the agent by all agencies it passes through.

Another interesting aspect is that the profile can be certified by the agency with the help of the TPM. But, moreover, the profile could include a history of all agencies the agent passed by, and these could certify its operation. This scheme takes advantage of the Trusted Computing model as well as the possibilities of rich profiling, providing a secure execution environment for agents. Besides, we allow a self-configuration of the agent execution environments (agencies) prior to its arrival to the host.

```
<?xml version="1.0" encoding="UTF-8" ?>
<agent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:noNamespaceSchemaLocation="D:\informacion\agentProfile.xsd">
   <generalReq>
      <memoryReq>128kb</memoryReq>
      <communication>none</communication>
<!-- more general requirements -->
   </generalReq>
   <securityReq>
      <cryptoMechanisms>
         <cryptoMechanism>
            <type>AsymmetricEncryption</type>
            <algorithm>RSA</algorithm>
            <keyLength>512</keyLength>
         </cryptoMechanism>
      </cryptoMechanisms>
      <tpmAccess>Yes</tpmAccess>
      <platformCert>Yes</platformCert>
<!-- more security requirements -->
   </securityReq>
   <platformConf>
      <trustedMode>
         <required>Yes</required>
         <confID>'71386cb5c2ed63f855d2533cc264bc2e'</confID>
      </trustedMode>
<!-- more platform configuration values -->
   </platformConf>
<!- dynamic part of the profile containing
    platform-generated information -->
   <platformInfo>
      <monitoring>
         <visit>
            <host url="http://www.agentHome.org">
            </host>
            <actions>
               <!-monitored actions -->
            </actions>
         </visit>
      </monitoring>
<!-- more platform-generated information -->
   </platformInfo>
   <extensions>
<!-for scenario-specific information -->
   </extensions>
</agent>
```

Fig. 3. Example of agent profile

4 Conclusions and Ongoing Work

Our main motivation is to provide a secure and auto-configurable environment for mobile agents in ubiquitous computing scenarios. In order to achieve this goal we have based our approach on two main building blocks: trusted platforms and profiles.

We have described how these two elements contribute to the proposed solution and have discussed some alternatives. We have also reviewed the advantages of the proposed approach and have shown how additional features are enabled by the use of this scheme. Among these additional features we must highlight the enhanced support for supra-agency monitoring. The importance of this feature is that it enables the detection of attacks that can not be identified by analyzing the actions performed in just one agency. Additionally we have illustrated the migration processes.

We are currently working on the implementation of agencies over platforms containing TPMs. We are also working on the FIPA-OS specifications in order to accommodate the new functionalities required by our system (e.g. access to the TPM).

Finally, we are also working on complementary protection mechanisms for the agents that do not involve the use of TCG technology [30].

References

1. Trusted Computing Group: TCG Specifications. 2005. Available online at https://www.trustedcomputinggroup.org/specs/
2. Resnick, P. and Varian, H., Eds.: Communications of the ACM: Special Issue on Recommender Systems 46. 1997.
3. Riecken, D., Ed.: Commun. ACM: Special Issue on Personalization 43. 2000.
4. Maybury, M., Ed.: Commun. ACM: Special Issue on News on Demand 43. 2000.
5. Maybury, M. and Brusilovsky, P., Eds.: Commun. ACM: The Adaptive Web 45. 2002.
6. Bygrave, L.: Electronic Agents and Privacy: A Cyberspace Odyssey 2001, Intl. Journal of Law and Information Technology, vol 9, no 3, p 280, Oxford University Press, 2001.
7. Pearson, S.: Trusted Agents that Enhance User Privacy by Self-Profiling. Proceedings of the AAMAS Workshop (Special track on privacy). 2002.
8. W3C: Resource Description Framework (RDF): Concepts and Abstract Syntax. 2004.

9. W3C: CC/PP: Structure and Vocabularies 1.0 January 2004. Available online at http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html
10. Wireless Application Protocol Forum: Wireless Application Group User Agent Profile Specification. Nov. 1999.
11. Foundation for Intelligent Physical Agents: FIPA Device Ontology Specification, December 2002. Available online www.fipa.org.
12. Chi, E.H.: Transient User Profiling. Proceedings of the workshop on User Profiling. 2004.
13. Sparacino, F.: Sto(ry)chastics: a Bayesian Network Architecture for User Modelling and Computational Storytelling for Interactive Spaces. Proceedings of the Fifth International Conference on Ubiquitous Computing. 2003.
14. Berkovits S, Guttman J, Swarup V.: Authentication for Mobile Agents. In Mobile Agents and Security volume 1419, pages 114-136. Springer-Verlag. 1998.
15. Maña, A.: Protección de Software Basada en Tarjetas Inteligentes. PhD Thesis. University of Málaga. 2003.
16. Hachez, G.: A Comparative Study of Software Protection Tools Suited for E-Commerce with Contributions to Software Watermarking and Smart Cards. PhD Thesis. Universite Catholique de Louvain. 2003.
17. Necula G.: Proof-Carrying Code. Proceedings of 24th Annual Symposium on Principles of Programming Languages. 1997.
18. Gunter Carl A., Homeier Peter, Nettles Scott.: Infrastructure for Proof-Referencing Code. Proceedings of the Workshop on Foundations of Secure Mobile Code. March 1997.
19. Yee, Bennet S.: A Sanctuary for Mobile Agents. Secure Internet Programming. 1999.
20. Schaumüller-Bichl1, I., Piller, E.: A Method of Software Protection Based on the Use of Smart Cards and Cryptographic Techniques. Proceedings of Eurocrypt'84. Springer-Verlag. LNCS 0209, pp. 446-454. 1984.
21. Stern, J. P., Hachez, G., Koeune, F., Quisquater, J. J.: Robust Object Watermarking: Application to Code. Proceedings of Info Hiding '99, Springer-Verlag. LNCS 1768, pp. 368-378. 1999.
22. Collberg, C., Thomborson, C.: Watermarking, Tamper-Proofing, and Obfuscation - Tools for Software Protection. University of Auckland Technical Report #170. 2000.
23. Wayner, P.: Dissapearing Cryptography. Information Hiding, Stenography and Watermarking. Morgan Kauffman. 2002.
24. Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S., Yang, K.: On the (Im)possibility of Obfuscating Programs. Proceedings of CRYPTO `01. Springer-Verlag. LNCS 2139. pp. 1-18. 2001.
25. Goldreich, O.: Towards a theory of software protection. Proceedings of the 19th Ann. ACM Symposium on Theory of Computing, pp. 182-194. 1987.
26. Sander, T., Tschudin C.F.: On Software Protection via Function Hiding. Proceedings of Information Hiding '98. Springer-Verlag. LNCS 1525. pp 111-123. 1998.
27. Pearson, S., Balacheff, B., Chen, L., Plaquin, D., Proudler. G.: Trusted Computer Platforms. Prentice Hall. 2003.
28. Arbaugh W., Farber D., Smith, J.: A Secure and Reliable Bootstrap Architecture. Proceedings of the 1997 IEEE Symposium on Security and Privacy, pp 65-71. 1997.
29. Ghosh, A., Schwartbard, A., Schatz M.: Learning program behavior profiles for intrusion detection. Proceedings of the Workshop on Intrusion Detection and Network Monitoring, Usenix. 1999.
30. Maña, A., Muñoz, A.: Mutual Protection for Multiagent Systems. Proceedings of the Third International Workshop on Safety and Security in Multiagent Systems (SASEMAS '06). 2006.

Connectivity Preservation and Key Distribution in Wireless Sensor Networks Using Multi-deployment Scheme*

David Simplot-Ryl[2] and Isabelle Simplot-Ryl[1]

[1] IRCICA/LIFL, Univ. Lille 1, CNRS UMR 8022
[2] INRIA Futurs, POPS research group
Bât. M3, Cité Scientifique
59655 Villeneuve d'Ascq Cedex, France

Abstract. Secure key establishment in wireless sensor networks has been shown to be efficient. For sake of computation power, predistribution of symmetric keys is recommended. In such schemes, a set S of keys is used and each deployed node knows a subset of S. However, the capture of nodes can quickly lead to the corruption of a significant part of the network and solutions like multi-deployment have been proposed. In multi-deployment schemes, sets of nodes – using different key sets – are sequentially deployed that limits the impact of the capture of nodes. In this paper, we consider the problem of connectivity of single deployment in regards to the size of key sets. We show that connectivity guaranteed solutions lead to network vulnerability when nodes are captured. Then we evaluate network robustness of different schemes.

1 Introduction

Recent advances in micro-electro-mechanical systems (MEMS), digital electronics, and wireless communications have enabled the development of low-cost, low-power, and multifunctional sensor devices. These nodes are autonomous devices with integrated sensing, processing, and communication capabilities. Sensor networks consist of a large number of sensor nodes that collaborate together using wireless communication and asymmetric many-to-one data flow [1]. Indeed, sensor nodes usually send their data to a specific node called the sink node or monitoring station, which collects the requested information. All nodes cannot communicate directly with the monitoring station, since such communication may be over long distance that will drain the power quickly. Hence, sensors operate in a self-organized and decentralized manner and message communication takes place via multi-hop spreading. To enable this, the network must maintain the best connectivity as long as it is possible. Sensor's battery is not replacable, and sensors may operate in hostile or remote environments. Therefore energy

* This work was partially supported by a grant from CPER Nord-Pas-de-Calais/FEDER TAC MOSAÏQUE, French National Research Agency RNRT SVP, and CNRS National platform RECAP.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 988–997, 2006.

consumption is considered as the most important resource, and the network must be self-configured and self-organized. The best energy conservation method is to put as many as possible sensors to sleep. The network must be connected to remain functional, so that the monitoring station may receive message sent by any of the active sensors. An inteliggent strategy for selecting and updating a set of active sensors that are connected is needed in order to extend the network lifetime. The network topology therefore changes frequently. Sensor networks are applied in military, environmental, health, home, and other environments. For instance, sensor networks can be used for battlefield surveillance, monitoring equipment, chemical attack detection, doctors and patients tracking and monitoring, for disaster prevention and monitoring (forest fire, flood), in air, water, or soil pollution surveillance, environmental control in office buildings, machines diagnosis, smart environment.

In such networks, security becomes an important issue to guarantee integrity and confidentiality of transported data. Because of node capabilities, traditional methods like public key cryptography cannot be used. Security relies on key management where favorite techniques is to use probabilistic key-predistribution [2]. In such schemes, before deployment, each sensor is initialized with a random subset of keys from a key pool. Hence, two nodes can securely communicate by using a common key with a given probability. Such techniques are evaluated by mesuring their robustness against node captures. The robustness is assimilated to the number of non-compromised links. In order to increase the network livespan, Durresi et al. proposed to used a multi-deployment scheme where each deployment uses a different key pool [3]. Nodes from previous deployments can be reused by the introduction of stronger nodes called bridges. In this paper, we consider the problem of connectivity of networks using key pre-distribution. First, we consider the problem of connectivity of single deployment in regards to the size of key sets. We show that connectivity guaranteed solutions lead to network vulnerability when nodes are captured.

The remaining of the paper is organized as follows. First, we give an overview of existing solutions and related works. Then, in Section 3, we introduce notations, basical definitions and a study of key pre-distribution and induced communication graphs. In the next section, we present an evaluation of network robustness for three schemes and we introduce key deletion which significantly decreases the number of corrupted links. The last section contains conclusion and future works of this study.

2 Related Works

Wireless environments have introduced new security problem due to the typical lack of central authority or administrator, and the low reliability of the communication medium. A survey of security problems in ad hoc networks is presented in [4]. Major problems are for example authentication, or cooperation [5,6]. Sensor networks have similar problems since for example an adversary can also hear communications. Some issues are nevertheless different and results obtained in

general ad hoc networks cannot be directly applied. One of the major difference is that all the network nodes belongs to the same authority, thus one can *a priori* rely on the identity of nodes and on their cooperation. So the major security problems in distributed sensor networks are the fault injection and the pervasive listening.

To ensure communication secrecy, communications are encrypted. Asymmetric cryptography, used in ad hoc networks like for example in [7], is often to heavy for sensor strong resources constraints. As all the nodes of a distributed sensor network belong to the same authority and are initialized before deployment, most of the solutions proposed in the literature to secure communication use symmetric cryptography.

The problem is that nodes are potentially deployed in hostile environments and can be physically captured, thus keys known by a given node can potentially be revealed to an adversary. To address this problem, works on key predistribution propose scheme to load nodes with different subsets of a set of keys before deployment to avoid the whole network to be compromised by a small-scale attack. If deployment information are available, key pre-distribution can be optimized [8]. But the nature of distributed sensor networks makes it most of the time impossible to decide before deployment which nodes will be neighbors, thus pre-distribution schemes must ensure that nodes will be able to communicate despite of a particular deployment. A distributed key establishment mechanism has been proposed in [2], that relies on the probability for nodes of a graph to share a key. The mechanism was extended and enriched in several works (*e.g.* [9]). Recently, [10] proposed a solution, which supports anonymous nodes and addition and removing of nodes.

All these solutions have to deal with the number of keys, the cost of the solution, the number of links compromised by the capture of one node, and the network connectivity. For example, [9] is an extension of [2] in which the number of keys required for two nodes to be able to communicate is increased: this increases the network security since an adversary requires more keys to compromise a link. Anyway it decreases the network connectivity since the probability for two nodes to share enough keys to communicate is lower than in the general case.

To increase the life duration of the network, some works (*e.g.* [11]) introduce activity scheduling: more nodes than needed to cover the target area are deployed and algorithms select nodes that have to switch off to safe power for future use. These algorithms are shown to increase the network lifetime since they distribute the area coverage between different set of nodes at different times. However, this solution does not increase the network security since nodes are deployed with pre-distributed key sets: nodes that are switched off can be captured as well, and nodes that switch on can use already compromised links.

Another way to increase life duration of the network is to use multiple deployments. Some applications use new deployments to replace nodes which have no power anymore. This solution also permits to re-establish secure links when too many links have been compromised. In [3], authors propose to use multiple

deployment schemes to introduce new nodes with new sets of keys in the networks, and to use stronger nodes called "bridges" to establish communication between nodes deployed at different phases. They shows that the use of this scheme increases the number of uncompromised nodes regarding to a traditional multiple deployment solution using same sets of keys.

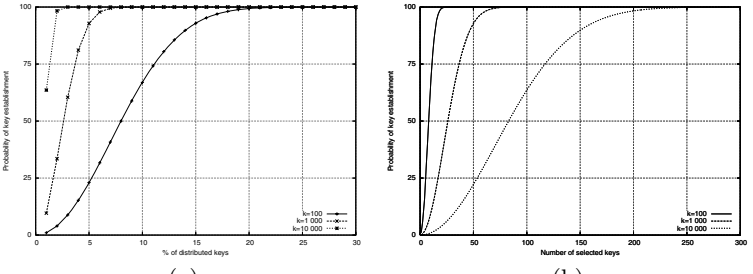
3 Key Pre-distribution and Connectivity

A sensor network is represented by a graphe $G = (V, E)$ where V is the set of vertices for sensor nodes and $E \subseteq V \times V$ is the set of edges. The couple (u, v) belongs to E if and only if the node v can receive messages of the node u. In this paper, we limit our study to *unit disk graph* (UDG) but most of results can be easily extended to symmetrical graphs. In UDG, the edge set is defined by physica proximity of nodes. Let $u, v \in V$ be two nodes. We denote by $d(u, v)$ the geographical distance between node u and note v. The set E is then defined by $E = \{(u, v) \in V^2 \mid u \neq v \wedge d(u, v) < R\}$ where R is a constant representing the communication range.

Let K be the key pool which contains k keys ($|K| = k$). Let $G = (V, E)$ be a UDG. For a given non-null integer n, a n-key assignment is a function α from V to 2^K where $\forall u \in V$ we have $|\alpha(u)| = n$. If the key sets are uniformly set, the probability that two distinct nodes share at least one key is:

$$p_{k,n} = \begin{cases} 1 & \text{if } k < 2n \\ 1 - \binom{k-n}{n} \times \binom{k}{n}^{-1} & \text{otherwise} \end{cases}$$

where $\binom{p}{q}$ denotes the binomial coefficient corresponding to the number of q-subsets possible out of a set of p distinct items. The probability $p_{k,n}$ is also called probability of key establishment and is also denoted by p when we do not need to consider the number of keys. This probability $p_{k,n}$ is illustrated in Figure 1.



(a) (b)

Fig. 1. Probability of key establishment (a) versus number of distributed keys and (b) versus percentage of distributed keys for different amount of available keys k

Since all couples of nodes cannot establish a session key, some links of the communication graph G are not available. Hence, the secured graph induced by the key assigment α is denoted by $G_\alpha = (V, E_\alpha)$ and is defined by $E_\alpha = \{(u, v) \in E \mid \alpha(u) \cap \alpha(v) \neq \emptyset\}$.

In Figure 2, we give some examples of randomly generated secured graphs. For these graphs, N nodes are randomly dropped in a square of fixed size S. The density represents the number of nodes per communication area: $d = \frac{\pi.R^2.N}{S}$.

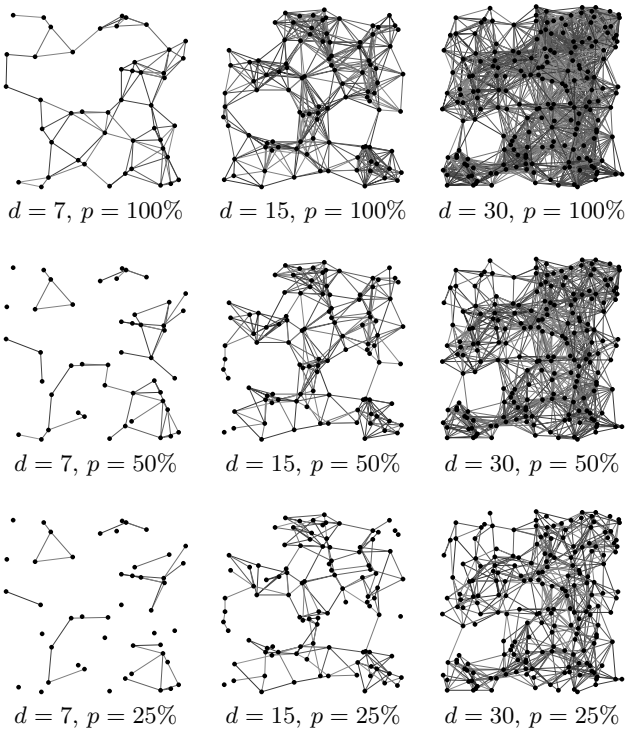


Fig. 2. Some examples of considered secured graphs (d is the density, p is the probability of key establishment)

Figure 3 shows the probability of graph connectivity versus the network density. It is experimental results with a 1000 m$times$1000 m area S with $R = 10$ m. The density and the number of nodes which are required to obtain 90% of connected graphs are summarized Table 1.

4 Evaluation of Network Robustness

In this section, we describe three schemes of sensor deployment. The first one corresponds to the usual sensor deployment which uses only one deployment. We distinguish a variation of this scheme with the idea that sensors can erase unused

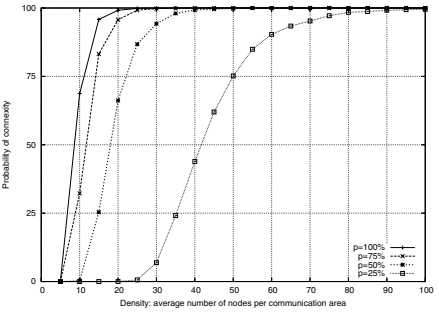


Fig. 3. Probability of network connectivy versus density for different probabilities of key establishment p

keys. The second one is the protocol which uses activity scheduling in order to increase the network lifetime. The third protocol is the multi-deployment scheme without communication between nodes of different rounds.

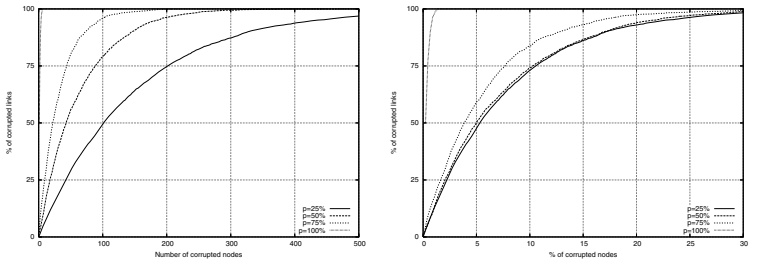
For all these schemes, the aim is to monitor an area S and we want to minimize the number of necessary nodes N while minimizing to number of corrupted links when nodes are captured. We consider that n, the number of keys embedded in the sensors, is a fixed parameter. But one can modify k, the number of keys in the key pool K, in order to adjust the probability of key establishment p. For the first scheme (simple deployment), the lifetime of the network corresponds to the lifetime of the battery. For the other schemes, we can control the lifetime by using different strategies.

Scheme 1: Simple deployment. The algorithm is straightforward:

1. at initialization, each sensor randomly chooses n keys (function α),
2. randomly drop N sensors,
3. each sensor discovers its neighborhood and establishes a key when possible (*i.e.* when nodes share a common key),
4. ignores neighbors without common keys.

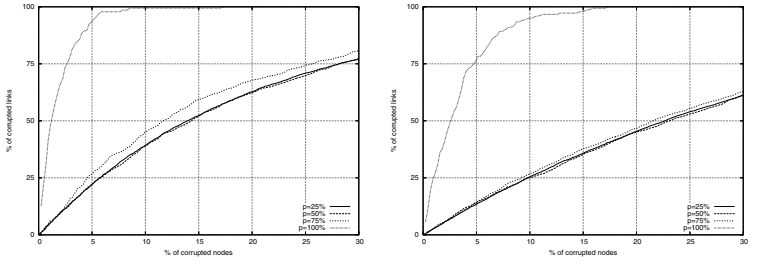
The process can be represented by a single graph G as described in previous section. The key establishment can be represented by a function β from E to K: $\forall (u,v) \in E$, $\beta(u,v)$ is set to a key of $\alpha(u) \cap \alpha(v)$ if non-empty and is undetermined otherwise. Let $C \subseteq V$ be the set of corrupted nodes. The number of corrupted links is $|\{(u,v) \in V_\alpha \mid \exists w \in C \quad \beta(u,v) \in \alpha(w)\}|$.

The number of necessary nodes N is imposed by connectivity criteria according to the probability of key establishment p. When N is suffisantly high to ensure the connectivity, it is clear that the number of corrupted links is linear with the number of nodes and with the number of captured nodes. The percentage of corrupted links is illustrated Figure 4. We give two ways to present the results and we believe that one have to use percentage of captured nodes instead (b)of number of captured nodes (a).



(a) (b)

Fig. 4. Percentage of corrupted links (a) versus number of captured nodes and (b) versus percentage of captured nodes for different probabilities of key establishment p



(a) $n = 50$ (b) $n = 100$

Fig. 5. Percentage of corrupted links versus percentage of captured nodes with key erasing

A variation consists in the deletion of unused keys after step 3. It means that after deletion, each node u contains a subset $\alpha'(u) \subseteq \alpha(u)$ defined by: $\alpha'(u) = \{\beta(v) \mid v \in V \wedge (u, v) \in E\}$. Then the number of corrupted links is computed by using α' instead of α.

This variation is very efficient as shown Figure 5 where we distinguish two configurations for the number of embedded keys. It appears that it is easy to improve the network robustness by increasing n.

Scheme 2: Single deployment with activity scheduling. In this scheme, we want to extend the lifespan of the network by activate-desactivate nodes

Table 1. Required density and number of nodes for a connectivity probability of 90%, percentage of corrupted links when 5% of nodes are captured

p	d	N	without deletion	with deletion
25	60.0	1909	47.7%	13.5%
50	27.1	931	50.1%	14.3%
75	17.7	563	59.6%	14.3%
100	13.9	443	100.0%	78.3%

while maintaining the surveillance function. We split the process in sequential rounds and we write V_i the set of nodes which are active in round i, $1 \leq i \leq r$ where r is the maximal number of rounds we want to reach. A node can be activate only during one round ($\forall 1 \leq i < j \leq r$, we have $V_i \cap V_j = \emptyset$). Moreover, each V_i has to be connected and covers the monitored area. There exist several algorithms for such activity scheduling problems [11]. We will use the simple algorithm presented in [12]. In such algorithms, for each round, we determine a connected area dominating set. Each node computes a timeout function, and listens messages of other nodes before deciding its dominating status at the end of timeout interval. The status is then sent in an advertising message. The protocol at each node u runs as follows.

1. u computes the area covered by each transmission he receives, and includes each node that sent advertizing in the subset,
2. at the end of timeout interval, u computes subgraph of 1-hop neighbors that sent advertisements (they are exactly neighbors with higher priority),
3. if this subgraph is connected and if the nodes in the subgraph fully cover the area centered at u, the node decides to be non-gateway (sleeping) status. Otherwise, the node chooses the gateway (active) status. Active node also transmits a message to all its neighbors, which is referred to as the 'positive advertising'. If the node decides not to monitor its area, there are two variants that can be considered, depending on whether or not such information is transmitted to all its neighbors. If it decides to transmit, the message is referred to as the 'negative advertising'. The two variants are consequently called 'positive and negative advertising' and 'positive only advertising'.

In this paper, we consider only 'positive and negative' variant of this protocol. In this protocol, the connectivity is ensured by the verification of connectivity of neighbors with higher priority. It means that the node u is able to know wether or not two neighbors can communicate. It is then necessary to take key establishment into account. The algorithm is the following one:

1. at initialization, each sensor randomly chooses n keys (function α),
2. randomly drop N sensors,
3. each sensor discovers its neighborhood and establishes a key when possible (*i.e.* when nodes share a common key),
4. ignores neighbors without common keys,
5. for each round:
   (a) each node randomly chooses a timeout,
   (b) during timeout, listen to advertising messages, each node which sends an advertising message is inserted in a set A,
   (c) after timeout expired, look if there exists a subset B of A such that B is connected (by using connection information included in advertising messages) and such that nodes in B cover the monitoring area of the current node:

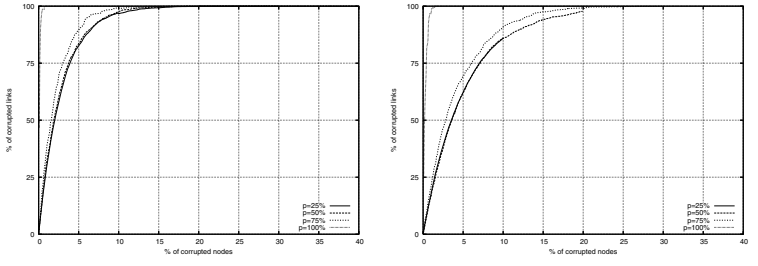
- if yes, the node sends a 'negative advertising message' and switch off until next round,
- otherwise, the current node sends a 'positive advertising message' and remains active for the round.

An advertising message has to contain information about neighbors that share a common key.

The question is then to determine N and p in order to be able to obtain all the V_i which cover the area and are connected. However, the number of corrupted links evolves similarly to previous case. It means that activity scheduling cannot improve the robustness of the networks because all nodes are dropped in the same time.

We can also derive a variation of this scheme by using key deletion. For a given round i, a node u which belongs to V_i removes all non-necessary keys by selecting $\alpha'(u) = \{\beta(u,v) \mid v \in V_i \wedge (u,v) \in E_\alpha\}$. In the same time, a node u which belongs to a round $j > i$ also remove non-mandatory keys: $\alpha'(u) = \{\beta(u,v) \mid \exists l > i \quad v \in V_l \wedge (u,v) \in E_\alpha\}$.

The performance of these two schemes is shown Figure 6. In these experiences, we run 4 rounds and during each round, 10% of nodes are captured. However, in order to be able to run 4 rounds, we have to put 3 times more nodes than for a single round (*i.e.* like in Scheme 1). This leads to a low robustness of the network even when we erase keys.



(a) without deletion (b) with deletion ($n = 50$)

Fig. 6. Percentage of corrupted links versus percentage of captured nodes with activity scheduling

Scheme 3: Multi-deployment. In this sheme, we simply repeat the single deployment of sensors with disjoint key pools. The performance of this solution is good but it does not take into account the fact that we have to re-deploy a sensor set. Moreover, the number of necessary sensors is exactly the number of sensors needed for a single deployment multiplied by the number of rounds we want – while in previous case, we see that the number of nodes can be reduced.

5 Conclusion and Futures Works

In this paper, we give a robustness evaluation of key pre-distribution algorithms and we introduce the possibility to erase keys. The simple feature significantly reduces to number of corrupted links. We think that it is promising to combine activity scheduling and multi-deployment in order to reduces the vulnerability of activity scheduling scheme while reducing the number of nodes required for multi-deployment.

References

1. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Computer Networks **38** (2002) 393–422
2. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In Atluri, V., ed.: ACM Conference on Computer and Communications Security (CCS 2002). (2002) 41–47
3. Durresi, A., Bulusu, V., Paruchuri, V., Barolli, L.: Secure and continuous management of heterogeneous ad hoc networks. In: Proc. 20th International Conference on Advanced Information Networking and Applications - (AINA'06). Volume 1., IEEE Computer Society (2006) 511–516
4. Michiardi, P., Molva, R.: Ad hoc network security. In Basagni, S., Conti, M., Giordano, S., Stojmenovic, I., eds.: Mobile Ad Hoc Networking. Wiley-IEEE Press (2004) 329–354
5. Buchegger, S., Boudec, J.Y.L.: Performance analysis of the CONFIDANT protocol (Cooperation Of Nodes: Fairness In Dynamic Ad-hoc neTworks. In: Proc of MobiHoc, ACM (2002) 226–236
6. Hauspie, M., Simplot-Ryl, I.: Cooperation in ad hoc networks: Enhancing the virtual currency based models. In: Proc. of the 1st International Conference on Integrated Internet Ad hoc and Sensor Networks (InterSense 2006). (2006)
7. Zhou, L., Haas, Z.J.: Securing ad hoc networks. IEEE Network **13**(6) (1999) 24–30
8. Du, W., Deng, J., Han, Y.S., Chen, S., Varshney, P.K.: A key management scheme for wireless sensor networks using deployment knowledge. In: 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004). (2004)
9. Chan, H., Perrig, A., Song, D.X.: Random key predistribution schemes for sensor networks. In: IEEE Symposium on Security and Privacy (S&P 2003), IEEE Computer Society (2003) 197–213
10. Eltoweissy, M., Wadaa, A., Olariu, S., Wilson, L.: Group key management scheme for large-scale sensor networks. Ad Hoc Networks **3**(5) (2005) 668–688
11. Gallais, A., Carle, J., Simplot-Ryl, D., Stojmenovic, I.: Localized sensor area coverage with low communication overhead. In: 4th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2006). (2006)
12. Carle, J., Simplot-Ryl, D.: Energy efficient area monitoring by sensor networks. IEEE Computer **37**(2) (2004) 40–46

A Practical Solution to the (t, n) Threshold Untraceable Signature with (k, l) Verification Scheme

Jen-Ho Yang[1], Chin-Chen Chang[1,2], and Chih-Hung Wang[3]

[1] Department of Computer Science and Information Engineering,
National Chung Cheng University,
160 San-Hsing, Ming-Hsiung, Chiayi 621, Taiwan
jenho@cs.ccu.edu.tw
[2] Department of Information Engineering and Computer Science,
Feng Chia University,
100 Wenhwa Rd., Seatwen, Taichung 40724, Taiwan
ccc@cs.ccu.edu.tw
[3] Department of Computer Science and Information Engineering,
National Chiayi University,
300 University Rd., Chiayi 621, Taiwan
wangch@mail.ncyu.edu.tw

Abstract. The (t, n) threshold signature scheme can delegate the signing capability to all n participants in a group, and at least t participants can cooperatively sign a message on behalf of the group, where $t \leq n$. For the group communication in the real society, the verification site also needs to be restricted in its associated access privileges. Therefore, this paper proposes a practical solution to the (t, n) threshold untraceable signature with (k, l) verification scheme, which requires that k out of l verifiers or more can verify the threshold signature on behalf of the verification group. Compared with the previous works, such as Wang et al.'s scheme and Chang et al.'s scheme, our proposed scheme is more practical and expansible. Our scheme allows each group to be both a signing party and a verification party, and the shadows of all group members are no need to be redistributed after the initialization has been finished. In addition, the share distribution center (SDC) is not required in our scheme.

1 Introduction

With the growth of the transactions on Internet, computer cryptography becomes more and more important. Many digital signature schemes have been proposed (e.g. RSA [7] and ElGamal [8]) for the security of network transactions. Some security tasks such as authenticating electronic contract, identifying the owner of an electronic message, and protecting the ownership of the electronic document can be accomplished with these digital signature schemes. Thus, various digital signature schemes are proposed for different applications of electronic commerce on Internet. An important one of these schemes is the (t, n) threshold signature scheme.

In the (t, n) threshold signature scheme, n shadows of a signing key are shared among n participants in a group, and at least t or more participants can sign a message on behalf of the group, where $t \leq n$. The (t, n) threshold signature scheme has many applications on electronic transactions, such as network payment, electronic voting,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 998 – 1007, 2006.

and contract signing. For example, suppose that there are n managers in a company, and at least t or more managers can sign a contract on behalf of this company.

In 1991, Desmedt and Frankel [10] proposed a (t, n) threshold signature scheme based on RSA, and afterwards Harn [5] in 1994 proposed a threshold signature scheme based on Shamir's perfect secret sharing scheme [1]. Harn's scheme employs the Lagrange interpolating polynomials to construct a (t, n) threshold module that divides the group secret key into n shadows among n participants. The group secret key is hidden in the polynomial and at least t or more participants can assemble the group secret key by reconstructing the polynomial, and thus these t or more participants can sign the message on behalf of the group. In 2000, Wang et al. [3] proposed a (t, n) threshold signature scheme with (k, l) threshold verification. In their scheme, a verification group can verify a (t, n) threshold signature only if k or more members in the verification group agree to cooperate. Therefore, their scheme is especially suitable for signing a contract between two companies. In 2000, Lee et al. [6] proposed a (t, n) untraceable threshold signature scheme based on Ohta-Okamoto signature scheme [4]. The actual t signers of the signing group cannot be traced in their scheme for safeguarding the signer's privacy. Moreover, their scheme additionally provides actual signers the ability to prove, of their own free will, that they really made the signature. In 2004, Chang et al. [9] proposed a (t, n) threshold signature scheme with (k, l) threshold verification based on Lee et al.'s scheme. Unlike Wang et al's scheme, they used the extended Euclidean algorithm to construct their scheme.

Our Contributions. We find that the previous related researches about the (t, n) threshold signature with (k, l) threshold verification, such as Wang et al's scheme [3] and Chang et al.'s scheme [9], have the following common problems. The signing group and the verification group are a fixed pair, which is impractical for many applications. The signing group uses the shadows distributed by a trusted SDC to make a threshold signature, which can only be verified by a specific verification group. The SDC needs to redistribute a set of new shadows to all group members if the pair of signing group and verification group has been changed. An alternative approach to solve the above problems is that each group keeps different shadows for all other groups. However, this approach cannot be applied to a large number of groups.

Therefore, in this paper, we propose a practical solution on (t, n) threshold signature with (k, l) threshold verification scheme. Our scheme can be more expansible because there is no predetermination of the role played by each group and less secret shadows kept by each group member. It can be seen that our scheme is more applicable to the situation of existing a large number groups who want to communicate with each other. The major contributions of our scheme are listed below.

1. Each group can be both a signing party and a verification party.
2. The shadow of each member in the group need not be redistributed after the initialization phased has been finished.
3. Both threshold values of signing and verification can be dynamically changed according to the requirements of the practical applications.
4. It is not necessary to maintain a trusted share distribution center.
5. Each group is required to hold only one secret shadow for signing and verification.

Models Comparison. Fig. 1 and Fig. 2 show two models of the previous related schemes and our proposed scheme, respectively. In Figure 1, the signing group must

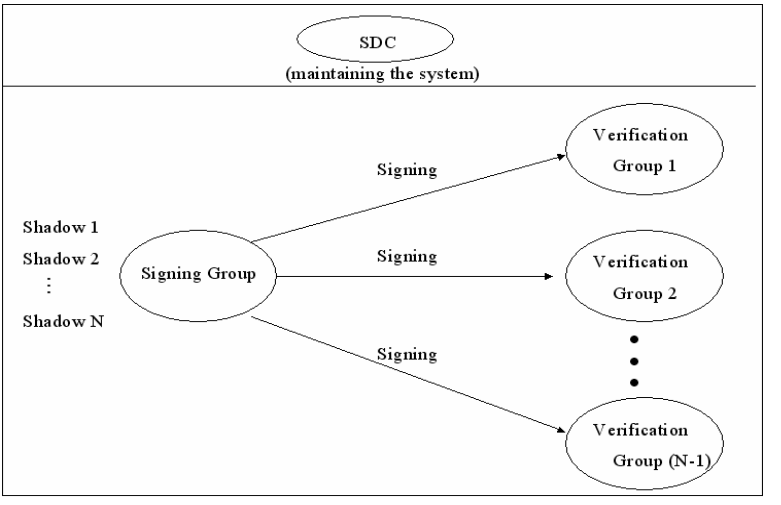


Fig. 1. The model of the previous related schemes

keep N-1 shadows for N-1 corresponding verification groups. If the signing group wants to verify the signatures signed by other N-1 groups, it needs to keep N-1 additional shadows. That is, to sign and verify the signatures for other N-1 groups, one group must keep 2(N-1) shadows in this system. On the other hand, in Figure 2, regardless of how many groups there are, each group just keeps one shadow to sign and verify the signatures for other N-1 groups.

The rest of our paper is organized as follows. First, a review of Chang et al.'s scheme is presented in Section 2, and our proposed algorithm is shown in Section 3. Then, the security analysis is shown in Section 4. Finally, the conclusions are given in Section 5.

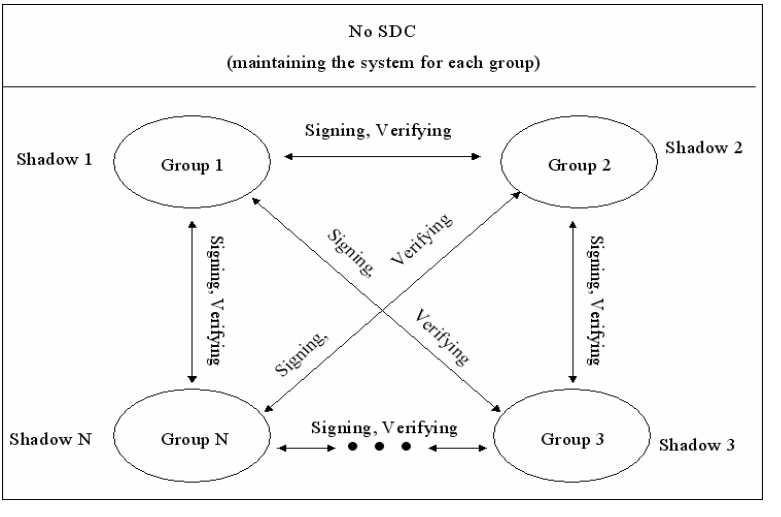


Fig. 2. The model of our proposed scheme

2 Chang et al.'s Scheme

For the reader further understanding the difference between two models mentioned above, we review Chang el al.'s untraceable (*t*, *n*) threshold signature with (*k*, *l*) threshold verification scheme. There are three roles in their scheme, a trusted SDC, a signing group, and a verification group. The SDC is responsible for initializing the parameters and distributing the shares to both signing group and verification group. Let G_s ($|G_s| = n$) be a signing group with *n* members and g_s ($|g_s| = t$) be a subset of G_s with *t* signers. Similarly, G_v ($|G_v| = l$) is defined as the verification group with *l* members and g_v ($|g_v| = k$) is the subset of G_v with *k* verifiers. Chang et al.'s scheme has three phases: Parameter Generating Phase, Individual Signature Generating Phase, and Threshold Signature Generating and Verifying Phase. The scheme is shown as follows.

2.1 Parameter Generating Phase

The SDC initializes the system and the parameters in the following:

Step 1. Chooses two large secret primes $p = 2p'+1$ and $q = 2q'+1$, and computes a public number $N = p \times q$, where p' and q' are also two secret primes.

Step 2. Chooses a public number $W \approx 10^{50}$ that satisfies $\gcd(\lambda(N), W) = 1$, where $\lambda(N) = 2p'q'$ is the Carmichael function.

Step 3. Chooses a secret primitive α in both $GF(p)$ and $GF(q)$.

Step 4. Randomly chooses two numbers *a* and *b* satisfying $a \cdot c + b \cdot h = 1$, where $\gcd(a, b) = 1$. These integers can be computed with Euclidean algorithm.

Step 5. Chooses two secret polynomials $f_s(x) \bmod \lambda(N)$ with degree $t-1$ and $f_v(x) \bmod \lambda(N)$ with degree $k-1$, where $f_s(0) = d \cdot a \cdot c$, $f_v(0) = d \cdot b \cdot h$, and $\gcd(\lambda(N), d) = 1$.

Step 6. Computes $S = \alpha^d \bmod N$ as G_s's group secret key and the associated G_s's group public key $Y = \alpha^{-d \cdot W} \bmod N$. Then, chooses a public collision-free one-way hash function $h(\cdot)$.

Step 7. Randomly selects *n* public and odd integers x_{si} with even $f_s(x_{si})$ for each participant in the signing group G_s. Then each participant in G_s has the secret key $K_{si} = \alpha^{s_i} \bmod N$, where $s_i = \dfrac{f_s(x_{si})/2}{[\prod_{j \in G_s, j \neq i}(x_{si} - x_{sj})]/2} \bmod p'q'$.

Step 8. Randomly selects *n* public and odd integers x_{vi} with even $f_v(x_{vi})$ for each participant in the verification group G_v. Then each participant in G_v has the secret key $K_{vi} = \alpha^{v_i} \bmod N$, where $v_i = \dfrac{f_v(x_{vi})/2}{[\prod_{j \in G_v, j \neq i}(x_{vi} - x_{vj})]/2} \bmod p'q'$.

2.2 Individual Signature Generating Phase

In this phase, each participant $P_{si} \in g_s$ generates an individual signature as follows:

Step 1. Each P_{si} randomly selects an integer r_{si} and computes $u_{si} = r_{si}^{W} \bmod N$. Then, P_{si} broadcasts u_{si} to other participants of g_s.

Step 2. After receiving u_{sj} $(j \in g_s, j \neq i)$, each P_{si} computes $U_s = \prod_{i \in g_s} u_{si} \bmod N$ and $e = h(U_s, m)$, where m is the message that the signing group wants to sign.

Step 3. Each P_{si} uses his secret key K_{si} to generate his individual signature z_{si} as $z_{si} = r_{si} \cdot K_{si}^{\prod_{j \in G_s, j \neq g_s} (x_{si} - x_{sj}) \cdot \prod_{j \in g_s, j \neq i} (0 - x_{sj}) \cdot e} \bmod N$.

Then, each P_{si} sends (z_{si}, m) to a designated clerk $C_s \in g_s$ who is responsible for computing the threshold signature of the signing group.

2.3 Threshold Signature Generating and Verifying Phase

Step 1. The clerk C_s computes $Z_s = \prod_{i \in g_s} z_{si} \bmod N$ after receiving t individual signatures z_{si}'s. Then, C_s sends the threshold signature (Z_s, e) on the message m to the verification group G_v.

Step 2. After receiving the threshold signature (Z_s, e), each $P_{vi} \in g_v$ randomly selects an integer r_{vi} and computes $u_{vi} = r_{vi}^{W} \bmod N$. Then, each P_{vi} sends his u_{vi} to a clerk C_v arbitrarily selected from g_v.

Step 3. Each P_{vi} computes $z_{vi} = r_{vi} \cdot K_{vi}^{\prod_{j \in G_v, j \neq g_v} (x_{vi} - x_{vj}) \cdot \prod_{j \in g_v, j \neq i} (0 - x_{vj}) \cdot e} \bmod N$ and sends it to C_v.

Step 4. When C_v receives t individual u_{vi}'s and t individual z_{vi}'s, he computes $U_v = \prod_{i \in g_v} u_{vi} \bmod N$ and $Z_v = \prod_{i \in g_v} z_{vi} \bmod N$.

Step 5. Finally, the threshold signature can be verified by checking if $e = h(\tilde{U}_s, m)$ holds, where $\tilde{U}_s = (Z_s \cdot Z_v)^{W} \cdot (U_v)^{-1} \cdot Y^{e} \bmod N$.

Chang et al.'s scheme satisfies the following properties:

1. t out of n members or more in G_s can generate the threshold signature (Z_s, e) on behalf of the signing group.
2. k out of l members or more in G_v can verify the threshold signature (Z_s, e) on behalf of the verification group.
3. The actual signers of the threshold signature cannot be traced.

3 Our Proposed Scheme

In this section, we propose a new scheme. Assume that each group has a manager responsible for initializing the parameters and distributing the secret shadows to the group participants. In the proposed scheme, the shadows only need to be distributed once in the initialization stage. Our scheme has three phases: Initialization Phase, Threshold Signature Generating Phase, and Threshold Signature Verification Phase. The scheme is shown as follows.

3.1 Initialization Phase

In this phase, each group manager generates and initializes parameters for the group members. The details are shown in the following.

Step 1. Each group manager chooses four large primes p, q, $\tilde{p}$, and $\tilde{q}$ such that $p = 2p' + 1$, $q = 2q' + 1$, $\tilde{p} = 2\tilde{p}' + 1$, and $\tilde{q} = 2\tilde{q}' + 1$, where p', q', $\tilde{p}'$, and $\tilde{q}'$ are also large primes. Then, the group manager computes two public numbers $N = p \times q$ and $N' = \tilde{p} \times \tilde{q}$, where $N \in [\,2^{c_1}, 2^{c_1'}\,]$ and $N' \in [\,2^{c_2}, 2^{c_2'}\,]$. Note that c_1, c_1', c_2, and c_2' are four integers that satisfy $c_1' > c_1$, $c_2' > c_2$ and $c_2 > c_1'$.

Step 2. Each group manager randomly selects a public number W such that $\gcd(\lambda(N), W) = 1$, where $\lambda(N) = 2p'q'$.

Step 3. Each group manager chooses a secret polynomial $f(x) \bmod \lambda(N)$ with degree t-1, where $f(0) = d$, $\gcd(\lambda(N), d) = 1$, and t is a threshold value decided by the group manager. Then, the group manager selects a public number e, where $e \times d = 1 \bmod \lambda(N')$.

Step 4. Each group manager selects a primitive α in both $GF(p)$ and $GF(q)$ to compute $S = \alpha^d \bmod N$ as the signing key and the associated public key $Y = \alpha^{-d \cdot W} \bmod N$, where $\gcd(\lambda(N), d) = 1$.

Step 5. Each group manager selects a primitive α' in both $GF(\tilde{p})$ and $GF(\tilde{q})$ to compute an additional public key $Y' = S(\alpha')^S \bmod N'$ for verification.

Step 6. Each group manager randomly selects n public and odd integers x_i with even $f(x_i)$ for each participant, where n is the number of members in the group.

Then each participant has the secret key $K_i = \alpha^{s_i} \bmod N$, where $s_i = \dfrac{f(x_i)/2}{[\prod_{j=1, j \neq i}^{j=n} (x_i - x_j)/2} \bmod p'q'$.

Step 7. Each group manager selects a public one-way hash function $h(\cdot)$.

When the phase is finished, each group has a signing key $S = \alpha^d \bmod N$ and the associated public key $Y = \alpha^{-d \cdot W} \bmod N$ for constructing the threshold signature. Thus, each group can be a signing party. In addition, each group has an additional public key $Y' = S(\alpha')^S \bmod N'$ for verification. Thus, each group can also be a verification party. According to our design of this phase, each group member just keeps a secret shadow.

3.2 Threshold Signature Generating Phase

Assume that G_s, with *n* members, is a group who wants to sign a threshold signature for another group G_v with *l* members. Besides, we denote that g_s are *t* out of *n* or more participants in G_s, and g_v are *k* or more participants in G_v. After the parameter generating phase, the group G_s has the parameters: α_s, N_s, W_s, $f_s(x)$, d_s, S_s, Y_s, K_{si}, α_s', N_s', Y_s', and $h_s(\cdot)$. Similarly, the group G_v has the parameters: α_v, N_v, W_v, $f_v(x)$, d_v, S_v, Y_v, K_{vi}, α_v', N_v', Y_v', and $h_v(\cdot)$. The detail steps of this phase are shown in the following.

Step 1. Each participant $P_{si} \in g_s$ randomly selects an integer r_{si} and computes $u_{si} = r_{si}^{W_s} \bmod N_s$. Then, P_{si} broadcasts u_{si} to others.

Step 2. After each P_{si} receives the other participant's u_{sj} ($j \in g_s$ and $j \neq i$), he computes $U_s = \prod_{i \in g_s} u_{si} \bmod N_s$ and $b = h(U_s, m)$, where *m* is the message.

Step 3. Each P_{si} uses his secret key K_{si} to generate his individual signature z_{si} as $z_{si} = r_{si} \cdot K_{si}^{\prod_{j \in G_s, j \neq g_s} (x_{si} - x_{sj}) \cdot \prod_{j \in g_s, j \neq i} (0 - x_{sj}) \cdot b} \bmod N_s$.

Step 4. Each P_{si} sends (z_{si}, m) to the designated clerk $C_s \in g_s$ to generate the threshold signature.

Step 5. The clerk C_s computes $Z_s = \prod_{i \in g_s} z_{si} \bmod N_s$ after receiving *t* individual signatures. Then, the clerk randomly chooses a secret number *r* and computes the session key $K = ((Y_v')^{e_v} \cdot (\alpha_v')^{-1})^r \bmod N_v'$. Then, he computes the cipher texts $C_1 = (\alpha_v')^{e_v r} \bmod N_v'$, $C_2 = K \cdot Z_s \bmod N_v'$, and sends (C_1, C_2, b) to the verifying group.

3.3 Threshold Signature Verification Phase

Step 1. Each verifier $P_{vi} \in g_v$ computes $k_i = (C_1)^{K_{vi}^{\prod_{j \in G_v, j \neq g_v} (x_{vi} - x_{vj}) \cdot \prod_{j \in g_v, j \neq i} (0 - x_{vj})}} \bmod N_v'$ after receiving (C_1, C_2, b), and he sends it to the verification clerk C_v.

Then, C_v computes the session key $K = \prod_{i \in g_v} k_i \bmod N_v'$ and $Z_s = (C_2 \cdot K^{-1} \bmod N_v') \bmod N_s$, and sends Z_s to P_{vi}.

Step 2. Each P_{vi} computes $U_s' = Z_s^{W_s} \cdot Y_s^b \bmod N_s$. Then, each P_{vi} can verify the signature by checking if $b = h(U_s', m)$ holds.

In our scheme, each group can be both a signing party and a verification party, and the shadows of all group members do not need to be redistributed after the initialization has finished. Besides, our scheme does not need a trusted SDC to distribute the secret shadows, so that the threshold value can be changed easily and dynamically. If the signing group reveals the session key K, the specific-verified threshold signature can be converted into a primitive threshold signature, which can be verified by an arbitrary group, for different practical applications. Moreover, each group member just keeps one secret shadow in our scheme even if there exists a large number of groups who want to communicate each other.

4 Security Analysis

In this section, we discuss the security of our proposed scheme. Basically, the security of our scheme is based on RSA cryptosystem [7]. Now, we analyze several possible attacks in the following.

1. An attacker wants to obtain the signing key S from the related group public key Y.

   If the attacker wants to obtain the signing key S from $Y = S^W \bmod N$, he must face the difficulty of breaking RSA cryptosystem. Thus, this kind of attack is infeasible.
2. *t* or more shareholders want to cooperate to obtain the system secrets s_i by using their secret keys K_i's.

   In our scheme, the signing shareholder's secret key is computed as $K_i = \alpha^{s_i} \bmod N$, where $s_i \bmod \lambda(N)$ is unknown to any shareholder. However, deriving s_i from K_i is as difficult as breaking RSA cryptosystem. Thus, *t* or more shareholders cannot cooperate to obtain the system secrets s_i.
3. An attacker wants to obtain the signing key S_s from a valid signature (Z_s, *b*).

   The threshold signature Z_s is computed by $Z_s = \prod_{i \in g_s} z_{si} = R_s \alpha_s^{d_s} \bmod N_s$, where $z_{si} = r_{si} \cdot K_{si}^{\prod_{j \in G_s, j \neq s} (x_{si} - x_{sj}) \cdot \prod_{j \in g_s, j \neq i} (0 - x_{sj}) \cdot b} \bmod N_s$ and $R = \prod_{i \in g_s} r_{si} \bmod N_s$. Assume that an attacker wants to obtain the signing key $S_s = \alpha_s^{d_s} \bmod N_s$ from Z_s, he may compute

$\alpha_s^{d_s} = R_s^{-1} \cdot Z_s \bmod N_s$. It is infeasible because the attacker does not know the random number R_s so that he cannot compute $R_s^{-1} \bmod N_s$.

4. An attacker wants to forge a valid threshold signature (Z_s, b).

   To forge a valid threshold signature, the attacker has to construct Z_s' and b' that satisfy the verification equations $U_s' = Z_s'^{W_s} \cdot Y_s^{e'} \bmod N_s$, and $b' = h(U_s', m)$. However, the attacker does not know the signing key $S_s = (Y_s)^{-1} \bmod N_s$. It is infeasible to construct Z_s' satisfying $U_s' = Z_s'^{W_s} \cdot Y_s^{b'} \bmod N_s$ and $b' = h(U_s', m)$ without knowing the signing key $S_s = (Y_s)^{-1} \bmod N_s$.

5. An attacker wants to impersonate the shareholders in G_s and G_v.

   Assume that the attacker wants to impersonate the shareholder in the signing group, he randomly selects an integer $r_{si}' \in [1, N_s - 1]$ and broadcasts $u_{si}' = (r_{si}')^{W_s} \bmod N_s$. Then, the partial threshold signature b' is computed as $b' = h(U_s', m)$, where $U_s' = (\prod_{j \in g_s, j \neq i} u_{sj}) \cdot u_{si}' \bmod N_s$. However, it is infeasible for the attacker to generate an individual signature z_{si}' such that $u_{si}' \cdot (\prod_{j \in g_s, j \neq i} u_{sj}) = (z_{si}' \cdot \prod_{j \in g_a, j \neq i} z_{sj})^{W_s} \cdot Y_s^{b'} \bmod N_s$ since the secret share K_{si} is unknown to the attacker.

5 Conclusions

In this paper, we provide a practical solution to the (*t*, *n*) threshold untraceable signature with (*k*, *l*) verification scheme. Our scheme is more expansible and practical than the previous related researches. Because of the advantages of high expansibility and practicability, our scheme is suitable for many applications on Internet, such as e-cash, electronic transactions, and online voting. In the future work, we are planning to use this scheme as a basic model to design a variety of applications on Internet. Moreover, we can improve our scheme according to different requirements in practice.

References

1. Sharmir, A.: How to Share a Secret. Communications of the ACM, Vol. 22, (1979) 612-613.
2. Fiat, A., Shamir, A.: How to prove yourself. Proceedings of Advances in Cryptology-Crypto'86, (1987) 186-199.
3. Wnag, C. T., Chang, C. C., Lin, C. H.: Generalization of Threshold Signature and Authenticated Encryption for Group Communications. IEICE Transactions on Fundamental of Electronics Communications and Computing, E83-A, Vol. 6, (2000) 1228-1237.
4. Ohta, K., Okamoto, T.: A Modification of the Fiat-Shamir Scheme. Proceedings of Advances in Cryptology-Crypro'88, Santa Barbara, CA, USA, (1988) 232-243.

5. Harn, L.: Group-Oriented (t, n) Threshold Signature Scheme and Digital Multisignature. IEE Proceedings on Computer Digital Technology, Vol. 141, No. 5, (1994) 307-313.
6. Lee, N. Y., Hwang, T., Li, C. M.: (t, n) Threshold Untraceable Signatures. Journal of Information Science and Engineering, Vol. 16, (2000) 835-846.
7. Rivest, R. L., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signatures and Public Key Cryptosystems. Communications of the ACM, Vol. 21, No. 2, (1978) 120-126.
8. Elgamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE Transactions on Information, IT-31, (1985) 469-472.
9. Chang, T. Y., Yang, C. C., Hwang, M. S.: Threshold Untraceable Signature for Group Communications. IEE Proceedings on Communications, Vol.151, No.2, (2004) 179-184.
10. Desmedt, Y., Frankel, Y.: Shared Generation of Authenticators. Proceedings of Advances in Cryptology-CRYPTO'91, Santa Barbara, CA, USA, (1991) 457-469.

On Studying P2P Topology Construction Based on Virtual Regions and Its Effect on Search Performance*

Yufeng Wang[1], Wendong Wang[2], Kouichi Sakurai[3], and Yoshiaki Hori[3]

[1] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[2] State Key Laboratory of Networking & Switching Technology, Beijing University of Posts and Telecommunications (BUPT), Beijing 100876, China
[3] Department of Computer Science and Communication Engineering, Kyushu University, Fukuoka 812-0053, Japan

Abstract. The virtual region-based P2P architecture was provided in this paper, which extended the proximity concept in overlay network. By virtual region it meant various related factors should be integrated into the design of P2P topology, such as peer interest, peer locality, and heterogeneity in peer capacity etc. But, the shared contents in P2P are vast and miscellaneous, it is very difficult to recognize the interest similarity among peers, dynamically form the interest region and direct peer to appropriate interest region. In this paper, the Jensen-Shannon Divergence (JSD) was used to characterize interest/semantic similarity among peers, and the interest region formation and location mechanism based on Dynamic Interest Landmark (DIL) was offered to facilitate to dynamically guide peers to join the appropriate interest region corresponding to peers' content semantic. Considering that interest region is composed of geographically sporadic peers, each interest region was organized according to peer proximity in underlying infrastructure. The theoretical and simulated results illustrated that our architecture can significantly reduce the query overhead, and improve the search performance.

1 Introduction

Recently, P2P technologies have attracted the interest of the research community. Generally, P2P overlay network is virtual topology superposed on the existing Internet infrastructure, in which all participants (always at edge of the Internet) have identical responsibilities, and organize them into a network in ad hoc way. In terms of approaches to locating resource, P2P systems can be divided into two classes: structured and unstructured [1]. The technical meaning of structured is that the P2P overlay network topology is tightly controlled and contents are not placed at random peers but at specified locations that will make subsequent queries more efficient. In theory, if contents exist in P2P systems, it is guaranteed that they can be located

* Research supported by the NSFC Grants 60472067, JiangSu education bureau (5KJB510091) and State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications (BUPT).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1008 – 1018, 2006.

within average O(log*N*) hops in overlay network, in which *N* represents the number of peers in systems. But, the main disadvantage of structured P2P systems (that is, they don't take into account the inherent interest pattern hidden in peers. For example, certain peers are only concerned about special content topics) makes their unsuitable for large-scale file sharing application. By comparison, in unstructured P2P systems, topology of the overlay network and placement of files are largely unconstrained. Due to the extra flexibility, they can be designed to be more adaptive and resilient than structured P2P systems. But, traditional unstructured P2P systems adopt flooding-like or random walker to send queries across overlay network within a limited scope, so the load on each peer grows linearly with the total number of queries, which in turn grows with system size. Thus, the approach is clearly not scalable.

It is shown that clustering peers with similar interest can improve the search performance greatly. Semantic Overlay Network (SONs) [2] used static profile to cluster peers with semantically similar contents in P2P data-sharing networks. The interest-based clustering architecture was also introduced in [3]. But those approaches above were built on a fundamental assumption that there already exists certain interest-based classification hierarchy. But, in high dynamic P2P environment, it is very difficult or even impossible to find out the proper interest classification statically, so, how to dynamically identify content semantic and interest among peers is a fundamental problem. Ref. [4] proposed to create interest-based shortcut to locate content quickly, in which the peers with similar interest established shortcut superposed on the P2P network to improve search performance, and peers used these shortcuts to locate content. The Adaptive P2P topologies (APT) protocol used direct outcomes count to organize the network, which enabled the network to self-organize into clusters of peers that are mutually benevolent, so that collaborative peers obtain relative better QoS than malicious peers [5]. But most of those approaches, which only made use of the direct interaction between peers to construct or change the P2P topology, neither considered the peers' interest in whole P2P system, nor utilized the peer proximity in underlying network to construct more efficient P2P topology.

Ref. [6] proposed a novel mechanism, mOverlay, for constructing an overlay network that takes account of peer locality, which can significantly decrease the communication cost between end hosts. But their research didn't consider the discrepancy in peer interests. Ref. [7] adopted two-level hierarchy to divide the construction of overlay network into two independent subtasks: clustering and mesh management. Similar with Ref. [6], the idea was only based on peer locality in underlying network.

The P2P overlay network construction mechanism based on virtual region was provided in this paper. By virtual region it means various related factors should be integrated into the design of P2P topology, such as peer interest, peer locality, and heterogeneity in peer capacity etc. The main contributions of this paper included:

- JSD-based dynamic interest region formation and interest region locating mechanism based on DIL (Dynamic Interest Landmark). For, static interest classification structure breaks the fundamental features of distributed systems, the dynamic interest region formation mechanism was provided in this paper, and, based on the formed interest regions, the DIL was proposed to dynamically guide peers to join corresponding interest region.

- Based on the observation that interest region is composed of geographically sporadic peers. So, each interest region was organized into several connected geographical groups according to peer proximity in underlying infrastructure.

This paper was organized as follows: In section 2, we introduced our research motivation and the definition of JSD. In section 3, The DIL was provided to dynamically form the interest region and direct peer to corresponding region. In section 4, we provided the theoretic analysis and simulation results to evaluate our approach. Finally, we briefly concluded the paper and pointed out the future work in section 5.

2 Research Motivation and Basic Models

2.1 Research Motivation

P2P architectures can be viewed as a virtual (logical) overlay networks built on top of the underlying physical links of the Internet. Two peers are said to be neighbors if there is a connection (for communication purpose) through the overlay. The intuitive idea is that, if nearby peers in underlying network are constructed as neighbors and neighbor groups are connected, the maintenance messages/other application specific information delivery time will be significantly decreased.

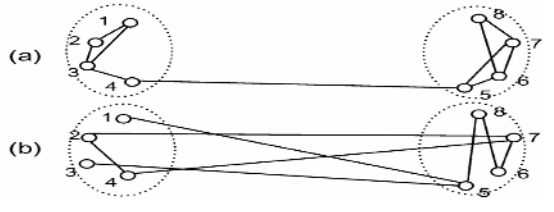


Fig. 1. (a) Illustration for locality-aware overlay (b) randomly connected overlay

Ref. [6] used Fig.1 to illustrate that locality-aware overlay topology is better than randomly connected overlay topology. For example, nodes 1 and 4 can communicate with each other through peer 3 in Fig. 1(a), and through peer 5, 8, 6, 7 in Fig. 1(b), traveling two unnecessary long inter-group links: link 15 and link 74. But, the above locality-aware overlay neglected the inherent correlation existing in peers' shared contents. For example, if peer 1 and peer 8 belong to the same interest region (that is, they share similar contents, thus, the traffic load between these two peers is relatively high), the overlay network topology constructed by locality-aware algorithms is not very ideal. Thus, the better way to construct overlay network topology follows two steps: firstly, according to the content semantic of peers, The P2P topology is organized into connected virtual interest regions; then, in each virtual region, peers are organized according to locality-aware mechanism.

The key point to construct virtual region-based P2P topology depends upon how to dynamically form interest region, and direct peers to proper region corresponding to their content interest. This paper adopted the JSD to characterize the semantic

similarity between peers, and provided the DIL to form interest regions and direct peers to appropriate region.

2.2 Jensen-Shannon Divergence (JSD)

In information theory, the Kullback-Leibler Divergence (KL) and the Jensen-Shannon Divergence (JSD) are usually used to measure the divergence between two probability distributions P(x) and Q(x) on certain data set X. (refer to [8][9]).

The following mapping pattern was used to measure the semantic distance between peers: Let V be the set of all words in the vocabulary of all peers, P_A(V) and P_B(V) denote two word frequency histograms in peer A and B respectively (e.g., the word frequency in a peer's queries or shared content). Let $v \in$ V be a word in the vocabulary. $p_A(v)$ (and $p_B(v)$) is the percentage of the words in P_A(V) (and P_B(V)) that is equal to v respectively. Then, the above equations can be transformed as:

$$D_{KL}\left(P_A(V) \parallel P_B(V)\right) \overset{def}{=} \sum\nolimits_v p_A(v) \log \frac{p_A(v)}{p_B(v)} \tag{1}$$

$$JSD\left(P_A(V), P_B(V)\right) = \frac{1}{2}\left(D_{KL}\left(P_A(V) \parallel \left(\frac{P_A(V)+P_B(V)}{2}\right)\right) + D_{KL}\left(P_B(V) \parallel \left(\frac{P_A(V)+P_B(V)}{2}\right)\right)\right) \tag{2}$$

3 Overlay Topology Construction Based on Virtual Region

Before describing the procedure of constructing an overlay topology based on peers' interest, we first introduce several concepts.

Definition 1: An interest region: *R*: consists of a set of peers that are close to each other in content semantic, which can exchanges messages with other interest regions referred to the neighbors of interest region *R*.

Definition 2: The Criterion of forming interest region: When the semantic distance (measured with JSD) between a new peer *A* and *R*'s neighbor regions is the same as the distance between region *R* and region *R*'s neighbor groups, then peer *A* should belong to region *R*. Here, the neighbors of an interest region are acting as the ***Dynamic Interest Landmarks***, ***DIL*** (used in the interest region forming criterion). When a new peer arrives, it uses a locating method to join a nearest interest region or form its own virtual region according to the interest region forming criterion. Based on the proposed DIL, the adaptive interest region locating mechanism was provided in following subsection.

3.1 Process of Locating Interest Region

Just like other P2P systems, the proposed architecture needed "bootstrapping" mechanism to connect new arrival peers to some active peers in P2P system. As illustrated in Fig. 2, a new peer *A* first communicates with certain peers in interest region 1, which are returned as boot peers by bootstrapping mechanism. Then, interest region 1 measures the semantic distance from itself to peer *A*. At the same time, Interest region 1 selects its neighboring region sequentially, measure the distance

between peer *A* and the selected interest region, and determine the closest region. If the criterion of forming interest region is met, peer *A* joins interest region 1 and the process terminates. Otherwise, the current closest region to peer *A* and the current shortest distance, are recorded as *Distance*[1] (*Distance* is vector used to record the shortest semantic distances between peer *A* and various interest regions in each iteration). In Fig. 2, interest region 2 is the closest to peer *A*, and become the current region for peer *A*. In region 2, the similar steps are conducted. Note that the shortest distance among the neighbor regions of region 2 to peer *A* will be calculated and recorded as *Distance*[2]. The vector *Distance* is sorted according to increase order, and the region with current shortest semantic distance from peer *A* will be remembered as the current region. Otherwise, region 2 is still the current region which has the shortest distance to peer *A*. The same procedure will be repeated until the interest region forming criterion is met. If the region forming criterion has not been met after visiting all the regions, then a new interest region should be formed with peer *A* as member. The newly formed interest region selects several neighbor regions from the sorted vector *Distance* (in top-down order). Furthermore, in order to maintain P2P connectivity, a few regions are also randomly chosen as new region's neighbors. The total number of the new region's neighbors doesn't exceed certain threshold (describe in next section in detail). In Fig. 2, peer *A* will join region 4 (The dotted line in Fig. 2 illustrates the process of peer *A* joining proper interest region).

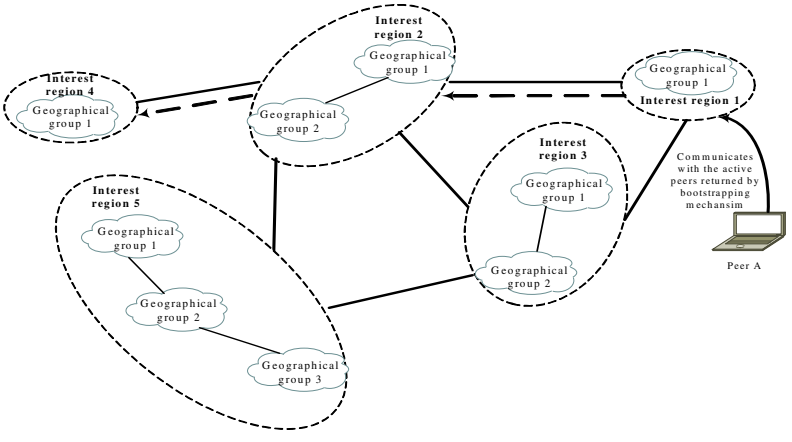


Fig. 2. Illustration of the process of locating appropriate interest region

Considering each virtual interest region composed of geographically sporadic peers, so interest region can be organized according to locality-aware algorithms provided in Ref. [6]. More important, the locality-aware topology construction algorithm run in interest region has the following advantages: the number of query in interest region is great larger than query among interest regions, so the search efficiency in our approach is higher; the number of peer in a interest region is much less than the whole peers in P2P system, so the overhead incurred by running locality-aware mechanisms in interest region is smaller than in the whole P2P system.

3.2 Related Data Structures

Three types of data structures need maintained in our virtual region-based P2P architecture. As far as interest region concerned, each region will maintain a local peer cache, which contains H_I peers in its region. These peers are responsible for communication with peers in other interest regions. The interest region also maintains information about its M_I neighbor interest regions. Such information may be the measured semantic distances based on JSD, and peers in their neighboring regions' peer cache, etc. For the geographical group located in specific interest region, each geographical group will maintain a local peer cache, which contains H_G peers in its current geographical group. These peers are responsible for communication with peers in other geographical groups located in same interest regions. The geographical group also maintains information about its M_G neighbor groups. Such information may include the underlying network distances, or peers in their neighbor geographical groups' host cache, etc. For each peer in specific geographical group, each peer will maintain the following information: status of each neighbor peer (whether the peer represents the current interest region to communicate with other regions, whether the peer communicates with other geographical groups on behalf of its current geographical group. Namely, determine whether the neighbor peer belongs to the member of M_I and/or M_G), the ratio of its neighbor peer's capacity to the neighbor peer's degree, etc.

4 Performance Analysis and Simulation

4.1 Overhead of Locating Interest Region

Obviously, one key performance metric of our constructed overlay network is the complexity of the algorithm to locate the nearby interest region. Generally, the constructed P2P architecture based on virtual interest region can be thought as a special abstract graph, in which each node corresponds to a virtual interest region in the constructed overlay, and neighbor relationship between two interest regions is represented by edges between two nodes in the corresponding graph. Based on the definition of the abstract graph, each node has a fixed number of neighbors (M_I). The problem of finding a nearest interest region in the overlay is similar to finding a specific node from another node through edges in the graph. One can prove that the average distance L between any two nodes follows the inequality: $L < \log_{M_I} N + 3$, where N denotes the number of nodes in abstract graph (that is, the number of interest regions in overlay network).Therefore, the average distance could be considered roughly at the level of $\log_{M_I} N$. The proof is similar as appendix in Ref. [6].

4.2 Search Performance Metrics

The search performance in our virtual region-based P2P architecture is measured from the following metrics [10]: Q_t denotes probability of successful search; S_t represents the average number of hops used to find out the resource (for, the overlay network is constructed as two-level hierarchy: interest region and geographical groups, the hop

number is measured in terms of geographical hop); M_t denotes the average number of message transmissions. Subscript t is the maximum allowed number of hops.

Case 1. Considering the organized overlay network only based on locality-aware algorithm offered in Ref. [6] (here, called geographical groups), Assume there are N geographical groups, and the contents are distributed into those groups randomly. Thus, the probability of locating a specific resource in a specific group is: PC(1) = $1/N = 1\text{-}a$ (Namely, a=1-1/N), then, the probability of locating the resource in total j groups is given: $PC(j) = 1 - (1 - PC(1))^j = 1 - a^j$.

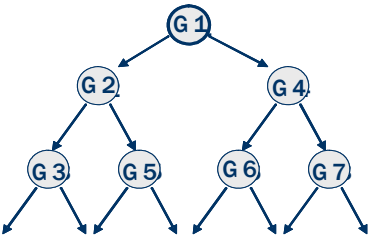


Fig. 3. Search in geographic group-based overlay

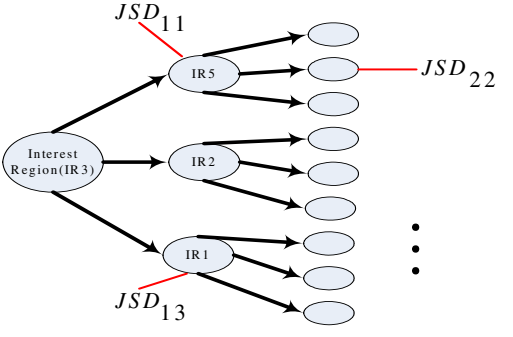


Fig. 4. Search in interest region-based overlay

Adopting flooding search mechanism, each group sends query to all neighboring groups, as illustrated as Fig.3, the overlay network is formed as complete d-ary "tree". In level i of the tree, d^i groups are queried (Note, we assume the approximation that all these groups are regarded as distinct, which does not introduce significant error).

Then the probability of locating resource at hop i in Fig. 3 is:

$s_i = PC(d^i)\prod_{j=0}^{i-1}(1-PC(d^j))$. Thus, $Q_t = \sum_{i=0}^{t} s_i$, then

$$\overline{S_t} = \sum_{i=0}^{t} i\frac{s_i}{Q_t} = \frac{1}{Q_t}\left(a - (t+1)(1-Q_t) + \sum_{i=2}^{t+1} a^{\frac{d^i-1}{d-1}}\right) \tag{3}$$

$$\overline{M_t} = c^t + c + c(c+1-a)\frac{c^{t-1}-1}{c-1} \quad \text{where } c=a*d \tag{4}$$

Case 2. Considering the organized overlay network based on virtual region.

Let $PC(d^i) = 1 - \prod_{j=0}^{d^i} JSD_{ij}$, where JSD_{ij} is the semantic distance measured in JSD between the query and the jth interest region at level i in abstract tree (illustrated in Fig. 4). Then the probability of locating resource at exactly hop i in Fig. 4 is given as:

$s_i = PC(d^i)\prod_{j=0}^{i-1}(1-PC(d^j))$. Thus, $Q_t = \sum_{i=0}^{t} s_i$.

Each interest region is composed of several geographical groups, which implies that a hop in interest region will be performed by several geographical hops in this virtual region. So, in order to compare with the approach above [6], this paper converted the hop measured in interest region into geographical hop. Thus, in terms of geographical hops, $\overline{S_t} = \sum_{i=0}^{t} i\frac{s_i}{Q_t} * \overline{S_t^i}$, where $\overline{S_t^i}$ represents the average number of geographical hops experienced in interest regions in level *i* in Fig.4, which can be approximately evaluated using equation (3).

The average number of messages (including the messages sent in geographical groups located in same interest region) is evaluated in following way: If specific resource is not in root's interest region (that is, the current region that query peer belongs to), the required number of message equals *d* transmissions plus all messages within the *d* subtrees:

$$\overline{M_t} = PC(1) * \overline{M_g} + (1 - PC(1))(d + dm(t-1)) \quad (5)$$

where $\overline{M_g}$ denotes the average number of message transmission in current interest region, which may include several geographical groups, which can be approximately evaluated using equation (4).

For each of the *k* subtrees:

$$m(t-1) = PC(1) \times \overline{M_g} + (1 - PC(1))(d + dm(t-2)) \quad (6)$$

Using the above iterative equations, the total messages can be evaluated. Note that, because each interest region will be organized according to peer proximity in underlying network, the average number of geographical hops and average number of messages in current interest region can be evaluated similarly as the way in Case 1.

4.3 Numerical Simulations

The search performances in our virtual region-based P2P architecture were simulated, and compared with the geographical group-based P2P architecture. The simulation parameters were given in following table (the initial JSD between query and its original interest regions was assumed to be 0.3).

Table 1. The parameters used in simulation

Overlay type parameters	Geographical-based overlay network	Virtual region-based overlay network
Total # geographical groups (or interest regions)	5000 (groups)	500 (regions), each region includes average 10 geographical groups
Average # neighbors	4	3, geographical groups in same regions maintain average 3 geographical neighbors.

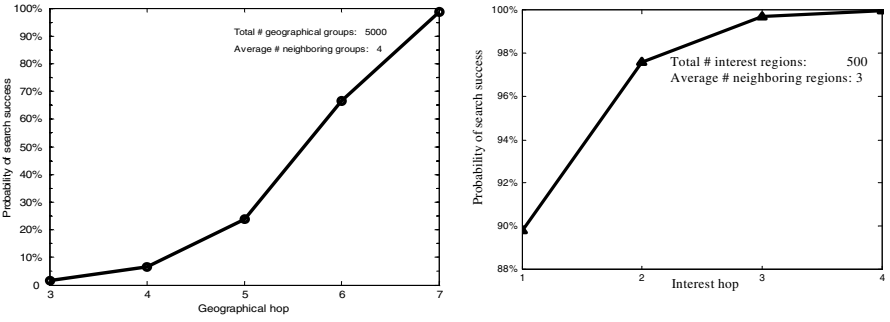


Fig. 5. Geographical hop vs. search success prob. in group (and region) based overlay

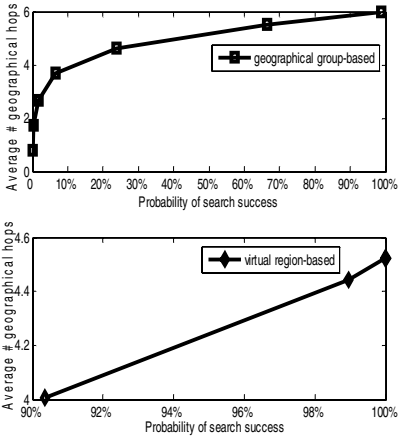


Fig. 6. Success Prob. vs. average # hops

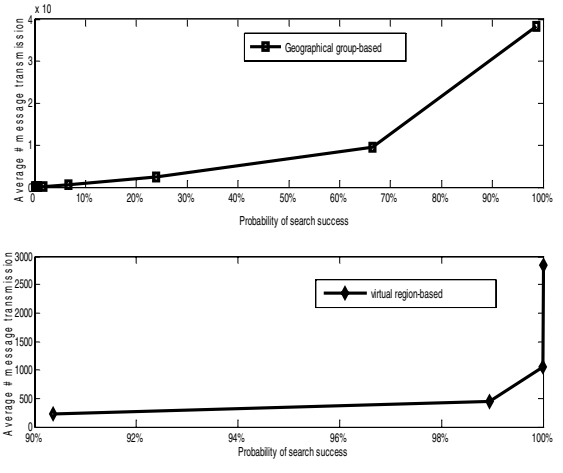


Fig. 7. Success Prob. vs. average # messages

From Fig.5 we can obtain that the search success probability in virtual region-based overlay is greatly larger than geographical-based overlay. The main reason is that our P2P architecture directs peer to the appropriate interest regions corresponding to its content semantic, so most queries can be satisfied by its current interest regions.

Fig. 6 (and Fig. 7) respectively denotes the relation between search success probability and average number of geographical groups (and average number of message transmission) in those two overlay networks. It is obtained that search overhead in our approach is significantly less than geographical-based overlay network. We think the reason why we achieve so great improvement, besides the inherent interest structure found out in our approach, is that we also neglect the maintenance overhead incurred in our architecture, which is a litter more than overhead in geographical-based overlay maintenance cost.

5 Conclusion

Generally speaking, organizing peers with similar shared contents into virtual community can improve the P2P search performance significantly. But, presently, most research on interest locality assumed certain static, pre-existing classification hierarchy, and then clustered peers according to this static structure, which breaks the inherent decentralized feature of P2P systems. Furthermore, constructing overlay network mapping well to the underlying infrastructure can greatly reduce the communication cost. Based on those observations, the virtual region-based P2P architecture was provided in this paper. In detail, this paper provided the adaptive interest region forming mechanism based on the JSD, and designed the DIL (Dynamic Interest Landmark) method to automatically direct peer to join the interest region closest to its content semantic. Furthermore, each virtual interest region was organized into several connected geographical groups according to locality-aware algorithm. The theoretical analysis and simulation results demonstrated the search performance in our virtual region-based architecture was significantly better the P2P architecture only based on locality-aware mechanism.

References

1. Eng Keong Lua, Jon Crowcroft, Marcelo Pias, Ravi Sharma, Steven Lim: A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. IEEE COMMUNICATIONS SURVEY AND TUTORIAL, 2004
2. Arturo Crespo, Hector Garcia-Molina: Semantic Overlay Networks for P2P Systems. Available at: http://www-db.stanford.edu/~crespo/publications/op2p.ps
3. Xiaole Bai, Shuping Liu, Peng Zhang, Kantola R.: ICN: Interest-based Clustering Network. In Proc. of the Fourth International Conference on Peer-to-Peer Computing (P2P'04)
4. Sripanidkulchai K., Maggs B., Zhang H.: Efficient Content Location Using Interest-based Locality in Peer-to-Peer Systems. In Proc. of IEEE INFOCOM'03
5. Tyson Condie, Sepandar D.K., Hector Garcia-Molina: Adaptive Peer-to-Peer Topologies. In Proc. of the Fourth International Conference on Peer-to-Peer Computing (P2P'04)

6. Xin Yan Zhang, Qian Zhang, Zhensheng Zhang, Gang Song, Wenwu Zhu: A Construction of Locality-Aware Overlay Network: mOverlay and Its Performance. IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, 2004.
7. Jain S., Mahajan R., Wetherall D., Borriello G.: Scalable Self-Organizing Overlays. Available at: http://www.cs.washington.edu/homes/sushjain/overlays.html
8. Majtey A.P., Lamberti P.W., Prato D.P.: Jensen-Shannon Divergence as a Measure of Distinguishability between Mixed Quantum States. arXiv:quant-ph/0508138, 2005
9. Atip A., Ramayya K., Smith M.D., Rahul T.: Interest-Based Self-Organizing Peer-to-Peer Networks: A Club Economics Approach. Available at http://ssrn.com/abstract=585345
10. Dimakopoulos V.V., Pitoura E.: A Peer-to-Peer Approach to Resource Discovery in Multi-Agent Systems. In Proc. the 8th International Workshop on Cooperative Information Agents, 2003 (CIA'03)

Scalable Resources Portfolio Selection with Fairness Based on Economical Methods*

Yu Hua[1], Dan Feng[1], and Chanle Wu[2]

[1] School of Computer Science and Technology
Huazhong University of Science and Technology
Wuhan, 430074, China
[2] School of Computer, Wuhan University
Wuhan, 430079, China
yhuastar@hotmail.com

Abstract. The fairness of scheduling resources are important to improve the whole performance. In this paper, we study the economy-based approach, *i.e.*, portfolio selection, to realize the dynamic allocation of distributed and heterogeneous resources. The portfolio selection method emphasizes the mean-variance model, which can evaluate the final return and help the scheduler to adjust the allocation policy. We present the practical algorithms for network nodes and Bloom filter-based surveillance, which can support the efficient adjustment of a scheduler.

1 Introduction

Portfolio selection is derived from the economical field and introduced by [1]. In nature, a market is a decentralized dynamic system, which holds all kinds of distributed and heterogeneous resources with the characteristic of competition. The purpose of the portfolio optimization problem is to present the feasible solutions for support an individual investor's decisions so as to allocate his wealth among various expenditures and investment opportunities over time, typically the investor's expected lifetime.

The mean-variance (M-V) portfolio selection model [2] aims to maximize the final wealth with the mean time to minimize the risk. The criterion of the risk was the variance in order to enable investors to seek highest return with the acceptable risk level. In addition, transaction cost has been also considered and become the important factor in decision making. The portfolio selection utilizes the mean of returns as the whole return and the variance of returns as the whole risks. Thus, the mean-variance model can evaluate the whole performance from the view of return and risks.

1.1 Applications

The portfolio selection is a kind of efficient method to coordinate and schedule the distributed and heterogeneous resources. It has been widely applied in the

* The work was supported by National Basic Research 973 Program under Grants 2004CB318201.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1019–1027, 2006.

economical fields and we utilize it into the network environments to support the efficient resources management.

Generally speaking, we need to allocate different amounts of subtasks of a task into the available network nodes and adjust the policy from time to time in order to adapt itself to new changes. The process should be fair and adapted in order to achieve the satisfactory performance. The portfolio selection method is fair because it is market-based approach and uses the price-based implementation. Meanwhile, the portfolio selection utilizes the mean-variance model to compute the current achievement and proceeds to schedule the allocation plan according to the results. Thus, it is a multi-stage and adapted approach.

1.2 Motivations

Although the portfolio selection is efficient in scheduling and allocating resources, the communications are often overloaded with higher costs and thus we need a kind of simple and efficient data structure, which can reduce the communication cost and provide the quick responses from servers or clients. Meanwhile, the nodes themselves need to compute the effect of their changed returns. They can maintain many copies of returns from other nodes. When the current node finds the new return is larger than the threshold, it can invoke a new adjustment.

1.3 Our Contributions

In this paper, we make the main contributions as follows. First, we show the portfolio-based economical model, which can represent the capacities of current nodes and support the fair and adaptive resources portfolio selection in the network environments. The capacities can be modeled into the returns, which are random variables.

Second, we present the practical approach, which uses the Bloom filter to exchange information. The Bloom filter contains the hashed values of the returns from all nodes in the previous adjustment, which are transmitted in the whole network. When the return value of a node does not appear in Bloom filters, Bloom filters need to count it.

Once the number of lost nodes reaches a threshold, Bloom filters may trigger an adjustment, in which a node gets a new proportion. Meanwhile, a node can also maintain a record of returns and compute the new final return when it itself produces a new return. Once the value of the new return minus old one is larger than a certain threshold, the node may request the adjustment for Bloom filters. Thus, in our proposed approach, we can launch an adjustment from the values of Bloom filters or the nodes themselves.

1.4 Roadmap

In section 2, we introduce the related work and some backgrounds on existing approaches. The preliminaries in Section 3 includes the basic definitions and explanations about the portfolio selection model and Bloom filters. Section 4

shows the analytical model, which consists of a scheduler, Bloom filter and network nodes. We describe the practical operations in Section 5 and show two algorithms to support the operations in Bloom filter and network nodes. Section 6 concludes our paper.

2 Related Work

In this section, we will present the literatures about the economical methods and a Bloom filter, which is space-efficient data structure as the carrier of transmission.

2.1 Economical Methods

Decision of resources selection in commercial context is necessary and important because the network system needs timely to allocate and schedule all kinds of sub-tasks based on the information of distributed and heterogeneous resources, which have different time intervals available. As a result of the cost and time constraints, the adjustment between time periods was necessary and should satisfy the rule of maximized return and minimized risks.

In [3], authors presented a decentralized auction-based approach to the edge-allocated bandwidth and conducted a game theoretic analysis. The proposed method could derive optimal strategies for buyers and brokers, and achieve the market-based equilibria. Furthermore, authors in [4] revealed the connections between discrete-time models and continuous-time counterparts. They introduced an aggregated process with smaller state-space for realizing the Markov chain and reducing the complexity and formulated the underlying portfolio selection as a two-time-scale problem.

In [5], authors considered the portfolio selection problem with interval coefficients as a multiple objective problem. Moreover, the uncertainties could be classified into optimistic or pessimistic solutions. Authors in [6] considered the transaction costs of portfolio selection and supposed that there were charges on all transactions with a fixed value. They also considered that the transactions were related to a nonlinear free boundary problem, which was solved by their proposed algorithm.

2.2 Bloom Filter

A Bloom filter is a kind of space-efficient data structure to represent a set and realize the membership query, which was introduced in [7]. Counting Bloom filters [8] replace an array of bits with counters in order to count the number of items hashed to that location. This approach can support the deletion operation very well.

Authors in [9] proposed the compressed Bloom filters, which can achieve a smaller false positive probability using a function of compressed size. Furthermore, space-code Bloom filters [10] and spectral Bloom filters [11] can be used to describe a multiset. Based on the methods, they could realize the occurrence query based on the analysis of a data set.

3 Preliminaries

In this section, we introduce the problem statement and basic definitions about the portfolio selection and Bloom filters. These contents are the foundation of our analysis and practical operations.

3.1 Problem Statement

We consider n network nodes with certain computation and storage capacities as the choices for finishing a special task. The network nodes are heterogeneous, which have different capacities and risks of use. Hence, we need to select the available resources and combine them according to the feasible proportion. Meanwhile, the network environments with the dynamic character need the adaptation to support and improve the whole performance.

The problem can be represented as follows. There are n network nodes and the ith $(1 \leq i \leq n)$ node can represent its computation and storage capacities as a return, r_i, which is a random variable. The expected value and deviation are $E(r_i)$ and $D(r_i)$, respectively. We need to select the available resources and assign $x_i, (0 \leq x_i \leq 1)$ proportional subtasks to the ith node. Our optimal purpose is to maximize the final return with the acceptable risk level. The approach can support to finish the task in an efficient way.

3.2 Portfolio Selection

In the network environments, the optimization of portfolio selection is rather important to realize the resource management and scheduling efficiently. Real market and network environments have similar characteristics in resources management and tasks scheduling, which have the resources owners, resources users and various available resources.

The return and risk are the mean of resource ability and covariance between capacities. Concretely speaking, $E(r_i)$ stands for the mean of the return of node i. The covariance between node i and j can be represented as $\delta_{i,j}$. Because the operation of adjustment needs to occupy many extra resources, such as computation, storage and transmission ones.

Thus, the adjustment should satisfy the certain constraints, *i.e.*, thresholds. When the current values are larger than the thresholds, we need to execute the operation of adjustment. Portfolio optimization aims to realize the resource selection efficiently and task scheduling dynamically.

The classical portfolio model, which appeared in [2], is

$$Maximize\{(1-\omega) \times \sum_{i=1}^{n} r_i x_i - \omega \times \sum_{i=1}^{n} \sum_{j=1}^{n} \delta_{ij} x_i x_j\} \tag{1}$$

$$\sum_{i=1}^{n} x_i = 1, x_i \geq 0, i = 1, 2, \ldots, n \tag{2}$$

where n is the number of securities, x_i is the proportion of investment in security i, r_i is the expected returns in security i, δ_{ij} is the covariance of expected returns

between security i and j, ω is used to describe the aversion degree to risks, $0 \leq \omega \leq 1$.

The different values of ω can reflect the trends of investors for achieving the returns with different approaches. For example, the investors may be extremely conservative and only emphasize the risk of the investments. Thus, they might ignore the potential the returns. We can use $\omega = 1$ to describe the trend. On the other hand, if the investors want to pursue the returns of the investments regardless of the risks. The scenario may be described using $\omega = 0$.

3.3 Bloom Filter

The paper [7] introduced a Bloom filter, which is a bit array of m bits for representing a set $S = \{a_1, a_2, \ldots, a_n\}$ of n items. In order to record the hashed values, all bits in an array are initially set to 0. A Bloom filter uses k independent hash functions $\{h_1, \ldots, h_k\}$ to map the set to the bit address space $[1, \ldots, m]$. For each item a, the bits of $h_i(a)$ are set to 1.

We use the following result from a popular theorem to describe the characteristic of Bloom filters. The theorem is introduced in a survey of Bloom filter [12].

Theorem 1. *The false positive probability in the standard Bloom filter is* $f = (1 - e^{-\frac{kn}{m}})^k$. *The probability can obtain the minimum,* $(1/2)^k$ *or* $(0.6185)^{m/n}$, *when* $k = (m/n)\ln 2$.

The proof has been shown in [12]. For clearly describing the following parts, we abbreviate the proof and give some important conclusions.

First, we assume that hash functions are perfectly random and the hashed values are uniformly distributed. Then, when a particular bit in the standard Bloom filter is set to 1 by a hash function, the probability is

$$1/m \tag{3}$$

Furthermore, when a bit is not set by any of k hash hash functions, the probability becomes

$$(1 - \frac{1}{m})^k \tag{4}$$

After inserting n items into the Bloom filter, we can deduce that the probability that a bit is still 0 is

$$(1 - \frac{1}{m})^{kn} \approx e^{-\frac{kn}{m}} \tag{5}$$

Therefore, the false positive probability of Bloom filter is

$$f = (1 - (1 - \frac{1}{m})^{kn})^k \approx (1 - e^{-\frac{kn}{m}})^k \tag{6}$$

Moreover, it is easy to check that the derivative is 0 when

$$k = \ln 2(m/n) \tag{7}$$

Correspondingly, we can obtain the minimum of the false positive probability

$$minimize(f) = (1/2)^k \approx (0.6185)^{m/n} \tag{8}$$

4 Analytical Model

The analytical model consists of three parts: a scheduler, Bloom filters and network nodes, which are shown in Figure 1. Scheduler is responsible for allocating the proportions of subtasks executed, *i.e.*, $x_i, (1 \leq i \leq n)$, to different network nodes. Bloom filters contain the return values of all network nodes, *i.e.*, $r_i, (1 \leq i \leq n)$ in the previous adjustment.

The Bloom filter routes all the network nodes one by one according to the special path, which is derived from the predefined routing table. Furthermore, the Bloom filters are used to check how many return values have been changed. At the same time, a network node maintains its current return value and stores the previous values of other nodes. Thus, a node can compute the current final return according its latest value.

4.1 Economical Method

In the analytical model, we represent the capacities of network nodes as their returns, which can support the portfolio selection. We can compute the final return according to the equation 1 and 2. Thus, each node can compute the current return, which includes the changes of its return.

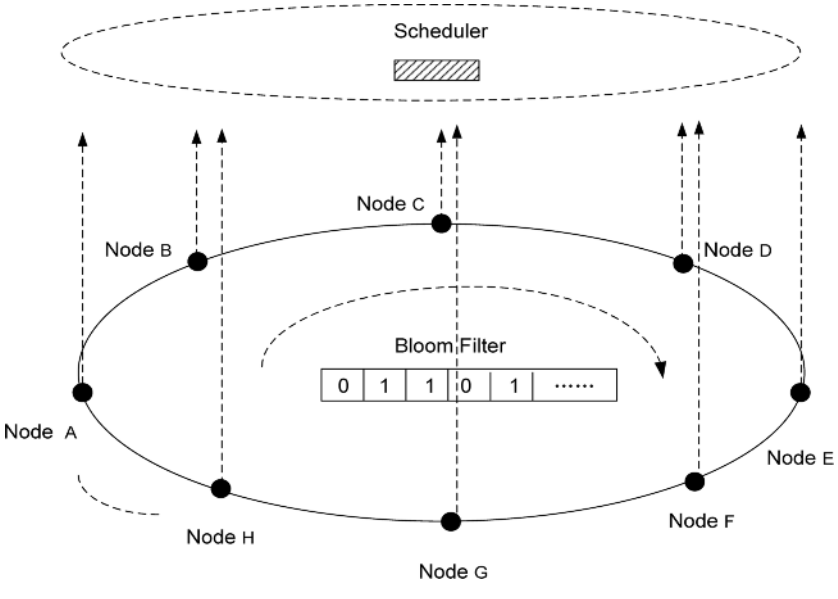


Fig. 1. The analytical model based on economical methods

Once the new value exceeds the predefined threshold, ξ, the node will trigger a request for adjustment to the scheduler. Afterwards, the scheduler reallocates the new proportion to each node.

4.2 Bloom Filter

We consider the Bloom filter as an auxiliary structure, which is responsible for checking the degree of changed returns among all the network nodes. First, we store the initial return values into the Bloom filters based on the computation of hash functions. Second, the Bloom filter routes the special path and checks each node in order to verify whether the return of a node has been changed.

In this process, we use the return value r_i of node i as the input of hash functions $\{h_1, \ldots, h_k\}$ and check whether all $h_j[r_i], (1 \leq i \leq n, 1 \leq j \leq k)$ equal to 1. When all $h_j[r_i]$ are 1, we can say that the return r_i is unchanged with a certain false positive probability. Otherwise, we can say that the node has changed its return value.

Meanwhile, Bloom filter needs to record the number of nodes with changed values. Furthermore, when the number exceeds the threshold, ζ, Bloom filter can send a request to the scheduler for adjusting the allocated proportions.

5 Practical Operations

In this section, we present the practical operations in network nodes and Bloom filter, which can trigger the adjustment of a scheduler.

Figure 2 shows the surveillance algorithm of network nodes and its main function is to compute the current final return. Consider a network node, a, with return r_a and its mean and variance values are M_a and V_a, respectively. Thus, the proposed algorithm can compute the final return, R_a. When R_a is

Node_Surveillance (Input: r_a)

Initialize $V_a = 0, M_a = 0$
Update the return value, r_a
for $(i = 1; i \leq n; i++)$ **do**
  for $(j = 1; j \leq n; j++)$ **do**
    Compute $V_a = V_a + \delta_{i,j} x_i x_j$
  end for
  Compute $M_a = M_a + r_i x_i$
end for
Compute $R_a = (1 - \omega) M_a - \omega V_a$
if $R_a > \xi$ **then**
  Send a request message to scheduler
end if

Fig. 2. The algorithm for network nodes surveillance

Bloom Filter_Surveillance (input: r_a, λ)

```
Initialize V_a = 0, Flag = True
for (j = 1; j ≤ k; j++) do
  Compute H_[j](r_a)
  if BF[H_[j](r_a)]==0 then
    Flag = False
    break
  end if
end for
if Flag = False then
  λ := λ + 1
  if λ > ζ then
    Send a request message to scheduler
    λ := 0
  end if
end if
Return λ
```

Fig. 3. The algorithm for Bloom filter surveillance

larger than the predefined threshold, ξ, the node can send a request message to the scheduler for executing a new policy of proportion allocation.

Figure 3 shows the algorithm of Bloom filter surveillance for verifying the hashed values in each node. The algorithm needs the current return of network node a, *i.e.*, r_a and the number of changed nodes, λ. We need to compute the hashed values of return r_a based on the computation of hash functions. Each location of Bloom filter, $BF[H_{[j]}(r_a)]$, is probed and if any value equals to 0, we can determine that the return value has been changed. Otherwise, the return is invariable with certain false positive probability. Afterwards, we proceed to add the number of changed nodes to λ. If λ is larger than the predefined value, ζ, the Bloom filter can send a request message to the scheduler for an adjustment operation.

6 Conclusion

Fairness is an important characteristic of the efficient scheduling policy. The economical methods can provide the fair approach and realize the adjustment with adaptation. In this paper, we study the problem of scalable resources portfolio in the network environments. We present the practical mechanisms and algorithms based on the theory of portfolio selection for providing the efficient adjustment.

Each node can execute the computation based on the changed returns and when the changes reach a certain threshold, the node can trigger a request message to the scheduler for conducting a new allocation of assigned proportion. Meanwhile, we utilize the Bloom filter to collect the number of nodes with changed returns. In the similar way, when the number of nodes collected is larger

than a certain threshold, Bloom filter can initiate an adjustment, which leads to a reallocation of the proportions.

References

1. Markowitz H.: Portfolio Selection. The Journal of Finance, **7** (1952) 77-91
2. Markowitz H.: Portfolio Selection: Efficient Diversification of Investments. Yale University Press, (June 1971)
3. Semret, N., Liao R.F., Campbell A.T., Lazar A.A.: Pricing, provisioning and peering: dynamic markets for differentiated Internet services and implications for network interconnections. IEEE Journal on Selected Areas in Communications, **18** (Dec. 2000) 2499–2513
4. Yin G., Zhou X.Y.: Markowitz's mean-variance portfolio selection with regime switching: from discrete-time models to their continuous-time limits. IEEE Transactions on Automatic Control, **49** (March 2004) 349–360
5. Ida M.: Mean-variance portfolio optimization model with uncertain coefficients. Proceedings of the 10th IEEE International Conference on Fuzzy Systems, (Dec. 2001) 1223–1226
6. Davis M. H., Norman A. R.: Portfolio Selection with Transaction Costs. Mathematics of Operations Research, (Nov. 1990) 676–713
7. Bloom B.: Space/time Trade-offs in Hash Coding with Allowable Errors. Communications of the ACM, **13** (1970) 422–426
8. Fan L., Cao P., Almeida J., Broder Z.A.: Summary cache: a scalable wide area web cache sharing protocol. IEEE/ACM Trans. on Networking, **8** (2000) 281–293
9. Mitzenmacher M.: Compressed Bloom filters. IEEE/ACM Trans. on Networking, **10** (2002) 604–612
10. Kumar A., Xu J., Wang J., Spatschek O., Li L.: Space-Code Bloom filter for efficient per-flow traffic measurement. Proceedings of the twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), **3** (2004) 1762–1773
11. Saar C., Yossi M.: Spectral Bloom filters. Proceedings of the 2003 ACM SIGMOD international conference on Management of data, (2003) 241–252
12. Broder A., Mitzenmacher M.: Network applications of Bloom filters: a survey. Internet Mathematics, **1** (2005) 485-509

Personalized u-Portal System with Ontology and Web Services

Eun-Ha Song[1], Yang-Seung Jeon[1], Dae-Keun Si[1], Laurence T. Yang[2], Young-Sik Jeong[1], and Sung-Kook Han[1]

[1] Dept. of Computer Eng., Wonkwang Univ., 344-2 Shinyoung-Dong, Iksan, 570-749, Korea
{ehsong, globaljeon, sdk124, ysjeong, skhan}@wku.ac.kr
[2] Department of Computer Science, St. Francis Xavier University, Antigonish, NS, B2G 2W5, Canada
lyang@stfx.ca

Abstract. Most portal systems provide simple web services in existing web environment but cannot support semantic web environment. Moreover, current portal systems are not adequate to compose personalized services. We also need semantic portlet search methods using portlet ontology, which can support ubiquitous environment, as well as portlet API that supports users' access to u-Portals through various types of terminal equipment. This paper constructs a user-centered customized web service u-portal system that supports semantic web service search and composition through a ubiquitous portal engine.

1 Introduction

Many researches are going on today in order to improve the quality of web services, and some of them on creating and exchanging information. As this concept is getting extended, now the web is providing infrastructure environment that has not been available in simple information infrastructure in the past. An example of such infrastructure is a semantic portal that uses portal, web service and ontology technologies.

A portal system is a next-generation enterprise information technology that bases integrated information systems, and its market is an arena of hot competition for preoccupation. Portal systems support functions essential to portals such as information resource management, collaboration support and personalization. Various portlet APIs have been developed for portal components in portal systems. However, the hardly interchangeable component interface brings many problems to application suppliers, portal consumers and portal server developers. To solve these problems, JSR168 Java portlet API and WSRP(Web Services for Remote Portals) standard were proposed. These standards provide interoperability among web services and between portlets and portal frameworks through visual interface.

Ontology is a key technology for upgrading the system function from existing information processing to knowledge processing, and enables web-based knowledge processing, knowledge sharing among application programs and knowledge reuse. The application of ontology to a portal system also allows the transition of existing

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1028 – 1037, 2006.

web sites into knowledge portals. That is, web sites mainly for the exchange and expression of information evolve into knowledge portals where users can collaborate and interact with one another. Moreover, ubiquitous environment, which enables the real-time exchange of information at any time and in any place, is emerging as a new information communication paradigm in the world. In the ubiquitous age, all electronic devices are compounded, fused and networked for more convenient and affluent life. We need to make ground research for realizing customized u-Society that implements personalized web services, which are new IT systems, by applying next-generation technologies such as web service technologies and ubiquitous environment [8,9].

This paper applies web service automatic generation system, ontology technology and ubiquitous environment to ordinary portal systems. We design a *Personalized u-Portal system* that can be executed simply by serving all and supporting web service composition, through which users can compose web services regardless of time, place and terminal equipment. Portlet ontology enables the upgrade of system from information processing to knowledge processing and semantic web service search and composition. Moreover, it is designed for the application of semantic web technology, which is the base of ubiquitous systems, in consideration of future functional expansion. Furthermore, we design a user-centered *Personalized u-Portal system* that supports semantic web service search and composition by providing portlet API, which enables users to access personalized u-Portal through different types of terminal equipment.

2 Related Works

There are active researches on semantic portal system and efforts to apply semantic web technologies.

Esperonto uses metadata as the platform of WebODE ontology, and utilizes file systems to store information. However, it cannot support groupware for the environment of collaboration and composition environment.

OntoWeb, which is based on ZOPE/CMF framework, maintains workflow for publication, user concepts and folders for personalization. Ontology is developed and utilized based on ZOPE contents and it provides user interface of relatively high completeness.

Empolis K42 is based on topic maps but its development has not been finished. A number of APIs are proposed to use the portal system, and there are editors for developing and editing topic maps and independent graph viewers in hyperbolic tree structure.

Most systems do not have ontology that can manage portlets, and although there have been many researches on web service composition none of them has been applied to portals. This paper designs a *personalized u-Portal system* embedded with a web service composition system for portlet ontology and personalization. Table 1 compares the proposed system with other semantic portal systems under study or development.

Table 1. Comparison of semantic portal site

System	Remark
Esperonto	WebODE Ontology platform ▪ Ontologies as meta data structure and organization principle Pure file system to store documents No group ware / collaboration features Poor user interface No Composition
OntoWeb	Based on ZOPE / CMF Framework ▪ Publishing workflow, user concept, private folders, … Ontology classes mapped to ZOPE content types Matured user Interface
Empolis K42	Infrastructure – not a complete portal ▪ Based on Topic Maps ▪ API to access it ▪ Topic Maps editing component, stand visualization techniques like hyperbolic tree
Mondeca ITM	Different connectors to DMS Meta-Ontologies ▪ Modeling: Structure, Constraint, Access Gather Meta data: ▪ Linguistic tools(automatic) ▪ MS Office meta data(semi automatic) ▪ Via forms(manual)
others	SWWS ▪ Protégé + RDQL Mind Swap ▪ Embedded RDF docs OntoWeb Edu. ▪ CGI script generating HTML forms from an Ontology
Personalized u-Portal	Jakarta Struts Framework ▪ Protégé Implementation of Portlets ▪ JSR168 ▪ WSRP ▪ OWL Ontology Parsing ▪ Jena 2.1, Xerces 2 XML API Axis Engine

3 Personalized u-Portal System

3.1 Architecture of Personalized u-Portal System

The architecture of the personalized u-Portal system designed in this paper is as in Fig. 1, and the details of its major components are as follows.

JSR168 standard provides interoperability among web services, portlets and portals through visual interface. Struts is framework for the construction of web applications, and a flexible control layer based on standard technologies such as Java Servlet, Java Beans, Resource Bundles and XML. Struts itself provides controller components, and incorporates with other technologies to provide models and views. In the aspect of

model, it interacts with standard data access technologies such as Enterprise Java Beans, JDBC and Object Relational Bridge. If a SOAP message arrives, the engine that supports old type web services simply plays the role of a broker connecting the message to a backend object. Accordingly, it is not easy to do preprocessing, which should be done before the object is called. Having a handler to deal with the shortcoming, AXIS Engine is utilized as a container in charge of web service of EIP. Cocoon provides flexible web publication environment by separating contents, logics and styles, and is a part supporting ubiquitous technology. Cocoon does the job through the centralized setting system and the sophisticated casing function, and it helps the development, distribution and maintenance of robust XML server applications. Cocoon uses almost every kind of data resources including file system, RDBMS, LDAP, native XML database and network-based data. Moreover, it supports ubiquitous system operation environment by transmitting contents in the WML format. Portlet ontology uses OWL (Web ontology language) to describe ontology. Ontology processing is done using Jena 2.0 ontology API of HP Research Institute, RDF API of Stanford University, and Xerces 2 a Java XML parser.

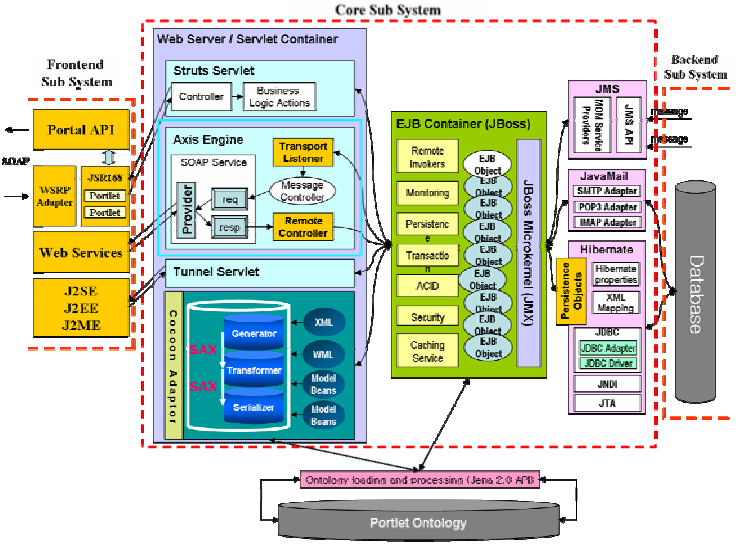


Fig. 1. Architecture of personalized u-portal system

3.2 Cockpit Functions

The present system includes the basic functions of a portal system, and maximizes interoperability among heterogeneous systems by adopting WSRP and JSR168 standard technologies for the conversion of portlets to web services. In addition, it collects web services opened to the public on the web and provides environment for the use of new web services through the composition of web services with limitation. That is, users can compose and use various services utilizing the visual web service composition system. Functions provided by the system are as follows.

- Personalization: Personalization is a key function for the usability of EIP system. A user constructs a system suitable for his/her purpose by dragging and dropping various services in the service repository.
- Collaboration: Collaboration is a useful tool for enhancing the work efficiency of individuals and organizations. It supports essential functions for collaboration environment such as information sharing and the integration of business processes.
- Search: Because a portal system supports the generation, sharing and distribution of diverse information resources – typical/atypical documents and contents, business processes and applications – these resources need to be managed efficiently. In addition, the search function is essential in information resource management.
- System integration: A portal system requires the integration not only of formative heterogeneity but also of semantic heterogeneity.
- System security and management: System security is connecting users to portal system resources and, at the same time, managing resources and users in order to protect data in the portal and portlets from unauthorized accesses. Moreover, it provides functions for protecting and managing resources without lowering productivity.
- Publishing and distribution: The system provides functions such as the separation and publication of contents so that users without expert knowledge in IT technology can produce, process and manage information with ease.
- Community: This function is to collect users' opinions and post them in the bulletin board.
- Single Sign On: All resources in a portal system should be accessible through a single logging on. This provides perfect security and reduces the burden of memorizing ID and password. If these are written down they can be exposed to others, but management through SSO reduces the risk.

3.3 Portlet Ontology

The scope of portlet ontology includes portlet system classification and portlet information management. Fig. 2 shows the conceptual elements and relational structure of portlet ontology. Rectangles denote classes of portlet ontology, and cylinders are XML schema types. Lines connecting two classes or a class and an XML scheme type indicate class properties. In this paper, portlet ontology is divided into 13 classes and 65 sample portlets are extracted.

In portlet ontology, the most essential classes are category and portlet. The category class describes basic information for categorizing portlets, and the portlet class describes basic information of each portlet. Besides, there are classes such as supports that describes portlet MIME-TYPE and portlet operation mode, init-params that describes initial parameters for executing portlets, security-role-ref that describes information on portlet security and authority, portlet-preferences that describes preferred environment for the operation of portlets, and user-attribute that describes basic information on users necessary for the use of portlet information in the portal system. Portlet ontology in Fig. 3 shows a part of ontology.

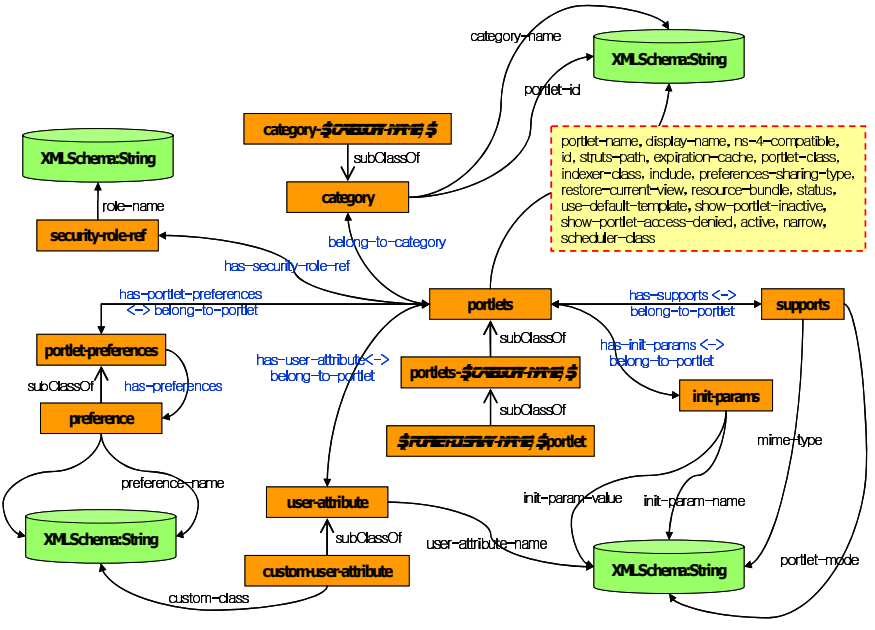


Fig. 2. Portlet ontology relational diagram

```
<?xml version="1.0"?>
<rdf:RDF xmlns="urn: sws.wonkwang.ac.kr/swpt#"
       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
       xmlns:owl="http://www.w3.org/2002/07/owl#"
       xml:base="urn:sws.wonkwang.ac.kr/swpt">
  <owl:Ontology rdf:about="urn:sws.wonkwang.ac.kr/swpt"/>
  <owl:Class rdf:ID="polls-portlet">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="portlets-polls"/>
    </rdfs:subClassOf>
</owl:Class>
       … …
<owl:ObjectProperty rdf:ID="has-supports">
    <rdfs:range rdf:resource="urn:sws.wonkwang.ac.kr/swpt#supports"/>
    <rdfs:domain rdf:resource="urn:sws.wonkwang.ac.kr/swpt#portlets"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="category-finance">
    <rdfs:subClassOf rdf:resource="urn:sws.wonkwang.ac.kr/swpt#category"/>
</owl:Class>
</rdf:RDF>
```

Fig. 3. Portlet ontology

3.4 Personalized Generation and Composition

With the development of semantic web technology, web resources became accessible based on their contents without using keywords. The most important web resources

are services. Services do not mean just the provision of information through web sites but they also include explicit actions such as driving physical equipment. This paper adopts a technology that collects established web services and generates portlets in the form of web services. That is, established web services are stored in the web service repository using a crawler. In addition, web services stored or composed are provided by users' request. Fig. 4 is a scenario of web service generation and composition in our system.

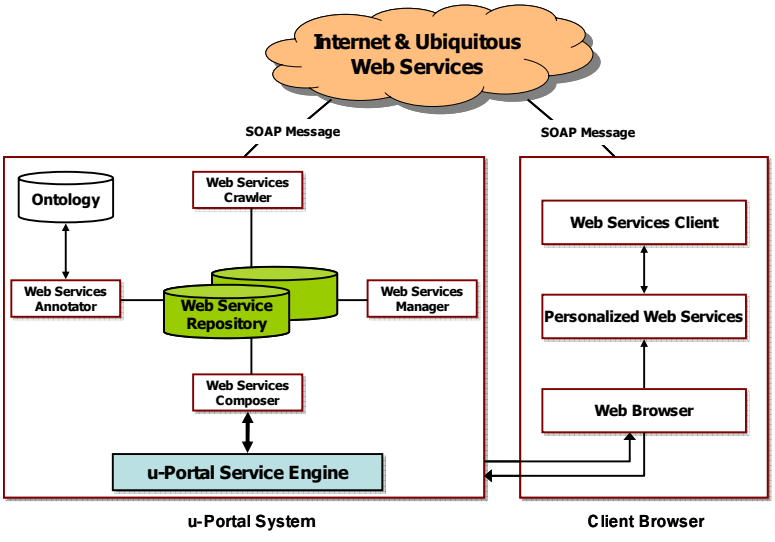


Fig. 4. Control flow of personalized web service generation and composition

The web service crawler agent collects web services opened to the public, converts them into portlets and construct them in the web service repository using ontology. Personalized web services are generated automatically by the u-Portal engine, and users compose and execute various web services by themselves. Created web services are converted to XML through Cocoon framework and provided to users regardless of mobile equipment type. Web services requested by users are searched for and, if not found, new web services are composed and the results are returned to the users. To compose a new web services, services to be composed are selected.

4 An Example

Fig. 5 is a screen for service composition. The Action and Object of services to be composed are selected. When an Action and its corresponding Object are selected and the 'Add' button is clicked, the web service composer analyzes web services of the selected Action and Object, extract them and add them to the composition service list. Because 'composition' implies the combination of at least two services, two or more services should be added to the composition service list.

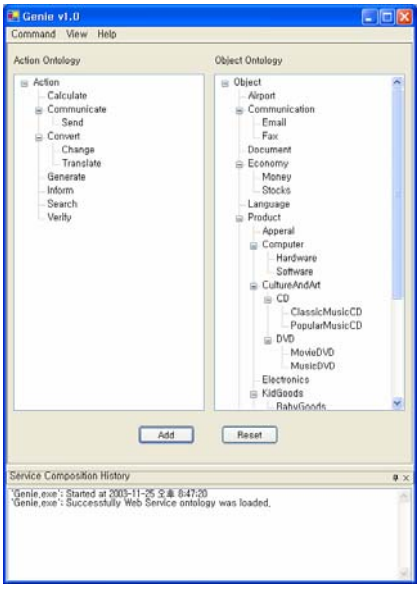


Fig. 5. Web services composition view

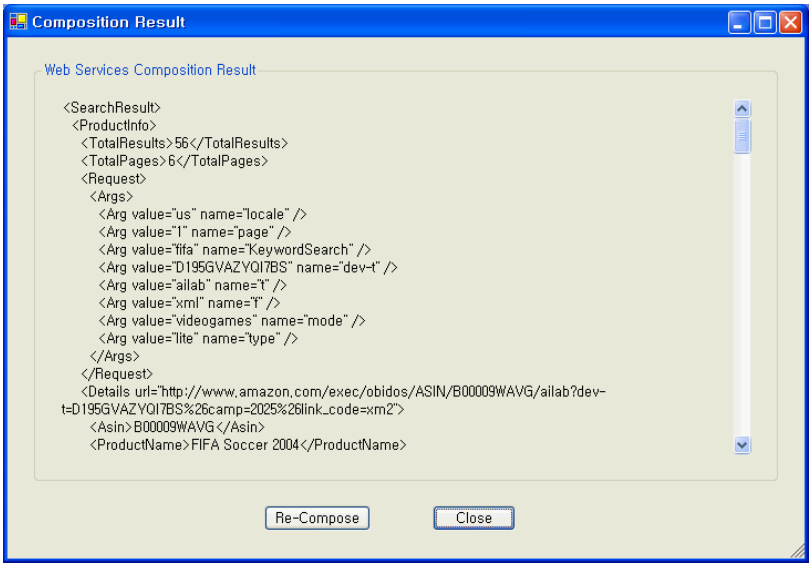


Fig. 6. The result of services matching

Fig. 6 shows the result of matching and composing AirportHumidity service that provides data on humidity in airports, Amazon GameSearch service that finds game products in the Amazon site, and EmailSend service that sends emails. Tag <TotalResults> is the total number of results, <Args> the type of results, and < Details url> the results of composition.

5 Conclusions

Current portal systems are constructed following standard technologies such as JSR and WSRP. Most of EIP integration solutions, however, adopt neither JSR168 and WSRP nor ontology technology and ubiquitous environment.

This paper constructs portlets following JSR168 standard and uses WSRP 1.0 protocol for communication between the portal and portlets. To classify and manage portlet information in the portal system, we constructs portlet ontology. The application of portlet ontology to a portal system enables organic interoperation between the portal and each portlet based on semantic processing and supports knowledge-based portlet processing. In addition, portlet ontology supports ubiquitous environment utilizing Cocoon framework. In this way, when personalized u-Portal system with ontology-based web services is introduced to corporations, e-commerce and enterprise portal companies, it can save the cost of development through the reuse of existing resources and enhance R&D capacities and competitiveness by providing advanced information services.

Acknowledgement

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (the Center for Healthcare Technology Development, Chonbuk National University, Jeonju 561-756, the Republic of Korea).

References

1. Java Community Process(JCP), JSR 168: Portlet Specification, http://jcp.org/en/jsr/ detail?id=168
2. OASIS Web Services for Remote Portlets TC, http://www.oasis-open.org/committees/ tc_home.php?wg_abbrev=wsrp
3. Christopher C. Shilakes and Julie Tylman, Enterprise Information Portals, Merrill Lynch, Inc., NY (1998)
4. Viador, "Enterprise Information Portals: Realizing The Vision Of Information At Your Fingertips, AViador, Inc. White Paper, San Mateo, CA (1999)
5. Tim Berners Lee, J. Hendler, and O. Lassilla, The Semantic Web, Scientific American, Vol.284, No.5 (2001) 34-43
6. Ying Ding and Dieter Fensel, The Semantic Web: Yet Another Hip?, Data & Knowledge Engineering, Vol.41, No.2-3 (2002)
7. Dean, M., D. Connolly, F. Harmelen, J. Hendler, I. Horrocks, D. McGuinness, Web Ontology Language (OWL) Reference Version1.0., http://www.w3.org/TR/owl- ref/ (2002)
8. Shankar R. Ponnekanti, Brian Lee, Armando Fox, Pat Hanrahan, and Terry Winograd ICrafter: A Service Framework for Ubiquitous Computing Environments, Proceedings of Ubicomp (2001)
9. Jean Scholtz and Asim Smailagic. Workshop on Evaluation Methodologies for Ubiquitous Computing, Ubicomp (2001), Report published in SIGCHI Bulletin

10. The Apache Struts Web Application Framework, http://jakarta.apache.org/struts/.
11. Apache Cocoon, The Apache Cocoon Project, http://cocoon.apache.org/.
12. Axis, Apache Axis (2001) http://xml.apache.org/axis/.
13. S.Staab et al. AI for the Web - Ontology-based Community Web Portals. Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, Austin, Texas, USA(2000)

Augmented Video Services and Its Applications in an Advanced Access Grid Environment

Ying Li, Xiaowu Chen, Xiangyu Ji, Chunmin Xu, and Bin Zhou

The Key Laboratory of Virtual Reality Technology, Ministry of Education,
School of Computer Science and Engineering, Beihang University,
100083 Beijing, P.R. China
{liying, chen}@vrlab.buaa.edu.cn

Abstract. The video services of most Access Grid (AG) toolkits still only employ outdated Video Conferencing Tool to transmit real-time video, and hardly have efficacious way to mix the 3D graphics model of grid results and the live videos of AG session into one scene for grid users' collaborations. This paper introduces the research work about augmented video services (AVS) of CGAG (ChinaGrid Access Grid) based on the AG framework. It shows up a reasonable way on providing augmented video services in an advanced Access Grid environment, and also a new approach to share grid resources in Grid applications. At the end of this paper, a typical sample is given to demonstrate the capabilities of this AVS, and the result proves that it yields superior quality compared to the traditional webpage of grid portal and usual interaction modals of AG.

1 Introduction

Access Grid (AG) is a typical network-based collaborative environment over high-performance networks, providing group-to-group interactions across the Grid with an ensemble of resources including multimedia large-format displays, interactive conferencing tools, and interfaces to grid middleware [1]. With most of generic AG toolkits, the video services for the AG still employ outdated VIC (Video Conferencing Tool) to transmit real-time video only, hence it gives participants low interactivity of pure video conferencing without good QoE (quality of experience), and lacks of interactions with grid resources or grid services. For example, grid users not only receive some results including a 3D graphics model in a grid application, but also participate in an AG session through live videos displaying the real environments to make discussions, interactions, and references together. Furthermore, these grid users hope to collaboratively refer and operate a part of the 3D graphics model in the live videos

This paper is supported by National Natural Science Foundation of China (60503066), Program for New Century Excellent Talents in University, Project of National Development and Reform Commission of China (Application Demonstrations of Next Generation of Internet, CNGI-04-15-7A), National Research Foundation (5140*305*), the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, Program for New Stars in Science & Technology at Beijing (200411A), China Education and Research Grid Program (ChinaGrid).

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1038 – 1047, 2006.

environment of AG session. So it is necessary to mix the 3D graphics model of grid results and the live videos of AG session into one scene for grid users' collaborations, and also it shows up how to find a reasonable way on these mixtures for providing this kind of augmented video services included by an advanced Access Grid environment.

ChinaGrid (China Education and Research Grid Program) has been launched to provide the nationwide grid computing platform and services for education and research purpose among hundreds of universities in China since 2003 [2]. In the first stage of this program, it includes five main grid computing application platforms, such as bioinformatics grid [3], image processing grid [4], computational fluid dynamics grid [5], course online grid [6], and MIPGrid (massive information processing grid) [7].

UDMGrid (University Digital Museum Grid), a typical information grid included by MIPGrid, has been developed to provide one-stop information services about kinds of digital specimens in the form of grid services [8]. Now most results of grid services in UDMGrid usually describes the digital specimens in the form of word, data, image, graphics, audio, video, and so on. For example, the 3D graphics model of a Dragon-in-Cloud crock of Qing Dynasty shows up in the results of UDMGrid Services.

CGAG (ChinaGrid Access Grid) project founded by ChinaGrid program targets an advanced collaborative environment [9] to provide interactive, collaborative, and immersive access of grid resources based on grid middleware CGSP (ChinaGrid Support Platform) [10], AG framework, and some advanced Human-Computer Interaction technologies. It includes a very important component, which provides above kind of augmented video services in an advanced Access Grid environment.

This paper introduces the research work about the augmented video services (AVS) of CGAG, it presents an AVS couple composed of one augmented video producer service (AVPS) and one augmented video consumer service (AVCS) in CGAG based on the AG framework. Finally, a sample for UDMGrid application is given, running in CGAG for demonstrating the capabilities of our AVS, and the result proves that it yields superior quality compared to the traditional webpage of grid portal and usual interaction modals of AG.

The remainder of this paper is organized as follows. Section 2 briefly surveys related works. Section 3 presents the architecture of CGAG, explaining the AVS' position in CGAG. Then, Section 4 details technical issues to be considered to enable AVS for CGAG. Section 5 discusses an application instance for the UDMGrid. Finally, Section 6 summarizes the paper and suggests future work.

2 Related Works

Based on the growing of high-speed networks CERNET (China Education and Research Network) [11], Access Grid and Augmented Reality technologies, the research work of this paper is brought closer to the aim above.

Firstly, the AG Toolkit, which has been developed by U.S. Argonne National Laboratory since 1998, includes streamlined user interfaces, robust middleware and low-level services that enable participants to share experiences through digital media [12]. But it hasn't supported the mixtures of the 3D graphics model of grid results and the live videos of AG session into one scene for grid users' collaborations, which falls

short of users' expectations of sharing these resources in live videos displaying the participants' real environment to improve the QoE level of collaborative tasks.

Secondly, Augmented Reality (AR) has the characteristics to make up for above deficiency of AG. In an AR environment the user can see the real world with virtual objects superimposed upon or composited with the real objects of the real world. CSCW (computer-supported collaborative work) is one of the evident application domains that Milgram et al.'s definition of AR suggests [13]. Since AR shows great potential use for developing new types of collaborative interfaces, this technique can be used to enhance face-to-face and remote collaboration in ways that is difficult with traditional technology [14]. For example, an AR Videoconferencing System, explained by Istvan Barakonyi et al. [15], is a novel remote collaboration tool combining a desktop-based AR system and a videoconference module. Another Mixed Reality conferencing application which uses the overlay of virtual images on the real world to support three dimensional remote computer supported collaborative work is described by Mark Billinghurst et al. [16].

Despite the obvious potential of AR for remote collaboration used in videoconferencing, only a limited amount of attention has been paid to the use of AR in the area of AG to construct an advanced AG environment in which AVS is built up for mixing the 3D graphics model of grid results and the live videos of AG session into one scene for grid users' collaborations.

3 CGAG Architecture

According to OGSA standard [17], the architecture of CGAG is divided into three distinct layers, as depicted in Fig. 1.

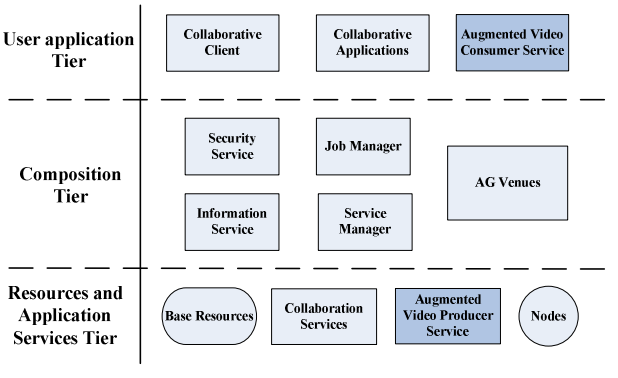


Fig. 1. The architecture of CGAG

The bottom tier of the representation comprises the resources that can implement various applications. Base Resources are supported by some physical or logical entities or artifacts in ChinaGrid. One part of our AVS, AVPS is implemented in this tier. The AVPS and the collaboration services are used to initiate an application session, establish the interaction channels for participants, and integrate the input messages

and allocate output messages. And a Node is a particular entrance point into the collaborative environment.

The middle tier is the composition tier composed of two parts. One part is supplied by CGSP, comprising Security Service, Job Manager, Information Service and Service Manager [2]. The other part is the AG Venues. As a spatial metaphor of rooms, it is responsible for controlling the scope of who is attending meetings and collaborating [18]. Through the AG Venues, the participants are integrated with other grid resources into grid computing environment. Thus, these components in middle tier are used to organize the participants, AG nodes, base resources, collaborative services and AVPS to implement grid-based augmented video applications.

At the top tier are the collaborative applications, the collaborative client and the AVCS. Thereinto, the AVCS provides an interface to grid services for interacting with services in composition tier. Using the AVCS, user could choose the application and submit corresponding request to start the application.

4 Augmented Video Services and Its Applications in CGAG

This section expands detailed technical issues in each part of AVS: AVPS and AVCS. Some basic AR problems and applications requirements are discussed.

4.1 Augmented Video Producer Service

The AVPS is exploited to initiate the application session, establish the interaction channels between participants and grid resources, integrate the inputs and allocate outputs, and what is the most important is to accomplish the 3D registration and occlusion handing of the 3D graphics model of grid results and the videos of the AG session, and then transmit the augmented video over the AG.

3D registration is a pivotal technical issue in our proposed AVS, and it is performed in the AVPS. Concretely speaking, 3D registration is the process of determining the virtual objects' position and orientation, and also one basic problem AR technique need to resolve. The perspective projection transformation and modelview

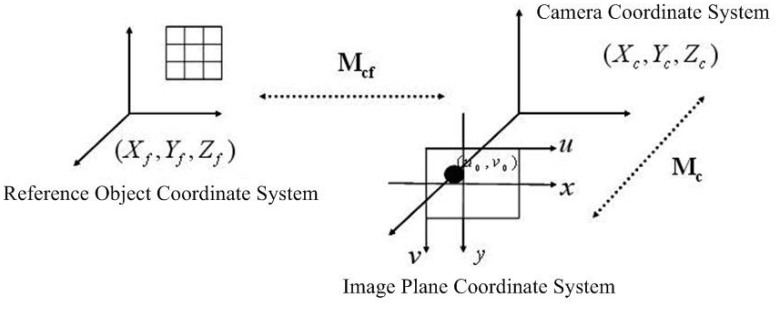


Fig. 2. The coordinate system of camera calibration

transformation for rendering virtual objects are required to implement the 3D registration of virtual objects on real world images. The perspective projection transformation parameters rest with camera intrinsic parameters, and are evaluated through calibration. Modelview transformation parameters are computed according to the camera's position and orientation.

According to the coordinate systems illustrated in Fig. 2, the relationship between 3D real environment and 2D video image can be expressed by the following equation [19]:

$$Z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{f}{dx} & 0 & u_0 & 0 \\ 0 & \frac{f}{dy} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{f}{dx} & 0 & u_0 & 0 \\ 0 & \frac{f}{dy} & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} X_f \\ Y_f \\ Z_f \\ 1 \end{pmatrix} = \mathbf{M}_c \mathbf{M}_{cf} \begin{pmatrix} X_f \\ Y_f \\ Z_f \\ 1 \end{pmatrix},$$

where $\mathbf{M}_c$ is completely determined by f, dx, dy, u_0, v_0, called intrinsic parameters; $\mathbf{M}_{cf}$ is completely determined by the position and orientation of the camera with respect to the world coordinate system, $\mathbf{R}$ and $\mathbf{t}$ in $\mathbf{M}_{cf}$ are called extrinsic parameters. From the above equation, we can see that the camera's intrinsic and extrinsic parameters matrices, $\mathbf{M}_c$ and $\mathbf{M}_{cf}$, are respectively the perspective projection matrix and the modelview matrix, which are just the matrices needed by 3D registration.

The intrinsic parameters is computed based on a camera calibration algorithm proposed by Zhengyou Zhang [20][21], and the camera position and orientation relative to markers are dynamically calculated using ARToolKit [22] which utilizes computer vision method based on the 2D marker identification, thus the projection matrix and the modelview matrix can be provided according to the intrinsic and extrinsic parameters to accomplish 3D registration.

Occlusion handing is another technical issue that AR needs to resolve to exactly estimate the spatial occlusion relationship between the real objects and the virtual ones. The distances between the real/virtual objects and the user's viewpoint, which is so-called depth information, are indispensable to the right estimation of the occlusion relationship.

4.2 Augmented Video Consumer Service

The AVCS provides functions including the access of grid middleware, the control of access authority which coordinate and synchronize the users in the collaboration work, and the reception and processing of the augmented video produced by the AVPS.

The AVCS gives an interface to the grid middleware, through which requests can be submitted to the grid middleware. The AVCS allows participants to join the application session, and to apply for the authority of accessing the grid resources. It also provides a window with the responsibility of not only rendering the received augmented video, but also responding the users' inputs and producing local events.

4.3 Streams Distributions for Different Applications

Different grid applications such as remote equipment sharing and UDMGrid make different demands of our AVS in CGAG. For the former, the simulation models are built for shared equipments as their virtual images, which are used to access, manage and schedule the remote resources, so as to solve the problem of distributed equipment resources sharing via network [23][24]. As the interface between users and the real shared equipment, the simulation models need to be rendered on the users end to provide interaction capabilities with the virtual objects so as to control the real equipments well and truly. For the latter, there may be some valuable 3D graphical models of precious artifacts, such as high-resolution digital scans of cultural heritage objects, may require protection to prevent piracy or misuse, hence the blending of reality and virtuality should be completed before transmission, and the mixed result video, i.e. the sequence of output images is streamed to the remote participants. Thus the digital rights could be well preserved.

In order to meet the requirements mentioned above, our proposed AVS are designed to support alternative schemes of streams distribution for each requirement. The AVS are composed of an AVPS and an AVCS which respectively rely on different video stream senders or receivers.

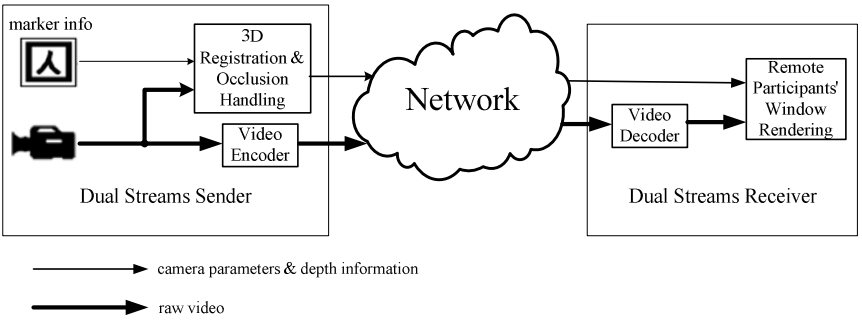


Fig. 3. Streams distribution in Dual Streams Sender/Receiver

For remote equipment resource sharing, the dual streams (i.e. pure video plus camera parameters & depth information) will be produced and transmitted by the AVPS, and Fig. 3 describes the streams distribution. 3D registration and occlusion handing are performed locally based on the raw video and the marker information. Then the derived camera's intrinsic and extrinsic parameters and the depth information are transmitted along with the video stream. The rendering of the virtual counterparts of the shared equipments is done on the dual streams receiver end according to these parameters. The graphics representation rendered in participants' window could respond to the users' inputs and produce local events so as to make it feasible to share equipment resource in an AR-based environment.

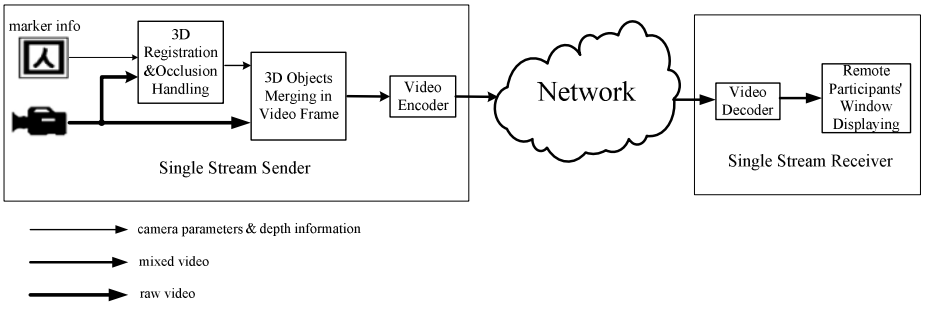


Fig. 4. Streams distribution in Single Stream Sender/Receiver

Aiming at UDMGrid application, streams distribution will be performed as Fig. 4 depicts. Similarly, after calibration, the single stream sender estimates the camera's position and orientation with respect to the marker and the depth information of the real scene in real time, but merging 3D models of the digital specimens into the video frame of real world is accomplished locally, and then the mixed video stream is compressed and transmitted to network for remote display. This solution is similar to what VPC (Virtual Presence Capture) [25] does, and suitable for sharing archives of 3D models while protecting the 3D geometry from unauthorized extraction. Accordingly, the AVCS will turn to the single stream receiver to receive the mixed output. Since the 3D graphics models have merged into the live video from the real world, it is practically impossible to reconstruct the 3D models of specimens from the received images.

As has been mentioned, the streams distribution in AVS is implemented with optional schemes, relying on either dual streams sender/receiver or single stream sender/receiver according to specific applications. Video stream is encoded using H.261 video compression standard to meet the available bandwidth requirements, and transmitted using RTP (Real-time Transport Protocol).

5 Sample

At present, UDMGrid provides information services to users by integrating the relative information into web pages, but it needs a better form to share the 3D models of the digital specimens, therefore, we choose a sample for UDMGrid application for demonstrating the capabilities of our AVS.

An experimental environment for UDMGrid is set up like Fig. 5. In the configuration, the 3D models of the digital specimens are located as the form of VRML files in different databases distributed in four university digital museums of UDMGrid. Through the interaction between the AVPS and the UDMGrid data services, the requested VRML files could be acquired according to their XML descriptions. The AVPS is deployed for properly rendering the acquired data files on the real time captured video through 3D registration and occlusion handing. In the mean time, the AVCS is deployed to receive the augmented video stream. The data channels are

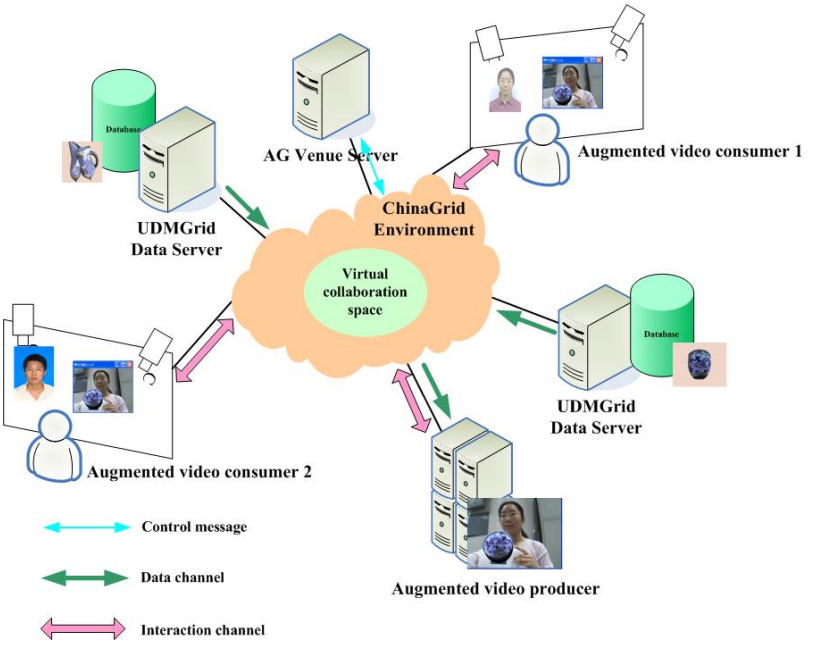


Fig. 5. The experimental environment for UDMGrid

established by FTP (File Transfer Protocol) to transfer the data files. Then, the security policies are defined for the virtual collaboration space, including a global administrative domain and the access policy.

Fig. 6 shows the traditional grid portal through which we access the grid resources in UDMGrid, the result of UDMGrid services such as the 3D graphics model of a Dragon-in-Cloud crock of Qing Dynasty is returned with a bald webpage of the

Fig. 6. UDMGrid application using the traditional grid portal

Fig. 7. UDMGrid application running in CGAG with the AVS

portal. In contrast, the UDMGrid application running in CGAG with our AVS is shown in Fig. 7. The same result of the UDMGrid services is merged into the live video of the presenter as if it is held in her hand with a tangible physical artifact, as shown in the left of Fig. 7. We can clearly feel the obvious merit of using augmented video for grid resource sharing, and the sample demonstrates the preliminary purpose perfectly.

6 Conclusion and Future Works

In this paper, we present an AVS couple by means of AR techniques based on AG framework to enable augmented video support for grid resource sharing in CGAG. The proposed services have been implemented as the prototype associated with the AG. The prototype has verified better users' QoE than traditional webpage of grid portal by demonstration in the above experimental environment. To make the proposed prototype better, our future plans include the following:

- Synchronizing the pure video stream and the corresponding camera parameters in the case of dual streams transmission, to avoid the non-consistency between the virtual objects and the real objects, a possible problem arising from an awful network condition.
- At this stage, we could only apply our prototype to UDMGrid application, and we are about to make equipment sharing available in CGAG and extend more grid applications.

References

1. Access Grid Web Pages. http://www.accessgrid.org
2. The ChinaGrid project. http://www.chinagrid.edu.cn
3. ChinaGrid Bioinformatics Grid. http://166.111.68.168/bioinfo/tools/index.jsp
4. ChinaGrid Computational Fluid Dynamics (CFD) Grid. http://grid.sjtu.edu.cn:7080/grid
5. ChinaGrid Course Online Grid. http://realcourse.grids.cn.
6. ChinaGrid Image Processing Grid. http://grid.hust.edu.cn/ImageGrid
7. ChinaGrid Massive Information Processing Grid. http://athena.vrlab.buaa.edu.cn/gcc
8. Xiaowu Chen, Zhi Xu, Zhangsheng Pan, Xixi Luo, Hongchang Lin, Yingchun Huang, Haifeng Ou: UDMGrid: A Grid Application for University Digital Museums. Grid and Cooperative Computing, Wuhang, China (2004)
9. The Advanced Collaborative Environments (ACE) Research Group. https:// forge.gridforum.org/ projects/ace-rg
10. Hai Jin: ChinaGrid: Making Grid Computing a Reality. Digital Libraries: International Collaboration and Cross-Fertilization - Lecture Notes in Computer Science, Vol.3334. Springer-Verlag (2004)
11. The China Education and Research Network. http://www.edu.cn
12. The AG Toolkit. Software available online, September 2004. http://www-unix.mcs.anl.gov/fl/research/accessgrid
13. P. Milgram, H. Takemura, A. Utsumi, F. Kishino: Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum. Proceedings of Telemanipulator and Telepresence Technologies, SPIE 2351 (1994) 282-292

14. Mark Billinghurst, Hirokazu Kato: Collaborative Augmented Reality. Communications of the ACM, Vol. 45, No. 7 (2002)
15. Istvan Barakonyi, Tamer Fahmy, Dieter Schmalstieg: Remote Collaboration Using Augmented Reality Videoconferencing. Proc. of Graphics Interface. London, ON, Canada (2004)
16. M. Billinghurst, H. Kato, S. Weghorst, and T.A. Furness: A Mixed Reality 3D Conferencing Application. Technical Report R-99-1 Seattle: Human Interface Technology Laboratory. University of Washington (1999)
17. I. Foster, H. Kishimoto, A. Savva, D. Berry, et al: GFD-I.030 Open Grid Services Architecture. Global Grid Forum (2005)
18. L. Childers, T. Disz, R. Olson, M. E. Papka, R. Stevens, and T. Udeshi: Access grid: Immersive group-to-group collaborative visualization. Proceedings of the 4th International Immersive Projection Technology Workshop (2000)
19. Lin Hongchang, Chen Xiaowu, Ge Xuedong: Consistent 3D Registration Based on Multiple Videos' Images in Collaborative Augmented Reality. Journal of Computer Research and Development, Vol.42, Suppl.A. (2005) 701-707
20. Z. Zhang: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence (2000) 1330-1334
21. Zhengyou Zhang: A flexible new technique for camera calibration. Technical Report MSR-TR-98-71. Microsoft Research, Microsoft Corporation (1998)
22. The ARToolKit Software available online, September 2005. http://n www.hitl.washington.edu/artoolkit
23. Yuexuan Wang, Cheng Wu, Xixiang Hu, Lianchen Liu: The Study of Equipment Grid Based on Simulation Modeling. Computer Integrated Manufacturing Systems (CIMS), Vol. 9 (2004)
24. Yuexuan Wang, Lianchen Liu, Xixiang Hu, Cheng Wu: The Study on Simulation Grid Technology for Equipment Resource Sharing System. Proceedings of the 5th World Congress on Intelligent Control and Automation. Hangzhou, P.R. China (2004)
25. Rhys Hawkins, Yifan Lu: Introducing VPC: A mixed reality VideoProducer for the Access Grid. AG Retreat 2005

RINDY: A Ring Based Overlay Network for Peer-to-Peer On-Demand Streaming*

Bin Cheng, Hai Jin, and Xiaofei Liao

Cluster and Grid Computing Lab
Huazhong University of Science and Technology, Wuhan, 430074, China
{showersky, hjin, xfliao}@hust.edu.cn

Abstract. Using peer-to-peer overlay network to provide video-on-demand service has been a promising solution due to its potential high scalability and low deployment cost. However, it remains a great challenge to construct an efficient overlay network for peer-to-peer video-on-demand systems owing to their inherent dynamicity caused by frequent VCR operations or joining/leaving operations. In this paper, we propose a *ring based overlay network* to handle this problem, called *RINDY*, in which each peer maintains a *gossip-ring* to explore appropriate data suppliers and several *skip-rings* with power law radius to assist the quick relocation of VCR operations. Our simulation results show that RINDY achieves better load balance in the control overhead than tree based overlay. As compared with the traditional client/server model, it saves more server bandwidth and achieves lower start-up latency when lots of users watch a same video simultaneously.

1 Introduction

With the exponential expansion of Internet resource and users, Video-on-Demand has become one of the most attractive services over Internet [15]. In order to provide large scale on-demand streaming services, most of existing systems deploy content distribution networks (***CDN***s) [5][16] and distributed proxies [6][11] to extent their capacities. Unfortunately, the bandwidth and I/O capacity of servers inevitably become their performance bottleneck as the number of online user increases constantly.

Peer-to-peer multimedia streaming systems [1][2] have become popular both in academic and industry. Compared with the traditional client/server model, they can sufficiently utilize the capacity of end nodes to improve their scalability. There have been a number of successful research projects on peer-to-peer live streaming, such as CoolStreaming [21], ESM [4], SplitStream [3], AnySee [18]. But there are few peer-to-peer on-demand systems due to the following difficulties. *First*, for peer-to-peer on-demand streaming systems, peer nodes have more obvious ***dynamicity***. Beyond joining or leaving, users can perform all kinds of VCR operations at will, such as PAUSE, REWIND and FORWARD, which lead to frequent changing of topology. *Second*, each peer has ***heterogeneous*** capacities, such as different inbound/outbound

* This paper is supported by National Science Foundation of China under grant 60433040, and CNGI projects under grant CNGI-04-12-2A and CNGI-04-12-1D.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1048 – 1058, 2006.

bandwidth, memory size. In this case, it still is a problem to determine where and when to fetch an expected data segment from multiple suppliers with non-uniform bandwidth and data availability, under the limitation of the playback deadline. *Third*, it is also very difficult to achieve high ***reliability*** and guarantee the quality of streaming under the condition that peer nodes join, leave or seek frequently.

In order to solve these problems, we propose a novel ***ring based overlay*** network for peer-to-peer on-demand streaming service, called ***RINDY***, in which each peer keeps a set of concentric rings to implement efficient membership management and fast relocation of VCR operations under a low control overhead. It maintains a *gossip-ring* to explore appropriate data suppliers and several *skip-rings* with power law radius to assist quick relocation of VCR operations. Compared with tree-based overlay, it is more reliable because it only needs to maintain a loosely consistent ring overlay and can tolerate the failure of many peer nodes.

The rest of this paper is organized as follows. An overview of RINDY system architecture is presented in section 2. In section 3, we give an insight of ring based overlay, including membership management and overlay maintenance. The performance of RINDY is evaluated by simulation in section 4. Related work is discussed in section 5. Finally, we end this paper with some conclusions and future work.

2 Overview of RINDY

As shown in Figure 1, the infrastructure of RINDY is consisted of four components: *tracker server, source server, peer,* and *web portal*. For the tracker server, it is a well-known Rendezvous Point (RP) to help each newly joining peer bootstrap. We deploy a backup tracker server. There are many source servers distributed in different ISP networks and organized in a logical ring. When an incoming peer logins our overlay network, the tracker server will allocate a near source server for it. Peer is the most

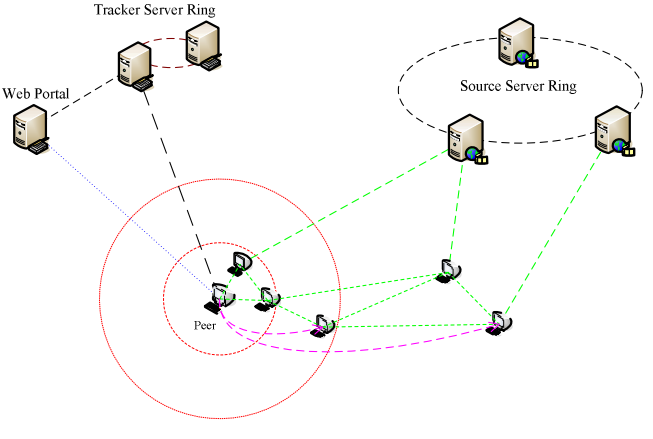


Fig. 1. Architecture of RINDY System, including Tracker Server, Source Server, Peer and Web Portal

complicated component in our peer-to-peer on-demand streaming system. For any peer, it can not only request data packets from its source server, but also exchange data packets with other peers. Each peer caches the recent watched media data and announces its buffer map to all neighbors periodically. At the same time, it schedules data requests to fetch its required media data packets to support playback continuously. The web portal is the entry of the system, providing the latest channel list to show the number of online peers and the rank of all movies.

3 Ring-Based Overlay

In this section, we focus on the discussion on the construction and maintenance of ring based overlay, including neighbor distribution, ring based gossip protocol, and relocation algorithm for new neighbors.

3.1 Overlay Construction

A. Ring

In RINDY, each peer node keeps track of a small and partial view of the total overlay and organizes all neighbors into concentric, non-overlapping logical rings according to their relative distance. The radius of each ring is represented with $a2^i$, where a represents the length of buffer window ($a = w$) and i is the layer number ($0< i <= B$, B is the maximum layer number). The ith ring denotes the zone between the inner radius ($r_{i-1} = a2^{i-1}$) and the outer radius ($r_i = a2^i$), where $i >= 1$. The distance between any two peers can be calculated with their playing positions, i.e., given any two peers i and j and their playing positions cur_i and cur_j, respectively, the distance from peer j to peer i (d_j) equals $cur_j - cur_i$. If d_j is negative, the playing position of peer j is behind that of peer i and we call peer j a *back-neighbor* of peer i. Otherwise if d_j is positive, peer j is a *front-neighbor* of peer i. The distance remains unchanged until some VCR operations occur. For all neighbors of peer i, their locations are up to the distances from them to peer i. If the distance of neighbor j meets the inequation $a2^{k-1} < |d_j| <= a2^k$, peer i will place neighbor j in the $\boldsymbol{k}$th ring.

B. Neighbor Distribution over Ring

For any peer, a large q will help it get larger member list, increasing its knowledge about the total overlay network and the locality of lookup operations. But at the same time, a large q also entails more memory and more bandwidth and causes worse network traffic. A reasonable neighbor distribution always brings more efficient performance for VCR relocation and member discovery. For peer-to-peer video-on-demand systems, nodes that are temporal diversity instead of clustered together can forward a query to a wider region.

In RINDY, we define two kinds of rings, gossip-ring and skip-ring, and provide different neighbors distribution rules for them respectively. For the innermost ring ($i = 1$), it collects some peer nodes with close playing positions and overlapped buffer windows with peer i. We call this ring *gossip-ring*. For all outer rings ($i>= 2$), they sample some remote peer nodes to help peer i look up new neighbors after seeking. These rings are mainly used to improve the locality of lookup operations and decrease the

load of tracker server. We call these rings *skip-rings*. In the gossip-ring, each peer node keeps *m* ring members by the received gossip message, called near-neighbors. In each skip-ring, a peer node keeps track of the *k* primary ring members and *l* secondary ring members which serve as *backup candidates* for primary ring members. All of these primary remote members are called far-neighbors. When any far-neighbor leaves, a new far-neighbor will be selected from *l* back candidates quickly.

Figure 2 illustrates the member distribution of peer *i*. Node A is a near-neighbor in the gossip-ring and node B, D, E, G and H are far-neighbors in the skip-rings. Node C is the backup candidate of far-neighbor B. Peer *i* can propagate gossip messages through its near-neighbors and explore new members by receiving gossip messages. For any peer, to find good neighbors as its partners is critical to continuous playback. The join and departure messages need to announce to other peers in the gossip-ring. So we try to maintain as many near-neighbors as possible at the permission of overhead. According to the analysis in [9][10][19], log(*n*) is enough to make most of near peers receive join and departure events. The ring based overlay only requires loose consistent. For skip-rings, they function as the bridge to look up new proper neighbors and good partners. It is enough to ensure that there is one connection in each skip-ring. Considering users may seek back or forward, we remain two primary far-neighbors pointing to the front and back of current position respectively. Here we configure $k = l = 2$.

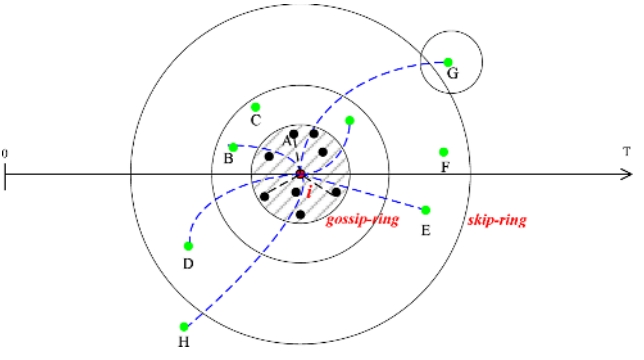


Fig. 2. Neighbor distribution of peer node i, node A in the gossip-ring and node B, C, D, E, F, G, H in the skip-ring

3.2 Overlay Maintenance

A. Discovery of New Members

In RINDY, each peer can leave the overlay network or seek new positions so that some peers may lose lots of neighbors due to the departure of neighbors, which results in the damage of the normal ring structure. In previous work, there are many projects importing a gossip protocol to discover new members, such as CoolStreaming [21], AnySee [18]. The principle of traditional gossip protocol is simple, that is, a node sends a newly generated message to a set of randomly selected nodes and these nodes do similarly in the next round. Thus contents will spread throughout the network in an epidemic fashion. The random choice of gossip targets achieves resilience to random

failures and enables decentralized operations. However, a great trouble for gossiping based streaming is how to minimize delay and avoid the date outage of messages. It is difficult for peer-to-peer on-demand streaming to handle this problem because the playing position of each peer is moving constantly and even changing unexpectedly. In order to address this problem, we design a *temporal and spatial restraint* gossip protocol based on ring structure to explore new members and maintain stable distribution of neighbors in all rings.

In RINDY, each peer sends a gossip message to announce its existence and its buffer windows status periodically. At the same time, it updates its member table according to the member information carried in the received gossip messages. The gossip message format is shown in Figure 3, where *GUID*, *Peer Address*, *near-neighbor* and *Cur* represent the global identifier, the network address, neighbor number and the playback offset of source peer, respectively. *TTL* is the remaining hop number of this message. *Max, Min, Want* and *Avail* denote the snapshot of moving buffer together. All gossip messages are divided into two kinds, ANNOUNCE and FORWARD. The difference between them is that ANNOUNCE messages carry the buffer snapshot while FORWARD messages not. Because the main purpose of gossip is to help online peers to discover more data suppliers with close playback offset, we restrict all gossip messages spreading in the range of gossip-ring to decrease the overhead of gossip propagation. For a peer, it sends a ANNOUNCE message to all of its near-neighbors in each gossip round, notifying its position and buffer map. After its near-neighbors receiving this message, they update their mCache and randomly choose one near-neighbor with the playback offset t_{forward} from their near-neighbors, where t_{forward} meets the relation $t_{\text{forward}} - m <= t_{\text{source}} <= t_{\text{forward}} + m$ and t_{source} is the current playing position of source peer of this gossip message.

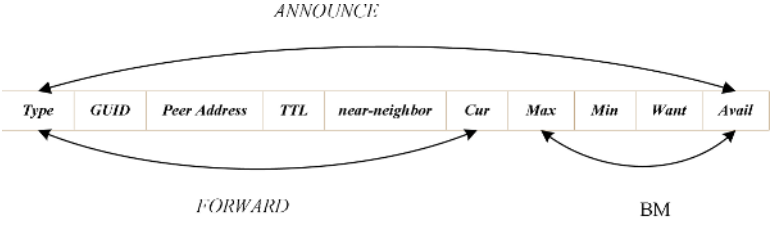


Fig. 3. Format of gossip message

B. Lookup of New Position

The lookup operations based on skip-rings mainly occur in the following two cases. First, when users perform VCR operations lookup operations will be triggered to find some new neighbors close to the destination. Second, when a peer finds that there are not enough neighbors in a ring with the simultaneous leave of all primary far-neighbors, it also executes a lookup procedure to find more backup candidates for this ring. As Figure 4 illustrates, when peer i with the playing position p wants to seek to the destination d it first judges whether the destination d is located on its gossip-ring or not. If $|d - p|$ is less than or equal to a, that means the destination d is in its gossip-ring, it will execute a local search to find out enough near-neighbors and far-neighbors for peer i. If

$|d - p|$ is more than a, it will find out a neighbor as its next hop from its rings closest to the destination d and then forward this query to the next hop neighbor. When the next hop neighbor receives this query it will execute the same procedure and the procedure will be iterated until this request is forwarded to a peer that is in the same gossip-ring with the new peer i. As Figure 4 shows, if d is in the 3rd ring of peer i and P_1 is the neighbor closest to d, the query q will forward to the neighbor P_1 at first. Then P_1 will forward this query to its neighbor P_2 in its 2nd ring closest to d. Finally, when P_2 finds out that the destination d is in its gossip ring it will add all neighbors in its gossip ring into the result set and return it to the source peer i, following the forwarding path.

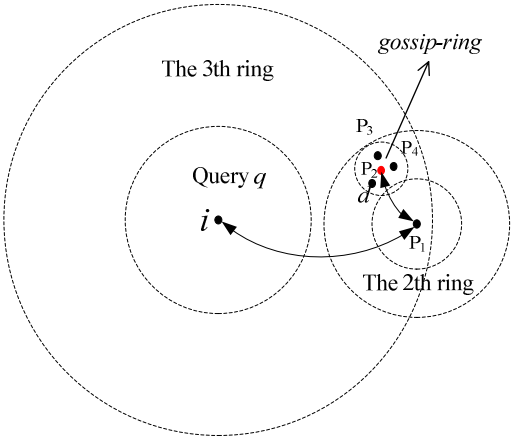


Fig. 4. Procedure of lookup operation for peer i

4 Performance Evaluation

In this section, we investigate the performance of RINDY in server bandwidth, control overhead and start-up latency by simulation. At the same time, we also give some comparisons between RINDY and other on-demand systems.

4.1 Simulation Setup

We use GT-ITM topology generator [20] to create a topology of 1000 peer nodes based on transit-stub model. The network consists of 3 transit domains, each with 5 transit nodes and a transit node is connected to 6 stub domains, each with 12 stub nodes. In this set of experiment, peers can be located on any stub nodes in the topology. We randomly choose 1000 stub nodes as peer clients and place the source server on a transit node. The bandwidth between two transit nodes, a transit node and a stub node, two stub nodes are 100Mbps, 10Mbps, and 4Mbps, respectively. In addition, we choose a movie with 60 KB/s streaming rate and 60 minutes content as our testing stream. We modify the source code of NICE simulator (http://www.cs.umd.edu/~ suman/research/myns) to implement a simulator of RINDY, in which each peer remains a ring structure instead of a layer structure. Some important parameters for our ring based overlay network are presented as Table 1.

Table 1. Some configuration parameters

Parameters	Value and Meaning
w	*300 seconds, buffer window size*
B	*5, maximal layer number*
R	*60KB/s, average streaming rate*
α	*300, radius coefficient of rings*
t	*30 seconds, gossip period*
m	*12, near-neighbor number in each gossip-ring*
k	*2, far-neighbor number in each skip-ring*
l	*3, back candidates in each skip-ring*
L	*60 Minutes, length of a sample movie*

4.2 Simulation Result

A. Control Overhead

We first evaluate the control overhead for overlay construction and maintenance in our ring based overlay network. Here the control overhead is represented by the number of processed messages. In order to investigate the overhead of ring based gossip, we record the average message number of each peer in a gossip period when there are no any peer failed and any VCR operation occurred, then we calculate the average message number of all peers with different scales. The result is illustrated in Figure 5. We can see that the average messages number increases very slowly after the total peer number of the same channel reaches 400. When the total peer number exceeds 1000, the average message number basically remains about 100, in a restricted constant range. Since the gossip period t is 30 seconds, each peer just processes about three or four messages in a second. Compared to the streaming overhead, the control overhead is very low and can be ignored.

Figure 6 shows the average message number of each peer when 10 percent peers join, leave RINDY network or perform VCR operations randomly. When the total peer number reach a specified scale, about 300 nodes, the average control overhead to accommodate the overlay changing effected by peer joining, leaving or seeking basically keeps a constant. That illustrates that the control overhead induced by peer joining, leaving or seeking does not increase with the scale of peer nodes.

Figure 7 shows the distribution of lookup request message processed by each peers in ring based overlay and tree based overlay when the total peer number is 600 and 10 percent peers perform VCR operations randomly. We can see that, in the ring based overlay, most peer nodes process 0~8 messages and a few peer nodes reach 12 messages but none of them exceeds 20 messages. However, in the tree based overlay, most peer nodes process 0~3 messages but the lookup message number of some peer nodes reaches 30 or 40 and the root node processes 60 messages. Compared with the tree based overlay, our ring based overlay gets better load balance among peers during the relocation of VCR operations. To the best of our knowledge, there are two main reasons for this result. First, in our ring based overlay network, each peer has an individual neighbor distribution and its lookup operations begin with its local rings. However, for the tree based overlay network, all of

lookup operations travel from the root node firstly. The root node and the peer node of upper layer always process more lookup request messages, which results the unbalance of control overhead among peer nodes. Second, for each peer, its skip rings construct a quick index based relative playback offset, just like a binary index, which decreases the length of lookup path.

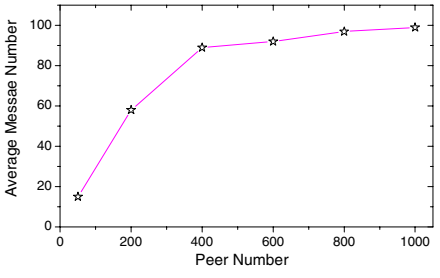


Fig. 5. Message cost of stable status

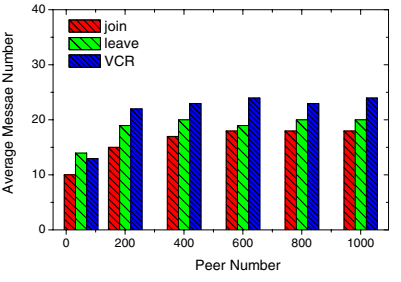


Fig. 6. Message cost of joining/leaving and VCR

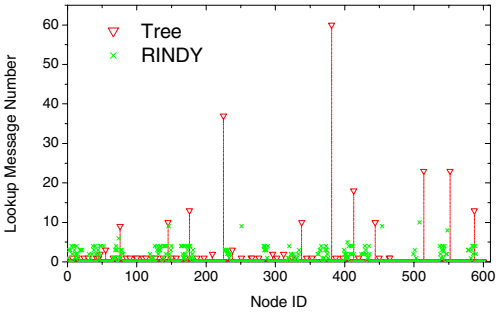


Fig. 7. Lookup message distribution when 10% peers randomly seek

B. Server Bandwidth Consumption

Since the main purpose to deploy peer-to-peer on-demand streaming is to alleviate the bandwidth bottleneck, we compare the server bandwidth consumption between traditional C/S model and our RINDY. Figure 8 reports their bandwidth cost with the increasing of peer client number. When the peer client number arrives at 200, the traditional solution nearly consumes all bandwidth of the central server, about 89Mbps. But for our RINDY, the increase of peer client number has no obvious effect to the server bandwidth consumption. The main reason is that peers can exchange their data packets efficiently. In our testing case, the server bandwidth has been the bottleneck when the peer client number reaches about 200, while RINDY just uses 6~8Mbps, 10 percent of C/S model. RINDY has great potential for bandwidth saving.

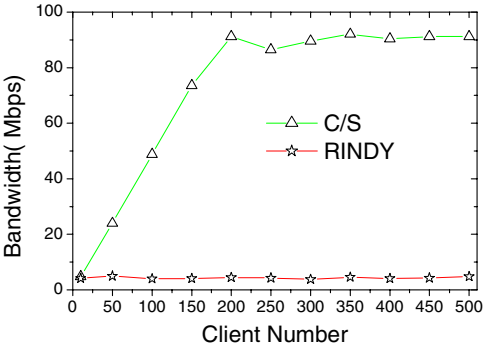


Fig. 8. Comparison of server bandwidth between C/S model and RINDY

C. Start-up Latency

The start-up latency is the time that users must wait before playback since users start a channel. Users always expect shorter waiting time, however, it is not easy for most peer-to-peer systems to find initial corresponding data supplier quickly, which always take longer time than expected. In addition, it must try to avoid fetch data from the central server. Otherwise, the central server becomes the bottleneck easily. In RINDY, we use a scheme to solve this problem. We make each peer buffer the initial one minute stream packets and the SDP packet need by RTSP protocol. For a newly incoming peer, it retrieves the initial data from any neighbors. Figure 9 gives a comparison of start-up latency between C/S model and RINDY. The start-up latency of RINDY has little change with the increasing of peer client number, keeping about 30 seconds. But the start-up latency of C/S model increases quickly with the system scale. When the server is overloaded, the start-up latency will make users unacceptable.

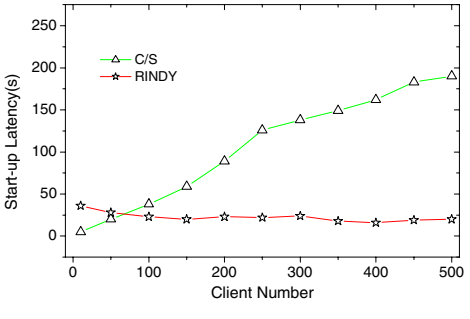


Fig. 9. Comparison of start-up latency between C/S model and RINDY

5 Related Work

There are mainly following three methods to provide peer-to-peer on-demand streaming systems. The first method is to construct *tree based overlay* network, such as P2Cast [12], P2VoD [8], oStream [7], DirectStream [13]. In these systems, peers on

the upper layer always play an important role in the whole overlay network. Their departure will lead to the lower layer network fluctuation. Moreover, each peer has only one data supplier, which will cause inefficient utilization of available bandwidth in a heterogeneous and highly dynamic network environment.

Another method is to deploy *mesh based overlay* to organize all joined peers, such as PROMISE [14]. Although they can fetch packets from many data suppliers [17], they meet another problem, that is, how to relocate the destination efficiently to support random VCR operations. Because of the shortage of structure, mesh based overlay always consume longer time to find appropriate close partners, therefore they hardly meet the real-time requirements of VoD services and always induce an unacceptable waiting time after seeking. Combing the advantages of tree based overlay and mesh based overlay, TAG [22] presents a *hybrid overlay*, called Tree Assistant Gossip. It uses a balance binary tree to index all joined peer nodes and search many partners from the tree when a peer joins. It integrates the advantages of tree overlay and mesh overlay to avoid the problems above. However, the maintenance algorithm of the distributed AVL tree is complicated. In addition, there is only one tree structure in the global overlay and each lookup operation begins from the root node, so the control overhead among peer clients is unbalanced seriously.

6 Conclusions and Future Work

In this paper, we present a novel ring based overlay network for peer-to-peer on-demand streaming, called ***RINDY***, which maintains a gossip-ring to explore appropriate data suppliers and several skip-rings with power law radius to assist quick relocation of VCR operations. We investigate the control overhead of RINDY by simulation experiments. The results illustrate that RINDY achieves better load balance and faster relocation of VCR operations compared with tree based overlay. We also compare the bandwidth consumption and start-up latency between the traditional C/S model and RINDY. The results are so exciting that we believe that RINDY has great potential in bandwidth saving and QoS guarantee. In future we plan to study a distributed cache scheme to improve the performance of peer-to-peer on-demand streaming further.

References

1. Andersen, H. Balakrishnan, M. Kaashoek, and R. Morris, Resilient overlay network, In Proceedings of ACM SOSP'01, Banff, Canada, 2001.
2. S. Banerjee, B. Bhattacharjee, and A. Srinivasan, Resilient multicast using overlays, In Proceedings of ACM SIGMETRICS'03, Jun. 2003.
3. M. Castro, P. Druschel, A. M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, SplitStream: High-Bandwidth Content Distribution in Cooperative Environments, In Proceedings of ACM SOSP'03, Oct. 2003.
4. Y. H. Chu, S. G. Rao, and H. Zhang, A Case for End System Multicast, In Proceedings of ACM SIGMETRICS, Jun. 2000.

5. L. Cherkasova and J. Lee, FastReplica: Efficient Large File Distribution within Content Delivery Networks, In Proceedings of 4th USENIX Symposium on Internet Technologies and Systems, Mar. 2003.
6. S. Chen, B. Shen, S. Wee, and X. Zhang, Adaptive and Lazy Segmentation Based Proxy Caching for Streaming Media Delivery, In Proceedings of ACM NOSSDAV'03, Jun. 2003.
7. Y. Cui, B. Li, and K. Nahrstedt, oStream: asynchronous streaming multicast, IEEE J. Select Areas in Communication, Jan. 2004.
8. T. Do, K. A. Hua, and M. Tantaoui, P2VoD: providing fault tolerant video-on-demand streaming in peer-to-peer environment, In Proceedings of ICC'04, 2004.
9. P. Eugster, R. Guerraoui, A. M. Kermarrec, and L. Massoulie, From epidemics to distributed computing, IEEE Computer Magazine, 2004.
10. J. Ganesh, A. M. Kermarrec, and L. Massoulie, Peer-to-peer membership management for gossip-based protocols, IEEE Transaction on Computer, 52(2), Feb. 2003.
11. L. Guo, S. Chen, S. Ren, X. Chen, and S. Jiang, PROP: a scalable and reliable P2P assisted proxy streaming system, In Proceedings of ICDCS'04, Mar. 2004.
12. Y. Guo, K. Suh, J. Kurose, and D. Towsley, P2Cast: peer-to-peer patching scheme for VoD service, In Proceedings of WWW'03, May 2003.
13. Y. Guo, K. Suh, J. Kurose, and D. Towsley, A peer-to-peer on-demand streaming service and its performance evaluation, In Proceedings of ICME'03, 2003.
14. M. Hefeeda, A. Habib, B. Botev, D. Xu, and B. Bhargava, Promise: Peer-to-Peer Media Streaming Using Collectcast, In Proceedings of ACM Multimedia, 2003.
15. M. Hefeeda and B. Bhargava, On-demand media streaming over the Internet, In Proceedings of FTDCS'03, May 2003.
16. S. S. Kien A. Hua, and Y. Cai, Patching: A Multicast Technique for True Video-on-Demand Services, In Proceedings of ACM Multimedia, Sep. 1998.
17. Kostic, A. Rodriguez, J. Albrecht, and A. Vahdat, Bullet: High Bandwidth Data Dissemination Using an Overlay Mesh, In Proceedings of ACM SOSP, Oct. 2003.
18. X. Liao, H. Jin, Y. Liu, L. M. Ni, and D. Deng, AnySee: Peer-to-Peer Live Streaming, In Proceedings of IEEE INFOCOM'06, Apr. 2006.
19. B. Wong, A. Slivkins, and E. G. Sirer, Meridian: A Lightweight Network Location Service without Virtual Coordinates, In Proceedings ACM SIGCOMM'05, Aug. 2005.
20. E. Zegura, K. Calvert, and S. Bhattacharjee, How to model an internetwork, In Proceedings of IEEE INFOCOM, Mar. 1996.
21. X. Zhang, J. Liu, and B. Li, CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming, In Proceedings of IEEE INFOCOM'05, Mar. 2005.
22. M. Zhou and J. Liu, Tree-Assisted Gossiping for Overlay Video Distribution, Technical Report, 2005.

A Multi-layered Assessment Model for Evaluating the Level of Ubiquitous Computing Services

Ohbyung Kwon and Jihoon Kim

College of Management and International Relations, KyungHee University
Seochoen, Kiheung, Yongin, Kyungggi-do, 446701, South Korea
{obkwon, hdlamb}@khu.ac.kr

Abstract. Despite a variety of service scenarios and prototype systems that adopt ubiquitous computing technology, sophisticated methodologies for assessing the level of ubiquitous service and systems are still very rare. Hence, we propose a multi-layered model to assess levels of ubiquitous computing services.

1 Introduction

Ubiquitous business is a business which fully makes use of the abilities embedded in ubiquitous computing and is characterized by intelligent, automated and personalized transactions. Moreover, the ubiquitous business model runs on top of mobile business services, which means that the capabilities of mobile, e-, and IT-enabled business are also applicable to the ubiquitous business. Although to date researchers have suggested several service scenarios and prototype systems that adopt ubiquitous computing technology, methodologies for assessing the service levels and/or systems' performance levels are still very rare.

Hence, this paper aims to propose an approach to assessing the level of ubiquitous computing services. Two-layered assessment model is proposed: capabilities of ubiquitous computing technology for evaluating whether the service is ubiquitous or not and level of ubiquity in technical perspectives and level of service quality in behavioral perspectives. Evaluating levels of service is useful for analyzing who may be potential services users.

The rest of this paper is organized as follows. Section 2 describes the notion and framework of multi-layered assessment model. The layer 1, 2-1, and 2-2 models are addressed in Section 3, 4, and 5, respectively. Finally, we conclude in section 6.

2 Evaluation of Ubiquitous Computing Systems and Services

Until now, there have been neither standardized nor generic criteria of evaluating the quality of ubiquitous computing services. Criteria suggested by most of the researchers also do not consider the generic issues but rather tend to focus on side issues about evaluating ubiquitous computing services solely based on their own research fields. The representative models and arguments for assessing ubiquitous computing systems and services are listed in Table 1.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1059 – 1068, 2006.

Table 1. Evaluation of Ubiquitous Computing Systems and Services

Authors	Content	Perspective
Scholtz and Consolvo [9]	Framework of Ubiquitous Computing Evaluation Areas (UEAs)	Behavioral
Riekki et al. [8]	'Level of Calmness' of ubiquitous computing systems	Technical
Mankoff et al. [4]	Evaluated ubiquitous computing focused on a specific filed, such as sensing systems	Technical
Bellotti et al. [1]	Five interaction challenges: address, attention, action, alignment, and accidence	Technical and Behavioral

Despite the improved availability that ubiquitous services has in overall framework and example services, methodologies for evaluating service levels are so far quite rare. Moreover, they didn't provide explicit decision criteria to see to some extent a specific service is based on ubiquitous computing. Hence, this paper aims to propose a three dimensional view of ubiquitous computing services, by extending one more dimension: level of ubiquitous computing services. A Multi-layered approach to ubiquitous computing service assessment is considered in this paper as shown in Fig. 1 and Table 2.

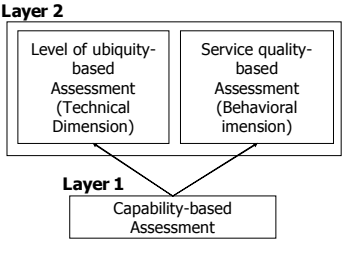


Fig. 1. Multi-layered assessment model

Table 2. Explanation of each layer of assessment

Layer	Assessment	Explanation
1	Capability based	Check whether the provided service is ubiquitous or not.
2	Ubiquity based	Evaluate the technical requirements that the ubiquitous computing service should have
	Service quality based	Evaluate the service quality based on the perception when the customer receives the ubiquitous computing service.

3 Layer 1: Capability-Based Assessment of Ubiquitous Computing Services

Assessment by layer 1 model is mainly used to determined whether a service could be regarded as 'ubiquitous computing' service or not. This assessment is crucial because many ubiquitous computing services have appeared in electronic markets may be actually not based on ubiquitous computing technology but just mobile or pervasive computing technology.

We looked up the contents referred about capabilities of IT from all the articles published at the Communications of the ACM Journal between 1994 and early half of 2004. As results, we found 294 terms and summarized the frequency of each term. From this IT capability list revealed in the previous step, we asked the experts to choose the capabilities that related to ubiquitous computing service using focused group interview.

We made up the questionnaire for the experts in ubiquitous computing area to analyze that these capabilities are how important to ubiquitous computing services practically. We added the terms such as invisibility, reconfigurability. Even though those terms were not shown in the Communications of the ACM, those have been introduced continuously as developing ubiquitous computing issues. We chose 121 project managers and professors who were involving in the projects of developing the ubiquitous computing appliances or services and sent them the questionnaire through e-mail. After all, we received 35 responses (return ratio = 28.9%). Then we conducted the descriptive statistics and got the results as listed in Table 3.

Table 3. Descriptive Statistics about Capabilities of Ubiquitous Computing Service

Capability Name	Frequ-ency	Weight	Capability Name	Frequ-ency	Weight
Security	23	6.97	Invisibility	13	3.94
Connectivity	18	5.45	Proacitiveness	13	3.94
Sensibility	18	5.45	Customizability	12	3.64
Ubiquity	17	5.15	Credibility	10	3.03
Embeddedness	16	4.85	Interoperability	10	3.03
Personalization	16	4.85	Configurability	8	2.42
Accessibility	15	4.55	Integrability	8	2.42
Mobility	15	4.55	Reconfigurability	8	2.42
Reliability	14	4.24	Usablity	8	2.42
Adaptability	13	3.94	Adjustability	7	2.12
			Misc.	68	20.63

We conducted an assessment with the persons who are specialties in the ubiquitous computing technology so that they can understand what the capabilities appeared in the statistics derived in this layer. Seven Likert scale can be used for a certain IT-based service to identify to what extent the service can be regarded as ubiquitous computing service.

4 Layer 2-1: Assessing Level of Ubiquity in Technical Perspectives

The explanations of '3 keywords' along with the items are shown at Table 4 and 5, respectively.

Table 4. Explanations of 3 keywords

3 Keywords	Explanations
Situation Sensing/Decision	Resolves the problems with sensing the various contexts and inferring the human intention.
Autonomic Computing	Meets the goal through the autonomous cure and re-structuring commissioned authority from human.
Self-growing Intelligence Engine	Has the aim the user's purpose or goal.

Table 5. Items to evaluate the level of ubiquity

Items			
User preference	Fault tolerance	Context reusability	Ease of use
User profile	Negotiation	Inferred context	Seamlessness
User context	Trust	Service coverage	Response time
Location tracking	Self-control	Learning	Scalability
Time tracking	Authentication	Reasoning	Durability
Identity tracking	Authorization	Autonomy	Standardization
Entity tracking	Usability	Automation	

Using this evaluation method, the level of ubiquity measures as follows:

First, give points from 1 to 7 to each service item; then multiply the weight of an item to know the final score of each. Now add all final scores according to the keyword, and divide them to the sum of all weights of a keyword. With this score, get the evaluated score for each keyword: autonomy, self-growing, and community computing. For the last step, multiply the weight of a keyword to the keyword's evaluation, add all, and divide it to the sum of weights of each keyword. This is the final score of services. The example of this evaluation appears on the section 6 and 7.

5 Layer 2-2: Assessing Service Quality in Behavioral Perspectives

Over the past twenty years, a few ideas have emerged for assessing service quality. Among those, SERVQUAL, suggested by Parasuraman et al. ([5], [6]) for assessing it, has been adopted most widely among other measurements and it has extended its application to various fields. In the IS field, many researchers have adopted the SERVQUAL model, and tried to build up the amended model in order to evaluate the information systems based on service quality ([2], [3]). Looking ahead, if we may want to adopt the SERVQUAL model to assess newly designed information system-based

services, such as ubiquitous computing service, then we must determine whether either the conventional dimensions of SERVQUAL model could be reused or a new assessment model is required. Researchers in IS field have tried to match the conceptual correspondence in order to fit the dimensions of SERVQUAL to those of IS-SERVQUAL ([3], [7]).

For evaluating a ubiquitous computing service, we created an 18-item questionnaire that includes all of Parasuraman et al.'s items [6] except those of dimension 'Tangibles', with reference to the items of Kettinger and Lee's and Jiang et al.'s ([2], [3]). Many results of research in IS field have showed that the dimension of Tangibles should be eliminated because they are less explainable or supportable than those of other dimensions.

To obtain samples for this research, we performed a survey for two weeks. We sent 200 questionnaire sheets to the general users and 129 sheets were returned (return rate = 64.5%). Among respondents, 45% were female and 96% respondents had over bachelor degree. 98% of the respondents ranged from teenagers to thirties, and were open to new technology including ubiquitous computing. Therefore, respondents are knowledgeable about information technology and have experience of using it. Factor analysis of measuring items was performed. To derive the measurement model, LISREL 8.50 and SPSS for Windows 10.0 were used. As a result, the implication of four-dimensional Ubi-SERVQUAL model consists of Reliability, Responsiveness, Assurance, and Empathy. Based on the experimental results described, assessing items of ubi-SERVQUAL for behavioral dimension of assessing service quality are proposed as listed in Table 6.

Table 6. ubi-SERVQUAL: an amended SERVQUAL for assessing the quality of ubiquitous computing service

Dimension	**Items**	Dimension	**Items**
Reliability	Promise Fulfillment	Assurance	Confidence
	Dependability		Consistent Courteousness
	Time Accuracy		Knowledgeablility
Responsiveness	Prompt Service	Empathy	Individual Attention
	Willingness to Help		Personal Attention
	Scalability		Sincereness
			Request Understandability

Using the items listed in Table 6, give points from 1 to 7 to each service. And then, get the weighted average value of each case. The example of this evaluation appears on the next section.

6 Illustrative Example 1: GPS-Based Location Announcing Service

To evaluate the service value of Location-Based Services (LBS), five actual services were selected from location announcing services using GPS-based contexts to high value added navigation services. Classifying the navigation services in two categories,

one of the services was chosen from GPS services, the remaining services from navigation services. The selected services for test are described in Table 7.

Table 7. The selected services for evaluation

Category	Services	Service Description
GPS	RoadMate Pro (RM) http://www.road-mate.co.kr	Backward Alarm Receiver / Road-danger and Traffic Information Storage and Voice Guide / Place Register by User-Own
Navigation	ALMAP NAVI (AN) http://www.almap.co.kr	Navigation / Overspeed Area Warning and Traffic Information Guide by Voice / Driving Simulation and Track-log Management
	I-NAVI (IN) http://www.inavi.co.kr	Accident-Frequent Area and Safety Speed Block Voice Guide / Buzz for Overspeed Alarm when Over Assigned Speed / Real-Time Route Voice Guide / Near Facility Search / Frequent Visit Place and Route Secession Alarm and Route Re-Setup
	Nate Drive (ND) http://drive.nate.com	Route Guide using Voice and Map / Watch-Camera Location Guide using Cell-phone Screen and Voice / Real-Time Traffic Information / Information Gathering about Facilities Around / Restaurant/Travel Place Recommendation
	Mozen (MO) http://www.mozen.com	Burglary Alarm and Trace / Fast-Route Guide, Real-Time Traffic Information, Danger-Area Alarm / Car Remote Diagnosis, Parking Location Alarm / Restaurant/Travel Place Recommendation

The domain experts are supposed to give the weight for location-based service to each item of ubiquity by observing each service. After the observation, the experts are asked to give the scores to each item of capability and ubiquity to evaluate those levels. Finally, interpret the scoring results of the assessments.

The results of the level of capability are shown at Table 8. Since the features such as 'Traffic Information Storage' and 'Place Register by User-Own' that RoadMate has indicate the lower level of sensibility, proactiveness, and learability, RoadMate scores the lowest. In the contrary, Mozen stands on the highest place because some of the features such as 'Car Remote Diagnosis' and 'Burglary Alarm and Trace' that Mozen has indicate the high level of 'Personalization' and 'Sensibility'. Meanwhile, the evaluation results in terms of the level of ubiquity are that RoadMate is 1.54, Almap Navi is 2, I-Navi is 2.15, Nate Drive is 2.43, and Mozen is 3.16. The scores of the items such as 'Location Tracking', 'User Preference', and 'User Profile' showed the gap of the level of ubiquity between RoadMate and Mozen. The evaluation results in terms of the level of service quality are also shown at Table 8. These result shows the four services classify as the three levels except RoadMate that eliminated at the layer of level of capability. Almap Navi and I-Navi are classified as the same level, and Nate Drive and Mozen shows the one level each. In the case of Nate Drive

evaluated the highest level of mobility, the level of 'Time Accuracy' showed higher point than anything else. These implies that Nate Drive provides service using cell-phone has advantage that is easy to request and response the service individually than other services. Finally, Mozen evaluated that had the highest level of service quality seems to provide the higher level of service quality to the users than other services through the different services such as 'Burglary Alarm and Trace' and 'Danger-Area Alarm' *etc*. Table 8 shows the overall results of illustrative example 1.

Table 8. Overall results of illustrative example 1

Layer	Services				
	RM	AN	IN	ND	MO
1	2.9	3.54	3.79	4.65	5.43
2-1: Technical Dimension	1.54	2.00	2.15	2.43	3.16
2-2: Behavioral Dimension	1.49	2.27	2.24	2.75	3.30

As shown in Figure 2, since the level of capability of RoadMate (2.90) is less than the threshold (θ = 3.00), RoadMate was not regarded as a kind of ubiquitous service and hence was excluded for further assessment. Almap Navi and I-Navi provide the lower level of ubiquitous service than Nate Drive and Mozen. Nate Drive runs on cell-based service and guarantees the higher level of mobility which is substantially required by the ubiquitous services. Since Mozen adopts intelligent services such as 'Restaurant/Travel Place Recommendation', 'Burglary Alarm and Trace' and 'Car Remote Diagnosis', higher level of security and ubiquity is doable.

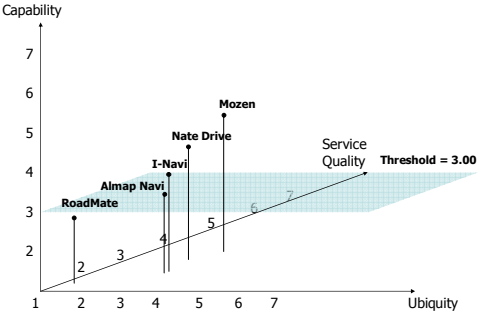


Fig. 2. Service Valuation of LBS

7 Illustrative Example 2: Ubiquitous Services in Apartments

To show the feasibility of the evaluation methods described in this paper, suppose a practical assessing example using the three scenarios, which are based on the actual content shown as advertisements in 2005, Korea. The advertisements show the IT-based services which have been or will be actually installed in apartments built by the leading companies of building industry in Korea.

7.1 Cases

Case D is a type of service that has been actually installed and used, Case Z is a type of service that has been provided partially, and Case R is a type of service that is going to be provided shortly.

Case D
Jamey lives in D apartment equipped the high speed internet connection network accesses to the homepage of the apartment to get served to pay for the rental fee of this month. Logging on with her own ID, she gets the up-to-date 'bill'. If she presses the 'go to my bank' button, the homepage changes to the website of my favorite bank. It helps her to pay over those expenses in the easier manner.

Case Z
At 7:00pm, Susan leaves her office. On the way home, her home server at her intelligent apartment checks her current location to access the LBS system installed in her cellular phone. Recognizing her car approaches near by, the home server automatically turns on the room heating system to reach the temperature according the current temperature, climate, and Susan's preference. As she reaches to the front door, the home server recognizes her and unlocks the door. At the same time, it turns on the lights of the porch and living rooms.

Case R
At 5:00pm, Grace is choosing a dress in the dressroom to go to a party tonight with her husband. When she stands in front of Media-mirror, an augmented reality device, the mirror displays a list of dresses suited for the party. By clicking on the dress shown in the mirror, she checks how she looks like if she wears the dresses. After all, when she chooses a dress to wear and press a button to connect to her husband's mobile device such as PDA, Media-mirror sends the preview to her husband's mobile device to show her look.

7.2 Results

To assess whether the provided services are 'ubiquitous' or not, Layer 1 assessment was performed. As a result, since the scores of Case L and Case R are which are larger than the threshold, they are regarded as ubiquitous computing services. In case of Case D, on the other hand, since the score is less than the threshold, the case is not accepted as ubiquitous computing services, and is excluded from further assessment steps. However, as a matter of course, the fact that the Case D is excluded for further assessment does not mean that the service of Case D is not acceptable to the users, but means that the service could be another type of IT-based service. Nevertheless, layer 1 assessment is very useful for efficient assessment in that the assessment prunes 'so called' ubiquitous computing service which is not essentially ubiquitous. As the result of layer 2-1, the scores of technical dimension of Case L and Case R are derived. As the result of the layer 2-2, the final score of Case L is much lower than Case R. This means that the service quality of Case R is perceived much better than that of Case L.

In summary, the results of layer 1 and 2 are shown as Table 9.

Table 9. Overall Results of Illustrative Example 2

Layer	Case D	Case L	Case R	
1	1.472	3.870	4.872	*threshold = 3.0*
2-1	-	3.259	3.519	
2-2	-	2.064	3.065	

Finally, we can conclude that Case L has the more value than Case R for Ubiquitous Computing Service (See Fig. 3).

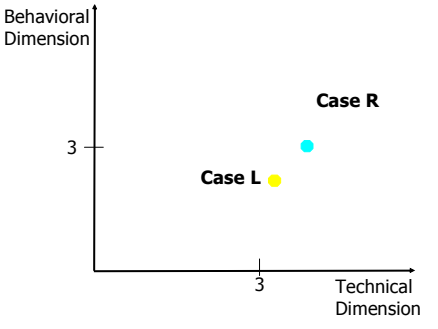


Fig. 3. Positioning Map of Each Case

8 Conclusion

In this paper, we have proposed a method to assess ubiquitous computing services and systems in terms of level of service. A multi-layered approach is proposed to represent the assessment model: capability-based ubiquitous service assessment as layer 1, technology-driven assessment and behavioral-driven criteria as layer 2-1 and 2-2, respectively. One of the main contributions of this paper is that we include both behavioral aspects, as well as technical aspects, for consideration in the assessment. Evaluating levels of service is useful for analyzing the potential service value. We believe that the success of ubiquitous computing technology would rely on how the level of service itself is suited to the user's level, as well as conventional business success factors such as financial, operational, and technical feasibility. Especially, the level of capability and the level of ubiquity at layer 1 and 2-1 can be used the method for understanding the degree of u-transformation of the legacy services that tries to provide the service using the ubiquitous computing technology. Despite the future works remained, the proposed assessment model of this paper would be to some extent provide the practitioners with the insights to discern what are actually ubiquitous computing services in comparison from legacy mobile or pervasive computing based services, and to build the ubiquitous computing services to be used in the future electronic markets.

Acknowledgement

This research is supported by the ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

References

1. Bellotti V. et al.: Making Sense of Sensing Systems: Five Questions for Designers and Researchers. Proceedings of Conference of Human Factors in Computing Systems-C. (2002) 415-422
2. Jiang, J., Klein, G., Carr, C.: Measuring Information System Service Quality: SERVQUAL from the Other Side. MIS Quarterly, 26(2) (2002) 145-166
3. Kettinger W.J., Lee C.C.: Perceived Service Quality and User Satisfaction with the Information Services Function. Decision Sciences, Vol. 25(5) (1994) 737-766
4. Mankoff, J., Dey, A.K., Hsieh, G., Kientz, J., Lederer, S., Ames, M.: Heuristic Evaluation of Ambient Displays. Proceedings of the Human Factors in Computing Systems. ACM Press, (2003) 169-176.
5. Parasuraman A., Zeithaml V.A.: Alternative Scales for Measuring Service Quality: A Comparative Assessment Based on Psychometric and Diagnostic Criteria. Journal of Retailing, 70(3) (1994) 201-230
6. Parasuraman, A., Zeithaml, V.A., Berry, L.L.: SERVQUAL: A Multiple-Item Scale for Measuring Customer Perceptions of Service Quality, Journal of Retailing, Vol. 64(1) (1988) 12-40
7. Pitt, L.F., Watson, R.T., Kavan, C.B.: Service Quality: A Measure of Information Systems Effectiveness, MIS Quarterly, Vol. 19(2) (1995) 173-187
8. Riekki, J., Isomursu, P., Isomursu, M.: Evaluating the Calmness of Ubiquitous Applications, Proceedings of Production Focused Software Process Improvement: 5th International Conference, PROFES 2004, Kansai Science City, Japan, Vol. 5(8) (2004)
9. Scholtz J., Consolvo S.: Toward a Framework for Evaluating Ubiquitous Computing Applications, Pervasive Computing. Vol. 3(2) (2004) 82-89

UPmP: A Component-Based Configurable Software Platform for Ubiquitous Personalized Multimedia Services

Zhiwen Yu[1,2], Xingshe Zhou[2], Changde Li[2], Shoji Kajita[1], and Kenji Mase[1]

[1] Information Technology Center, Nagoya University, Japan
zhiwen@itc.nagoya-u.ac.jp, kajita@nagoya-u.jp, mase@nagoya-u.jp
[2] School of Computer Science, Northwestern Polytechnical University, P.R. China
zhouxs@nwpu.edu.cn, changde_lee@126.com

Abstract. As multimedia contents are becoming widely used in ubiquitous computing environments among many application fields, e.g. educational content management, entertainment, and live surveillance, the demand of personalized access to these contents has increased dramatically. Delivering ubiquitous personalized multimedia services (UPMSs) is a challenging task, which relies on many different functions. In this work, we propose a three-layer software platform, called UPmP to support efficient development and deployment of UPMSs. It fulfills the core functions for UPMS including service management, multimedia recommendation, adaptation, and delivery. We adopt component-oriented approach in building the platform. Therefore the configurability of the platform is inherently achieved. A representation model is introduced to hierarchically organize components and describe meta-level information about components. We also present a visual configuration tool together with a XML-based language for the purpose of platform configuration. The experimental results show the UPmP is flexible to be configured under different settings, and the overheads are acceptable.

1 Introduction

With rapid development of multimedia and communication technologies, it becomes possible to offer multimedia content to people whenever and wherever they are through different devices, such as personal computer, personal digital assistants (PDAs) and mobile phones. Multimedia content is widely used in ubiquitous computing environments among many application fields, such as digital course management, entertainment, and live surveillance. The number of multimedia content to access can be quite overwhelming. To quickly and effectively provide content from large amounts of media information, in the right form, to the right person, the multimedia content need to be personalized based on the user's preferences and his current contextual information, such as time of day, user location, and device conditions. These services are so-called ubiquitous personalized multimedia services (UPMSs).

Delivering UPMSs is a challenging task. It relies on many different functions, such as service management, multimedia adaptation, multimedia recommendation,

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1069 – 1079, 2006.

multimedia delivery, etc. Software infrastructures are needed to enable such functions to be achieved easily and systematically so that the service providers and application developers just need to concentrate on the application itself. In this paper, we present a software platform, namely UPmP (**U**biquitous **P**ersonalized **m**ultimedia **P**latform) to support efficient development and deployment of UPMSs. We adopt component-oriented approach in building the platform. Therefore the configurability of the platform is achieved inherently. Component-based software design has been widely utilized in many fields to implement complex functions. A software component is a unit of composition that can be deployed independently and is subject to composition by a third party [1]. Three major component models are presented and used successfully today: COM, CORBA, and Javabeans.

There are several benefits from using the UPmP software platform. First, it integrates third-party software to accomplish software reusability and complex function consummation. Second, the platform is configurable and allow service provider to select different functions based on the service needs. Third, the atomic components within the platform can be taken from pre-existing applications. It facilitates service development so as to reduce the cost of development as well as the time to market.

2 UPmP Architecture

The UPmP architecture consists of three layers: multimedia resources, service function components, and service instances (UPMSi), as shown in Fig. 1.

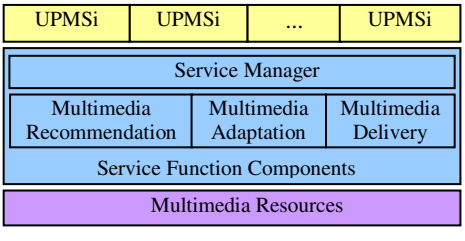


Fig. 1. UPmP architecture

2.1 Multimedia Resources

Multimedia resources are composed of multimedia content and corresponding description metadata. For the sake of interoperability with third-party services and applications, we adopt MPEG-7 description schema to represent multimedia metadata. The MPEG-7 *Creation DS* and *Classification DS* are used to describe information about the media item, such as the title, keyword, director, actor, genre, and language. This information is used to match user preferences. The *Variation DS* is used to specify variations of media content as well as their relationships. The *Variation DS* plays an important role in our content recommendation by allowing the selection among the different variations of the media content in order to select the most appropriate one in adapting to the specific capabilities of the terminal devices and network conditions.

The multimedia resources layer can integrate varied multimedia repository from a wide range of content providers by leveraging the O.K.I (Open Knowledge Initiative) Repository OSID (Open Service Interface Definition), which gains access to content in a manner that hides the technical detail by which that content is provided [2]. The O.K.I Repository Specification has been used to successfully integrate several applications with multiple content repositories, such as Sakai [3], which aims at building a collaboration and learning environment for higher education.

2.2 Service Function Components

The service function components are deployed as two sub-layers. The top layer is Service Manager. The bottom layer contains three components: Multimedia Recommendation, Multimedia Adaptation, and Multimedia Delivery. The Service Manager is responsible for lifecycle management of services. It interacts with services directly, and invokes functionalities supported by the function components in the bottom layer. Multimedia Recommendation is to select the right content in the right form for a service request. It takes user preference, terminal capability, and network condition into account. Multimedia Adaptation adjusts multimedia content to different requirements from service manager. It mainly involves two kinds of processes: content summarization and content transcoding, e.g. video-to-image conversion. Multimedia adaptation can be statically done at authoring time prior to delivery or dynamically done on-the-fly if needed. Multimedia Delivery is responsible for streaming or downloading media content to various terminals through different networks. If the modality recommended is continuous video or audio, the media deliverer streams the content to terminals. On the other hand, if the modality is static image or text, the media deliverer just downloads the content.

2.3 Service Instances

The service instances are concrete ubiquitous personalized multimedia services requested by a wide range of devices in ubiquitous computing environment. There are two typical scenarios for ubiquitous personalized multimedia services. One is providing a recommendation list with top L items. The other one is directly presenting the item with the highest score according to user preferences.

3 Component Representation

Components are functional units forming the UPmP platform. They can be composed to provide ubiquitous personalized multimedia services. A component comprises two parts: a metadata description and a processing entity. Component description presents detailed information of the component including component name, category, programming language, interface, hardware requirement, and software requirement (e.g. libraries, depended components). The component description is mainly used in platform configuration as well as service composition. Component entity is a software program (code) to accomplish a particular function.

For the sake of efficient organization, we model the components as a hierarchy. We give 3-layer definition to UPmP component hierarchy classification. The root element UPmPComponent is abstraction for all components. It is mainly divided into four categories, i.e. the second layer includes four items, which are ServiceManaging, ContentRecommendation, ContentAdaption, and ContentDelivery. The leaf components are different implementation algorithms or mechanisms of the abstract function in the upper layer. The UPmP component hierarchy structure is shown in Fig. 2. For space consideration, we here merely present the detailed structure of ContentAdaptation component.

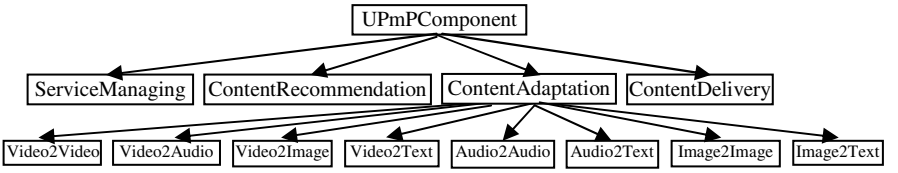


Fig. 2. Component hierarchy

For efficient discovery and reuse, information about the component should be described and advertised. Then the system can lookup and match desired components for a particular service according to the component metadata. We propose a service component description language (SCDL) based on XML to describe component. SCDL defines three kinds of information of a component: general information, public interface, and composition logic. The general information indicates a component's name, category, parent class, and also gives a brief annotation for the component. The category structure is the same as component hierarchy presented above. Category and annotation are very useful for component search and match. The public interface describes input and output formats, and also the real running entities of the component. The composition logic is provided to support service composition. It indicates the order of a component when it is composed with other categories of components using the Following and FollowedWith elements. It also indicates whether a component can be combined with its brother component. The components with the same parent class are regarded as brother components. For instance, Video2Image and Video2Text are brother components.

Fig. 3 shows a component metadata example. The component's name is *SComponentExample* and its category is *Video2Image*. The annotation and parent class are also given. From PublicInterface part, it can be seen that the input of the component is *MPEG* file, and the output is *JPEG* and *BMP* files. Two executors, *mpeg2jpeg.jar* and *mpeg2bmp.jar* are specified as the real running entities of the component. The composition logic indicates that the component can follow the components of *ContentRecommendation* and *ContentDelivery*, and be followed by the components of *ContentDelivery* in composing a service. However, it cannot be combined with its brother component.

```
<?xml version="1.0" encoding="UTF-8"?>
<SComponent xmlns="http://www.dcel.nwpu.edu.cn/SComponent_Schema">
<SComponentDescription>
    <GeneralInformation>
      <Name>SComponentExample</Name>
      <Category>Video2Image</Category>
      <Annotation>Transforming a video content into images.</Annotation>
      <ParentClass>ContentAdaptation</ParentClass>
    </GeneralInformation>
    <PublicInterface>
      <Input>
            <Format>MPEG</Format>
      </Input>
      <Output>
            <Format>JPEG</Format>
            <Format>BMP</Format>
      </Output>
      <Executors>
            <Executor>mpeg2jpeg.jar</Executor>
            <Executor>mpeg2bmp.jar</Executor>
      </Executors>
    </PublicInterface>
    <CompositionLogic>
      <Following>
        <Category>ContentRecommendation</Category>
        <Category>ContentDelivery</Category>
      </Following>
      <FollowedWith>
        <Category>ContentDelivery</Category>
      </FollowedWith>
      <CombinedWithBrotherComp>NO</CombinedWithBrotherComp>
   </CompositionLogic>
</SComponentDescription>
</SComponent>
```

Fig. 3. Component metadata (example)

4 Platform Configuration

The UPmP platform offers a set of optional functionalities, which can be switched off at the platform initializtion time. This flexibility is useful because not all the systems need to exploit the completed capabilities offered by the platform. In the simplest cases, the platform should support the development of lighter systems, which reduce the overhead during the interaction with the user. In particular, the developer may choose the multimedia recommendation techniques best suiting the requirements of the application domain. For instance, collaborative filtering efficiently support recommendation of multimedia, but it only works if ratings of the items are available. In contrast, content-based filtering is more suitable to the cases where meta-level information about the items is available.

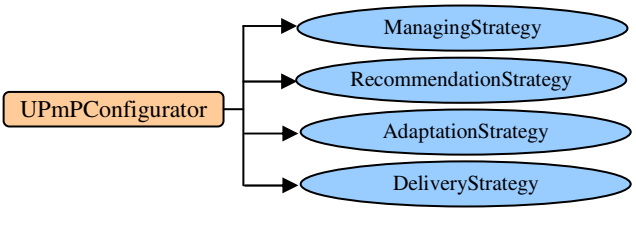


Fig. 4. UPmP configurator

To provide this flexibility, we have made the platform configurable so that the developers can select the functionalities offered by the platform. A GUI-based tool, called UPmPConfigurator, is proposed to customize platform functionalities according to needs or characteristics of different systems. The UPmPConfigurator mainly includes four parts: service managing strategy, content recommendation strategy, adaptation strategy, and delivery strategy as shown in Fig. 4.

For purpose of configuration at initializtion time, an XML-based platform configuration language (XPCL) is designed. The Extensible Markup Language (XML) is an ideal configuration language, because it is the universal format for structured documents and data on the Web and also extensible. However, XML itself does not tell platform administrator how to specify platform configuration parameters for his/her services. We define a set of suitable tags to specify platform running policies based on XML syntax. The XPCL is accordingly divided into four parts. The <UPmPConfiguration> tag is the root tag. It contains the entire four parts parameter or policy configuration. The <ServiceManagingStrategy> tag contains two tags: <EnterBlockedState> and <EnterWaitingState>. They specify whether the services enter *Blocked* or *Waiting* state respectively. The services are modeled with a lifecycle management model based on FSM (Finite State Machines), which is presented in our early work [4]. The <RecommendationStrategy> contains at least one <RecommendationAlgorithm> tag, which specifies a particular algorithm, e.g. content-based recommendation. The <AdaptationStrategy> contains at least one <Transcoding> tag, which is also a container tag. The <Transcoding> tag has one required attribute, "type", which identifies the type/class of the transcoding. It contains at least one <Transcoder> tag. The <DeliveringStrategy> contains at least one <DeliveringMode> tag.

```
<?xml version="1.0" encoding="UTF-8"?>
<UPmP xmlns="http://www.dcel.nwpu.edu.cn/UPmP_Schema">
<UPmPConfiguration>
  <ServiceManagingStrategy>
     <EnterBlockedState>YES</EnterBlockedState>
     <EnterWaitingState>NO</EnterWaitingState>
  </ServiceManagingStrategy>
  <RecommendationStrategy>
    <RecommendationAlgorithm>Content-based Recommendation</RecommendationAlgorithm>
      <RecommendationAlgorithm>Rule-based Recommendation</RecommendationAlgorithm>
  </RecommendationStrategy>
  <AdaptationStrategy>
    <Transcoding type="Video2Image">
      <Transcoder>MPEG2JPEG</Transcoder>
    </Transcoding>
    <Transcoding type="Video2Text">
      <Transcoder>MPEG2TXT</Transcoder>
    </Transcoding>
    <Transcoding type="Audio2Text">
      <Transcoder>WAV2TXT</Transcoder>
    </Transcoding>
  </AdaptationStrategy>
  <DeliveryStrategy>
    <DeliveringMode>Streaming</DeliveringMode>
    <DeliveringMode>Downloading</DeliveringMode>
  </DeliveryStrategy>
</UPmPConfiguration>
</UPmP>
```

Fig. 5. Platform configuration (example)

Fig. 5 gives an example of UPmP platform configuration. The services are set to enter the Blocked state but not the Waiting state. Two recommendation algorithms, content-based recommendation and rule-based recommendation are included. For multimedia adaptation, this configuration sets three transcoding mechanisms: Video2Image, Video2Text, and Audio2Text. The corresponding transcoders are MPEG2JPEG, MPEG2TXT, and WAV2TXT. The delivery strategy includes audio/video streaming and image/text downloading.

An XPCL-based description file is generated after a user completes the configuration through the visual UPmPConfigurator tool. The configurator interprets the configuration file, loads specified components, and builds a running platform.

5 Service Composition

A multimedia service is built as the composition of UPmP supported functional components. In our system, service composition is the selection of suited service components in order to deliver the service to a terminal in appropriate form. It consists of two steps. The first step is abstract function selection, e.g. multimedia adaptation. The second one is concrete handler selection, e.g. Video-to-Image transcoder.

Fig. 6 illustrates some composition paths for multimedia services under different conditions. S denotes the start point of services, while T stands for the terminal point. The rectangle contains components for recommendation purposes including components A, B and C. The oval contains adaptation components D, E, F, G, and H. The delivery components I and J are represented in the rounded rectangle. In Fig. 6(a), since the appropriate variation already exits, the service goes directly to deliver it. In Fig. 6(b), the service firstly performs video-to-text transcoding, and then

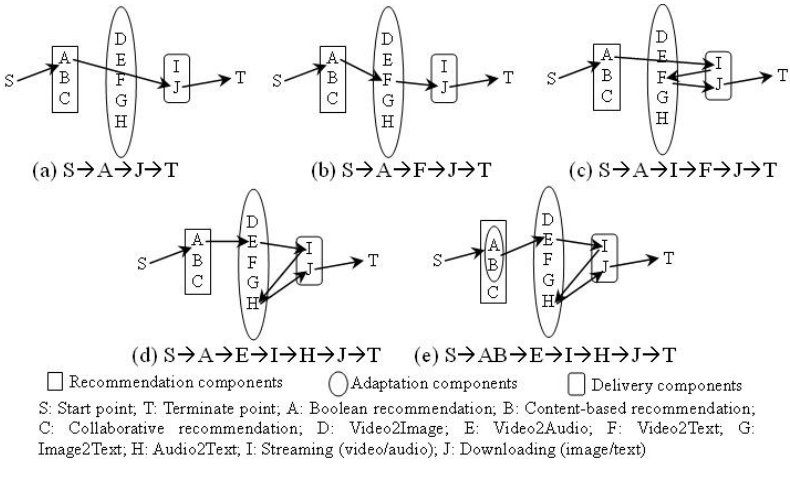


Fig. 6. Service composition examples

downloads the text. In Fig. 6(c), the service first performs video streaming under high bandwidth, then for the decrease of bandwidth, it performs video-to-text transcoding, and then delivers the text. In Fig. 6(d), the service first performs video-to-audio transcoding, then audio streaming, and then for bandwidth decrease dramatically it has to perform audio-to-text transcoding, last sends the text. In Fig. 6(e), component A and B are combined for the recommendation.

6 Implementation and Experiment

We have implemented a prototype of UPmP. The visual platform configuration tool UPmPConfigurator is implemented in Java Swing. The components are all implemented in Java and built as .jar files. Some media adaptation components are integrated from third party. To handle component dependency during platform configuration and service composition, we can utilize the Dependency Injection [5] pattern provided by the Spring framework [6]. Fig. 7 shows the main interface of UPmPConfigurator. It allows the developer to choose different strategies for service management, content adaptation, content delivery, and content recommendation. When a content adaptation button, e.g. Video-to-Image, is clicked, a list of transcoders will be presented for user's choice. For ubiquitous multimedia recommendation, rule-based technique is always combined with the other recommendation algorithms. It is used to infer the appropriate presentation form of a selected content from network condition and device capability [7]. The developers can directly choose some existing rules or define specific rules of their own.

We mainly evaluate our system by measuring the overhead of UPmP's configuration in terms of time. It includes the time to parse and interpret the configuration XML file, load specified components, and link them together. The experiment is conducted on a PC with 1.6 GHz Pentium 4 CPU, and 1 GB RAM running Windows XP. There are five different setup configurations involved in this experiment. The configuration details are presented in Table 1. Configuration 1 is a completed setup with all the four categories of components selected. Configuration 2, 3, and 4 test the overheads of different types of components. Configuration 5 is the configuration for a real running system. When selecting a category of content adaptation component, e.g. Video-to-Image, we simply select all the transcoders of it.

Table 1. Configuration details

Configuration	Selected components
1	all the four categories of components
2	all the service managing and recommendation components
3	all the content adaptation components
4	all the content delivery components
5	Enter Blocked State, Video-to-Image, Video-to-Text, Audio-to-Text, Streaming, Downloading, Content-Based Recommendation, and Rule-Based Recommendation

The experimental results are shown in Fig. 8, in which the time for each configuration is an average value of 10 runs. The completed configuration takes about 8.5 seconds. We can also observe that the main configuration overhead comes from linking content adaptation components, which takes nearly 75% of the total configuration time. Fortunately, this overhead is merely generated during the platform setup. It does not affect the performance of service delivery at running time. Through this experiment, we could conclude that the UPmP is flexible to be configured under different settings, and the overheads are acceptable.

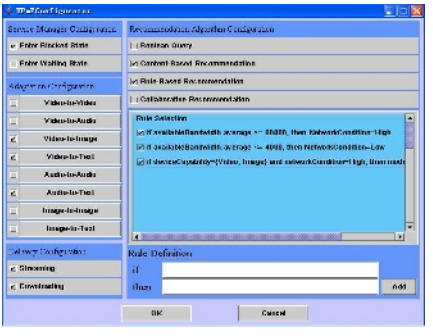


Fig. 7. Main interface of UPmPConfigurator

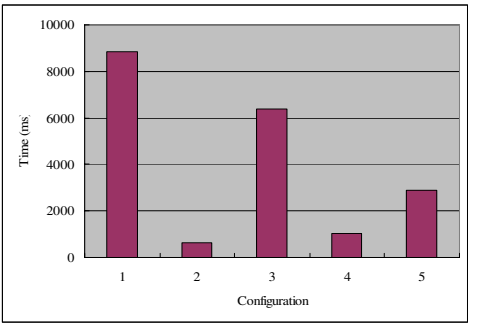


Fig. 8. Overhead of UPmP configuration

7 Related Work

There has been much research work done specifically to provide systematic architectural support for ubiquitous multimedia delivery. Gamma [8] is a content-adaptation server for wireless multimedia applications, which supports the automatic and transparent transcoding for individual users according to their pre-configured user profiles. CANS [9] is an application-level infrastructure for injecting application-specific components into the network. It supports automatic deployment of transcoding components for ubiquitous and network-aware access to Internet services. Dali [10] is a set of reusable libraries, which can be used for building processing-intensive multimedia software. Gamma, CANS, and Dali are mainly towards content adaptation without the support of service management and multimedia recommendation. QCompiler [11] is a programming framework to support building ubiquitous multimedia applications, which are mobile and deployable in different ubiquitous environments, and provide acceptable application-specific Quality-of-Service (QoS) guarantees. However, QCompiler does not involve server platform configuration and personalization functionalities. CAPNET [12] is a context-aware middleware for mobile multimedia applications. It fulfils broad functionalities including service discovery, event management, context storage, media content retrieval and adaptation to various mobile devices. But CAPNET is not built based on component and is not configurable in setup.

Recently, to deliver personalized multimedia to ubiquitous devices, some researchers have considered both user preference and device/network capability to

generate appropriate presentation to terminals e.g. [13] and [14]. However, none of them are proposed from the middleware perspective.

8 Conclusion

We described the architecture and key features of the UPmP, a general-purpose platform to support the deployment of ubiquitous personalized multimedia services. The major contributions of this paper include: (1) proposing a three-layer architecture for the general-purpose software platform; (2) introducing a component representation model that is helpful for component organization, indexing, and description; (3) presenting a configuration tool and an XML-based platform configuration language; (4) illustrating service composition under different conditions.

Future work is planned on the extension of the platform to support re-configuration at running time. We also envision deploying function components in a network of computers to improve performance in terms of throughput and scalability. It will call for the incorporation of load sharing and task scheduling mechanisms.

Acknowledgment

This work is partially supported by the Ministry of Education, Culture, Sports, Science and Technology of Japan under the “Development of Fundamental Software Technologies for Digital Archives” project, and the Doctorate Foundation of Northwestern Polytechnical University of China.

References

1. Szyperski, C.: Component Software: Beyond Object-Oriented Programming. Addison-Wesley, Reading, Mass (1998)
2. Thorne, S., and Sim, S.: Integrating Applications with Repositories Using the O.K.I Repository OSID. JA-SIG Conference. (2005)
3. Sakai, http://sakaiproject.org/
4. Yu, Z.W., Zhou, X.S., Zhang, D.Q., Lugmayr, A., and Yu, Z.Y.: A Ubiquitous Personalized Multimedia Service Model Based on FSM, Proc. Of the 6th IEEE Intl. Conf. on Information and Technology: Coding and Computing, USA, 801-802 (2005)
5. Fowler, M.: Inversion of Control Containers and the Dependency Injection pattern. http://www.martinfowler.com/articles/injection.html. (2004)
6. Spring Framework, http://www.springframework.org/
7. Yu, Z.W., Zhang, D.Q., Zhou, X.S., Chin, C.Y., Wang, X.H., and Men, J.: Supporting Context-Aware Media Recommendation for Smart Phones, IEEE Pervasive Computing Magazine, Vol. 5, No. 3, July-September (2006)
8. Lee, Y.W., Chandranmenon, G., and Miller, S.C.: Gamma: A Content-Adaptation Server for Wireless Multimedia Applications. Lucent Technologies white paper. (2003)
9. Fu, X., Shi, W., Akkerman, A., and Karamcheti, V.: CANS: Composable, Adaptive Network Services Infrastructure. USENIX Symposium on Internet Technologies and Systems (USITS), March 2001, 135-146 (2001)

10. Ooi, W.T., et al.: Dali: A Multimedia Software Library. Proceedings of 1999 SPIE Multimedia Computing and Networking, 264-275 (1999)
11. Wichadakul, D., Gu, X.H., and Nahrstedt, K.: A Programming Framework for Quality-Aware Ubiquitous Multimedia Applications. ACM Multimedia 2002, 631-640. (2002)
12. Davidyuk, O., et al.: Context-aware middleware for mobile multimedia applications. The 3rd International Conference on Mobile and Ubiquitous Multimedia, 213-220 (2004)
13. Steiger, O., Ebrahimi, T., and Sanjuan, D.M.: MPEG-based Personalized Content Delivery. IEEE Intl Conf. on Image Processing, 45-48 (2003)
14. Lemlouma, T., and Layaida, N.: Encoding Multimedia Presentations for User Preferences and Limited Environments. IEEE ICME, 165-168 (2003)

An End User Tool for Customising Personal Spaces in Ubiquitous Computing Environments

Jeannette Chin, Vic Callaghan, and Graham Clarke

IIEG, Computer Science Department, Essex University, United Kingdom
{jschin, vic, graham}@essex.ac.uk
http://iieg.essex.ac.uk

Abstract. We present a variant of end-user programming targeting ubiquitous computing environments that allows non-technical users to create "programs" to customise their personal living spaces. Using this end-users do not need to write program code, or follow a rigid sequential list of actions in order to achieve results Rather they only need to show the system the required behaviour via physical interactions with the environment. Finally, we report on a user evaluation that indicates end-users find this approach to be a useful and enjoyable experience.

1 Introduction

End-User programming employs techniques that allow users of an application program to create "programs" without any technical expertise [9]. One way to achieve this is to create a "scripting language" by abstracting conventional algorithms of functionalities into representations (eg graphical objects) and then operate on these representations to create a program. Another way to achieve this goal is to employ an approach introduced by Smith in the mid-seventies, Programming-by-Example (PBE), where the functionality required is demonstrated via concrete examples by the end-users, rather than presented in the form of abstractions [23]. Current PBE work is based on single desktop environment [21, 14]; in our work we take PBE forward by applying it to highly distributed ubiquitous computing environments. In this we employ a "show-me-by-example" approach allowing non-technical end-users to create "programs" for customising their personal space. End-users are neither required to write program code, nor follow a rigid sequential list of actions in order to 'program' their personal space. The end-user *shows* the system the required behaviour by demonstrating it via physical interactions with the environment. Using a *show-me-by-example* approach, end-users are able to create "programs" that encapsulate the actions that they perform. These "programs" can be retrieved on demand, or be terminated on request. We called this approach Pervasive interactive Programming (PiP).

2 Ubiquitous Environments – An Area Suitable for End User Programming

Ubiquitous Computing [28] means our living environments are being populated with an increasing number of networked devices and smart sensors, offering a variety of

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1080 – 1089, 2006.

services. The availability of network delivered services opens up the possibility of assembling composites of services enabling the environment as a whole to take on a collective behaviour, creating intelligent ubiquitous environments.

The home of the future might contain tens or even hundreds of networked devices offering services. Some of these devices could take the form of physical appliances whilst others might be virtual appliances made up of aggregated services from numerous separate devices. Remote access could allow manufacturers to support customers in new ways, or to gather usage data that would help them improve the design of their products. From a customer's perspective the environment could function in a smarter way, as appliances and services could coordinate actions in ways designed by manufacturers or the customers themselves.

Technical infrastructures are vital for supporting the dynamic nature of ubiquitous environments. Research in this area has been directed at physical development [1], through system architecture [15], to middleware [3, 13, 18]. Additionally attention has been directed to the user's perspective and needs with research aimed at helping end-users control the environment [10, 17, 26], build high-level abstractions to support tasks [13, 27] and apply_task-based paradigms [19, 22]. Automating the environment to learn from the user, using intelligent agents has been addressed by a number of groups such as the University of Essex [5], the University of Colorado [20] and the University of Texas [8]. More recently, the theme, 'empowering end-users' has emerged with much attention focused on developing end-user programming tools [12, 25], allowing the end users to program their environment.

The overall objective of ubiquitous computing is to make life easier for the end users, but the problems of privacy, robustness and usability have come to the fore. Such considerations have led to a debate on issues such as agent versus human control which touches on areas of ethics and human values such as who, what, when (the "3 W's") should monitor, program and control our environments [6]; to these ends, the approach adopted in this paper seeks to maximise system transparency and end user control.

3 Motivations

Currently most end-user programming tools for ubiquitous environments are focused on streamlining the use of the input languages or metaphor-based GUI interfaces, aimed at simplifying the use of applications for the users. They are largely based on the procedural programming metaphor requiring the user to mentally manipulate constructs that would be familiar to most programmers, thereby placing a significant cognitive load on the user. We have been inspired by the ease that people perform daily routine tasks (eg. switch on the light when the room gets dark, mute the TV when the telephone rings etc), we decided to direct our approach at finding a way of programming that was natural and mimicked familiar everyday practices as much as possible. Thus the motivation behind PiP was to create a system that maximized user's trust and empowered their creativity by developing control and operational transparency and enabled them to customise their own environment, without the need for detailed technical knowledge. This puts the user firmly in control of the '3 W's'.

4 Pervasive Interactive Programming (PiP)

PiP [UK patent: GB 0523246.7] provides a platform that utilises the environment as the programming space enabling the user to customise the functionality of their personal space. We assume that the services offered from networked devices are supported by an underlying protocol layers, discovery/registration process etc none of which are described in this paper. The term "program" in our approach refers to a representation of a collection of operations in the environment. Our meaning of a "program" is closer to that of a non-terminating process which may also be graphically represented. However, whilst such a "program" is constructed by an end user, who has no technical expertise, it produces an effect that is normally achieved by conventional programming.

4.1 A Deconstructed Model – Virtual Device

In ubiquitous environments devices and their services are discoverable and accessible. Consequently there is the potential to group communities of services together enabling the creation of a "virtual device". This new model of "virtual devices" offers a radical alternative to the conventional perception of a "device", as the functional units that make up current devices may be shared. A "virtual device" made up of other devices' functionalities could accomplish tasks that individual devices were not capable of. Moreover "virtual devices" could have an impact on developers and how they develop their products. Most importantly, end users could leverage this device and service rich ubiquitous environment to create their own "virtual devices" to suit their needs. We refer to such communities as **M**eta**Ap**pliance**s (MAps),** the representation of the virtual deconstructed device model.

4.2 MetaAppliances (MAp)

The concept of a MAp is a core concept in PiP. From a logical perspective, a MAp has primitive properties and a collection of *Rules* that determine the behaviour of the environment, which is the end user's personal space. Rules are essentially a marriage of two different types of behaviour, namely 'Antecedent' (condition) and 'Consequent' (action). Each behaviour has the property of a "virtual device". The 'Antecedent' of a Rule can be described as "if" while the 'Consequent' of a Rule can be described as "then". A Rule can contain 0-n 'Antecedent' and 1-n 'Consequent', and a MAp can contain 0-n Rules. A UML representation of a MAp relationships with *Rules* is shown in Figure1.

From the end users' viewpoint a MAp is a "program" that would create the sort of environmental behaviour they want. As individual end users have their unique preferences and their particular needs, it makes sense to let the users define their own MAps. Thus *MAps are created under the directions of end users*. MAps can have a graphical representation and thus can be visible to the user who created them, either at the time of creation or later when they can be retrieved, edited, shared, executed, or removed.

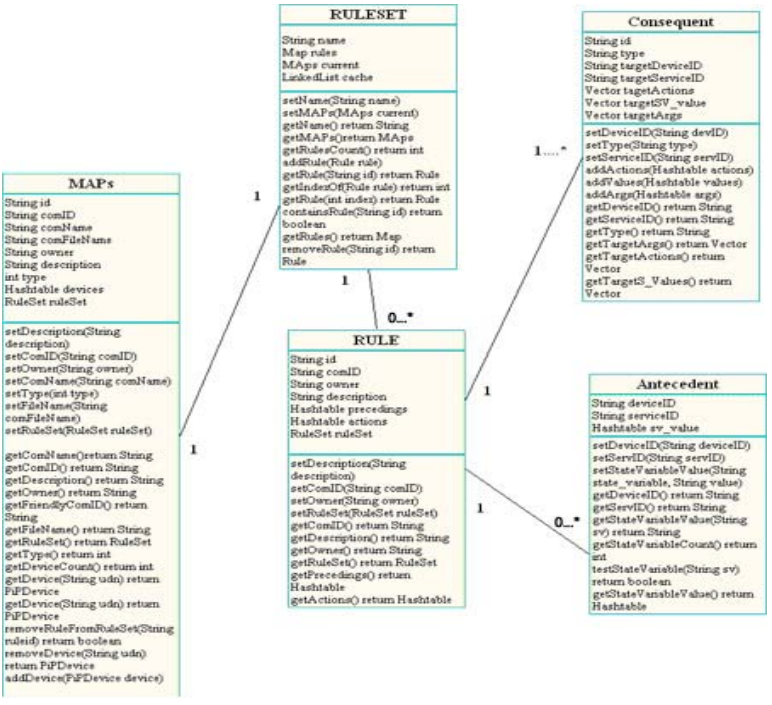


Fig. 1. The UML representation of the MAp object structure and its Rules

4.3 System Architecture

PiP is designed to work in real time within a ubiquitous computing environment. It has an event-based object-oriented asynchronous architecture. Unlike macro languages, where sequences of instructions or actions are significant, PiP assumes the logical sequence of actions is not important. It employs a rule policy to maintain a MAp process in which "a set of conditions is satisfied if the conditions defined within the context of this set are all satisfied". PiP leverages UPnP™ technology as its middleware and communication protocol, enabling simple and robust connectivity among devices and PCs.

It has modular framework (Figure 2), comprises six core modules as follow:

1. "Interaction Execution Engine" (IEE) – this module has a control point and is responsible for device discovery, service events subscription, and performing network actions requests.
2. "Eventing Handler" (EH) – this module is responsible for interpreting low-level network events (eg device discovery), device service events (eg service state changes) and high-level events caused by user interactions.
3. "Knowledge Engine" (KE) – this module is responsible for assembling and instantiating a "virtual device" before storing them on to the Knowledge Bank.
4. "Real-time MAp Maintenance Engine" (RTMM) — is a process that maintains the records of current and previously created MAps.

5. "Real-Time Rule Formation Engine" (RTRF) – this module is responsible for assembling rules based on user interactions within the "demonstration" mode[1].
6. "GUI" – A graphical interface called "PiPView" that the user can use to inspect the environment, compose/delete Maps/Rules etc, interact with the system and control physical environment.

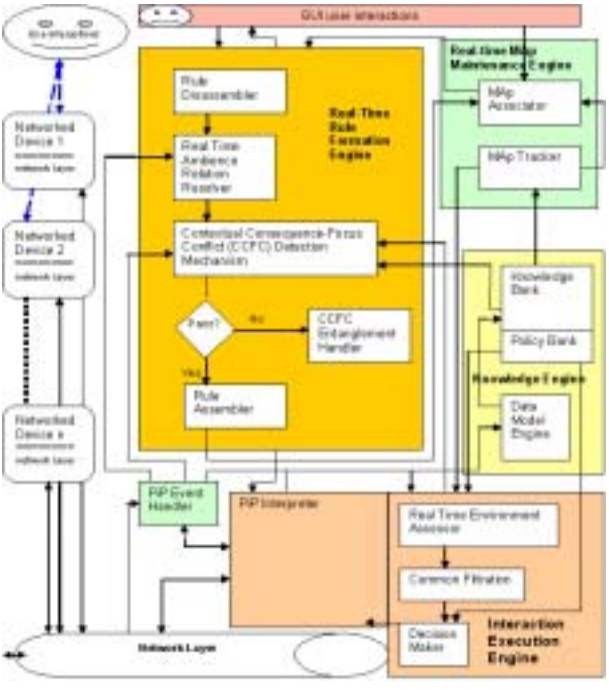


Fig. 2. PiP modular framework [UK patent: GB 0523246.7]. 05

In PiP, all networked items that exist in the ubiquitous environment can be regarded as user interfaces. Through these user interfaces, users can interact intuitively with the environment and the procedure for configuring their personal space is very simple: the user just needs to create a MAp and show the system the functionalities that the MAp should have by interacting with the environment and PiP will do the rest for the user.

5 Related Work

Visualisation techniques are often employed for making computer systems more comprehensible and easy to use. For example some X10 clients provide graphical interfaces that allow users to specify the behaviour of various devices or objects in their homes

[1] A "demonstration" mode begins when the user clicks "ShowMe" button and ends when user clicks "Done!" button.

based on events or conditions [29]. SiteView [4], has incorporated tangible techniques for programming active environments with predictive visualizations. The Speakeasy system [11] supports the *ad hoc*, end-user configuration of devices and applications through a set of patterns defined in mobile code via web interfaces. Research has extended information interfaces to physical mechanisms such as Ubi-Finger, a gesture-input device developed by Tsukada [26], which uses sensor techniques together with an infrared transmitter allowing the user to control an ubi-device by first pointing at it and then using a finger gesture. In addition, the iCAP system [24] allows users to prototype context-aware applications rapidly using a pen-based interface for specifying input and output devices, as well as behavioural rules through 'drag and drop' interactions and pie menus. In general, the above developments offer users alternatives on how to use and customize ubiquitous applications but they have not developed enough to offer non-technical users the capability/ flexibility to design and make "programs" in their own user defined environments.

Task driven computing, proposed by Wang and Garlan [27], is aimed at helping mobile users automate their service configurations in different environments. The AURA [13] project aimed at constructing a global environment where task-based configurations were mobilised and automatically restored within all environments visited by a user. Semantic "Task Computing", developed by Fujitsu Laboratories [19] took this concept forward significantly by enabling end-users to perform more complex tasks in ubiquitous environments. A variant of this concept called the Personal Operating Space (POS) was explored by Shahi [22]. From a computing perspective, Task Computing has the advantage of eliminating the need for the user to know how to achieve the end results, thereby allowing them to focus on the results which, in turn, has led to claims of significant productivity gains compared to conventional approaches [27]. However, there remain a few issues to be addressed for this approach to be more widely adopted. For instance the tasks themselves need to be created in advance and, given the ambiguous nature of the abstractions which form the basis of the Tasks, a degree of guesswork on what users might need and technical expertise is required for their definition. To date end users, have not been able to design and customise their own tasks, instead users use the pre-defined tasks built by experts.

End-User programming is characterised by the use of mechanisms to allow non-technical end-users to create "programs". Earlier work in this area was targeted primarily at desktop computing. Recently this vision has found its way into technology-rich ubiquitous environment where a number of different approaches are being explored. For instance, Humble [17] use a jigsaw, metaphor, enabling users to "snap" together puzzle-like graphical representations as a way of building applications. The HYP system [2] enables users to create applications for context-aware homes using a mobile phone based graphical interface. Media Cubes [16] offers a tangible interface for programming an environment in which each face of a cube is represented by a set of program structures. In this approach "Programming operations" are achieved by turning the appropriate face of the cube towards the target device. Truong's CAMP project [25] placed the end-users at the centre of the design experience by using a fridge magnet metaphor together with a pseudo-natural language interface that collectively enabled end-users to realize context-aware ubiquitous applications in their homes. The Alfred project proposed the concept of "goals", and "plans" to allow users to compose a program via

"teaching-by-example". The system utilised a macro programming approach which could be created by the user via verbal or physical interactions. This differs significantly to PiP which spawns non-sequence dependent processes based on a virtual device metaphor rather than macros and a procedural programming metaphor. According to Gajos no formal studies were completed and the work appears to have been cut short when he moved from MIT to the University of Washington. [12]

6 End User Evaluation

The evaluation was carried out in iDorm2[2], a newly built two-bedroom flat at the University of Essex to be an experimental ubiquitous computing environment. During the evaluations, PiP was set-up to run on a winXP tablet PC using a WIFI connection.

6.1 Evaluation Design and Procedure

The end-user evaluation was designed as a preliminary appraisal of PiP. Our objectives for the evaluation were to (1) assess whether PiP met its design objectives (2) assess how easy or not the end users found PiP to be in customising their personal space and (3) gather a view from a broader background of PiP users. Our strategy was to set-up open ended trials giving the end-users as much freedom as possible so that we could get a better idea of how participants would like to use the system, and how the system coped with different users. User freedom included time, methods, and tasks.

Eighteen participants (ages ranged from 22 to 65) drawn from a diverse set of backgrounds were invited for the evaluation. All participants had some minimal computing experience; Whilst 21.3% of the participants had a very good knowledge of programming, 57.4% of them had none at all.

An evaluation questionnaire was developed to assess the participants' judgements about the usability of PiP. It consisted of a set of seventeen statements, measuring attitudes over six usability dimensions[3]. The questionnaire used a five-point Likert scale with responses from "Strong Agree" through to "Strongly Disagree". Each of these dimensions consisted of a series of statements (from 2 to 4) and each statement offering a range of ratings (from 1 to 5). A higher rating score on the dimensions contributes towards the greater usability of PiP. The questionnaire was then iterated for checks and revised any ambiguous statements before it was piloted on 3 users.

Each trial was preceded by a 20-minutes training session to allow participant to familiarise themselves with the system, the nature of the task and the environment. The task for the trials was that the participant should use PiP to customise the environment as a personal space in the way they wanted; the participants were free to create one or more MAps of their own. No time limit was set for the participants to customise the space. Assistance was provided where needed. Following completion of the evaluations, the questionnaire was administrated to measure the participants' subjective judgements of PiP. Data was analysed using SPSS.

---

[2] iDorm2 at http://iieg.essex.ac.uk/idorm2/index.htm

[3] Six usability dimensions: Concept, User Controls, Cognitive load, Information Retrieval, Affective experience and Future Thoughts.

6.2 Results

Space doesn't allow detailed presentation of the results. However, in summary we found that 83% of participants were able to use PiP to customise their personal space with little or no assistance. Of the two methods available for demonstrating examples to the system 11% of the participants chose to customise their personal space wholly via GUI controls but 72% of them used physical interactions with the environment, the rest used a mixture of both. None of the participants found it difficult to understand the basic principles of the system. Other findings suggested that users found the system enjoyable and easy to use without any undue cognitive load (the average cognitive load was 4.3, indicating in the participants found the PiP process relatively simple).

7 Conclusion and Future Work

We have successfully implemented a "proof of concept" system called PiP, using an event-based modular architecture design which enables non-technical end-users to create MAps for customising their personal space in ubiquitous environments.

We reported on our end-user evaluation results based on a group of 18 participants selected from different backgrounds to evaluate their subjective views on the value of the research. The end-users were not required to follow a rigid logical sequence when customising their personal spaces, although we observed a minority of the participants, notably those with a computing background, preferred using a logical sequence. The initial results are encouraging as they show that PiP served different users well, allowing them to customise their personal space without undue cognitive load. Based on the results gathered, we concluded that PiP was able to support users composing MAp for customising their personal space in ubiquitous computing environments. At the outset of our work, one of our contentions for PiP was that participants would find the manual experience of customising their personal space relatively easy; and supported by the results as the low cognitive loading results which indicates the participants found the PiP process relatively simple. In addition, since the system operations are determined by MAps which are created under the direction of end users, we argue that PiP provides more operational transparency and control, thereby addressing the ethical and human value concerns of those who worry about "who, what and when" access to systems in there environments.

For our future work we hope to conduct further work on knowledge representation at the MAp level. For MAps to be portable across environments it is essential that there is a generic way of describing the capabilities of collectives of devices and services. For this we propose to look at ontology's such as OWL-s and dComp[4] [7].

Acknowledgement

We are pleased to acknowledge financial support from the UK DTI Next Wave Technologies and Markets programme and the University of Essex. We also wish to record our thanks to our colleagues Martin Colley, Hani Hagras, Michael Gardner and Malcolm Lear for their strong support.

[4] dComp at : http://iieg.essex.ac.uk/dcomp/ont/dev/2004/09/

References

1. Ballagas, R. et al.: iStuff: A Physical User Interface Toolkit for Ubiquitous Computing Environments, Proceedings of ACM Conference on Human Factors in Computing Systems (CHI 2003). ACM Press, New York (2003) 537-544.
2. Barkhuus, L., Vallgårda, A: Smart Home in Your Pocket, Adjunct Proceedings of UbiComp 2003 (2003) 165-166
3. Becker, C. et al BASE: A Micro-broker-based Middleware for Pervasive Computing, Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom03), Fort Worth, USA 2003.
4. Beckmann, C., Dey, A. SiteView: Tangibly Programming Active Environments with Predictive Visualization, adjunct Proceedings of UbiComp 2003 (2003) 167-168
5. Callaghan V et al, Programming iSpaces: A Tale of Two Paradigms, in book Intelligent Spaces, The Application of Pervasive ICT part of the series Computer Communications and Networks, Steventon, A; Wright, S (Eds.) approx. 455 p. 162 illus. ISBN: 1-84628-002-8, Dec 2005.
6. Chin JSY et al. Pervasive Computing and Urban Development: Issues for the Individual and Society, UN Second World Urban International Conference on The Role of Cities in an Information Age, Barcelona, Spain, 13-17 September, 2004.
7. J, Chin et al, Virtual Appliances for Pervasive Computing: A Deconstructionist, Ontology based, programming-By-Example Approach, The IEE IE05, Colchester, UK, 28-29 June 2005.
8. D. J. Cook, S. Das, MavHome: Work in Progress *IEEE Pervasive Computing*, 2004.
9. Cypher A et al, Watch What I Do: Programming by Demonstration The MIT Press, Cambridge, Massachusetts, London, England 1993
10. Drossos N et al A Conceptual Model and the Supporting Middleware for Composing Ubiquitous Computing Applications, The IEE International Workshop on Intelligent Environments, University of Essex, Colchester, UK, 28-29 June 2005
11. Edwards, W.K. et al: Challenge: Recombinant Computing and the Speakeasy Approach. In: Proceedings of the Eighth Annual International Conference on Mobile Computing and networking (MobiCom 2002). ACM Press, New York (2002) 279-286
12. Gajos K. et al, End User Empowerment in Human Centred Pervasive Computing, Pervasive 2002, Zurich, Switzerland, 2002.
13. Garlan, D. et al; Project Aura: Toward Distraction-Free Pervasive Computing, IEEE Pervasive Computing Magazine, April-June 2002.
14. Guibert N. Girard P., Teaching and Learning Programming with a Programming by Example System, Int'l Symp on End User Development, Sankt Augustin (Bonn), Germany, 2003
15. Grimm, R. et al, Programming for Pervasive Computing Environment, Proceedings of 18th ACM, Symposium on Operating System Principles, Canada, Oct., 2001
16. Hague, R., et al: Towards Ubiquitous End-user Programming. In: Adjunct Proceedings of UbiComp 2003 (2003) 169-170
17. Humble, J. et al Playing with the Bits, User-Configuration of Ubiquitous Domestic Environments, Proceedings of UbiComp 2003, Springer-Verlag, Berlin Heidelberg New York (2003), pp 256-263
18. P R Limb et al, User interaction in a shared information space – a pervasive environment for the home, Perspectives in Pervasive Computing, 25th October 2005 – IEE, Savoy Place, London

19. Masuoka R et al, Task Computing - the Semantic Web meets Pervasive Computing, 2nd Int'l Semantic Web Conf (ISWC2003), 20-23 Oct 2003, Florida, USA
20. Mozer, M. C. The neural network house: An environment that adapts to its inhabitants In M. Coen (Ed.), Proceedings of the American Association for Artificial Intelligence Spring Symposium on Intelligent Environments (pp. 110-114). Menlo, Park, CA: AAAI Press, 1998
21. Myers B.A., Creating user interfaces using programming by example, visual programming, & constraints, ACM Trans Programming Languages & Systems (TOPLAS), Vol 12, Issue 2 (April 1990), pp 143 – 177
22. Shahi, A., et al: Introducing Personal Operating Spaces for Ubiquitous Computing Environments. Pervasive Mobile Interaction Devices 2005 (PERMID 2005), hosted by 3rd International Conference on Pervasive Computing, Munich 8-13, May, 2005.
23. Smith, D. C., Pygmalion: A Computer Program to Model and Stimulate Creative Thought, Basel, Stuttgart, Birkhauser Verlag. 1977.
24. Sohn, T., Dey, A. K.: iCAP: An Informal Tool for Interactive Prototyping of Context-Aware Applications. In: Extended Abstracts of ACM Conference on Human Factors in Computing Systems (CHI 2003). ACM Press, New York (2003) 974-975
25. Truong, KN. et al, CAMP: A Magnetic Poetry Interface for End-User Programming of Capture Applications for the Home, Proceedings of Ubicomp 2004, pp 143-160.
26. Tsukada, K. and Yasumura, M.: Ubi-Finger: Gesture Input Device for Mobile Use, Proceedings of APCHI 2002, Vol. 1, pp.388-400
27. Wang Z, Garlan D. Task-Driven Computing Technical Report, CMU-CS-00-154, Computer Science, Carnegie Mellon Univ, May 2000
28. Weiser, M. Some Computer Science Issues in Ubiquitous Computing, Communications of the ACM, 36(7), pp75-84.
29. X10 Client. http://x10controller.sourceforge.net/X10Controller/X10Client

Distributed Personal Storage System with Flexible Selection and Replication Mechanism

Tomohiro Inoue and Motonori Nakamura

NTT Corp., Network Innovation Laboratories,
3-9-11 Midori-Cho, Musashino-Shi Tokyo 180-8585, Japan
{inoue, motonori}@ma.onlab.ntt.co.jp

Abstract. Although many users want to access their personal data from anywhere in ubiquitous computing environments, distributed storage systems with data replication have not yet been accepted widely. We point out that one current problem of existing systems is the difficulty of determining which data is important and should be replicated in mobile devices. We propose a new personal storage system that uses RDF for metadata descriptions and for selections of replication candidates. We present a basic design and a prototype implementation of the system and confirm it's feasibility for large amount of accumulated data.

1 Introduction

One of the goals of ubiquitous computing technologies is to give users access to resources and services over the network at any time and from any place. Typical resources that users want to be able to access from anywhere are various kinds of personal data. In many situations in our daily work, we need to refer to personal data, such as email received within the last month or documents written in the last year. Thus, such data should be accessible at any time and from any place.

The above examples are typical usage models of distributed storage systems with data replication such as AFS [1], Coda [2], and OceanStore [3]. Many systems have been proposed and developed in the past decades of distributed computing research. However, distributed storage systems have not yet been accepted by many users who need ubiquitous access to their personal data.

We think that existing distributed storage systems do not have sufficient functionality for users in ubiquitous environments. One important function that we think is missing is an intelligent method of selecting data to be distributed or replicated over the network. In other words, the current bottleneck in using distributed storage is that users cannot easily determine which data should be distributed via networks or replicated on mobile terminals.

Meanwhile, the typical usage model of storage systems seems to have changed in recent years. The exponential increase in the capacity of hard disk drives has meant that many users do not erase data. As a result, users accumulate large amounts of data —all the data they have ever received or created— in their personal storage systems. Furthermore, the amount of electronic data referred

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1090–1101, 2006.

to at one time by a user is increasing rapidly as users perform various kinds of work using computers.

The growths in the amount of data is overwhelming the traditional scheme of data arrangement, which is the hierarchical directory trees of storage systems. Today, many users feel frustrated in spending much time finding their files or e-mail in trees composed of hundreds of subfolders. Recently developed search-based user-interfaces represented by Google Desktop [4], Apple Spotlight [5] and iTunes indicate the end of directory-based storage framework.

We believe that distributed storage systems based on the hierarchal data-arrangement model must be redesigned. Users have trouble digesting and compiling their personal data into the shape of a directory tree. Therefore, distributed storage requires novel functionality to help users designate the data to be replicated. We explain details of the problem in the next section.

To resolve the above problem, we propose a new approach for distributed storage systems based on the use of flexible metadata to determine candidates of data to be replicated. Our system uses a personal storage system with rich metadata, which is similar to the model proposed by the Haystack project [6]; we use RDF (resource description framework) [8] semistructured metadata for rich descriptions of native files stored in local file systems. This lets users easily find their stored data and determine which data is important and should be replicated on their mobile devices. In addition, our prototype system has a two-tier mechanism for fast parsing of RDF metadata to ensure system scalability with respect to the amount of stored data. Details of our system are described in sections 4 and 5. A preliminary test of the performance is presented in section 6.

The following are examples of the type of applications that our system can provide. One is a system that automatically replicates to a user's PDA or a mobile PC all the personal files that might be needed for reference in his or her current work. Another is an automatic diary system that copies and publishes to a personal blog (web log) the key data extracted from everyday activities.

2 Details of the Problem

As briefly mentioned above, the major problem in the use of distributed storage is the users' difficulty in determining data to be replicated. Users cannot easily select and compile data that is relatively important and thus needs to be copied onto their mobile devices. Such a selection task (referred to as *target selection* hereafter) would impose a heavy burden on each user of the distributed storage system.

In existing distributed storage systems, the provided options are somewhat limited. Existing systems use the hierarchical model in the name space of files to be replicated. That was adequate in the past for compatibility with the local file system, but it is not so useful in today's computer environment. Major existing systems provide two types of replication schemes [2]: a user can replicate beforehand the entire subdirectory of a file system tree (subdirectory model), or some data that has recently been used can be replicated in units of a file (cache model).

The cache model cannot easily satisfy users' needs in general cases. Practical computer tasks these days require many kinds of records for reference, e.g., an e-mail thread archive for a project or experimental source code written a year ago. We cannot expect all of them to be cached in a mobile device. Thus, users must designate, for replication in advance, the subdirectory containing any potentially relevant data that needs to be referred to afterwards.

However, generally speaking, users have trouble designating appropriate subdirectories to match their needs. Average users cannot properly arrange their data into a rational directory tree. The traditional directory model is based on the MECE (mutually exclusive, collectively exhaustive) rules, so an object must belong to a unique category. This restriction means that a user must store data using the correct path in the directory tree whenever he or she saves data for the first time. That is rather a burden for average users because they need to deal with a large number of files concurrently, so categories in trees might have hierarchies that have too many layers and are fragmented. Although some exceptions are allowed by using aliases of filenames, this cannot be a general solution to this problem.

Actually, the single-*view* philosophy of the directory model is not suitable for target selection. Hereafter, we regard a *view* as a data arrangement scheme. Although most data can belong to multiple categories and can be arranged in multiple viewpoints, a directory must be predesigned to have a single hierarchy of data arrangement. However, the view designed when a directory was created may not be an appropriate arrangement for a user who later wants to browse data or designate data to be replicated. Moreover, selecting multiple data sets from a single-view directory for different mobile devices by different views is difficult.

Therefore, a target selection framework that helps a user select data with little effort needs a mechanism that is independent of the restrictions of the underlying file system. Users of that framework should be able to select a subset of data from multiple views of the stored data.

3 Related Work

Many researchers have worked to enhance data accessibility in distributed storage systems. For example, the concept of cooperative caching among mobile devices has been used to increase the availability of data in mobile environments. Some systems have been developed by extending the Coda file system [10][11]. However, most of these studies assume the cache model, explained in the previous section, for data replication, rather than the proactive replication model used in our system.

Our motivation in this paper is to improve personal data processing in ubiquitous environments. We consider that proactive replication is a better solution for managing personal data provided that the storage system has a framework that enables users to easily designate the priority of data replication.

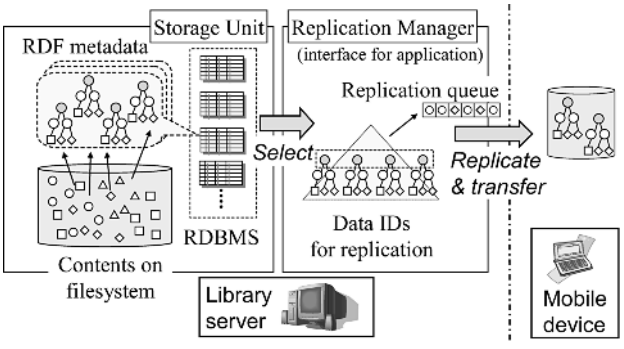


Fig. 1. Basic design of proposed system

Such a framework needs a rich metadata layer for the annotation of stored content. Haystack is a fine predecessor of personal storage systems. Major targets of the Haystack project are to build a new user interface and browser for personal data annotated with RDF metadata [6] and to create an environment for authoring metadata and applications [7]. We adopted the storage model of Haystack, added a little extension to the RDF framework, and proposed a new distributed personal storage system that accepts queries for proactive data replication.

Our system will be advantageous when the storage unit exhaustively accumulates personal data about daily activities and the data items are connected to each other by rich annotations. MyLifeBits [12] is a project for such accumulation of digital data gathered from the daily activities of users. They are developing "story-telling" methods to extract key data from personal data gathered exhaustively. Such methods can be applied to create an application's view in our personal storage system.

4 Approach and Design

This section describes our approach to resolve the problems explained in section 2. We also describe a basic design for a distributed personal storage system with an intelligent target selection mechanism.

4.1 Overview

We propose a distributed personal storage system called a *Library Server* (LS). A subset of data on an LS that is relatively important will be replicated on mobile devices. As shown in Fig. 1, the LS system consists of two units: a *Storage unit* and a *Replication manager*. The storage unit holds accumulated content data (files) and has a metadata overlay layer that describes annotations of the accumulated content. The replication manager processes the tasks of target selection and replication. Details are described in section 5.

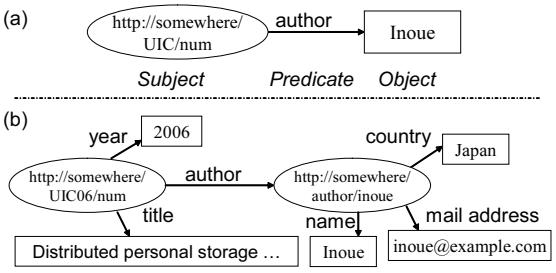


Fig. 2. Metadata description in RDF framework

We made two decisions about the basic design of our LS system. First, we chose to use the RDF metadata framework for our metadata layer. Second, we assumed that users or applications that want to select candidates for replication designate rough queries for target selection. These issues are described in the following sections.

4.2 RDF Metadata

RDF is a flexible framework for describing metadata using a semistructured data model. Our basic design using RDF for personal storage follows the idea introduced and prototyped in the Haystack project, which is constructing a personal storage system with rich annotation to help users handle their data in daily computer activities. They claimed that the benefits of using RDF are its ability to build a unified flexible metadata layer due to its high interoperability with applications having different data schemas and ontologies [6].

Structure of RDF metadata. Metadata descriptions in the RDF framework are based on directed labeled graphs [9]. Each RDF statement consists of the combination of a *subject*, an *object*, and a *predicate* arc between them. The simplest example of an RDF statement is shown in Fig. 2(a). This statement means that a resource identified by the ID 'http://somewhere/UIC06/num' has an author who is 'Inoue'.

Though each statement describes a minimal relationship between a subject and an object, the RDF framework can describe much more complex and rich annotations of content. As shown in Fig. 2(b), a resource identified by 'http://somewhere/UIC06/num' can have other properties such as "title" and "publication year". Furthermore, its author can be a complex resource with properties of 'Inoue' (name) and 'Japan' (country). In this way, complicated metadata is described as graphs consisting of a large number of RDF statements.

In the LS system, an RDF statement is used to describe a relationship between two items of stored content. Applications that create content data in LS annotate it using RDF graph structures. The content that is interpreted to have relationships is linked by RDF arcs. For example, a set of personal content

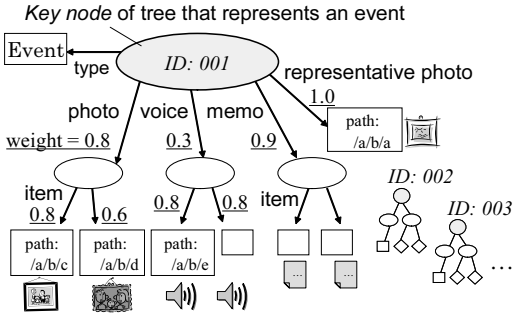


Fig. 3. Example view of proposed storage system

including files in various media taken at an "event" such as a party is structured, as shown in Fig. 3. Here, a leaf node's value means a reference (in practice, a file system path) to a content file.

Content in the data store will be structured as multiple trees in typical cases. Although these tree forms are similar to a directory model, they differ in that any content can be linked from multiple nodes, namely, any file or directory can have two or more parents. Moreover, an RDF graph does not need to have a root in its link structure.

While a traditional directory is normally structured to give a single and absolute view of a file system, a set of RDF-linked trees is created to describe the relative relationships between content using each aspect (view) of the user application. Thus, some applications might create different sets of RDF trees with different goals for the same content. Hereafter, in this paper, a situation is described in which an application arranges content in units of event trees, as shown in Fig. 3, but the arrangement units are not limited to any specific units.

Extension to RDF metadata. We made a simple extension to the standard RDF framework. Our extended RDF adds the attribute of *link weight* to each arc in RDF statements. A link weight, which is a number in [0-1], represents the strength of a relationship described by an RDF arc, where 1 means the strongest relationship. This extension enables applications and a replication manager to interpret and compare the relative importance of content linked from another node. These weights affect the target selection mechanism described in section 5.

4.3 Queries for Target Selection

To support target selection independent from a file system's view, we adopt a query model to select and designate candidates of data that a user regards as important. A query roughly designates a subset of data for replication. For example, a user who needs data about past important events to be copied onto his mobile device makes a query to select a list of IDs of event resources in RDF graphs. While the sub-directory model for target selection can only designate a data candidate within a predetermined view of an underlying file system, a

query model can use any views constructed by RDF graphs and freely designate data candidates for target selection.

Although there is no standard interface for querying RDF data yet, W3C is working to standardize a protocol called SPARQL (SPARQL query language for RDF) [13], which is a SQL-like language that selects various kinds of data from RDF data graphs. We did not use SPARQL for the query interface of the current version of our design because of performance concerns. We feel that SPARQL currently may exhibit insufficient performance, especially scalability with respect to accumulated data volume.

The current design simply uses SQL for the query interface, as described in section 5. Although SQL may not be able to represent very complex queries that correspond to functions of RDF, it can express simpler queries, such as queries about events in the past three years and projects currently in progress. For our system, we took an alternative approach to fulfill complex queries for target selection, as we describe in the next section.

5 Implementation

We have developed a prototype of LS. It is mainly implemented in Java and works on a Windows XP computer. We used a native file system as the data store of the storage unit and used PostgreSQL 8.0 for the metadata (RDF) store.

The main goal of this prototype implementation is to confirm the feasibility of the proposed approach with a large amount of accumulated data. The novel flexible mechanism for target selection must have sufficient scalability with respect to the volume of content. While traditional target selection methods such as a subdirectory model have obvious scalability, a flexible mechanism may not be scalable because it must parse a lot of metadata of accumulated content. To resolve this problem, our prototype system implements a two-tier algorithm for target selection.

5.1 System Components

Storage unit. As shown in Fig. 1, native content data in LS is stored as files in a file system. On the other hand, RDF statements in LS are stored on a local RDBMS. Some important metadata among all RDF descriptions, e.g., "event" IDs, will be indexed in an RDB table to accelerate the performance of applications that frequently access the important metadata. This indexed metadata is to be referred to by queries from applications.

Replication manager. A replication manager (referred to as manager hereafter) accepts SQL queries for target selection by users or applications and extracts required subsets of content from a storage unit that holds a huge volume of content data.

Reference IDs (paths) of selected data are put into a queue called the replication queue. When the manager detects a connection to a mobile device, the

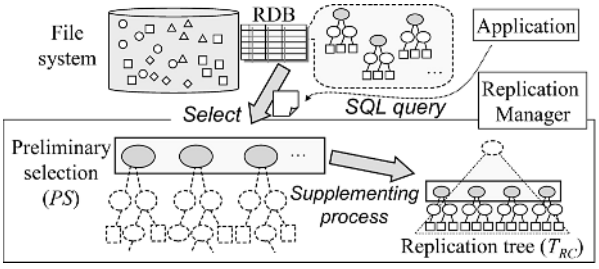


Fig. 4. Internal process of target selection

replication queue is read from the manager's replication subsystem, and each content item pointed to by the ID in the queue is replicated in the device in the same order as that of the queuing order.

5.2 Details of Target Selection

The internal data flow for the task of target selection is shown in Fig. 4. The manager maintains two sets of internal data to process target selections and works according to a two-tier model to enhance the scalability of the system.

First, when a query is received, the manager sends the query to the RDBMS and retrieves a data set from it. The retrieved data set has the form of a list of RDF nodes. We call this list the *preliminary selection* (PS). In typical cases, PS does not include individual small content, but key nodes that represent data units of an event in an RDF tree for example.

Next, the manager supplements the PS and expands it into tree-form data called a *replication tree*. The manager reads each item in the PS list, searches for RDF arcs from the item, and connects found content to the item. The resulting data takes the form of a tree having a pseudo root node. In this expansion process, the link weight of each RDF arc is considered to judge the priority of replication. Content data linked with a high weight means that the data has a high relevance to the item in the PS list and has high priority of replication.

Extraction of Preliminary Selection. When a manager receives a new or modified query from applications, it interprets the query and makes and sends a corresponding query to the RDBMS. The retrieved result is converted into a PS list. Each data item in PS has an attribute of *weight*, which is similar to the link weight of an RDF arc and is a value between 0 and 1 representing the replication priority.

The size granularity of each data item listed in PS depends on the query processed, tables in the RDB, and RDF trees created by applications. However, a query will be a operation that selects key data nodes from a large amount of accumulated data. Thus, in typical cases, the PS lists nodes of important data units arranged by a view. For example, nodes of ID 001, 002, and 003 in Fig. 3 would be listed in a PS. In this way, the manager extracts relatively important

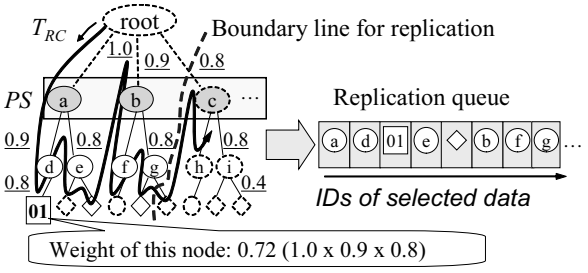


Fig. 5. Details of extraction of replication tree

RDF nodes from a huge amount of accumulated data in an LS and maintains an internal PS list of skimmed data.

Process of supplementing related data. Next, the manager expands the PS list into a replication tree by supplementing each listed item with data related to it. The manager judges relevance by tracking links (RDF arcs) of an RDF subtree whose root is the item. The manager repeats this for each item in a PS, and a PS is expanded into a tree form with a pseudo root, as shown in Fig. 4. Hereafter, we call the expanded replication tree T_{RC}. Data in T_{RC} becomes a candidate for replication on mobile devices.

However, RDF trees created by applications may not yield a strict tree model because any node can be linked from multiple parent nodes within the RDF framework. Thus, the link-tracking extraction will operate in infinite loops when processing general graphs. However, our system resolves this inifinite loop problem by regarding any graphs as trees using a depth-first algorithm with a link-weight cut-off technique.

Specifically, the manager scans a PS list from the start and tracks data linked to each item in the list with a depth-first search, as shown in Fig. 5. The manager sums the data size of tracked content and repeats the tracking until the summation of tracked data reaches the preliminary allocated capacity of a mobile device. In practice, the manager expands T_{RC} redundantly until about twice the allocated capacity is reached and sets a boundary for replication candidates.

Furthermore, the manager considers link weights of RDF arcs for discarding replication candidates during this process. The weight of a node in the replication tree is calculated as the product of each link weight on the path from the root to the node. For example, the weight of node 01 in Fig. 5 is calculated to be 0.72 (composed of $1.0 \times 0.9 \times 0.8$). If a node's weight is less than a certain threshold, the manager does not regard the node's content as a replication candidate and aborts the depth-first tracking of the following links. This termination of subtree expansion avoids loop tracking of the T_{RC} graph.

At the end of these processes, important content designated by a rough query and content highly related to queried content are selected as replication targets. Within the processes, a replication queue includes the IDs of selected content in the order of the replication priority determined by the above procedure.

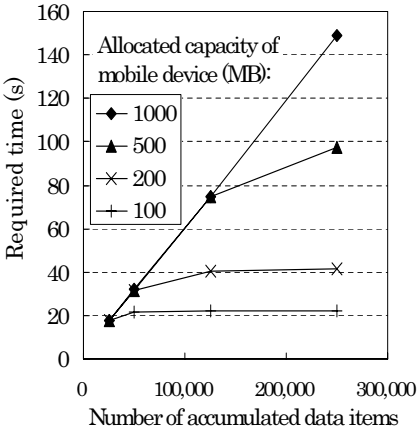


Fig. 6. Time required for initial startup

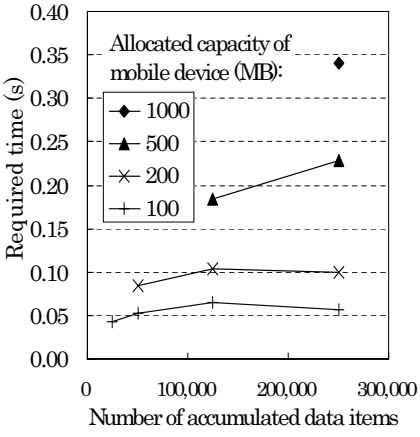


Fig. 7. Time required to maintain replication tree

6 Experiments

Results of simple experiments performed on our prototype system are shown in Fig. 6 and 7. In the experiments, the replication manager ran on a JDK 1.4.2 server VM. All processes of the LS ran on a PC with a 3.4-GHz Pentium 4 (HT enabled) with 2 GB of memory running under Windows XP. The LS accumulated up to 10,000 sets of event data. Each event consisted of 25 content items (twenty image files and five text-memo data sets) with a size of 1.6 MB on average. Each event was annotated by a three-tier RDF tree.

First, we measured the cold start time of the process of target selection. The measured times to prepare target selection, specifically, the times from the start of PS creation to the end of the replication tree expansion are shown in Fig. 6. As the plots indicate, increases in the preparation times are within linear growth with the size of accumulated content in the LS system. Actually, the preparation time depends on the size of T_{RC}, which is determined by the capacity allocated to a mobile device. Thus, the increase in the time saturates when the size of accumulated content exceeds a certain value corresponding to the allocated capacity.

Second, we observed the behavior in the steady state of the system operation. We measured the costs of maintaining internal data of the replication manager when accumulated content in the storage system was changed. Times from the detection of a change in the link weight in the RDB table to the end of replication tree modification are plotted in Fig. 7; these changes of link weights involved the replacement of some content in the replication candidates. The measured times were very short and approximately constant if each of T_{RC} were equal in scale. Therefore the proposed algorithm was confirmed to work with sufficient scalability.

These experimental results indicate that our prototype LS is feasible for accumulating up to about 250,000 items of content (10,000 events).

7 Discussion

We have pointed out that the current bottleneck in distributed data usage lies in the difficulty of selecting which data is important and should be replicated on mobile devices. In section 2, we have indicated the limitation of the single-view hierarchy model, and a target selection mechanism that cooperates with multiple views of accumulated content is needed for the convenience of users.

Our system uses a metadata model based on the RDF framework, which allows content data to be linked from one or more RDF nodes. Each application that works on our system annotates and arranges accumulated data in its own views. Thus, multiple views of metadata can be built onto native content thanks to the flexible architecture of the RDF framework.

Our system uses RDF link trees to extract relationships between content and uses the relationships to decide candidates for replication. Therefore, the system selects a data set defined by arbitrary views used by applications, independently of the file system's hierarchy.

In addition to multiple-view support, our system has the following features. It replicates data in the order of its importance because a manager transfers data in the queuing order in the replication queue. It also replicates data in accordance with the relevance of the data. Sets of data provided by applications, such as those of events shown in Fig. 3, are replicated in their clusters. These features are practical in the case where the network connection between an LS and a mobile device is often disconnected.

Furthermore, the flexible mechanism that uses user metadata for target selection has performance concerns. In particular, scalability with respect to the volume of accumulated content is a concern because the mechanism must parse a lot of metadata. Our system features a scalable algorithm for target selection. Our implementation uses a two-tier algorithm to avoid parsing a huge amount of RDF data. A manager first reduces the number of data candidates into a preliminary selection (PS) by a query to RDBMS and subsequently expands the PS into an enriched data set (T_{RC}) by tracking RDF links. A T_{RC} has a relatively small data size compared with the large amount of accumulated content in an LS, so the cost of maintaining a T_{RC} is rather small.

Our current algorithm may not provide the ideal accuracy for target selection. Although recall values of query results can be improved by modifying queries, precision values may remain rather small; results may contain much data which users do not want to replicate. Though we think small precision values will be acceptable to some extent compared to the average capacity of current mobile devices' storage, better algorithms, instead of the simple multiplication of link weight, may be appropriate for supplementing related data.

8 Conclusion

The main contribution of this paper is the proposal and feasibility confirmation of a new distributed storage system that provides the missing functionality of existing systems by using the RDF framework. We explained that one current problem of existing distributed storage systems is the difficulty of determining which data is important and should be replicated in devices.

To resolve this and other problems, we proposed a new personal storage system that uses RDF for metadata descriptions and presented the basic design of the system. Our prototype implementation has a two-tier mechanism to avoid parsing a huge amount of RDF metadata. We confirmed the feasibility of the prototype system for 250,000 accumulated data items.

References

1. Satyanarayanan, M.: Scalable, Secure and Highly Available Distributed File Access. IEEE Comput., **23** no. 5(1990) 9–21
2. Kistler, J., Satyanarayanan, M.: Disconnected Operation in the Coda File System. ACM Trans. on Comput. Systems, **10** no. 1 (1992) 3–25
3. Rhea, S., Wells, C., Eaton, P., Geels, D., Zhao, B., Weatherspoon, H., Kubiatowicz, J.: Maintenance Free Global Storage. IEEE Internet Computing, **5** Issue 5 (2001) 40–49
4. Google Desktop, http://desktop.google.com/about.html
5. Apple Spotlight, http://www.apple.com/macosx/features/spotlight/
6. Huynh, D., Karger, D.R., Quan, D.: Haystack: A Platform for Creating, Organizing and Visualizing Information Using RDF. In: Proceedings of Semantic Web Workshop, The Eleventh World Wide Web Conf. 2002 (WWW2002), 2002
7. Quan, D., Huynh, D., Karger, D.R.:Haystack: A Platform for Authoring End User Semantic Web Applications. In: Proceedings of 2nd Int. Semantic Web Conf. (ISWC), 2003
8. Resource Description Framework, http://www.w3.org/RDF/
9. Resource Description Framework (RDF): Concepts and Abstract Syntax, http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/
10. Inamura, H.: Extending the Coda File System to Handle Cache Misses on Isolated Clients. In: Proceedings of Workshop on Advances in Parallel and Distributed Systems, (1998) 336–340
11. Yasuda, K.: Cache Cooperation for Clustered Disconnected Computers. In: Proceedings of Int. Conf. on Parallel and Distributed Systems (ICPADS), 2002
12. Gemmell, J., Bell, G., Lueder, R., Drucker, S., Wong, C.: MyLifeBits: Fulfilling the Memex Vision. ACM Multimedia, (2002) 235–238
13. SPARQL Query Language for RDF, http://www.w3.org/TR/2006/WD-rdf-sparql-query-20060220/

Object Oriented vs. Agent-Based Oriented Ubiquitous Intelligent Mobile Managed e-Learning Environment

Elaine McGovern, Rem Collier, and Eleni Mangina

School of Computer Science & Informatics
College of Engineering Mathematical and Physical Sciences,
University College Dublin, Belfield, Dublin 4, Ireland
elaine.a.mcgovern@ucd.ie, rem.collier@ucd.ie,
eleni.mangina@ucd.ie

Abstract. The emergence of information technology the last decade is leading to changes in the way people work and learn. Computer networks and especially ubiquitous intelligent mobile learning environments offer new strategies the dissemination of course work. Following the current trend towards ubiquitous computers, networks, information and services, a new innovative mobile e-learning intelligent world is created, where software components embedded with intelligence assist the users in education and provide functionalities such as context awareness, automation, adaptivity and proactiveness. Within this paper a case study of the similarity and differences between an object oriented solution and an agent-based solution of ubiquitous intelligent mobile learning environment is discussed. The results of this research are presented in terms of preliminary user tests for the design of the Mini-ME Interface and performance testing for both implemented architectures.

1 Introduction

Managed learning environments (MLE's) have been designed with the aim of assisting academic intuitions in the management of online and distance learning courses for their student population [1]. This software solution provides an array of services including access control, syllabus provision, communication tools and a virtual administrator tool for faculty management. A virtual learning environment is a subset of the managed learning environment that enables the provision of online interactions. Such interactions include online lectures and tutorials as well as online student-teacher discussions. Students, depending on the software used, can also avail of the same facilities for use in discussions with their peers. It presents to the students, through a single, consistent, and intuitive interface, all the components required for a course of education or training.

The managed learning environment has proven itself desirable for economizing on the time of teaching and administrative staff. The extent of the economy over traditional "talk-and-chalk" teaching is not yet clear, but using an MLE almost certainly absorbs less instructor time (and requires less expertise, while producing a more professional result) than creating a home-grown website for a course. [2] For the purpose of this research paper two MLE's were studied, one commercial and one open source package were studied: Blackboard [3] and Moodle [4].

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1102 – 1113, 2006.

A wireless device can support the functionality of the MLEs and the Mini-ME application described in this paper has been designed to run on a handheld personal digital assistant. Students can use them as a graphing calculator, word processor, database, test prep tool, and reference resource. Preliminary studies, such as Multimedia Portables for Teacher's Pilot, have reported high levels of motivation and self-reliance among teachers who consider PDA's to be flexible and adaptable in providing a context for teacher professionalism [5]. The devices gave students "opportunities to connect questions and investigations to the data in a real time setting that enhances "systematic investigations, critical thinking and cooperation" [6]. Additional research suggests that PDA's facilitate group work, the immediate analysis of data particularly during laboratory exercises or when conducting scientific investigations in the field rather than in the classroom [7]. Collaboration and sharing of information and software is enhanced by PDA's as well.

Mini-ME, Mini Managed eLearning, delivers an enhanced learning experience in a mobile context. The virtues of ubiquitous interactive education are highlighted. The difficulties associated with effectively presenting and alerting students to personalized course material are discussed. A suitable MAS that emphasizes the human computer interactions via an intelligent assistant is utilized to overcome these issues. In order to highlight its effectiveness, a comparison is made with a similar object oriented application. Mini-ME has the ability to learn a user's preferences and working habits. This provides the user with a proactive and reactive assistant, thus enhancing their academic career. The remainder of this paper is structured as follows: Section 2 provides a brief description of the background work in the area of intelligent multi-agent systems in managed learning environments. Section 3 presents the case study of the object oriented and agent-based oriented paradigm of Mini-ME. Section 4 presents the results of the system. Finally, the conclusions and future developments are discussed in Section 5.

2 Background

There has been significant research carried out with regards to the attachment of intelligent interface agents to existing applications. Lieberman discusses the "marionette string" concept. The agent is given a set of "strings" corresponding one to one with user actions in the interface. The agent then "tugs" on the strings to make the external program perform [8]. Alan Kay [9] also highlights how utilization of intelligent user interfaces has resulted in a transition from the traditional direct manipulation of a system to indirect human-intelligent agent interactions. Intelligent agents enable the delegation of routine tasks that become tiresome when preformed on a daily basis. Kozierk and Maes [13] described a semi-intelligent agent that could undertake meeting scheduling tasks via Memory-Based Learning and Reinforcement Learning.

Today, many direct manipulation interfaces are centered around an interpreter called an "event loop" that accepts mouse and keyboard input, and tries to determine what functionality the input is requesting, changes the application data structures and updates the screen display accordingly. Mini-ME adheres to these concepts; it is more feasible to get different applications to agree on a standard format for input than to

specify invocations for each underlying function. This is due to a user's tendency not to remain at a single activity if it is not fully intuitive [11]. The more standardized the formats are, the more likely the user is to stay for the advantages. As far as the agent is concerned it is important that foreign objects are created in a lazy or on demand manner. An interface agent should not be required to retrieve or maintain objects that it is not required or expected to use. This makes the interface as efficient as possible.

The Baghera project [12] is a multi-agent architecture for human learning designed using computer modeling and conceptualization of learning environments. This multi-agent system is designed for teaching universally valid reference knowledge through collaboration between human and artificial agents. It considers the students knowledge base to be diverse, each student in a class having their own academic strengths and weaknesses. It streams the students learning based on an understanding that their knowledge of what resource is relevant is based on each item taken in context. The Baghera MLE provides individualized support for problem solving developed using JatLite. Each agent was extended by an interaction module, as this provided support for managing protocols among agents.

Multi-agent system (MAS) managed learning environments (MLE) along with other MAS currently under development, allow users to delegate tedious tasks that the average person would rather not perform, including: meeting scheduling, email filtering and reminder notification. MAS should have the overall effect of reducing the users workload. [8] They can be classified according to the role that they perform, technology they use and the domain that they inhabit. In order to develop a working Mini-ME application, it was necessary that an understanding be obtained of how a student interacts with a generic multi-agent system. An understanding of object oriented design methodologies and multi-agent system methodologies was required before design and implementation could be undertaken. Two MLE designs, commercial and open source, used in academic institutions were researched and assessed: Moodle and Blackboard. J9 and ActiveSync were researched, as these were the connection methods used for the object-oriented version of Mini-ME. Agent Factory and Agent Factory Lite [14] were studied as these were used in the development of the Mini-ME multi-agent architecture. Machine learning methods, such as memory based learning and version space algebra, for extracting user goals and intentions through observation were studied. Rules for extracting context were developed and contingencies for adapting to a user's changing objectives and how to reduce the initial training time were required.

In the development of the Agent-Based Oriented solution, Agent Factory toolkit has been employed. One of its primary concerns was the creation of a "cohesive framework that supports a structured approach to the development and deployment of agent oriented-applications" [13] for the creation of Belief-Desire-Intention agents that communicate via the Agent Communication Language.

3 Case Study – Mini Mobile E-learning (Mini-ME)

The Mini-ME application was designed to run on a HP iPAQ [14]. It contains a 400MHz PXA255 XScale processor with a 200MHz system bus. It runs on Microsoft's Windows Mobile 2003 for Pocket PC operating system. A study of the similarity and differences between the object oriented solution and an agent-based

solution of ubiquitous intelligent mobile learning environments is described in this section. In order to effectively compare the two, it was necessary to design an ubiquitous mobile managed learning environment for a handheld device that was capable of functioning to a standard level on both system architectures.

3.1 The Object Oriented Solution for Mini-ME

3.1.1 The Object Oriented Client-Server Architecture

Client-server network application architecture in Figure 1, describes the paradigm that separates the client from the server. The client-server application works because a client is created via a socket connection and a server is created that listens for socket connections. In order for the Mini-ME client and the server to communicate with each other, the client is required to have a host to connect to. The client and server communicate on the selected TCP/IP port. TCP/IP uses ports because it assumes that the servers will be doing more than one networked function at a time and ports provide a method to manage this. In the Mini-ME client-server model, a socket connection is made. A string buffer is used for reading the input stream, for example the login string. It is received from the interface and is modified so that it can be sent to the server via an output stream. The string passed to the server is based on the current operation being performed. The server receives the input stream and performs the actions required based on the current parameters.

The design of the Mini-ME client-server application is based on the generic client-server model, the user "updates" or "requests" information from the Mini-ME application backend. The client resides on the HP iPAQ. When the user performs an action, the event handler is notified and a method is preformed in the client logical layer. Based on the method called, a request is sent to the server. The servlet interprets the information that is received from the client and then calls a java method that can either enter data or retrieve information from the Moodle database, via a JDBC connector. The server sends a response to the client based on the results obtained and waits for the next message to be received from the client. Depending on the type of message the client received, an appropriate action is preformed: display login screen; welcome screen; inbox and messages; assignments screen or quiz screen. If the logged-in user is a tutor then Mini-ME application can also display a tutor's page.

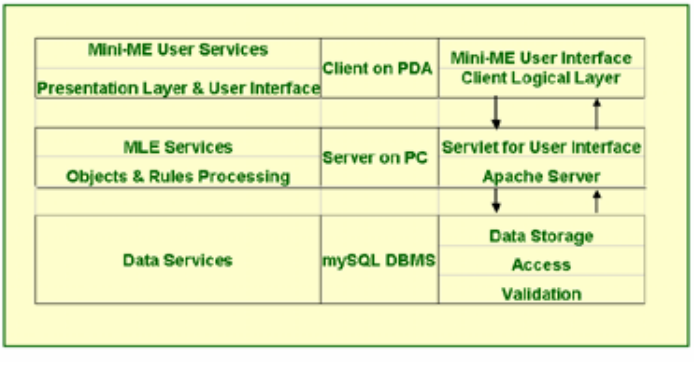


Fig. 1. Mini-ME Client Server Architecture

3.1.2 Object Oriented Mini-ME Class Diagram

The Mini-ME UML [15] class diagram displayed in *Figure* 2 is used the show the static structure of the applications client-side object-oriented model. It depicts each class, their internal structure and the relationships in which they participate. It does this by showing each classes associations.

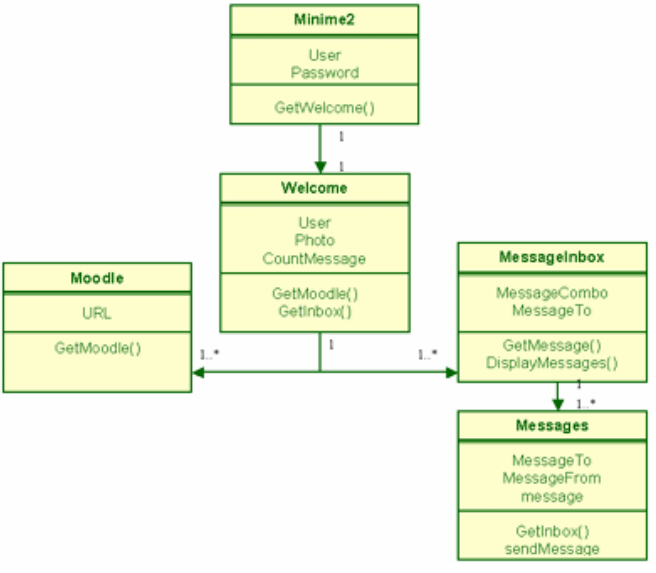


Fig. 2. Mini-ME UML Class Diagram for the Object Oriented Application

3.2 The Agent-Based Oriented Solution for Mini-ME

From the implementation viewpoint, the Mini-ME application has been developed according to FIPA specifications [18] by utilizing the Agent Factory [13] toolkit, using Java as the programming language. In particular, the Interface Agent runs on a HP iPAQ. The high-level communication protocols have been implemented using ACL messages, whose content refers to the Mini-ME ontology. The GAIA methodology was used to identify the agent structures, roles and interactions within the Mini-ME [16] MAS system.

The MAS Mini-ME application utilizes a FIPA compliant architecture, displayed in Figure 3, to fulfil the task of an intelligent application capable of autonomous human computer interaction for communication, event monitoring and the performance of menial tasks. The Mini-ME multi-agent system consists of a community of three agent types: wrapper, mediator and interface. They co-operate in order to communicate the MLE's messages.

The wrapper agent allows the other agents in the Mini-ME application to connect to the external software system, for example the system database. The role of the wrapper agent is to provide a single generic way for the other agents to interact with the external software system. These external systems may not be agent based. The wrapper agent, therefore, need be the only agent in the community with the capabilities to interface with the external resource.

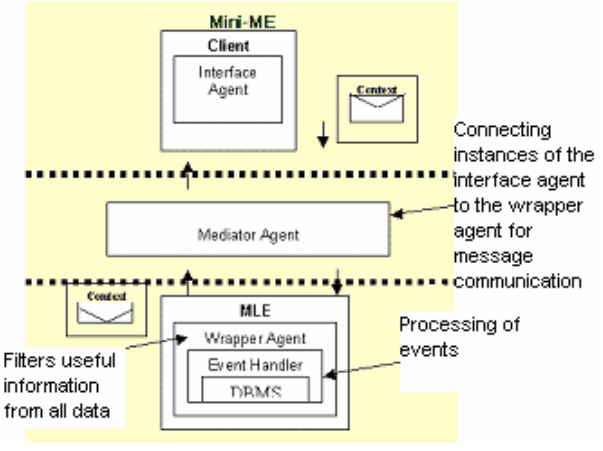


Fig. 3. Mini-ME Application Architecture

The wrapper agent has the ability to examine incoming messages. It filters these based on their content and rejects requests where applicable. The wrapper agent unbundles agent messages into multiple shorter messages. It can then match action requests with the functionality of the legacy system. This has proven to be practical, as there is often a mismatch with the ACL formats and the format of commands understood by Moodle and Blackboard. A wrapper that does not bundle agent messages allows for a simple more robust agent/application software communication bridge. This unbundling, however, leads to an increased message load. In order for messages to be received in a fair manner several message conveyance forms were considered.

The Mini-ME wrapper agent utilizes a scheduling mechanism based on a weighted priority allocation algorithm. Each message received is given a time code along with its priority rating. As the length of time a message is required to wait increases, so does its priority rating.

Mini-ME, aims to aid students at the first signs of trouble. The wrapper agent trawls through activity logs in order to glean an understanding of what students are participating at an adequate level consequentially deriving those students that are not. Once a student has been identified, the wrapper agent attempts contact. If that student is currently online, an alert notifies the student that a new message is awaiting them. If the student is not currently online then the wrapper agent undertakes the task of storing the message in the appropriate table so that when the student does enter the MLE, he/she will be alerted to the message. In effect, it is the task of the wrapper agent to maintain the MLE external software so that the Mini-ME application can provide the student with current information when logged in.

In summation, the role of the wrapper agent is extensible. It often takes on the role of monitor and event manager. It ensures that all communications are handled via its interface. It declines inappropriate or malicious data. Finally, it acts as an event manager and handler whenever an event is triggered.

The mediator in the Mini-ME environment assigns a provider for the service requested to the consumer requesting the information. The negotiation for service and

the actual service provided can be viewed as two distinct phases. The family of mediator agents can be subdivided into three basic forms; the matchmaker, the broker and the mediator. [9] These three agent pattern types can coexist as independent agent types or be concatenated in a single agent type. Universally, they act as intermediaries between a number of other agent types providing access to services and communication channels via the specified agent communication language. Utilization of a mediator agent works best when the interactions between agents are well defined. In order to coordinate interactions, the mediator need only know the valid sequence of events. One performance issue that has been discussed and dealt with is that of preventing the mediator agent becoming a bottleneck or a point of failure.

The Mini-ME application utilizes the HP IPAQ and is implemented in java. The user interface was implemented using AWT. Several things had to be considered in order that an intelligent interactive interface could be devised. Knowing the user was fundamental, and this involved studying his/her habits and being aware of certain preferences and work practices. For an assistant to correctly aid the student, it must know how the student tends to work. For example, if the student only works in the evenings, it can send a message reminder regarding an assignment due the evening it would normally send to a student that works during the day. Conversely, if the assistant is aware that the student is attending an online e-tutorial, no messages should be sent to the student until after the duration of the tutorial. The interface agent is concerned with interacting with the user of the system and the system itself. The use of personalization provides appropriate services, tailor-made to the students needs. This improves user interaction and usage.

4 Results

4. 1 Preliminary User Tests for the Design of the Mini-ME Interface

The purpose of the Mini-ME user interface evaluation was to ascertain the efficiency and effectiveness of the two potential interface designs on the hand-held device at the applications infancy. This study required a mixed corpus of twenty-one users to carry out a series of three basic tasks after no direction was given: logging in, viewing their messages and accessing the MLE on the separate user interface designs containing similar information. They were subsequently required to answer a brief questionnaire.

The first user interface contained an initial login screen followed by a personalized welcome screen, which directed the user to the external MLE home page and the MLE resources window. The second contained an initial login screen followed by a personalized welcome screen that contained a menu bar that directed the students to the MLE resource page or the Mini-ME messages window. The following is a small subset of the detailed pilot case study results taken [16]: 85.7% of those surveyed considered the applications structure and navigation functions conducive to revision at times and locations not previously considered. These findings reflect Staudt's [6], who noted that similar applications gave students "opportunities to connect questions and investigations to data in a real time setting" and enhances "systematic investigations, critical thinking and cooperation". 85% of the users believed that

Mini-ME would be useful in a traditional lecture; and also would be useful in the participation of course work when circumstances would otherwise prevent it, such as, in cases where students were isolated from the traditional classroom. Monahan [17] used this premise in the development of CLEV-R, a system that also attempted to shatter feelings of isolation. Due to unfamiliarity, only 57% of the users made specific reference to the benefits of the personalization facilities referring to the ranked message and resource notification or the intelligent reminder facilities.

4.2 Performance Testing

4.2.1 Component Interactions in the OOP and MAS Mini-ME Applications

Modest performance tests were applied to the MAS Mini-ME and the OO Mini-ME architectures. Using comparative time codes based on commencement and completion times of specific tasks. Overall, the object oriented Mini-ME model performed each task faster than the MAS model, as is evident from the chart in *Figure 4*. This was due to the inclusion of a mediator agent. The mediator agent was used in the MAS to act as a broker between the interface agent and the wrapper agent in order to further decouple the agent roles described in the design section.

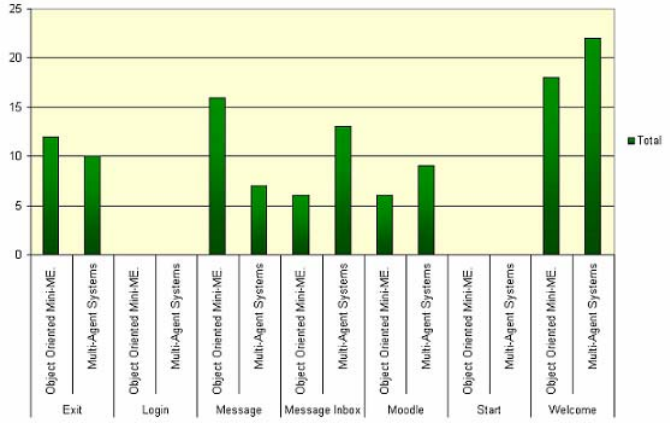


Fig. 4. Results from Interaction Performance Tests

There was only, however, a moderate increase in the time required before each successive interface was displayed. This was due to the mediator agents' message processing algorithm. A second evaluation analysed the multi-threaded nature of the MAS application versus the single-threaded nature of the object-oriented client-server approach. The results, as displayed below, indicated that the inclusion of the mediator agent effectively reduced the interval delay between each user interface. Conversely, the time delay significantly increased for the students simultaneously accessing each successive window.

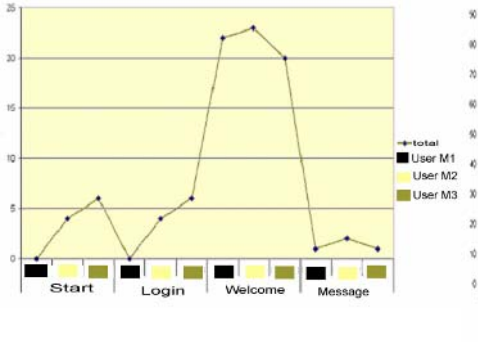


Fig. 5. MAS Multiple Time Increments

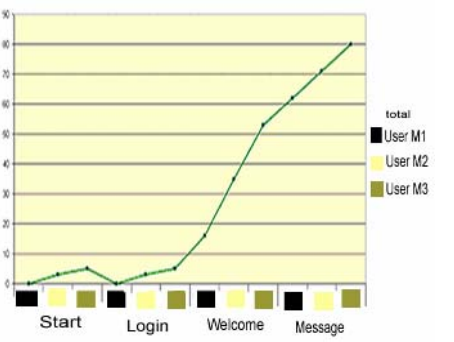


Fig. 6. Object Oriented Multiple Time Increments

As part of the weighted priority allocation algorithm test, results obtained have shown that the Mini-ME wrapper agent message is capable of maintaining a working weighted priority allocation function based on three factors; the type of message sent, the amount of time in the queue, and the sender of the message without affecting the overall functioning of the interface agent. Each message received is given a priority rating based on the above factors. As the length of time a message has been forced to wait increases, so does its priority rating. Thus preventing starvation ever occurring. Initially each weight was determined independently of the other. Later, it was ascertained that by combining the weighting factors, the interface agent's resources could be freed up. By setting a high priority level three, it was ensured that no messages would spend more than twenty seconds in the queue, regardless of their priority level and that starvation would never occur. *Figure 7* displays these results in graph form.

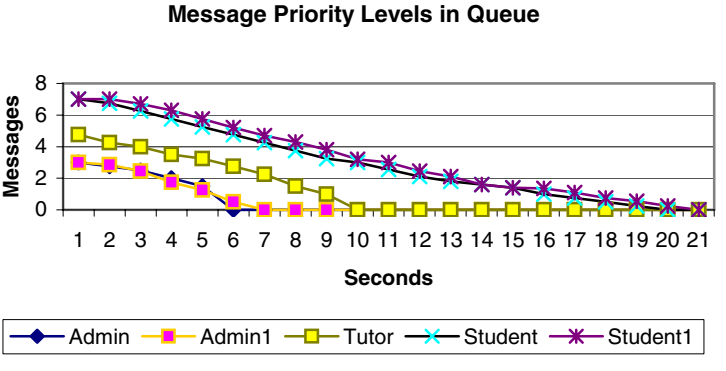


Fig. 7. Results from Weighting Priority Algorithm Tests

5 Discussion and Future Work

The Mini-ME application is an ubiquitous mobile managed learning environment developed for use by students in third level e-learning academic institutions. It was developed in order to present a 'comparative structure' depicting the variations between two design frameworks in a lucid and inclusive manner. This comparative study was applied to a standard MLE that is currently used in a number of academic institutions globally: Moodle [4]. Similarities were evident in the application design and analysis phase, through the use of modeling language schematics, as displayed in the design section: including UML and AUML. The MAS approach, however, facilitated the use of the GAIA methodology. These design schematics proved useful as the Mini-ME application required communication among its components in both the multi-agent system framework and the object-oriented framework.

Extensive coupling was not only alleviated in the MAS, but also proved difficult to implement through the use of a mediated communications network of agents. The object-oriented Mini-ME application is single-threaded, with a container class managing each instance lifecycle. The MAS Mini-ME is inherently a multi-threaded application using the FIPA compliant Agent Factory framework.

The two architectures required different component structures. In the case of the object oriented Mini-ME application, Servlets were used to communicate with the separate components. This methodology required that methods be invoked in a sequential manner. In the MAS, the interaction among components is asynchronous and can be done in parallel. The MAS framework is more suited to parallel processing than is the OOD framework. The J9 Mini-ME infrastructure relies on the data stored on the server side MySQL database to inform users of other students logging on. However, the mediator acts as an intermediary in this regard and maintains a registry of currently logged in participants. Furthermore it underlines the usefulness of the multi-threaded MAS framework.

The object-oriented Mini-ME application offers a less complex support structure for the reuse of code. Performance evaluations comparing the single threaded object oriented Mini-ME to the multi-threaded MAS application displayed concrete results. When the object-oriented and MAS applications were started simultaneously, the object-oriented Mini-ME completed its standard tasks on average one sixth times faster than the MAS Mini-ME. This has been attributed to the inclusion of the mediator agent that decoupled the user interface agent from the database wrapper agent. However, these results were reversed when multiple-users attempted to log on simultaneously in both variations of the application.

Another issue raised in the development of this application was the specification of the location of the mobile user interface at start time. The static nature of the object oriented Mini-ME interface meant that the user could not alternate between devices used during the course of a student's course management session. The MAS framework has been designed with the potential of implementing a mobile agent architecture that has the potential to migrate between multiple devices during the course of on session. A couple of issues that could be further investigated in the future include:

- Application of Mini-ME to other Managed Learning Environments;
- Scalability testing of Mini-ME, where a large corpus of students access the MLE resources;
- Development of migrating agents to Mini-ME in order to enhance the processing power;
- Improvement of security for Mini-ME during for storage and transmission of personalized information

Overall, if a mobile managed learning environment were inherently parallel, and the majority of communications within the system would be asynchronous, or if it were to be an open architecture application that would grow, dynamically, then it would be advisable for the developer to consider a FIPA-MAS framework.

References

1. Quinsee, S. Hurst, J. Blurring the Boundaries? Supporting Students and Staff within an Online Learning Environment. *In Turkish Online Journal of Distance Education-TOJDE* (2005)
2. Marriott, N and Shellard, E. Going online at the University of XYZ: Reflections on the Transformation from Traditional to Virtual Campus, *In Proceedings of Distance Teaching & Learning Conference* (2001)
3. Blackboard – Access to Learning Resources, http://www.blackboard.com/us/index.aspx
4. Moodle – Course Contect Management System, http://moodle.org/
5. Fisher, M and Solliday-McRoy, C. The European Computer Driving License: A Model for Teacher Education. *Journal of Educational Technology Systems Volume 27, Number 3* (1998-1999)
6. Staudt, C. Probing Untested Ground: Young students learn to use handheld computers. *Retrieved from: http://www.concord.org/library/1999fall/untested-ground.html* (October 1999)
7. Belanger, Y. Laptop Computers In the K-12 Classroom. *Eric Digest*. (2000)
8. Lieberman, H. Integrating User Interface Agents with Conventional Applications. *International Conference on Intelligent User Interfaces.* (1998)
9. Kay, A. "User interface: A personal view " Laurel. B. (ed.). *The art of Human-Computer Interface Design*, Addison-Wesley, 1990, 191-207
10. Kozierok, R., and Maes, P. A Learning Interface Agent for Scheduling Meetings. In Proceedings of the ACM SIGCHI International Workshop on Intelligent User Interfaces, 81-88. Orlando, Florida: ACM Press. (1993)
11. Lieberman, J and Breazeal, C. Improvements on action parsing and action interpolation for learning through demonstration. *In Proceedings of the IEEE-RAS/RSJ International Conference on Humanoid Robots (Humanoids 2004) (Santa Monica, CA, USA), IEEE, (*2004)
12. Webber, C. Bergia, L. Pesty, S Balacheff, N The Baghera project: a multi-agent architecture for human learning. Vassileva, J. (ed), Multi-agent architectures for distributed learning environments, 12, 1060-1069, (AIED workshop proceedings). (2001)
13. AgentFactory, URL: http://www.agentfactory.com
14. iPAQ5555, http://www.pdasupport.com/Ipaq5555.htm
15. UML diagrams, http://www.uml.org

16. McGovern, E. 2005 “A Comparative Study of Multi-Agent and Object-Oriente Methodologies for Mobile Managed Learning Environments”
17. Monahan, T. McArdle, G. Bertolotto, M. Mangina, E. 2005 “ 3D User Interfaces and Multimedia in E-Learning”. Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications. (ED-MEDIA 2005) Montreal, Canada.
18. FIPA, http://www.fipa.org

Evolution of Ubi-Autonomous Entities

Jason Hung[1], Kuan-Ching Li[2], Wonjun Lee[3], and Timothy K. Shih[4]

[1] Department of Information Management
Northern Taiwan Institute of Science and Technology
No. 2, Xueyuan Rd., Peitou, 112 Taipei, Taiwan, R.O.C.
jhung@cs.tku.edu.tw
[2] Parallel and Distributed Processing Center
Dept. of Computer Science and Information Engineering
Providence University
Shalu, Taichung 43301, Taiwan
kuancli@pu.edu.tw
[3] Department of Computer Science and Engineering
Korea University, Seoul, Korea
[4] Department of Computer Science and Information Engineering
Tamkang University
tshih@cs.tku.edu.tw

Abstract. Ubiquitous power includes one important factor that is the coordination among small instances of individualized smart programs. With the coordination, intelligent strategies can be established to realize computing topology and to serve different purposes of computing goals. With the development of cellular phones and internet, these individualized programs exist in the digital world where a new ecosystem can be formed. These smart entities create groups and consume natural resources in the digital world. Competition and collaboration thus exists among different groups of autonomous entities. We propose a model, based on a concept called food web in ecosystem. Food web exists in natural words for minions of years and serves as a balancing algorithm among different types of species. We define the food web model for autonomous entities and propose algorithms for formation and communication. As a consequence, the proposed model can be used to consider the development of intelligent strategies and the underlying communication topology such that ubiquitous intelligences can be evolved on the digital food web.

Keywords: Evolution Computing, Autonomous Agent, Food Web, Food Chain, Embedded Agents, Intelligent Networks, World Modeling and Semantics.

1 Introduction

Communication over wireless networks and Internet is growing increasingly and will have profound implications for our economy, culture and society. From mainframe-based numerical computing to decentralized downsizing, PCs and workstation computers connected by Internet have become the trend of modern computers. In addition to PCs, wireless communication further encourages personalized communication/computing

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1114 – 1123, 2006.

devices to be developed on small smart devices, such as PDAs, cellular phones, and other potable devices. These devices, along with Internet computers, had created a potential digital world for smart entities to survive, work, and evolve. The concept of ubiquitous intelligent can be redefined to include how these smart entities coordinate, compete, and consume/produce resources in the digital world. For instance, cellular network relies on individual units to pass information. Topology of such application is an intelligent exists on the infrastructure. The topology can be further enhances such that sharable small agent programs can be reallocated and reused in the topology, without a centralized control. Similar concepts exist in so called mobile agents on the Internet. Mobile agents can be cloned and reused for several purposes. For instance, to search for a particular digital on the Internet, it is necessary to copy a mobile searching agent to be broadcasted on the Internet topology. Coordination and communication among these cloned agents is an essential need to enable efficient and successful search. A mobile agent, in general, can be more than just a search program. For instance, a mobile agent can serve as an emergency message broadcaster, an advertising agent, or a survey questionnaire collector. A mobile gent should have the following properties:

- It can achieve a goal automatically.
- It should be able to clone itself and propagate.
- It should be able to communicate with other agents.
- It has evolution states, including a termination state.

The environment where mobile agents live is Internet and many types of cellular networks. Agents are distributed automatically or semi-automatically via some communication paths. Therefore, agents meet each other on the Internet. Agents have the same goal can share information and cooperate. However, if the system resource (e.g., network bandwidth or disk storage of a station) is insufficient, agents compete with each other. These phenomena are similar to those in the ecosystem of the real world. A creature is born with a goal to live and reproduce. To defense their natural enemies, creatures of the same species cooperate. However, in a perturbation in ecosystems, creatures compete with or even kill each other. The natural world has built a law of balance. Food web (or food chain) embeds the law of creature evolution. With the growing popularity of Internet where mobile agents live, it is our goal to learn from the natural to propose an agent evolution computing model over the Internet. The model, even it is applied only on the mobile agent evolution discussed in this paper, can be generalized to solve other computer science problems. For instance, the search problems in distributed artificial intelligence, network traffic control, electronic commerce, or any computation that involves a large amount of concurrent/distributed computation.

We propose a logical network for agent connections/communications called Agent Communication Network (or ACN) in section 3. ACN is dynamic. It evolves as agent communication proceeds. It also serves as a graph theoretical model of agent evolution computing. Given an ACN, the model finds which agent evolution policy produces the maximum throughput (i.e., the goal of agents achieved). Or, changing the structure of an ACN, the model is able to find out how to adjust the agent evolution policy in order to recover from the change (or how is the throughput affected). To service on the Internet and ACN, agents evolves and lives for particular goals. Evolution of agents

needs to be defined. The evolution of a particular agent species includes several states, as we should see in section 4. In our conclusion section, we point out potential applications of ubiquitous computing.

2 Related Works

The concept of agent-based software engineering is discussed in a survey paper [2]. The author presents two important issues: agent communication language and agent architecture. An open agent architecture for kiosk-based multimedia information service is proposed in [1]. Survey of different types of agents can be found in [5]. The author [5] also indicates that, Web-based intelligent agents are necessary and should be a trend to new engineering applications. However, application designers need to resolve the conflict between the client-server-based designs and the peer-to-peer protocol. The Java Agent Template (JAT) was proposed, which allows application designers to write Java programs run on heterogeneous computer environments.

The concept of mobile agent is discussed in several articles [4]. Agent Tcl, a mobile-agent system providing navigation and communication services, security mechanisms, and debugging and tracking tools, is proposed in [3]. The system allows agent programs move transparently between computers.

Many pervasive and ubiquitous researches combined with agent technology were found in the literature. A new smart meeting room system called EasyMeeting [7] explores the use of multi-agent systems. By the way, COBRA imports several different upper ontologies from the Standard Ontology for Ubiquitous and Pervasive Applications (SOUPA) [7]. The SOUPA is designed to model and support pervasive computing application. This ontology is expressed using the Web Ontology Language OWL and includes modular component vocabularies to represent intelligent agents with associated beliefs, desires, and intentions, time, space, events, user profiles, actions, and policies for security and privacy. In [9], the authors introduced a novel approach named Group-based Service Discovery (GSD). GSD combined the two fields by utilizing semantic service descriptions used in service matching to develop an efficient, distributed, scalable, and adaptive service discovery architecture for Ubiquitous and Pervasive computing environments.

Wireless communication, sensor technology, and miniaturization of electronics enable us to create new form factors of communication devices. Computers are no longer heavy and bulky and hidden away under a desk, but available as miniature notebooks with built-in electronics for wireless communications with a computer network. Speech technology, touch screens, heads-up displays, and pointing devices allow us to experiment with new form factors and to integrate these into the things we wear (wearable) or, in an extreme case, even as body implants. In heterogeneous wireless infrastructure, [10] provides the Session Initiation Protocol (SIP) to support addressing and localization of users and end devices.

The agent technology was applied widely in ubiquitous and context-aware environment. Context-aware Hospital Information System (CHIS) [8] is an example. Hospital information systems (HISs) that provide access to electronic patient records are a step in the direction of providing accurate and timely information to hospital staff in support of adequate decision-making.

3 The Agent Communication Network

Ubiquitous environments such as cellular networks may form an environment where small programs exist. These small programs are called agents. Agents communicate with each other since they can help each other. For instance, agents share the same search query should be able to pass query results to each other so that redundant computation can be avoided. An Agent Communication Network (ACN) serves this purpose. Each node in an ACN (shown in figure 1) represents an agent on a computer network node, and each link represents a logical computer network connection (or an agent communication link). Since it is a logical link, it is not necessary for the link to be direct. Agents of the same goal want to pass results to each other. They are modeled as a complete graph. A complete graph has a logical link between each pair of nodes. Therefore, results can be shared between two nodes without going through multiple agents. Each ACN holds different goals. We use lower case letters in figure 1 to represent agents and upper case letters to represent species. In figure 1, there are 8 types of agents. Agents of the same type run the same goal and form a species. Since agents can have multiple goals (e.g., searching based on multiple criteria), an agent may belong to different complete graphs. For instance, agent a belongs to the complete graphs of species A and B. Similarly, agent b belongs to B and C, both agent c and agent d belong to C and D, d also belongs to E, and e belongs to F and G. On the other hand, agents may have a unique goal (e.g., agents of species G, except agent e). Agent f of species H is the only one of its kind.

We define some terminology used through this paper. A host station (or station) is a networked workstation on which agents live. A query station is a station where a user releases a query for achieving a set of goals. A station can hold multiple agents. Similarly, an agent can pursue multiple goals. An agent society (or society) is a set of agents fully connected by a complete graph, with a common goal associated with each agent in the society. A goal belongs to different agents may have different priorities. An agent society with a common goal of the same priority is called a species. Since an agent may have multiple goals, it is possible that two or more societies (or species) have intersections. A communication cut set is a set of agents belong to two distinct agent societies, which share common agents (e.g., {a}, {b}, {c, d}, {d} and {e} in figure 1). The removing of all elements of a communication cut set results in the separation of the two distinct societies. An agent in a communication cut set is called an articulation agent (e.g., agents a, b, c, d and e). Since agent societies (or species) are represented by complete graphs and these graphs have communication cut sets as intersections, articulation agents can be used to suggest a shortest network path between a query station and the station where an agent finds its goal. Another issue is that an articulation agent can hold a repository, which contains the network

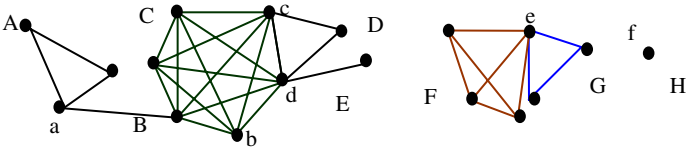


Fig. 1. Agent Communication Network

communication statuses of links of an agent society. Therefore, network resource can be evaluated when an agent checks its surviving environment to decide its evolution policy.

It is necessary for us to give formal definitions of the terminology to be used in our algorithms. In the following definitions, "= =" read as "is defined by" and ”Set_Of (X)” represents a set of object X:

Host_Station == URL × Resource × Set_Of(Agent)
Resource == Network × CPU × Memory × Information
Agent == Set_Of(Goal) × Policy
Goal == Query_Return_URL × Query × Priority
Agent_Society == Set_Of(Agent)
Species ⊂ Agent_Society

A Host_Station has a uniform resource locator (i.e., URL)1 which represents the station's unique network address. A host station has system resources (i.e., Resource) and can hold some agents (i.e., Set_Of(Agent)). Network represents the network facility available to a station. CPU represents the computation power of a station. Memory represents the storage of a station. It could be the main memory or the secondary memory. Information is available on a station. Each Agent has some Goals and a Policy, which is a set of application dependent factors the agent depends on to perform its evolution computation. Query_Return_URL is the URL where an agent should return its query results. Query is an application dependent specification, which represents a user request to the agent. Priority is an integer represents the priority of a goal. The larger the integer, the higher the goal priority. Agent_Society is a set of agents, which share a common goal. Species is an Agent_Society of the same goal priority.

4 The Evolution of Ubi-Autonomous Entities

An agent evolves. It can react to an environment, respond to another agent, and communicate with other agents. The evolution process of an agent involves some internal states of an agent, as well as a communication to other agents. As shown in figure 2, an agent is in one of the following states after it is born and before it is killed or dies of natural:

- Living: an agent is born with a goal to live
- Suspending: an agent is waiting for enough resource in its environment
- Dangling: the agent loses its goal of surviving, it is waiting for a new goal
- Mutating: the agent is changed to a new species with a new goal

We use two single layered circles, one in dark and one in white, to represent the initial (creation) and final (termination) of agents, respectively. We also use four double layered circles to represent the four states as discussed. Since agents do not exist alone in general, we use two example agents in figure 2 to illustrate the communication process in ACN in general. The procedure of agent evolution and communication can be discussed below:

[1] We could use an IP address. But, since our implementation of agents is based on the Web, a unique URL is used instead.

1. An agent is born with a goal in a Living state. The goal could be a task to search for an object, to request for a help, to make an announcement, etc.
2. When natural environment has no enough resource, the agent can be temporarily suspended (transit to the Suspending state).
3. When environment is changed with enough resource, the agent is resumed and transit back to the Living state.
4. In case that the environment run out of resources (below a low threshold), the agent must die. Thus, agent transition terminates.
5. If the agent achieves its goal, the transition id directed to a Dangling state. A Dangling state id different from a suspending state.
6. An agent in a Dangling state can wait for a new mission to be re-born.
7. An agent in the Mutating state can be relocated to another host station, along with its new goal. Thus, the agent is re-born in the Living state.
8. There is a duration in which the agent must receive the mission. If the deadline expires, the agent must die.
9. In most case, while the agent is performing its goal, additional agents of the same kind can be cloned, with the same goal.
10. The new agent communicates with the parent agent such that the entire species is working on the same goal.

The agent evolution state diagram shows how we keep the status of an agent. In order to maintain the activity of agents, in a distributed computing environment, we use message passing as a mechanism to control agent state transition.

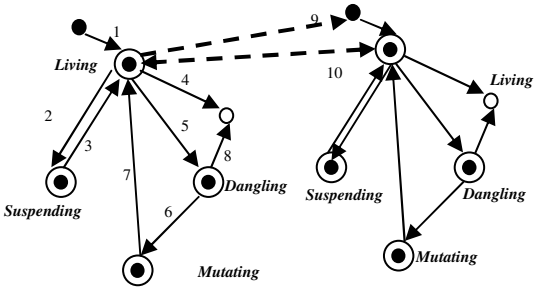


Fig. 2. Evolution States and Transitions

4.1 Agent Messages

Agent evolution computing is performed via actions response to agent messages. There are two categories of messages. The first type of agent messages is sent from a source agent to a destination agent. Agents of the same species are looking for the same goal. When an agent founds its goal, there is no reason for others of the same kind to redo the same goal. Therefore, the agent sends an abort_goal message to other agents of the same species. The message also includes the search results found by the source agent. After that, agents are ready to process a new goal. They can be re-located at a new host station, or stay at the original location.

Agent Messages

Messages from an Agent	Actions
abort_goal(DestAgent, Search_results)	The search process of DestAgent is abort. The search results are also passed to the DestAgent. After that, the source and the destination agents both transit to the dangling state.
new_goal(DestAgent, Goal)	A new search Goal is assigned to DestAgent.
new_host(DestAgent, Host_station)	The DestAgent is re-located at Host_station.

The new_host message is used in conjunction with the new_goal message. After a new_goal message is received, the destination agent transits to a mutating state. The second type of agent messages are sent form a host station. Upon resources available at a host station, the host station may suspend and later resume the search process of a specific agent. In the case that the resources are lower than a threshold, the host station may kill an agent.

Host Station Messages

Messages from an Station	Actions
suspend_goal(DestAgent)	The agent DestAgent is suspended.
resume_goal(DestAgent)	The agent DestAgent is resumed.
kill_agent(DestAgent)	The agent DestAgent is killed.

4.2 Species Food Web and Niche Overlap Graph

Agents can suspend/resume or even kill each other. We need a general policy to decide which agent is suspended of aborted. By our definition, a species is a set of agents of the same goal with a same priority. It is the priority of a goal rather than an agent we base on to discriminate two or more species. We need to construct a direct graph, which represents the dependency between species, w. r. t. a goal. We call this digraph the species food web. Each goal has a graph. Each node in the graph represents a species. All species of a connected food web are of the same goal. We assume that, different users at different host stations may issue the same query. Each directed edge has an origin represents a species of a higher goal priority and has a terminus with a lower priority. Since an agent (and thus a species) can have multiple goals, each goal of an articulation agent should have an associated food web.

We give an example of an ACN in figure 3, were 4 different species exist, called W, X, Y, and Z. Species are represented as complete undirected graphs in different line style and colors (see figure 3). Each species has several agents (from agent a to agent i) as the following:

W has agents a, b, c, d, e
X has agents f, g, b, c, e
Y has agents c, h, e
Z has agents e, c, i

Since agents with different goals may belong to different species, an agent can perform multiple goals in general. We use a general form, agent(w, x, y) to represent that the agent has goals w, x, and y. Figure 3 illustrates goals associated with each agent. Three agents (i.e., b, c, and e) are associated with multiple goals. Therefore, three food webs will be created, as illustrated in figure 4.

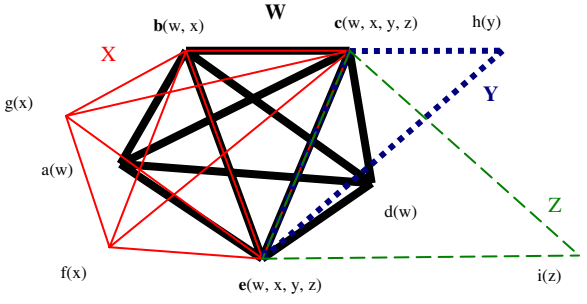


Fig. 3. An Agent Communication Network and Goals

In general, on the Internet, different users may launch same agents to perform same goals. However, it is possible that different users pay different price (with a priority) for the goal such that the same agents belong to the same species may have different priority. Thus, we need to define the priority of each agent associated with a user and a goal within the same species. For example, agent b in figure 3 belongs to two species, W, and X. The user who launches agent b in species W may pay more than another user who also launches agent b in species X. Thus, agent b in species W has a higher priority (denoted as 10) compare to the same agent b in species X (with priority 5). Similarly, the same agent c (and e) in species W, X, Y, and Z may have different priority. Each directed edge in the food web is from a higher priority to a lower priority. This food web structure is constructed in the ACN and can be used as a reference such that dependency relations can be established. The result of such a dependency will allow the host station to decide which agent to suspend or kill.

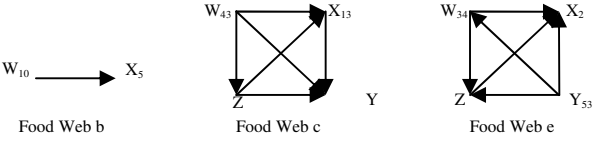


Fig. 4. Food Webs for goals b, c, and e in figure 3

5 Conclusions

Mobile agent based software engineering is interesting. However, in the literature, we did not find many similar theoretical approach to model what mobile agents should act on the Internet, especially how mobile agents can cooperate and compete. A theoretical computation model for agent evolution was proposed. However, there are other extensions to the evolution model. For instance, species in the natural world learn from their enemies. In our future model, agents can learn from each other. We can add a new state, the "learning" state, to the agent evolution state diagram. When an agent is in the dangling state, it can communicate to other agents via some agent communication languages. Computing methods can be replicated from other agents. And the agent transits to the mutating state to wait for another new goal. In addition, when a station lacks of system resource, an agent in the suspending state can change its policy to admit to the environment before it transits to the searching state. These are the facts that agents can learn. On the other hand, in the cloning process, two agents on a station sharing a common goal can be composed to a new agent (i.e., marriage of agents). An agent composition state could be added to the agent evolution state diagram.

The evolution of computers has changed from mainframe-based numerical computation to networked stations. In line with the success of Internet technologies, in the future, computation and information storage are not limited to a single machine. It is possible that, an individual buy a primitive computer that only has a terminal connected to Internet. Personal data and the computation power are embedded within the Internet. Mobile agent and agent evolution computing will be very interesting and important. Our agent evolution model addresses only a small portion of the ice field, which should be further studied in the societies of network communications, automatic information retrieval, and intelligent systems.

References

1. P. Charlton, Y. Chen, E. Mamdani, O. Olsson, J. Pitt, F. Somers, and A. Wearn, “An Open Agent Architecture for Integrating Multimedia Services,” in Proceedings of the Autonomous Agents 97 conference, Marina Del Rey, California, U.S.A., 1997, pp. 522 - 523.
2. Michael R. Genesereth, “Software Agents,” communication of the ACM, Vol. 37, No. 7, Jul. 1994, pp.48 -- 54.
3. David Kotz, Robert Gray, Saurab Nog, Daniela Rus, Sumit Chawla, and George Cybenko, “Agent Tcl: targeting the needs of mobile computers,” IEEE Internet Computing, Vol. 1, No. 4, July 1997, pp. 58 -- 67.
4. T. Magedanz and T. Eckardt, “Mobile software agents: a new paradigm for telecommunications management,” in Proceedings of the 1996 IEEE Network Operations and Management Symposium (NOMS'96), Kyoto, Japan, 1996, pp 360 -- 369.
5. Charles J. Petrie, “Agent-Based Engineering, the Web, and Intelligence,” IEEE Expert, Vol. 11, No. 6, 1996, pp. 24 - 29.
6. Peter S. Sapaty and Peter M. Borst, “WAVE: mobile intelligence in open networks,” in Proceedings of the 1996 1st Annual Conference on Emerging Technologies and Applications in Communications, Portland, OR, USA, 1996, pp. 192 -- 195.

7. H. Chen, T. Finin, Joshi Anupam, L. Kagal, and F. Perich, "Intelligent agents meet the semantic Web in smart spaces," IEEE Internet Computing, IEEE, Volume 8, Issue 6, Nov.-Dec. 2004, pp. 69 – 79
8. J. Favela, M. Rodriguez, A. Preciado, and V.M. Gonzalez, "Integrating context-aware public displays into a mobile hospital information system," IEEE Transactions on Information Technology in Biomedicine, Volume 8, Issue 3, Sept. 2004, pp.279 – 286
9. D. Chakraborty, A. Joshi, Y. Yesha, and T. Finin, "Toward Distributed Service Discovery in Pervasive Computing Environments," IEEE Transactions on Mobile Computing, Volume 5, Issue 2, March-April 2006, pp. 97 – 112
10. Theo G. Kanter, "HotTown, Enabling Context-Aware and Extensible Mobile Interactive Spaces," IEEE Wireless Communications, October 2002, pp. 18-27

Towards a Universal Knowledge Representation Language for Ubiquitous Intelligence Based on Mental Image Directed Semantic Theory

Masao Yokota

Department of System Management, Faculty of Information Engineering,
Fukuoka Institute of Technology,
3-30-1, Wajiro-higashi, Higashi-ku, Fukuoka-shi, 811-0295 Japan
yokota@fit.ac.jp
http://www.fit.ac.jp

Abstract. Towards an ideal ubiquitous computing environment, M.Yokota et al have already presented the concept of Distributed Intelligent Robot Network (DIRN) consisting of one brain node and numerous sensor and actor nodes with human-friendly interfaces. In order for well-coordinated DIRNs, it is essential to develop a formal knowledge representation language such as is universal for any kinds of devices and enlightens them enough to communicate with ordinary people naturally and to compute about and act upon their environments appropriately. For this purpose, the author has developed a formal language $\boldsymbol{L_{md}}$ and been applying it to integrated multimedia understanding and various cross-media operations in simulation of DIRN-world interaction.

1 Introduction

The author has already presented the concept of Distributed Intelligent Robot Network (DIRN [11]) towards an ideal ubiquitous computing environment. A DIRN is one kind of Wireless Sensor and Actor Network (WSAN) [8,9], consisting of one brain node and numerous sensor and actor nodes with human-friendly interfaces. It is assumed, for example, that sensors and actuators can collaborate autonomously to perform appropriate actions just like reflexive actions in humans and that the brain node works exclusively for complicated computation based on profound knowledge in order to control the other kinds of nodes, to communicate with people, etc.

In order to realize DIRNs coordinated well, it is essential to develop a systematically computable knowledge representation language [10, 15, 16] as well as efficient networking technologies [9]. This type of language is indispensable to *knowledge-based* processing such as *understanding* sensory events, *planning* appropriate actions and *knowledgeable* communication even with humans, and therefore it needs to have at least a good capability of representing spatio-temporal events that correspond to humans'/robots' sensations and actions in the real world.

As for human-robot communication, conventionally macro-commands such as 'move(10meters)', directly corresponding with certain computer sub-programs, were employed in order for deploying sensors/motors. However, these commands were

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1124 – 1133, 2006.

very specific to the devices and apt to have miscellaneous variants such as 'move(10meters, quickly)' and 'move(quickly, 10meters, leftward)', which is very inconvenient for communications especially between devices unknown to each other. This is also the case for spatial expressions such as 'left(x,y)' and 'left(x,y,z)', reading 'x is (z meters) to the left of y'. Therefore, it is very important to develop such a language as is universal among all kinds of equipments.

Yokota, M. et al have proposed a semantic theory for natural languages so called 'Mental Image Directed Semantic Theory (MIDST)' [1, 2]. In the MIDST, word concepts are associated with omnisensory mental images of the external or physical world and are formalized in an intermediate language $\boldsymbol{L_{md}}$ [10]. This language is employed for many-sorted first-order predicate logic with five types of terms. The most remarkable feature of $\boldsymbol{L_{md}}$ is its capability of formalizing both temporal and spatial event concepts on the level of human sensations while the other similar knowledge representation languages are designed to describe the logical relations among conceptual primitives represented by natural-language words [3, 4] or formally defined tokens [12-14].

The $\boldsymbol{L_{md}}$ was originally proposed for formalizing the natural semantics, that is, the semantics specific to humans, but it is general enough for the artificial semantics, that is, the semantics specific to each artificial device such as robot. This language has already been implemented on several types of computerized intelligent systems [1, 5, 10, 11] and there is a feedback loop between them for their mutual refinement, unlike other similar ones [6, 7].

In this paper we focus on the semantic processing of sensation, action and spatio-temporal knowledge represented in the formal language $\boldsymbol{L_{md}}$, simulating the interactions between robots and their environments including humans.

2 The Formal Language L_{md}

2.1 Mental Image Model

The MIDST treats word meanings in association with mental images, not limited to visual but omnisensory, modeled as "Loci in Attribute Spaces" as shown in Fig.1-a. An attribute space corresponds with a certain measuring instrument just like a barometer, a map measurer or so and the loci represent the movements of its indicator.

A general locus is to be articulated by "Atomic Locus" with the duration $[t_i, t_f]$ as shown in Fig.1-b and formalized as the expression (1). This is a formula in many-sorted first-order predicate logic, where "L" is a predicate constant with five types of terms: "Matter" (at 'x' and 'y'), "Attribute Value" (at 'p' and 'q'), "Attribute" (at 'a'), "Event Type" (at 'g') and "Standard" (at 'k'). Especially, the second Matter term is called 'Attribute Carrier (AC)'.

$$L(x,y,p,q,a,g,k) \tag{1}$$

The formula is called 'Atomic Locus Formula' and its intuitive interpretation is given as follows, where 'matter' refers to 'object' or 'event'.

"Matter 'x' causes Attribute 'a' of Matter 'y' to keep (*$\mathbf{p=q}$*) or change (*$\mathbf{p \neq q}$*) its values temporally (*$\mathbf{g=G}t$*) or spatially (g =Gs) over a time-interval, where the values 'p' and 'q' are relative to the standard 'k'."

When g=Gt and g=Gs, the locus indicates monotonous change or constancy of the attribute in time domain and that in space domain, respectively. The former is called a temporal event and the latter, a spatial event.

For example, the motion of the 'bus' represented by S1 is a temporal event and the ranging or extension of the 'road' by S2 is a spatial event whose meanings or concepts are formalized as (2) and (3), respectively, where the attribute is "physical location " denoted by 'A12'.

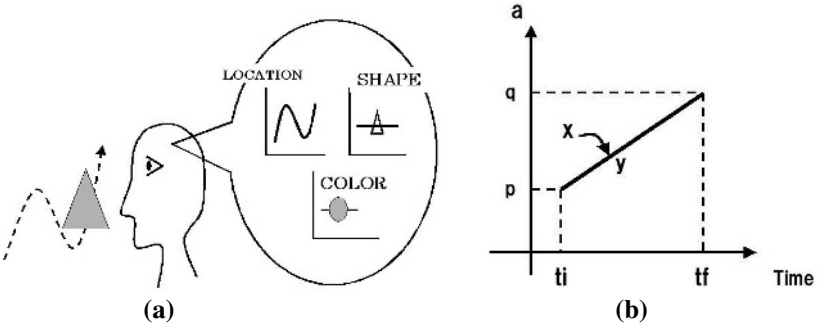


Fig. 1. Loci in Attribute Spaces as mental images (a) and Atomic Locus (b)

(S1) The bus runs from Tokyo to Osaka.

$$(\exists x,y,k)L(x,y,Tokyo,Osaka,A12,Gt,k)\wedge bus(y) \tag{2}$$

(S2) The road runs from Tokyo to Osaka.

$$(\exists x,y,k)L(x,y,Tokyo,Osaka,A12,Gs,k)\wedge road(y) \tag{3}$$

The difference between temporal and spatial event concepts can be attributed to the relationship between the Attribute Carrier (AC) and the Focus of the Attention of the Observer (FAO). To be brief, the FAO is fixed on the whole AC in a temporal event but *runs* about on the AC in a spatial event.

2.2 Tempo-logical Connectives

The MIDST has introduced tempo-logical connectives (TLCs), so called, in order to indicate both logical and temporal relations between loci at a time. A tempo-logical connective K_i is defined by (4), where τ_i, χ and K refer to one of the temporal relations indexed by *'i'*, locus, and an ordinary binary logical connective such as the conjunctive '$\wedge$', respectively. In general, a series of atomic locus formulas combined with these connectives is called simply 'Locus formula'. The 13 types of temporal relations between two intervals [12] are discriminated by τ_i ($-6\leq i \leq 6$). More exactly, the conventional notation for temporal relations [12] is exclusively for 'temporal conjunctives ($=\wedge_i$)' as described in 2.3.

$$\chi_1 \, K_i \, \chi_2 \leftrightarrow (\chi_1 \, K \, \chi_2) \wedge \tau_i(\chi_1, \chi_2) \tag{4}$$

2.3 Formal Description of Spatio-temporal Knowledge

The expression (5) is the conceptual description of the English verb "fetch", depicted in Fig.2-a, implying such a temporal event that 'x' goes for 'y' and then comes back with it, where '$\Pi(=\wedge_0)$'and '$\bullet(=\wedge_1)$' are instances of temporal conjunctives, 'SAND' and 'CAND', standing for "Simultaneous AND" and "Consecutive AND", respectively.

$$(\lambda x,y)fetch(x,y) \leftrightarrow (\lambda x,y)(\exists p1,p2,k)L(x,x,p1,p2,A12,Gt,k) \bullet ((\underline{L(x,x,p2,p1,A12,Gt,k)} \; \underline{\Pi L(x,y,p2,p1,A12,Gt,k)}) \wedge x \neq y \wedge p1 \neq p2 \quad (5)$$

Employing these TLCs, tempo-logical relationships between miscellaneous event concepts can be formulated without explicit indication of time intervals. For example, an event 'fetch(x,y)' is necessarily *finished by* an event 'carry(x,y)', whose concept is depicted as Fig.2-b and defined by (6) corresponding with the underlined part at (5). This fact can be formulated as (7), where '$\supset_{-4}$' is the '*implication* ($\supset$)' furnished with the temporal relation '*finished-by* (τ_{-4})'. This kind of formula is not an axiom but a theorem deducible from the definitions of event concepts in our formal system.

$$(\lambda x,y)carry(x,y) \leftrightarrow (\lambda x,y)(\exists p,q,k)L(x,x,p,q,A12,Gt,k)\Pi L(x,y,p,q,A12,Gt,k) \wedge x \neq y \wedge p \neq q \quad (6)$$

$$fetch(x,y) \supset_{-4} carry(x,y) \quad (7)$$

As for spatial events, our formal system has a very interesting and useful postulate for spatial inference, so called, ' ***Postulate of Reversibility of a Spatial Event*** (PRS) '. For example, the sentences S3 and S4 can refer to the same scene in the external world as shown in Fig.3-a as their semantic descriptions are given by (8) and (9), respectively, where 'A13', '↑' and '↓' refer to the attribute 'Direction' and its values 'upward' and 'downward', respectively. They are to be proven as semantically equivalent by applying the PRS in our formal system. This is also the case for S5 and S6. As easily imagined, the apparent movements of the ACs here reflect those of the FAOs.

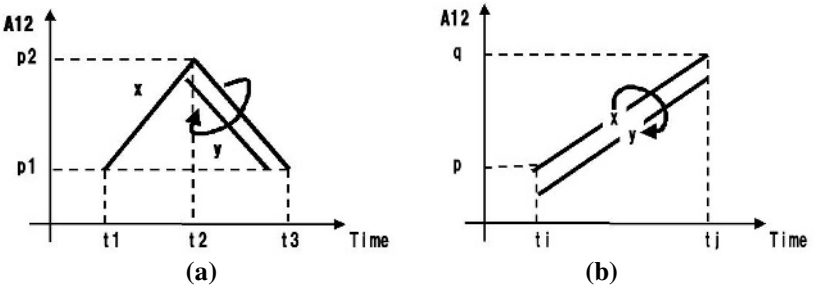


Fig. 2. Loci of 'fetch' (a) and 'carry' (b)

(S3) The path *sinks* to the brook.
(S4) The path *rise*s from the brook.
(S5) The roads *meet* there.
(S6) The roads *separate* there.

$$(\exists x,y,p,z,k1,k2)L(x,y,p,z,A12,Gs,k1)\Pi L(x,y,\downarrow,\downarrow,A13,Gs,k2)\wedge path(y) \wedge brook(z) \wedge p\neq z \quad (8)$$

$$(\exists x,y,p,z,k1,k2)L(x,y,z,p,A12,Gs,k1)\Pi L(x,y,\uparrow,\uparrow,A13,Gs,k2)\wedge path(y) \wedge brook(z) \wedge p\neq z \quad (9)$$

For another example of spatial event, Fig.3-b concerns the perception of the formation of multiple objects, where FAO runs along an imaginary object so called 'Imaginary Space Region (ISR)'. This spatial event can be verbalized as S7 using the preposition 'between' and formulated as (10) or (10'), corresponding also to such concepts as 'row', 'line-up', etc.

(S7) □ is between Δ and ○.

$$(\exists x,y,p,q,k)(L(x,y,\Delta,\square,A12,Gs,k)\Pi L(x,y,p,p,A13,Gs,k))\bullet(L(x,y,\square,\circ,A12,Gs,k) \Pi L(x,y,q,q,A13,Gs,k))\wedge ISR(y)\wedge p=q \quad (10)$$

$$(\exists x,y,p,k)(L(x,y,\Delta,\square,A12,Gs,k)\bullet L(x,y,\square,\circ,A12,Gs,k))\Pi L(x,y,p,p,A13,Gs,k) \wedge ISR(y) \quad (10')$$

Employing ISRs and the 9-intersection model [17], all the topological relations between two objects can be formulated in such expressions as (11) or (11') for S8, and (12) for S9, where '*In*', '*Cont*' and '*Dis*' are the values 'inside', 'contains' and 'disjoint' of the attribute 'Topology (A44)' with the standard '9-intersection model (*9IM*)', respectively.

(S8) Tom is in the room.

(S9) Tom exits the room.

$$(\exists x,y,k)L(Tom,x,y,Tom,A12,Gs,k)\Pi L(Tom,x,In,In,A44,Gt,9IM) \wedge ISR(x) \wedge room(y) \quad (11)$$

$$(\exists x,y,k)L(Tom,x,Tom,y,A12,Gs,k)\Pi L(Tom,x,Cont,Cont,A44,Gt,9IM) \wedge ISR(x) \wedge room(y) \quad (11')$$

$$(\exists x,y,k)L(Tom,x,y,Tom,A12,Gs,k)\Pi L(Tom,x,In,Dis,A44,Gt,9IM) \wedge ISR(x) \wedge room(y) \quad (12)$$

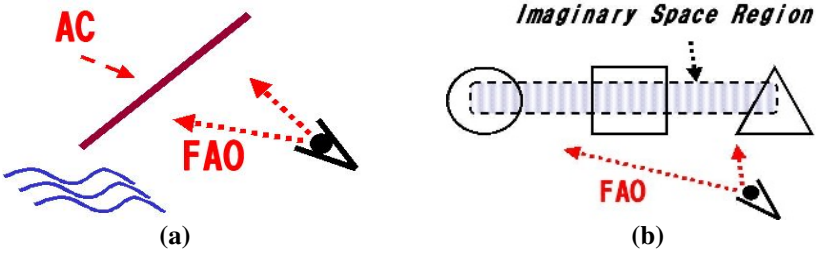


Fig. 3. FAO movements: 'slope' (a) and 'row' (b) as spatial events

3 Interaction Between a DIRN and the World

The integrated multimedia understanding system IMAGES-M [11] works as the main intelligence of the brain node of a DIRN. The intelligence of each sensor or actuator is a small-scaled IMAGES-M adapted for its specialized function. This system has employed locus formulas as intermediate conceptual representations in $\boldsymbol{L}_{md}$, through which it can integrally understand and generate sensor data, speech, visual image, text, and action data.

As shown in Fig.4, a DIRN is to solve some kinds of problems in its world. Such problems can be classified roughly into two categories as follows.

(CP) Creation Problem:

e.g.) house building, food cooking, etc.

and

(MP) Maintenance Problem:

e.g.) fire extinguishing, room cleaning, etc.

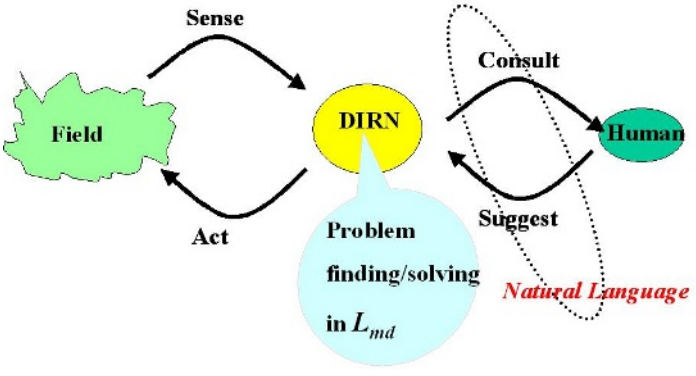


Fig. 4. Interaction between a DIRN and the world

In general, an MP is relatively simple one that the DIRN can find and solve autonomously while a CP is relatively difficult one that is given to the DIRN, possibly, by humans and to be solved in cooperation with them.

3.1 Definition of a Problem and a Task for a DIRN

A DIRN must determine its task to solve a problem in the world. In general, the DIRN needs to interpolate some transit event X_T between the two events, namely, 'Current Event (X_C)' and 'Goal Event (X_G)' as shown by (13).

$$X_C \bullet X_T \bullet X_G \tag{13}$$

According to this formalization, a problem X_P is defined as $X_T \bullet X_G$ and a task for the DIRN is defined as its realization.

The events in the world are described as loci in certain attribute spaces and a problem is to be detected by the unit of atomic locus. For example, employing such a

postulate as (14) implying 'Continuity in attribute values', the event X in (15) is to be inferred as (16).

$$L(x,y,p1,p2,a,g,k) \bullet L(z,y,p3,p4,a,g,k) \ .\supset. \ p3=p2 \tag{14}$$

$$L(x,y,p1,p2,a,g,k) \bullet X \bullet L(z,y,p3,p4,a,g,k) \tag{15}$$

$$L(z',y,p2,p3,a,g,k) \tag{16}$$

3.2 CP Finding and Solving

Consider a verbal command such as S10 uttered by a human. Its interpretation is given by (17), where 'A07' is the attribute 'Depth' and the goal event X_G is underlined. If the current event X_C is given by (18), then (19) with the transit event X_T underlined can be inferred as the problem corresponding to S10.

(S10) Make 'cave C9' 20 meters deep.

$$L(z1,C9,p,20m,A07,Gt,k) \bullet \underline{L(z,C9,20m,20m,A07,Gt,k)} \wedge cave(C9) \tag{17}$$

$$L(x,C9,q,q,A07,Gt,k) \wedge cave(C9) \tag{18}$$

$$\underline{L(z1,C9,q,20m,A07,Gt,k)} \bullet L(z,C9,20m,20m,A07,Gt,k) \wedge cave(C9) \tag{19}$$

For this problem, the DIRN is to execute a task deploying a certain depth sensor and actors 'z1' and 'z'. The selection of the actor 'z1' is performed as follows:

If 20m-q <0 then z1 is a burier, otherwise
if 20m-q >0 then z1 is a digger, otherwise
20m-q =0 and no actor is deployed as z1.

The selection of 'z' is a task in case of MP described in the next section.

3.3 MP Finding and Solving

In general, the goal event X_G for an MP is that for another CP such as S10 given possibly by humans and solved by the DIRN in advance. That is, the task in this case is to autonomously restore the goal event X_G created in advance to the current event X_C as shown in (20), where the transit event X_T is the reversal of such $X\text{-}_T$ that has been already detected as 'abnormal' by the DIRN.

For example, if X_G is given by the underlined part of (17) in advance, X_T is also represented as the underlined part of (19) while $X\text{-}_T$ as (21). Therefore the task here is quite the same that was described in the previous section.

$$X_G \bullet X\text{-}_T \bullet X_C \bullet X_T \bullet X_G \tag{20}$$

$$L(z1,C9,20m,q,A07,Gt,k) \wedge cave(C9) \tag{21}$$

3.4 Robot Manipulation as Problem Solving

At present, IMAGES-M, installed on a personal computer, can deploy SONY AIBOs, dog-shaped robots, as actors and gather information about the physical world through their microphones, cameras and tactile sensors. Communications between IMAGES-M

and humans are performed though the keyboard, mouse, microphone and multicolor TV monitor of the personal computer.

Consider such a verbal command as S11 uttered to the robot, SONY AIBO, named 'John'.

(S11) John, walk forward and wave your left hand.

Firstly, late in the process of cross-media translation from text to AIBO's action, this command is to be interpreted into (22) with the attribute 'shape (*A11*)' and the values '*Walkf-1*' and so on at the standard of '*AIBO*', reading that John makes himself walk forward and wave his left hand. Each action in AIBOs is defined as an ordered set of shapes (i.e., time-sequenced snapshots of the action) corresponding uniquely with the positions of their actuators determined by the rotations of the joints. For example, the actions 'walking forward (***W****alkf*)' and 'waving left hand (***W****avelh*)' are defined as (23) and (24), respectively.

$$L(John,John,Walkf\text{-}1,Walkf\text{-}m,A11,Gt,AIBO) \wedge L(John,John,Wavelh\text{-}1,Wavelh\text{-}n,A11,Gt,AIBO) \quad (22)$$

$$\boldsymbol{W}alkf=\{Walkf\text{-}1,\ Walkf\text{-}2,\ldots,Walkf\text{-}m\} \quad (23)$$

$$\boldsymbol{W}avelh=\{Wavelh\text{-}1,\ Wavelh\text{-}2,\ldots,\ Wavelh\text{-}n\} \quad (24)$$

Secondly, an AIBO cannot perform the two events (i.e., actions) simultaneously and therefore the transit event between them is to be inferred as the underlined part of (25) which is the goal event here.

$$L(John,John,Walkf\text{-}1,Walkf\text{-}m,A11,Gt,AIBO) \bullet \underline{L(John,John,Walkf\text{-}m,Wavelh\text{-}1,A11,Gt,AIBO)} \bullet L(John,John,Wavelh\text{-}1,Wavelh\text{-}n,A11,Gt,AIBO) \quad (25)$$

Thirdly, (26) is to be inferred, where the transit event, underlined, is interpolated between the current event and the goal event X_G (=(25)).

$$L(John,John,p1,p2,A11,Gt,AIBO) \bullet \underline{L(John,John,p2,Walkf\text{-}1,A11,Gt,AIBO)} \bullet X_G \quad (26)$$

Finally, (26) is interpreted into a series of the joint angles in the AIBO. Figure 5 shows AIBO's sitting down and waving his left hand that cannot be done simultaneously.

Fig. 5. AIBO behaving in accordance to the command 'Sit down and wave your left hand'

4 Discussion and Conclusion

This paper described the formal language $\boldsymbol{L_{md}}$ and its applications for ubiquitous intelligence. AI planning ("action planning") deals with the development of representation languages for planning problems and with the development of algorithms for plan construction [15, 16]. The most remarkable point of $\boldsymbol{L_{md}}$ resides on its descriptive power enabling systematic organization and computation of spatio-temporal knowledge including sensation and action. The authors formalized the performances of a DIRN as predicate logic in $\boldsymbol{L_{md}}$ and applied it to robot manipulation by text as a simulation of DIRN-world interaction.

Most of computations on $\boldsymbol{L_{md}}$ are simply for unifying (or identifying) atomic loci and for evaluating arithmetic expressions such as '$p=q$', and therefore we believe that our formalism can reduce the computational complexities of the others [12-14] when applied to the same kinds of problems described here.

Our future work will include establishment of learning facilities for automatic acquisition of word concepts from sensory data and human-robot communication by natural language under real environments.

Acknowledgements

This work was partially funded by the Grants from Computer Science Laboratory, Fukuoka Institute of Technology and Ministry of Education, Culture, Sports, Science and Technology, Japanese Government, numbered 14580436 and 17500132.

References

1. Yokota,M.: An approach to natural language understanding based on a mental image model. Proc. of the 2nd International Workshop on Natural Language Understanding and Cognitive Science (2005) 22-31
2. Yokota,M. et al: Mental-image directed semantic theory and its application to natural language understanding systems. Proc. of NLPRS'91(1991) 280-287
3. Sowa,J.F.: Knowledge Representation: Logical, Philosophical, and Computational Foundations. Brooks Cole Publishing Co., Pacific Grove, CA, (2000)
4. Zarri,G.P.: NKRL, a Knowledge Representation Tool for Encoding the 'Meaning' of Complex Narrative Texts. Natural Language Engineering - Special Issue on Knowledge Representation for Natural Language Processing in Implemented Systems, 3 (1997) 231-253
5. Oda,S., Oda,M., Yokota,M. : Conceptual Analysis Description of Words for Color and Lightness for Grounding them on Sensory Data. Trans. of JSAI,Vol.16-5-E (2001) 436-444
6. Langacker,R.: Concept, Image and Symbol, Mouton de Gruyter, Berlin/New York (1991)
7. Miller,G.A., Johnson-Laird,P.N.: Language and Perception. Harvard University Press (1976)
8. Haenggi,M.: Mobile Sensor-Actuator Networks: Opportunities and Challenges. Proc. of 7th IEEE Int. Workshop, Frankfurt,Germany (2002)283-290

9. Akyildiz,I.F., Kasimoglu,I.H.: Wireless Sensor and Actor Networks: Research Challenges. Ad Hoc Networks, 2 (2004)351-367
10. Yokota,M., Capi,G.: Cross-media Operations between Text and Picture Based on Mental Image Directed Semantic Theory. WSEAS Transactions on Information Science and Applications, 10-2 (2005) 1541-1550
11. Yokota,M., Capi,G.: Integrated Multimedia Understanding for Ubiquitous Intelligence Based on Mental Image Directed Semantic Theory. IFIP EUC'05 UISW2005, Nagasaki (2005)538-546
12. Allen, J.F.: Towards a general theory of action and time. Artificial Intelligence, 23-2 (1984) 123-154
13. Shoham,Y.: Time for actions: on the relationship between time, knowledge, and action. Proc. of IJCAI\89, Detroit, MI (1989) 954-959
14. Haddawy,P.: A logic of time, chance, and action for repesenting plans. Artificial Intelligence, 80-2 (1996) 243-308.
15. Wilkins,D.E, Myers,K.L.: A common knowledge representation for plan generation and reactive execution. Journal of Logic and Computation, 5-6(1995)731-761
16. Kabanza,F.: Synchronizing multiagent plans using temporal logic specifications. Proceedings of the First International Conf. on Multi-Agent Systems (ICMAS-95) (1995) 217-224
17. Egenhofer,M.: Point-set topological spatial relations. Geographical Information Systems, 5-2(1991)161-174

Resolving the Semantic Inconsistency Problem for Ubiquitous RFID Applications

Dongwon Jeong[1] and Younhee Han[2]

[1] Department of Informatics & Statistics, Kunsan National University,
San 68, Gunsan, Jeollabuk-do, 573-701, Korea
djeong@kunsan.ac.kr
[2] School of Internet-Media, Korea University of Technology and Education,
Cheonan, 330-708, Korea
yhhan@kut.ac.kr

Abstract. This paper proposes a noble RFID system architecture, named uRFID, to support application domain-independent semantic interoperability under ubiquitous environment. The objective of the uRFID is to let ubiquitous RFID applications independently use information of tags in a consistent way in a variety of application domains. Although many researches have been studying to support the RFID applications, they do not consider the aforementioned semantic inconsistency issue. The uRFID is based on the metadata registry to achieve the goal. It provides an infrastructure support for the semantic consistency and application domain-independent information utilization. As a result, it enables high quality RFID applications to be developed for ubiquitous environment.

Keywords: RFID, Semantic, Consistency, Interoperability, Ubiquitous computing, Metadata register.

1 Introduction

Ubiquitous computing is increasingly recognized as a new computing paradigm. In the ubiquitous computing environment, rich data of many sensors can be acquired and used by applications. Therefore, ubiquitous applications can provide users high quality services with context information.

Currently, we can also use data of sensors, but its application is restricted in a narrow and specific boundary. In this paper, the boundary refers to application/sensor domain/field. The ideal ubiquitous computing environment should support the function that applications can use context information anytime and anywhere regardless of its application domain.

Many issues should be resolved to realize the ideal ubiquitous computing environment. The issues are the energy management, protocol to gather data, data processing, independency on applications, and so on. Most of all, the application domain-independent semantic interoperability is one of key issues to maximize the usability of sensors [1].

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1134 – 1143, 2006.

One of representative ubiquitous applications is RFID (Radio Frequency IDentification) system. The RFID technology supports contactless and wireless information access of RFID tags attached to objects. It is used in various application domains [2].

Many researches on the RFID technology have been studying and many types of RFID system architecture have been introduced. However, the previous RFID system architectures still have limitations to support ubiquitous applications as follow: (1) Data dependency on a specific application domains, (2) static semantic management, and (3) semantic inconsistency between sensor/application fields/domains.

In this paper, we propose a noble RFID system architecture, called uRFID, to resolve the above issues. The most focused issue is how to support the consistent semantic interoperability. To do this issue, the uRFID uses the metadata registry that is the international standard developed to enhance the interoperability of data [3,4,5].

The metadata registry has many good points: dynamic metadata management, consistent semantic maintenance method, etc. The metadata registry has been applying to various application fields such as bibliographic field, environmental data management, e-Commerce, component management, scientific field, and so on [6,7,8,9,10,11].

The proposed RFID system architecture with the merits of the metadata registry extends the previous RFID system architecture. The uRFID provides the following advantages: Semantic independency on application domains, easy and consistent data sharing and exchanging between data processing systems, dynamic metadata management, improvement of the usability of RF tag information, and progressive integration and standardization of data used in various RFID applications.

This paper is organized as follows: Section 2 introduces the related technologies, the general RFID system architecture and the metadata registry. Section 3 describes the issues of the previous RFID systems for the emerging ubiquitous computing environment and the concept of the uRFID. Section 4 presents the key operations (processes) of the uRFID and the evaluation is discussed in Section 5. Finally, we conclude this paper in Section 5.

2 Related Work

In this section, we describe a general architecture of current RFID systems. And the metadata registry, which is a key component of the proposed architecture, is simply introduced.

2.1 General Architecture for Current RFID Systems

RFID standing for Radio Frequency IDentification is one of the most popular ubiquitous applications. RFID systems enable contactless and wireless access of objects using radio frequency. The RFID system is composed of three elements: RF tags (RFID tags, transponders), RFID readers (RFID readers, transceivers), and DPSs (data processing systems) [2,13].

The RF tag is the data carrier and includes a unique value (tag ID). It typically has a role as an identifier of objects to be identified. With the current RFID system architecture, the memory size of the RF tag is very small and most of RF tags hold only a

unique value to identify their corresponding object. The detailed information of an object is delivered from a data sever. The data server provides the details of the object with the unique value that is detected from the RF tag.

RFID systems can be classified into two types according to the RF tag communication function. One is the passive RFID system and the other is the active RFID system. In the passive RFID system, the RF tag is called the passive RF tag and its communication range is small or medium. In contrast, the active RF tag has larger communication range than the passive tag.

RFID readers detect RF tags and read data in RF tags. And then they transfer the data to data processing systems. In fact, most of current RF tags hold their unique ID. The RFID readers can read only the ID and send to data processing systems. The detailed object information is obtained from the data server using the ID.

Finally, the data processing system receives data from RFID readers, and processes and uses data for achieving given goals. As described above, the readers deliver only ID values to the data processing system. The processing system builds a connection to a data server using the IDs to get the detailed information of the objects. Fig. 1 shows a basic RFID system architecture [2,12].

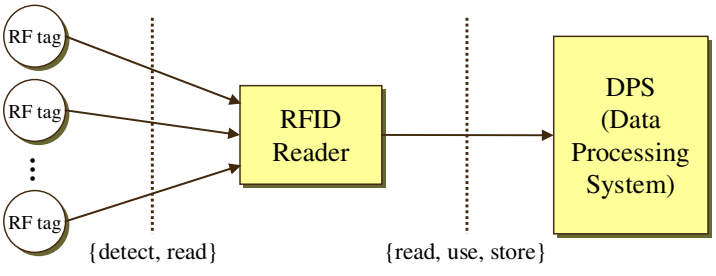


Fig. 1. General RFID system architecture

2.2 Metadata Registry

MDR (Metadata registry) is the most important component of ISO/IEC11179, which is the international standard developed by ISO/IEC JTC 1/SC 32. Its development goal is to provide the improved interoperability of data in various databases. The primitive unit of this standard is the data element.

A data element is a basic unit of data of interest to an organization. The data element consists of many attributes to describe the definition, identification, representation, and permissible values of data.

A metadata registry is a set of data elements including properties to describe data. Also there are many components for systemic metadata definition. Fig. 2 shows the high-level metamodel for the key regions/components [3].

The metadata registry has many advantages, and thus it has been applying to various application fields: bibliographic field, environmental data management, e-Commerce, component management, scientific field, and so on [6,7,8,9,10,11].

The merits of the metadata registry are summarized as follow:

- Dynamic metadata management
- Consistent semantic maintenance method
- Standardized management processes
- Semantic Interoperability between different fields

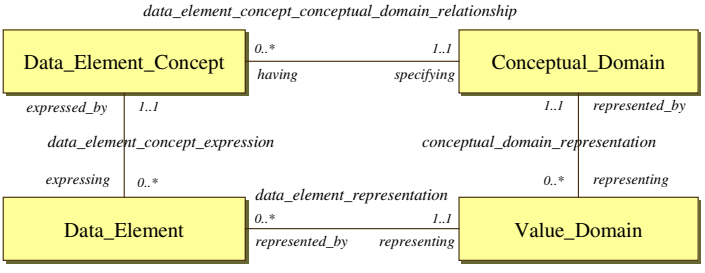


Fig. 2. High-level metamodel

3 Concept of uRFID

This section first introduces the problems of existing RFID systems for the forthcoming ubiquitous environment. Also, the concept of the uRFID is described and an architecture for the uRFID is proposed.

3.1 Problems of Existing RFID Systems

Many researches have been studying on developing improved RFID systems for the ubiquitous applications. However, there still remain several limitations as follow [14,15,16]:

- *Application domain-dependent semantic interoperability*: The existing RFID systems support that the semantic interoperability is available between RF tags and data processing servers in the only same application domain. It means that the interoperability between RF tags and data processing servers in different application domains each other is hardly achieved.
- *Static management of semantics*: Tags are located on objects and generally include only identification information to identify the objects. Recently, enhanced tags, which can include and carry much information such as historical data of objects, have been studying and developing. Therefore, a management policy of semantics of various data in tags is required. Especially, semantics can be created, updated, and deleted. However, current RFID system architectures do not support the issue.

3.2 Conceptual Model of uRFID

The goal of uRFID is to cover the forthcoming ubiquitous computing environment. It implies several assumptions are required. In this paper, we assume that (1) RF tag

memory size is powerful, and so (2) we can record more information (pairs of semantic and value) to RF tags.

Fig. 3 shows the concept of the uRFID in brief. In this figure, a tag, attached to an object, might be created and detected by a device in the same application domain (sensor field). Both are at the same application domain and the device can easily read data from the tag and interpret meanings (semantics) of the read data.

After the task, the tag can be moved to other application domains. In the previous RFID systems, devices in the other application domains detect the tag and read data of the tag. And the devices try to interpret the meanings. However, the devices cannot do that and also cannot use the read data. With the uRFID, the metadata registry supports the semantic interoperability. In other words, the devices in the other field also define their meanings according to the metadata registry. Therefore, the devices can interpret the semantics of the tag data.

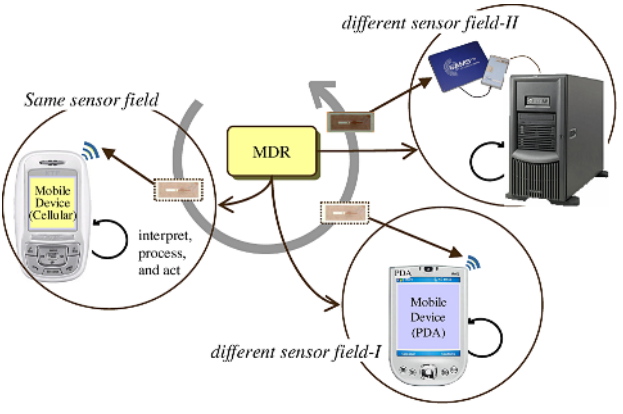


Fig. 3. Conceptual model of uRFID

4 Processes of uRFID

uRFID consists of three processes: Recording, interpretation, and creation process. Fig. 4 shows the overall process and relations.

In Fig. 4, the creation process is to define semantics to be used by RF tags and data processing systems. The semantics are defined in the same way with ISO/IEC 11179 creation process. The creation step is required when users cannot find proper semantics. In this case, the users can submit semantic candidates, and then use the certified semantics.

After building the semantics, the users design their semantics for RF tags or data processing systems according to the created semantics. Once the design is done, the users can write values to RF tags.

The final process is for data processing systems to use and store data of RF tags. The data processing systems are also design their semantics following the created semantics. Therefore, they interpret and use the data from RF tags in easy.

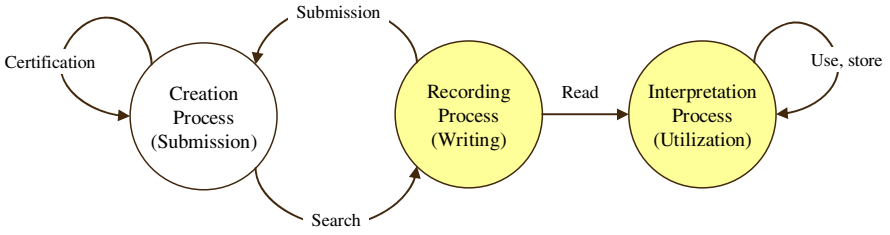


Fig. 4. Overall process of uRFID

Fig. 5 shows the overall step to create new semantics. When a data processing system cannot find proper semantics, the system submits semantic candidates. The metadata registry checks their requirements, and then registers the confirmed semantics. Finally, the metadata registry notifies the new semantics to the users.

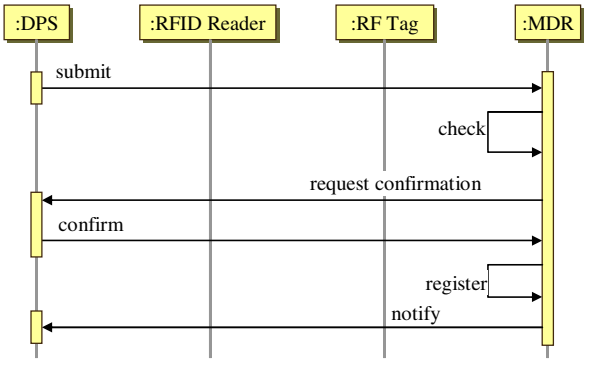


Fig. 5. Creation process of uRFID

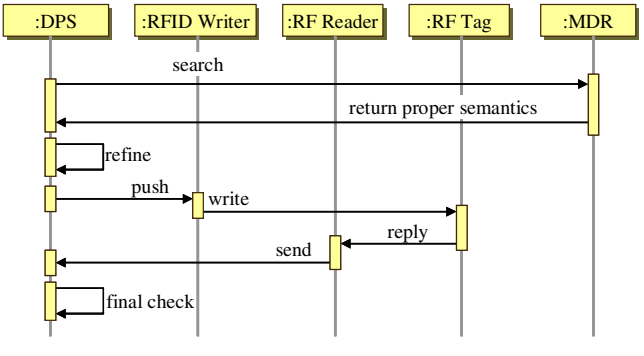


Fig. 6. Recording process

Fig. 6 shows the recording process. Actually, data volume at the RF tag level is less than at the traditional database level. Thus, we can apply the interoperability concept of ISO/IEC 11179. For recording values into a RF tag, we first define

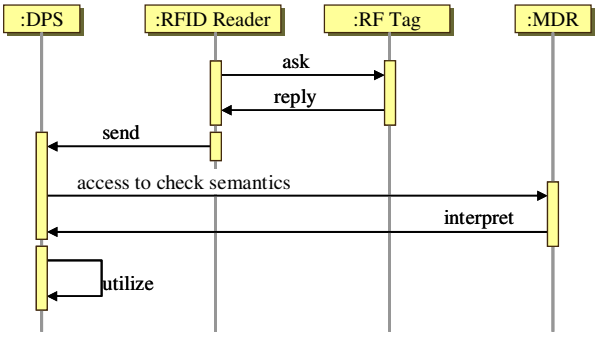


Fig. 7. Interpretation process

semantics for the values. Users find and define the semantics and then record the pairs of semantics and values to the RF tag. If there is no proper semantics, the creation process is required for providing additional new semantics.

Fig. 7 shows the interpretation process. A RFID reader accesses a RF tag to obtain data. When the reader obtains data of the tag (object), it transfers data to a data processing system. The server should interpret the semantics before utilizing the obtained data. In fact, under the proposed system architecture, the system predefines and uses their semantics according to the certified semantics in the metadata registry. Therefore, the server can easily interpret the semantics of the obtained data.

5 Evaluation and Discussion

We first show the contributions of the uRFID with an example in Fig. 8. A RF tag includes data, and its data can be interpreted and used by the two data processing systems in the same field. However, the third data processing system DPS-3 cannot understand its meanings. The existing RFID systems do not solve this issue.

In our system architecture, we can support two ways to achieve the issue. The first method is that the third data processing system initially defines its semantics. With the first method, the interpretation is automatically accomplished. The second is that the third system accesses to the metadata registry and understands the semantics of the RF tag. This method is processed manually or semi-automatically.

With the scenario example, Table 1 illustrates the qualitative evaluation result. Although this paper does not provide any simulations or experiments, we can intuitively understand the advantages of the proposed RFID architecture through the example above.

One of the critical problems of the previous RFID systems is that semantics are designed for a specific application domain. It causes a big problem that the semantic interoperability between tags and devices in various applications domains is not supported. Most of all, this limitation is not suitable for the ubiquitous computing environment.

The proposed RFID system architecture, uRFID considers the issue, and thus all devices can access and use data of tags from different application fields. In a word, the proposed architecture enables high usability of tag data. It means the uRFID supports application domain-independent semantic interoperability.

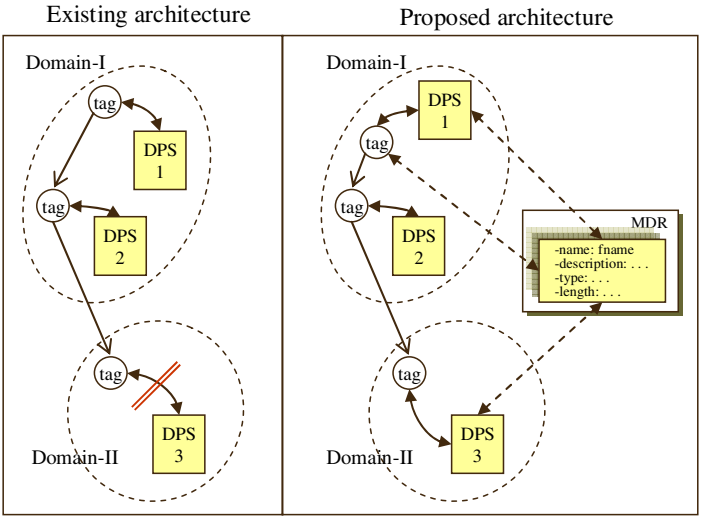


Fig. 7. An example for description of contributions

Table 1. Qualitative comparisons

Items	Existing Systems	uRFID
Dynamic management of semantics	N/A	Support
Usability of tag data	Low (Restricted)	High (Broad)
Exchanging between different applications	Complex mechanism (High cost)	Simple mechanism (Low cost)
Dependency to their own application	Dependent	Independent
Global semantic consistency	N/A	Support

6 Conclusion

Ubiquitous computing is increasingly recognized as a new computing paradigm. Many researches in various fields have been studying on applying and realizing the ubiquitous idea. One of the most famous and well-known applications is RFID and it has been implementing and applying to many real applications. However, previous RFID systems do not consider the consistent semantic management (interoperability) issue for the emerging ubiquitous computing environment. In other words, the existing RFID system architecture has several problems to apply to the ubiquitous computing environment: Semantic dependency on a specific application domain and static semantic management.

In this paper, we proposed a new noble RFID system architecture, uRFID. It is based on the metadata registry, and it is the international standard developed to enhance the interoperability of data. We first introduced the related technologies. The concept of the proposed RFID system architecture has been described, and also its main components have been illustrated. Finally, the processes of the uRFID have

been described, and then the qualitative comparisons with an example have been described.

The proposed system architecture provides several advantages. Most of all, the most important contribution is that the uRFID supports the application domain-independent semantic interoperability. It enables high tag data usability and the high quality application development under the ubiquitous environment.

As a further work, an access control mechanism on tag data should be defined and supported to improve the reliable usage. And the prototype implementation with an actual example is accomplished to explicitly show the contributions of the proposed system architecture.

References

1. Ilyas, M., and Mahgoub, I.: Handbood of Sensor Networks: Compact Wireless and Wired Sensing Systems, CRC Press, July (2004)
2. Finkenzeller, K.: RFID-Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, Wiley & Sons LTD, April (2003)
3. ISO/IEC JTC 1/SC 32, ISO/IEC 11179: Information Technology - Metadata Registries (MDR) - Part 1 ~ Part 6 (2004)
4. Jeong, D., In, H. P., Jarnjak, F., Kim, Y.-G., Baik, D.-K.: A Message Conversion System, XML-based Metadata Semantics Description Language, and Metadata Repository, Journal of Information Science (JIS), Vol. 31, No. 5, SAGE Publications (2005) 394-406
5. Jeong, D., Kim, Y.-G., In, H. P.: Quantitative Evaluation on the Query Modeling and System Integrating Cost of SQL/MDR, ETRI Journal, Vol. 27, No. 4, ETRI, August (2005) 67-376
6. Environmental Protection Agency (EPA), Environmental Data Registry (EDR) (2004) http://www.epa.gov/edr/
7. Australian Institute of Health and Welfare, Australian National Health Information Knowledgebase (2004) http://www.aihw.gov.au/
8. U.S. Department of Transportation, U.S. Intelligent Transportation System (ITS) (2004) http://www.dot.gov/
9. Environmental Protection Agency, Data Standards Publications and Guidances (2003) http://oaspub.epa.gov/edr/stddoc$document_type_vw.actionquery
10. Australian National Health Data Committee, National Health Data Dictionary (2003) http://www.aihw.gov.au/
11. ITS Architecture Development Team, ITS Logical Architecture—Vol. I, Vol. II: Process Specifications. Vol. III: Data Dictionary (2003) http://www.itsa.org/
12. Juels, A., Rivest, R. L., Szydlo, M.: The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy, The 8th ACM Conference on Computer and Communications Security, ACM Press (2003) 103-111
13. Sarma, S. E., Weis, S. A., Engels, D. W.: RFID Systems and Security and Privacy Implications, Lecture Notes in Computer Science, Vol. 2523, Springer-Verlag, Berlin Heidelberg New York (2002) 454-469
14. D Jeong, D., Kim, Y.-G., In, H. P.: New RFID System Architectures Supporting Situation Awareness under Ubiquitous Environments, Journal of Computer Science, Vol. 1, No. 2, Science Publications (2005) 114-120

15. Kim, J., Jeong, D., Baik, D.-K.: An Architecture for Semantic Integration between Data Elements in Sensor Networks, Korean Database Conference 2005, Seoul, Korea, Korea Information Science Society, May 20-21 (2005) 145-152
16. Powell, K.: Passive Radio Frequency Identification: Primer for New RF Regulations, November (2003) http://www.rfidjournal.com/whitepapers/0

Location-Based Services for Tourism Industry: An Empirical Study

Shuchih Ernest Chang[1], Ying-Jiun Hsieh[2,*], Chien-Wei Chen[3], Chun-Kuei Liao[3], and Shiau-Ting Wang[3]

[1] Institute of Electronic Commerce, National Chung Hsing University,
250 Kuo Kuang Road, Taichung City 402, Taiwan
eschang@dragon.nchu.edu.tw
[2] Graduate Institute of Technology and Innovation Management,
National Chung Hsing University, 250 Kuo Kuang Road, Taichung City 402, Taiwan
Tel.:+886 4 22840547; fax: +886 4 22859480
arborfish@dragon.nchu.edu.tw
[3] Institute of Electronic Commerce, National Chung Hsing University,
250 Kuo Kuang Road, Taichung City 402, Taiwan
ec@mail.nchu.edu.tw

Abstract. To cope with the fact of high competitiveness in tourism industry, some travel agencies are adopting the strategy of providing relatively more abundant and higher quality services to their clients. Since prior research has found that lack of information is one of the main barriers for people to travel, provide location-based information to people visiting and staying at a new location should be useful and valuable. This research attempts to find out travelers' attitude toward location-based service (LBS). "What factors will affect the travelers' willingness to adopt this new service" is what we tried to find out in this research. Our findings shows that perceived usefulness, perceived ease of use, and security & privacy are associated with travelers' attitude toward using LBS, but location information, context awareness, and device functions are not. Our findings can be referenced by tourism industry for the purpose of the design and development of successful business applications to catch the revolutionary opportunity and benefit of LBS.

1 Introduction

According to Taiwan Tourism Bureau, there were 2,428,297 foreigners visiting Taiwan in 2004, a 34.01% growth from 2003. Another interesting finding from the statistical data was that there were 2,038 travel agencies in Taiwan on October 31, 2005. The above facts reveal both vigorous development and high competitiveness in tourism industry. Lack of information is one of the main barriers for people to travel [1], and providing appropriate information for people visiting and staying at a new location should be useful and valuable, especially for travelers from overseas. With the advance in mobile technology, if mobile service providers know the end user's

* Corresponding author.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1144 – 1153, 2006.

exact location, providing useful information and location based services (LBS) at right place and right time can be beneficial to both businesses and their customers.

In this research, we attempt to find out travelers' attitude and demand on LBS, by investigating what factors will affect users' attitude toward adopting LBS. We believe that our findings can be referenced by tourism industry for the purpose of the design and development of successful business applications to catch the revolutionary opportunity and benefit of LBS. The remaining sections of this article are structured as follows. Section 2 reviews some prior researches about mobile commerce (m-commerce), LBS, location-based information that travelers need, technology acceptance model (TAM), and some security and privacy issues. Afterwards, Section 3 describes the applied methodology, including the hypotheses development, questionnaire design, and data collection. Section 4 shows our study result, and Section 5 concludes this paper after the discussions.

2 Research Backgrounds

2.1 Location-Based Service

Recently, LBS has been applied in various fields, including local traffic condition announcements, automatic route guidance based on onboard electronic maps and local traffic announcements, fleet management, etc. LBS can be defined as services that integrate a mobile device's location or position with other information so as to provide added value to users [2]. According to Kühn [3], a location based service is a service for mobile users (terminals) where the awareness of the current, past, or future location forms an integral part of the service. That is, we can define LBS as providing end users with right information at right time and right place by locating their mobile devices. The mobility is an added value to this new technology, which needs to service subscribers through multiple networks. Hence, in this research, we view location-based service as an application of mobile commerce (M-commerce) and information systems.

2.2 Location-Based Information That Travelers Need

Travelers may need several categories of location-based information about restaurants, accommodations, transportation, destination, traditional culture activities, security, and others [1]. In addition to travelers' information, e-coupon may serve as another feature that could be incorporated into the business design of offering LBS. As a potential marketing and advertising tool for promoting LBS business, E-coupon can help customers to acquire products or services at right places with attractive prices, induce more shoppers to visit targeted places or stores, and ultimately generate more revenues for LBS providers and related businesses.

2.3 Technology Acceptance Model

Users' attitudes towards the acceptance of a new information system have a critical impact on the success of information system adoption. The more accepting are the users for a new information system, the more willing they become to make changes in

their practices and spend their time and effort to actually use the new information system [4]. One of the most utilized models in studying information system acceptance, in terms of predicting user acceptance and usage behavior, is the technology acceptance model (TAM) [5]. The two specific beliefs that TAM uses to predict acceptance and actual system usage are perceived usefulness (PU) and perceived ease of use (PEOU). Other researches on PU and PEOU constructs also found strong support for these two determinants of technology usage [6]. To further strengthen the robustness of this model to the technology adoption process, Doll *et al.* [7] found that these two variables significantly contribute to adoption of technology even when there is limited exposure by the individual to the information system, and this finding supported Davis' original assertion that TAM could be used in pre-purchase decisions or during occasions when there was no application specific experience.

2.4 Mobile Commerce

Recently, many enterprises have started conducting business transaction via Internet. We may refer to this type of transactions and its related activities and operations as electronic commerce (EC). Through EC, companies can alleviate constraints (upon time, space, and cost) to enhance the way they connect to and interact with their EC counterparties by serving customers and collaborating with business partners electronically and intelligently. Instead of carrying transactions out via fixed and wired terminals in the traditional EC approach, the approach of mobile commerce (M-commerce) conducts EC activities via mobile and wireless devices such as mobile phone and personal digital assistant (PDA), so that the users can access M-commerce anytime, anywhere with their wireless devices. M-commerce can bring new opportunities, more profits, and a closer customer relationship for customer and companies. Some experts thought that M-commerce is a particularly promising field of business applications [8, 9]. In this research, we attempted to find out the potential benefits of serving LBS in the context of M-commerce for tourism industry.

2.5 Security and Privacy

Two important issues—security and privacy—are the topics in the growing number of discussion concerning the control of access to personal and sensitive information in information systems. Security can be defined as "providing the active or passive means to protect and preserve an environment which allows for conducting activities within the organization or society without disruption" [10]. The security of information serviced through mobile device is a key issue for the discussion of LBS. According to Westin [11], privacy can be defined as "the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is disclosed to others." Altman [12] stated that privacy is "the selective control of access to the self." Several studies show that many individuals are concerned about personal privacy. According to Robinson [13], privacy is the top remaining issue for LBS. In many countries there is legislation that requires consent for processing sensitive personal information.

3 Research Methodology

3.1 Research Framework

Based on literatures survey, a model for investigating the acceptance of LBS is developed. As shown in Figure1, our model consists of five factors postulated to evaluate the acceptance of LBS.

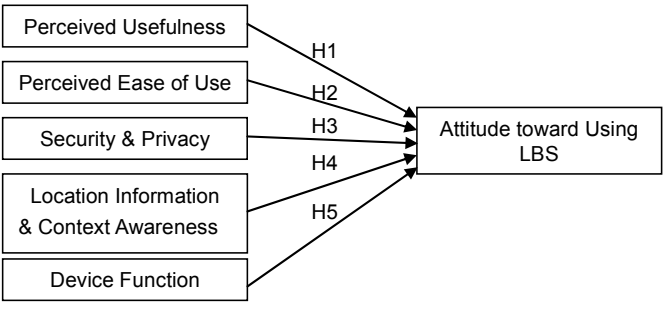


Fig. 1. Research Framework

Davis [5] defined PU as "the degree to which a person believes that using a particular system would enhance his or her job performance", and PEOU as "the degree to which a person believes that using a particular system would be free of effort". His research model of predicting acceptance and actual system usage has been widely used and adapted by numerous researches to show that an application perceived to be easier to use than another is more likely to be accepted by users. By applying these into LBS context, we postulate the following hypotheses:

H1: PU has a positive effect on consumer acceptance of LBS.
H2: PEOU has a positive effect on consumer acceptance of LBS.

Bisdikian *et al.* [14] stated that "people are justifiably concerned with the security of any information collected about them and the potential misuse of such information." Through previous researches conducted in a variety of online contexts, privacy has also been ranked as the top concern of Internet users [15, 16, 17]. With various surveys reporting large majorities of online users being concerned about security and privacy, we propose hypothesis 3.

H3: Security and Privacy have a positive effect on consumer acceptance of LBS.

Bisdikian *et al.* [14] stated that location information by itself is not sufficient. A broader notion of context needs to be considered. Context awareness provides substantial added value by enabling the delivery of desired content not only to the right place, but also at the right time and in the most appropriate manner, and it allows applications to infer user intention and to take proactive actions. Rao and Minakakis

[18] stated that LBS must deliver relevant, timely, and target information to consumers at the time and place for their choice. Competitive advantage will accrue to LBS providers who focus on superior customer experiences, distinctive, secure, and high-quality service. From this is hypothesis 4 derived.

H4: Location information and context awareness have a positive effect on consumer acceptance of LBS.

In addition to the factors described above, the accuracy, battery life, ability to penetrate buildings, weight, size, and function of the device are related to the consumer's comfort in using LBS [14, 15]. Therefore, hypothesis 5 is formulated.

H5: Device functions have a positive effect on consumer acceptance of LBS.

3.2 Questionnaire Design

According to the above hypotheses, the questionnaire is then designed as an instrument of data collection. We separate our questionnaire into four parts. The first part consists of demographic characteristics, including gender, age, education level, internet-using experience, on-line travel shopping experience, income level, and the purpose of last trip. The second part covers main categories of information that LBS may provide, such as the information about restaurant, hotel, transportation, scenic spot, cultural activity, police station and hospital, reply system, and the discount information of nearby shops. The third part includes the customer's acceptance toward portable device that has LBS, such as the dimension of perceived usefulness, perceived ease of use, security and privacy, and device function. Part 4 of the questionnaire focuses on the survey on users' attitude toward LBS. In total, the questionnaire consists of 25 items measuring the six variables as detailed in Table 1. Except for the demographic data in Part 1, questionnaire items are measured through five-point Likert scale, ranging "strongly disagree" (extremely unimportant) to "strongly agree" (extremely important). To ensure that the question items could be understood and measured validly, pre-test was conducted with small group. From the feedback derived from pre-test and the subsequent discussion with experts, the questionnaire was modified and refined.

3.3 Data Collection

We separated the sample into two groups. One focuses on online users who know how to use Internet and the other is aimed at random samples for those who have travel experiences. To collect those random samples, we distribute questionnaires at gas station, train station, college campus, and stock brokerage center. Out of 212 replied copies of questionnaire, 180 effective responses, including 100 on-line questionnaires and 80 face-to-face random samples, were collected. The effective response rate is about 85%.

Table1. Definition of the variables

Construct	Definition	Reference
Location Information & Context Awareness (LC)	Travelers may need several categories of location-based information about restaurants, accommodations, transportation, destination, traditional culture activities, security, etc.	[1]
Perceived Usefulness (PU)	The degree to which a traveler believes that using a particular system would enhance his or her job performance	[5]
Perceived Ease of Use (PEOU)	The degree to which a traveler believes that using a particular system would be free of effort	[5]
Security and Privacy (SP)	Travelers concern with the security of any information collected about them when they used LBS.	[19]
Device Functions (DF)	The functions of the portable device, e.g. accuracy, battery life, ability to penetrate buildings, weight, size, etc.	[14, 15]
Attitude toward using LBS (AT)	Indicate how strongly travel intend to perform different behavior	[20]

4 Results and Analysis

4.1 Descriptive Statistics

SPSS for Windows 10.0.7C was used in our study to analyze the sample. The formal questionnaire is used by confirmatory factor analysis to analyze collected data. Survey data were evaluated for their adequacy and construct validity, and the hypotheses were tested using correlation and regression analyses.

The characteristics of respondents were described using descriptive statistics. After a total of 212 responses were gathered, invalid survey results were identified by techniques such as the use of reverse questions. Overall, 180 valid questionnaires were collected and used for analysis. Among the 180 respondents, 41.7% were male, and 58.3% were female. The majorities (53.3%) were from 20 to 25 years old, and 45.6% of the 180 respondents had monthly incomes of less than NT$20,000. Most respondents (88.4%) had education level at college or above, and 111 out of the 180 respondents (61.7%) have more than 6 year experience in using Internet, but the majority (56.7%) of the respondents did not have any online shopping experience on travel website in one year. In terms of the purpose of their last trips, 86.1% respondents said that they traveled for leisure.

4.2 Data Analysis and Findings

The reliability of the questionnaire was checked using Cronbach's Coefficient (α) for all statement items. Unsuitable questionnaire items were deleted from the final version of the questionnaire, according to the guideline indicated by Nunnally and Bernstein [21], i.e., the value of 0.7 or above is an acceptable reliability coefficient. After our data were tested by factor analysis, validity and reliability test, we could get the completely standardized factor loadings that present the weight of the observed variables in Table 2.

Table 2. Standardized factor loadings and composite reliability estimates

Item	Measure	Factor loading	R^2	Composite reliability
LC1	LBS can provide nearby restaurants information.	0.670	0.5105	
LC2	LBS can provide nearby hotel information.	0.633	0.5355	
LC3	LBS can provide nearby transportation information.	0.498	0.5225	
LC4	LBS can provide nearby scenic spot/souvenir shop information.	0.549	0.4639	
LC5	LBS can provide nearby cultural activities information.	0.603	0.5283	
LC6	LBS can provide nearby police station/ hospital information.	0.752	0.5625	
LC7	LBS can provide the reply system. (ex: e-mail, telephone or FAQ information)	0.674	0.4495	
LC8	LBS can provide commodity discount information. (ex: e-coupon)	0.591	0.4291	0.7957
PU1	Using portable device enables me to utilize services more quickly.	0.752	0.7631	
PU2	Using portable device increase convenience in traveling.	0.818	0.7438	
PU3	Using portable device makes it easier to know services provided nearby.	0.742	0.7213	
PU4	Overall, portable device is useful for me to interact with the services.	0.695	0.6476	0.8668
PEOU1	Learning to use portable device is easy for me.	0.842	0.7554	
PEOU2	I find it easy to find what I need in using portable device.	0.628	0.6444	
PEOU3	My interaction with nearby services is clear and understandably.	0.465	0.4556	
PEOU4	It is easy for me to become skillful at using portable device.	0.869	0.7706	
PEOU5	Overall, I find LBS easy to use.	0.804	0.7411	0.8543
SP1	I trust in the ability of portable device to protect my privacy.	0.885	0.7381	
SP2	I trust in the technology a portable device is using.	0.697	0.6706	
SP3	I trust in a portable device as a service provider and collector.	0.710	0.6610	
SP4	I am not worried about the security of a portable device.	0.768	0.6214	0.8400
DF1	The weight and size of the portable device will affect my decision	0.808	0.5465	
DF2	The battery life of the portable device will affect my decision.	0.814	0.6149	
DF3	The signal strength of the portable device will affect my decision.	0.732	0.617	0.7642
AT1	After overall consideration, I like LBS.	1.000	1.000	1.0000

Using regression analysis, the results (shown in Figure 2) derived from this study confirm our hypotheses that Perceived Usefulness (PU), Perceived Ease of Use (PEOU), and Security & Privacy (SP) have positive influence on customers' attitude

toward using LBS. The results indicated that Perceived Usefulness (PU) has a significant effect toward users' attitude to use LBS (AT) (β=0.313, P<0.001), Perceived Ease of Use (PEOU) has a significant effect toward users' attitude to use LBS (AT) (β=0.366, P<0.001), and Security & Privacy (SP) also has a significant effect toward users' attitude to use LBS (AT) (β=0.238, P<0.01). As shown in Table 3, while H1, H2, and H3 were supported, the other two hypotheses, H4 and H5, are not supported by our empirical results, i.e., neither Location Information & Context Awareness (LC) nor Device Function (DF) has a significant effect toward customers' attitude toward using LBS (AT).

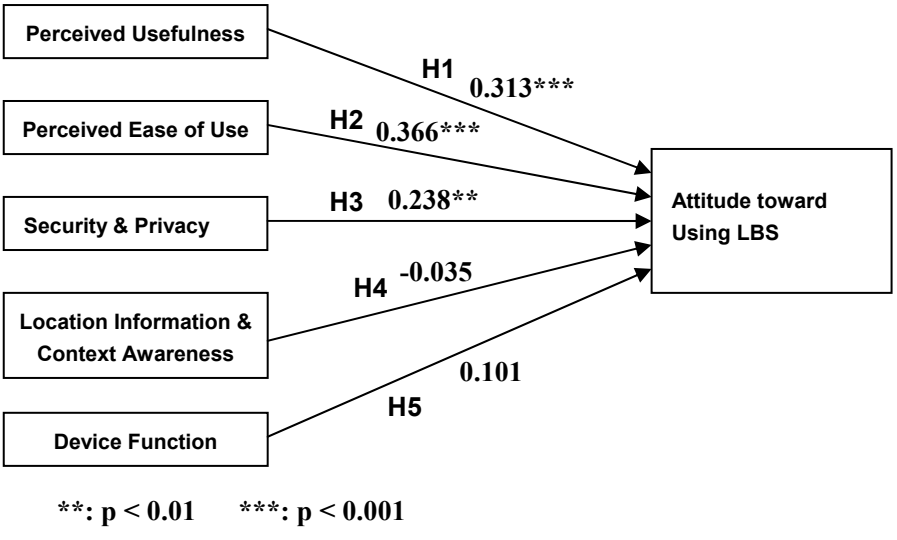


Fig. 2. The result of this empirical study

Nevertheless, the results of Location Information & Context Awareness (LC) and Device Functions (DF) are far away from our expectation. One possible reason for this result might be that travelers in Taiwan will prepare needed information related to their destination in advance; besides, they are quite familiar with this land. Therefore, they may not need the location information provided by LBS. For Device Function (DF), one possible explanation for the result might be the mature development in portable devices. Therefore, the weight, size, battery life, and signal strength are not important for their decision of using LBS.

Table 3. The result of hypothesis test

Hypothesis	β	p	Result
H1	**0.313**	**0.000**	**Supported**
H2	**0.366**	**0.000**	**Supported**
H3	**0.238**	**0.001**	**Supported**
H4	**-0.035**	**0.637**	**Not supported**
H5	**0.101**	**0.175**	**Not supported**

5 Discussions and Conclusions

Based on literature survey, this research formulated a user study model adapted from TAM to integrate security and privacy, location information as well as context awareness, and device functions into our research framework. Through this proposed model we investigated how these factors influence traveler's attitude toward using LBS. The research model together with five hypotheses was empirically tested using data collected from a survey of Taiwanese travelers. The validated model and its corresponding study results can be referenced by enterprise executives and decision makers to make favorable tactics for taking advantage of the opportunity available through LBS.

The results show that perceived usefulness, perceived ease of use, and security & privacy directly influence consumers' attitude toward using LBS. However, the location information & context awareness (H4) and device function (H5) were not significant through our study. Our findings suggest that travel agencies should put more focuses on consumers' security and privacy issues when releasing location-based service.

Location-based service is an important segment of M-commerce. This research studied customers' attitude toward location-based travel information service, which may suggest potential avenues for further work, both on technical and consumer behavior researches. However, the findings of this research are fully based on the selected sample, while different samples may have different results. In addition, in this paper, we assume this service will provide values to customers, but is it always true? The excess of information and the uselessness of information may need to be researched further. Besides, applying LBS concept to other industries and developing the framework of choosing the target customers of this service are also worth investigating. The restrictions of this research also include the time limitation and budget limitation, which all lead it to a narrowing down scope.

References

1. Huang, L., Tsai, H.-T.: The Study of Senior Traveller Behavior in Taiwan. Tourism Management 24(5) (2003) 561–574
2. Schiller, J., Voisard, A.: Location-Based Services. Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann (2004)
3. Kühn, P. J.: Location-Based Services in Mobile Communication Infrastructures. AEU - International Journal of Electronics and Communications 58(3) (2004) 159-164
4. Succi, M. J., Walter, Z. D.: Theory of User Acceptance of Information Technologies: An Examination of Health Care Professionals. Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS'99) (1999)
5. Davis, F.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly 13(3) (1989) 319-340
6. Chau, P. Y. K.: An Empirical Assessment of a Modified Technology Acceptance Model. Journal of Management Information Systems 13(2) (1996) 185-204
7. Doll, W. J., Hendrickson, A., Deng, X.: Using Davis's Perceived Usefulness and Ease-of-Use Instruments for Decision Making: A Confirmatory and Multi-Group Invariance Analysis. Decision Science 29(4) (1998) 839-869

8. Clarke I.: Emerging Value Propositions for M-Commerce. Journal of Business Strategies 18(2) (2001) 133-148
9. Buellingen F., Woerter M.: Development Perspectives, Firm Strategies and Applications in Mobile Commerce. Journal of Business Research 57 (2004) 1402– 1408
10. Post, R. S., Kingsbury, A. A.: Security Administration. 3rd edition. (1979) p. 14
11. Westin, A. F.: Privacy and Freedom. Athenaeum, New York (1967)
12. Altman, I.: The Environment and Social Behavior: Privacy, Personal Space, Territory, and Crowding. Brooks/Cole Publishing Company, Monterey, California (1975)
13. Robinson, T.: Location Is Everything. Internet Week Online, Tuesday September 12 2000 (2000). http://www.internetwk.com/lead/lead091200.htm
14. Bisdikian, C., Christensen, J., Davis, J., Ebling, M.R., Hunt, G., Jerome, W., Lei, H., Maes, S., Sow D.: Enabling Location-Based Applications. Proceedings of the 1st International Workshop on Mobile Commerce, ACM Press (2001) pp. 38-42
15. Koshima, H., Hoshen J.: Personal Locator Services Emerge. IEEE Spectrum, 37(2) (2000) 41-48
16. Snekkenes, E.: Concepts for Personal Location Privacy Policies. Proceedings of the 3rd ACM Conference on Electronic Commerce, ACM Press (2001) pp. 48-57
17. Minch, R. P.: Privacy Issues in Location-Aware Mobile Devices. Proceedings of the 37th Hawaii International Conference on System Sciences (HICSS'04) (2004) pp. 127-136
18. Rao, B., Minakakis, L.: Evolution of Mobile Location-Based Services. Communications of the ACM 46(12) (2003) 61-65
19. Dholakia R. R., Dholakia N.: Mobility and Markets: Emerging Outlines of M-commerce. Journal of Business Research 57 (2004) 1391– 1396
20. Ajzen, I., Fishbein, M.: Understanding Attitudes and Predicting Social Behavior. Prentice-Hall, Englewood Cliffs, NJ (1980)
21. Nunnally, J.C., Bernstein, I.H.: Psychometric Theory. McGraw-Hill, New York, NY (1994)

Towards Affective Collages of Presences

Jesús Ibáñez, David García, Oscar Serrano, Josep Blat, and Raquel Navarro

Department of Technology,
University Pompeu Fabra, Barcelona, Spain
{jesus.ibanez, david.garcian, oscar.serrano, josep.blat,
raquel.navarro}@upf.edu

Abstract. This paper describes ongoing work aimed to explore how to augment a person's work environment with information which enables her to feel the presence of intimate companions. The vehicle we deem to be appropriate for this situation is indirect communication (in some ways related to peripheral communication). The presence we intend is based on the activities of these intimate people. In short, with a certain periodicity the user is presented, on a peripheral user interface (windows desktop or digital picture frame), with a new collage composed of pictures indirectly triggered by their loved ones. In particular the pictures are triggered by the text they write and read, and the songs they listen to, while working on their PCs.

1 Contextualisation

In our globalised world many families, partners and friends live far apart. This socio-cultural phenomenon is becoming a daily reality for an increasing number of people. Technology has already helped immensely, not only by reducing the cost and increasing the quality of communication, as for example via msn, voice over IP et al., but also by creating new ways of establishing relationships. New languages evolve (mobile jargon) and new ways of expressing emotions (emoticons & smilies) can develop while users adapt and play with new technology. Technology has, however, not yet assisted in overcoming physical distance due to remote residence and hours of separation due to long working weeks.

Nowadays, many people spend long hours working alone in front of their PC. We frequently feel lonely, while some of our loved ones, working in the same city or farther away, feel lonely as well. Probably, the feeling of loneliness could be overcome if we could feel their presence.

Meanwhile, the digital world is overlapping the real world more and more, nurturing the emergence of new ways for distant people to communicate. However, established human communications through telecommunications systems are attention demanding. When two remotely located people communicate they have to focus their attention on the communication process. While using synchronous communication systems (phone, videoconference, Internet chat, etc) they concentrate on the conversation. While employing asynchronous systems (email, sms, etc) they concentrates on writing/reading messages.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1154–1163, 2006.

Furthermore, established human communication through telecommunications systems is episodic. Although chatting on the phone provides a strong feeling of connection with our interlocutor, for most of the remaining time we do not feel connected. There are, however, simple ways for keeping this connection. For example, some kindergartens publish, on the Internet, live images of the children taken with webcams. The parents of these children can keep an open window on the desktop of their PCs while they are working. In this way they can focus on their work, and at the same time feel the presence of their children. This may be a good solution for this case, which relates to children, but it is not applicable for adults. In general we prefer to maintain our privacy, and so we would prefer not to deliver live images of our daily life through the Internet. We do not like to feel observed by others who we cannot see. Furthermore, the live images of the children may not be attention demanding, but they still are distracting. Research work on the peripheral display area is intending to find solutions to these problems.

To avoid the problem of privacy and distraction we could employ indirect information. Rather than transmitting information captured directly from remote people (or directly triggered by them), we can work with information which is indirectly influenced by their actions.

In this sense, we have recently designed and developed Musimage (see figure 1), a novel visual interface which displays pictures according to the songs being played at the time [6]. Music triggers recollections. Listening to a particular song, we remember events that happened and feelings that we felt while listening to that song in the past. Our original idea was to design a user interface which on the one hand accompanies the user in this recollection process, and on the other hand is able to "illustrate" the song. By using the interface, the user selects the songs to be played, but the pictures are chosen automatically. For each song to be played, the system selects a set of pictures, according to various criteria corresponding to certain features of the song (namely lyrics and year). Both the topic and mood of a song are automatically extracted from its lyrics. These semantic features (topic, mood and year) are then used as cues for collecting pictures from both the user's personal picture collection and picture servers on the Internet. In this way the sequence of pictures is induced by user's actions (selection of songs), and the pictures themselves constitute indirect information about the user's actions.

In this document we set out to explore the use of this kind of indirect information as a means of reinforcing the feeling of being accompanied. In order to carry out this exploration we are developing an application, whose first prototype is detailed below. In short, with a certain periodicity the user is presented, on a peripheral user interface (windows desktop or digital picture frame), with a new collage composed by pictures indirectly triggered by their loved ones. In particular the pictures are triggered by the text they write and read, and some features (author, title, lyrics and year) of the songs they listen to, while working on their PCs.

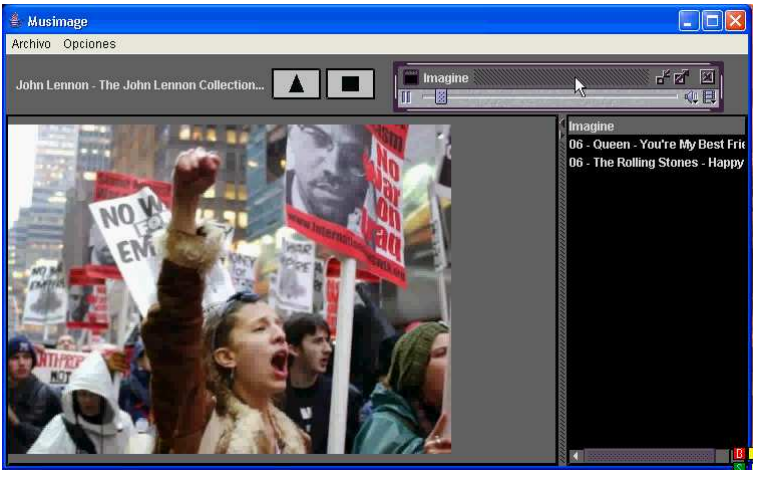


Fig. 1. Snapshot of Musimage

2 Related Work

In this section we survey work related to our proposal. In particular we review work on displays that sit on the periphery of a user's attention. Many public information displays which we encounter are usually in this category, including clocks, posters, and windows. Computationally enhanced variations on this theme are called peripheral displays. For example, one of the first such displays, created by an artist and technologist in collaboration, was a "dangling string" attached to a motor [18]. The string spun around at different speeds depending on network load.

Peripheral displays have the peculiar property that they are not meant to be the focus of a user's attention. They can be divided into two categories: ambient and alerting. Ambient displays are a subclass of peripheral displays that present information in such a way as to allow people to monitor a data source while not being distracted from their main task. They are continuous displays of information, and are often abstract, aesthetic, and non-disruptive. Alerting displays alert a user about salient information through more direct means. However, the division between these two categories is not always clear cut. Many alerting displays include an ambient component when they are not actively alerting the user, and similarly, an ambient display may at times alert a user about something.

The user interface of the system we propose is a peripheral display which is, more specifically, an ambient display.

A current trend is to explore symbolic representations [14] of captured activity data rather than just showing full video and audio. Symbolic representations are characterized by a certain intermediacy with respect to the events being represented, and reading the representations requires an interpretative effort by

the user. In our case, there is no direct mapping between the activities of loved ones and their symbolic representations. In our system the pictures are collected from the Internet, based on a textual simplification of the user activity. Thus, rather than symbolic representations, we employ indirect information.

Much effort has focused on designing peripheral displays in recent years. Many of the designed systems do not use screens. Some of them employ projectors [20] [16], while others are embedded in augmented objects [19][15] or in an architectural space [4] [11].

We, however, are particularly interested in peripheral displays on screens. Some well known peripheral displays of this kind are InfoCanvas [12], Informative Art [7], SideShow [5] and What's Happening [21].

InfoCanvas and Informative Art display information as an abstract representation. InfoCanvas uses eye-pleasing scenes such as a cartoon-like beach landscape in which certain elements convey information: the color of a woman's bathing suit may represent current traffic conditions, or the altitude of a bird may indicate a particular stock's activity. Informative Art, on the other hand, mimics famous paintings, subtly changing certain elements of the composition to convey awareness information. For instance, one such display borrows the composition of a Piet Mondrian painting to display the current weather in six different cities. By using abstract representations, InfoCanvas and Informative Art can securely display sensitive information such as stock activity.

Literal, iconic representation of information have been explored with SideShow [5] and What's Happening [21]. Whereas InfoCanvas and Informative Art both use separate LCD monitors, SlideShow and What's Happening try to make unobtrusive use of valuable main monitor screen real estate. Both of these approaches display information literally as either recognizable icons, pictures, or text.

Sideshow is a peripheral awareness interface that provides regularly updated peripheral awareness of a broad range of information from accessible web sites or databases. Sideshow provides awareness visually via a sidebar on one's primary display that cannot be covered by other applications. When a person installs Sideshow, the bar appears on the right side of the Windows desktop. The sidebar is filled with a variety of items called "tickets," each of which displays a small summary of information. For example, the ticket pointing to one's Outlook calendar shows how long one has until the next meeting, as well as the first few words from the meeting title. The ticket pointing to a local camera showing traffic conditions shows a small, static image from the camera. The goal of the tickets in the sidebar is to provide a relatively high-level summary of information in a small space. If users decide they want to find out more information about a particular item, they can hover their mouse over a ticket and a large tooltip window appears next to the ticket.

Our proposal, like Sideshow, is aimed to achieve peripheral awareness. However, while Sideshow has been applied to improve working environments by facilitating coordination among co-workers, our proposal is aimed at exploring the

use of indirect information as a way of reinforcing the affective feeling of being accompanied by our loved ones. In this sense, our proposal and Sideshow are complementary.

What's Happening (WH for short), is a set of two systems designed to help promote awareness of information and activities in a local community. The tools seek to do this in an unobtrusive, peripheral manner. The first tool is the WH Communication-Bar, a small corner-of-the-display interface that cycles through local community and general interest information blurbs that it has gathered. The second tool is the WH Screen-Saver that shows graphics and text excerpts from pages on web sites in the community.

The WH Communication-Bar's user interface has a small footprint on a person's computer display, and is often suitably placed in the corner. It is designed to remain visible and not be obscured by other windows. The system shows short "blurbs" of automatically collected local content such as official announcements and community events, as well as external ones such as news reports and weather forecasts. Users can contribute content either by posting new stories or by "following up" on existing content in the built-in chat rooms. Blurbs are shown one at a time, in a cycle. The WH Screen-saver is an screen-saver which shows collages composed by pictures and texts collected from the set of local web pages about people's research interests, hobbies, travels, family, etc. Thus, WH is aimed at providing local information (in particular announcements, discussions and information from the web pages of the community members) which provides improved opportunities for people to learn about each other.

Our proposal, as WH, utilises collages of pictures as a peripheral display. However, while in WH the collages are composed by explicit pictures from the web pages of the community members (explicit information showing static states), in our proposal the collages are composed by pictures collected from the Internet according to the current activity of our loved ones (indirect information showing dynamic states).

3 Our Proposal

In this section we describe the concrete application we are developing in order to validate our ideas. We start by describing the first prototype developed, and then move on to outline some important considerations relating to this research.

In describing the overall functioning of the prototype we provide some diagrams to support the explanation. For this description, we assume a scenario where a user wishes to feel the presence of 3 loved ones. Figure 2 shows the processes on the PC of each of these 3 loved ones.

While the loved one is working on the PC, data from the following sources of information are transparently collected:

- the text written by the user
- the text read by the user (the text in active window)
- some features (author, title, lyrics and year) of the mp3 song being played on the PC

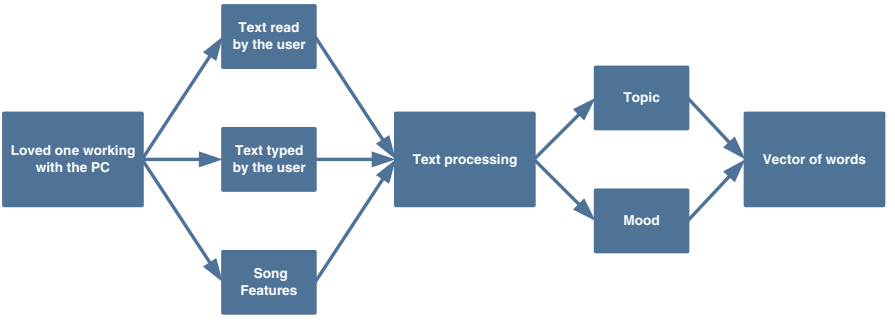


Fig. 2. Processes on the PC of each loved one

Note that the written and read text, as well as most of the features of the song being played (in particular author, title and year) can be directly captured on the PC itself. These song features can be obtained from the ID3 of the mp3 song. However, the lyrics of the song should be obtained from a lyrics server on the Internet. In theory the lyrics could be obtained from the ID3 as well, as ID3v2 includes a lyrics feature, but this feature of ID3 is seldom annotated in the mp3 files. Note that the format of all these data is text.

In the current prototype, the text written and read by the user is captured by employing the MSAA(Microsoft Active Accessibility) technology [3]. The system builds a tree of data reflecting both the structure of the original captured data and the temporal order and length (depending on the time frame the user wrote/read this data). Heuristic rules are also applied in this process. On the other hand, the ID3 of the mp3 song being played is obtained by using the Java MP3 Tag Library [2].

With a certain periodicity, a vector of words is automatically constructed from the data collected during that period. This vector of words reflects the topic and mood induced from the user activity during that period. Thus, finally the collected information is reduced to a vector of words.

In [6] we used statistical methods to extract topic and mood from the lyrics of the songs. In this new proposal we are employing ConceptNet [9] as a tool to extract the topic and mood from the text captured on the loved ones' PCs. ConceptNet is a semantic resource that is structurally similar to WordNet, but whose scope of contents is general world knowledge in the same vein as Cyc. ConceptNet is generated automatically from the English sentences of the Open Mind Common Sense (OMCS) corpus [17], and it is integrated with the MontyLingua natural-language-processing engine [10]. MontyLingua is an end-to-end integrated natural-language-understander for English.

The ConceptNet tool-kit supports various contextual commonsense-reasoning tasks. At present, three node-level functionalities are implemented, context-finding, analogy-making, and projection, as well as four document-level functions, topic-gisting, disambiguation and classification, novel-concept identification,

and affect sensing. The *topic-gisting* and *affect sensing* functionalities are relevant for us. Thus, we briefly describe these characteristics.

Using MontyLingua, a document is gisted into a sequence of verb-subject-object-object (VSOO) frames. Minor transformations are applied to each VSOO frame to massage concepts into a ConceptNet-compatible format. These concepts are heuristically assigned saliency weights based on lightweight syntactic cues, and their weighted contextual intersection is computed. ConceptNet used in this way serves as a nave topic spotter.

ConceptNet performs textual affect sensing over a document by employing an algorithm which is a simplification of Liu et al's Emotus Ponens system [8]. Its technical workings are quite easily described. Consider that a small subset of the concepts in ConceptNet are first affectively classified into one of six affect categories (happy, sad, angry, fearful, disgusted, surprised). The affect of any unclassified concept can be assessed by finding all the paths which lead to each of these six affectively known categories, and then judging the strength and frequency of each set of paths.

Figure 3 shows the complete scenario (a user being presented with a collage composed of pictures triggered by the actions of three loved ones). In this scenario, the vectors of words corresponding to the three loved ones are collected and employed as keywords to search for pictures in picture search engines on the Internet.

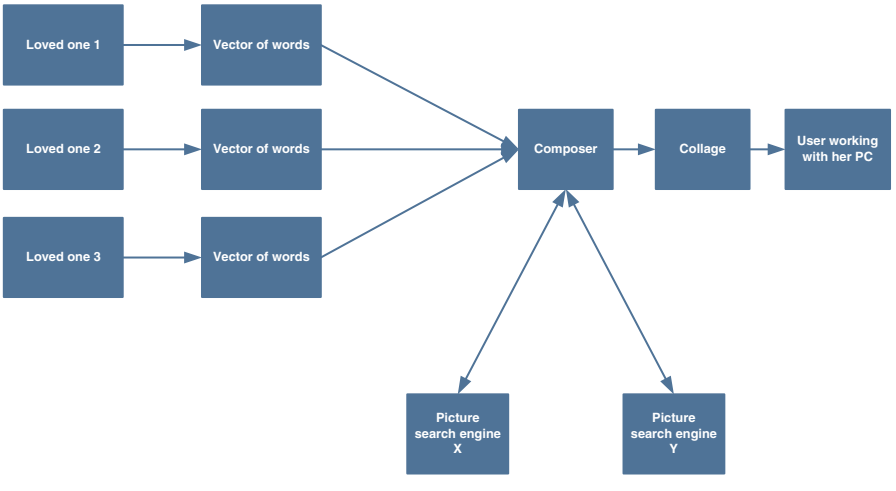


Fig. 3. The complete scenario

An automatic composer creates a new collage using the recently retrieved pictures. The collage consists of a simple matrix of $m \times n$ images (see figure 4). The collage is then presented to the user on a peripheral user interface (the

Fig. 4. Example of generated collage

windows desktop in this case). The actions to be carried out if the user presses on any of the pictures have not yet been determined, and will be defined through user studies.

As to the distributed infrastructure of the prototype, we first considered the multi-agent system approach. In general, designing the application as a multi-agent system facilitates the separation of different functionalities into different components (agents). It also facilitates the addition of new agents providing new functionalities.

We already designed and developed a multi-agent system in [6], on top of the JADE (Java Agent DEvelopment Framework) platform [1]. The multi-agent approach proved to be useful and extensible in that case. It was really appropriate for that application ([6]). However, the kind of system we propose in this paper fits even better the Peer-to-Peer (P2P) philosophy. Moreover, the P2P approach and technologies seems to be quite mature. In fact, the past years have seen a revolution in the P2P research community with the introduction of structured P2P overlay networks, which offer an efficient, scalable, fault-resilient, and self-organizing substrate for building distributed applications. Thus, after evaluating the state of the art in P2P systems, we decided to base our prototype on Dermi (Decentralized Event Remote Method Invocation) [13], which is inspired by several applications that have emerged as a result of these structured P2P substrates. Dermi is a completely decentralized event-based object middleware built on top of a structured P2P overlay. Its primary objective is to provide developers the necessary abstractions to develop wide-area-scale distributed applications. Dermi uses a P2P publish-subscribe event system and offers several services to the application layer: P2P call abstractions, a decentralized way to locate objects, and a distributed interception service.

Next we list important considerations about this research:

- For simplicity we only work with text sources of data. However, it should be noted that the system could be extended, in the future, to work with other media. For example, the mood could be extracted from the sound of the songs which the user listens to.
- For simplicity we concentrate on the case of people working with a PC. The potential for extending this technique to mobile/pda devices is clear.

– The functioning of the system is transparent for users (that is to say, it does not require them to carry out any special actions such as annotating media or interacting with a specific device).

4 Conclusions and Future Work

In this paper we have described ongoing work aimed to explore how to augment a person's work environment with information which enables her to feel the presence of intimate companions. The vehicle we deem to be appropriate for this situation is indirect communication (in some ways related to peripheral communication). The presence we intend is based on the activities of these intimate people.

We have described a particular application. In short, with a certain periodicity the user is presented, on a peripheral user interface (windows desktop or digital picture frame), with a new collage composed of pictures indirectly triggered by their loved ones. In particular the pictures are triggered by the text they write and read, and the songs they listen to, while working on their PCs.

The main issues we intend to explore by employing this application are whether the mechanisms we suggest do indeed convey a feeling of an active presence of the relationship; and which interactions are appropriate for this type of environment. Some issues to investigate with users, among others, are: Do users feel, at some degree, the presence of the loved ones by using the system? If so, to what degree? Do users feel more accompanied when using the system? Does the collage effectively evoke the related users who triggered the pictures? What kind of interaction would users expect from the prototype? Do users allow their loved ones to receive indirect information triggered by their actions or do they still feel their privacy is invaded?

Acknowledgements

We would frankly like to thank Leticia Lipp and Dai Griffiths for generously proofreading. This work has been partially funded by both the European Union IST program (through the project "ICING: Intelligent Cities for the Next Generation") and the Spanish Ministry of Science and Technology (via the project "PLANET: Plataforma de colaboracin aumentada para el acceso y distribucin de contenidos educativos", TIC-2003-09288-C02-00).

References

1. *Java agent development framework*, Available at http://jade.tilab.com/.
2. *Java mp3 tag library*, Available at http://javamusictag.sourceforge.net/.
3. *Microsoft active accessibility*, Available at http://msdn.microsoft.com/library/en-us/msaa/msaastart_9w2t.asp.
4. S. Antifakos and B. Schiele, *Laughinglily: Using a flower as a real world information display*, UbiComp, 2003.

5. J. J. Cadiz, G. D. Venoliaa, G. Jancke, and A. Gupta, *Designing and deploying an information awareness interface*, CSCW, November 2002, pp. 314–323.
6. David Garcia, *Multi-agent system for automatic presentation of multimedia elements depending on the context*, Master's thesis, University Pompeu Fabra, Barcelona, Spain, 2005.
7. L. E. Holmquist and T. Skog, *Informative art: Information visualization in everyday environments*, 1st international conference on Computer graphics and interactive techniques in Austalasia and South East Asia (Melbourne, Australia), 2003, pp. 229–235.
8. H. Liu, H. Lieberman, and T. Selker, *A model of textual affect sensing using real-world knowledge*, IUI 2003 (Miami, Florida), 2003.
9. H. Liu and P Singh, *Conceptnet: A practical commonsense reasoning toolkit*, BT Technology Journal **22** (2004), no. 4, 211–226.
10. Hugo Liu, *Montylingua v1.3.1, toolkit and api*, 2003, Available from http://web.media.mit.edu/ hugo/montylingua/.
11. S. Marti and D. Seetharam, *Weathertank: Interface for non-literate communities and ambient visualization tool*, MIT Media Lab, 2001.
12. Todd Miller and John Stasko, *Artistically conveying peripheral information with the infocanvas*, Technical Report GIT-GVU-02-11, Graphics, Visualization, and Usability Center, Georgia Institute of Technology, Atlanta, GA, June 2002.
13. Carles Pairot, Pedro Garcia, and Antonio F. Gomez Skarmeta, *Dermi: A new distributed hash table-based middleware framework*, IEEE Internet Computing **8** (2004), no. 3, 74–84.
14. Elin Ronby Pedersen, *People presence or room activity supporting peripheral awareness over distance*, Conference on Human Factors in Computing Systems (Los Angeles, California, United States), 1998, pp. 283–284.
15. Thorsten Prante, Carsten Rcker, Norbert Streitz, Richard Stenzel, Carsten Magerkurth, Daniel van Alphen, and Daniela Plewe, *Hello.wall - beyond ambient displays*, UbiComp, 2003.
16. R. Rodenstein, *Employing the periphery: The window as interface*, CHI, 1999.
17. P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu, *Open mind commonsense: knowledge acquisition from the general public*, First International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems, Lecture Notes in Computer Science, vol. 2519, Springer, 2002.
18. Mark Weiser and John Seely Brown, *Designing calm technology*, PowerGrid Journal **1** (1996), no. 1.
19. Craig Wisneski, Hiroshi Ishii, and Andrew Dahley, *Ambient displays: Turning architectural space into an interface between people and digital information*, First International Workshop on Cooperative Buildings (CoBuild), Springer, 1998.
20. Jim Youll and Dana Spiegel, *Ambient dayplanner: A tangible interface for public and private appointment calendars*, MIT Media Lab, 1999.
21. Q. A. Zhao and J. T. Stasko, *What's happening?: Promoting community awareness through opportunistic, peripheral interfaces*, Working Conference on Advanced Visual Interfaces (AVI) (Trento, Italy), 2002, pp. 69–74.

Automatic Trap Detection of Ubiquitous Learning on SCORM Sequencing

Chun-Chia Wang[1], H.W. Lin[2], Timothy K. Shih[2], and Wonjun Lee[3]

[1] Department of Information Management
Northern Taiwan Institute of Science and Technology, Peitou, Taipei, Taiwan, R.O.C.
[2] Department of Computer Science and Information Engineering
Tamkang University, Tamsui, Taiwan, R.O.C.
892190082@s92.tku.edu.tw
[3] Department of Computer Science and Engineering
Korea University, Seoul, Republic of Korea

Abstract. In order to adapt the teaching in accordance to individual students' abilities in the distance learning environment, more research emphasis on constructing personalized courseware. The new version of SCORM 1.3 attempts to add the sequence concept into this course standard. The sequencing describes how the sequencing process is invoked, what occurs during the sequencing process and the potential outputs of the sequencing process. However, the related research of sequence trap is lack. Sequence trap results from improper sequence composing. The more complex course is the higher trap-probability arises. When the sequence trap occurs, it will block any learning activities and cannot go on any course object. As a result, we apply the valuable features of Petri net to decrease the complexity of the sequencing definition model in the SCORM 1.3 specification and process the input sequencing information to detect the sequencing trap in advance.

1 Introduction

Distance learning provides a flexible environment to learn anytime and anywhere. Moreover, it provides a solution to solve the problem of traditional education such as personalize learning. More and more learner with the different background, various learning needs and diverse learning styles take part in the same course. Relatively, they need the course "their own"- at their pace, in their demand. As a result, SCORM (Shareable Content Object Reference Model) [1], a conventional model adopted in distance learning, adopts IMS Simple Sequence Specification [2] to describe how sequencing behaviors are applied to tracked individual learner progress [9].

SCORM compatible course is described in a tree-like structure. The structure defines clusters which are considered as the basic building block of learning activity. A cluster includes a single parent activity and its immediate children but not the descendants of its children. SCORM sequencing rules in the SCORM Sequencing Definition Model are especially applied to clusters. The parent activity of the cluster will contain the learning sequencing information. The child activity in the cluster will have its associated learning content and objects. Through the composing the elements of SCORM Sequencing Definition Model, each course content has its ability to handle the different learning behaviors.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1164 – 1173, 2006.

However, SCORM Sequencing Definition Model, used in modeling various sequencing behavior, is too complex to implement. To decrease the designing cost, there are some authoring tools [3, 4] can edit the sequencing rules by clicking in the comboBox of sequencing rules. Moreover, [5] proposed a SCORM sequence template to support sequence rules design. After editing the course sequence, designers can save their sequence template for reloading and reusing next time. Although those authoring tools offer the graphical user interface (GUI) to create SCORM course, the sequence of final course is hard to image. Some sequencing behaviors may not be intuitive, so content developers should focus on not only SCO itself but interactive relationship of each SCO. In [6, 7], Petri Net is applied to easily image the final course. They provide an efficient way to decrease the complexity on designing course sequence. However, how can validate whether the complex composition can result in effective course is important but lacks discussion. We consider not only designing course efficiently but tracing sequence effectively. As a result, trap preventing and detecting methods are proposed to avoid the sequence trap occurring.

Section 2 first describes the causes and characteristics of Sequencing Trap, and then illustrates the Sequence Trap with improper sequence composing. Section 3 propose our Petri Net Model and applied on SCORM Sequence. Continually, Section 4 expounds the trap preventing and detecting. Finally, conclusions are drawn in Section 5.

2 Sequencing Trap

2.1 Sequencing Trap Definition

When a learner is learning some courses, the following irregular situations may occur.

- Learning attempt is stuck in a certain activity: Even though the learning activity is completed, the learners are still blocked in that activity and cannot go on any other learning activity.
- Learning attempt is fallen into a vicious learning circle: The learning path is blocked in a certain loop (cluster) of sub activity tree.
- Deserved activities are not identified for delivery: Sequencing rules are not fired or LMS launch error activities.

We call the above situations that hinder the learning as Sequencing Trap. Learning disability of learner may results in Sequencing Trap. If some regular rules defined by designer are not satisfied, it is reasonable that the learner should learn the material over and over. How to make the rules appropriately to avoid the Sequencing Trap is an important topic of instruction research but not the focus in this paper. The other main cause of Sequencing Trap is improper sequencing setting during course designing. As mentioned in Section 1, many elements grouped into 11 categories in the Sequencing Definition Model apply specifically to clusters. The definition model elements are applied to learning activities within the context and reciprocal effect of an Activity Tree. It is much difficult to make sure the sequence composing will not lead to Sequencing Trap but result in what designers want.

Because a cluster is defined as a basic building block and learning sequencing strategy are applied to it, the problem is reduced to the simplest state, a single cluster, to prevent and detect Sequencing Trap under Sequencing Control Mode.

2.2 Sequencing Trap Within Sequencing Control Mode

In SCORM Sequencing and Navigation specification [9], all elements can divide into 11 categories. One of categories, Sequencing Control Mode, is basic elements to control the sequencing behavior for a cluster. There are six elements defined in Sequencing Control Modes. Except two elements for Tracking Model, Use Current Attempt Objective Information and Use Current Attempt Progress Information, all in Sequencing Control Mode affects the learner's learning sequence directly.

To deserve to be mentioned, Sequencing Control Choice Exit can be applied to any activity in the Activity Tree, however the Sequencing Control Choice, Sequencing Control Flow and Sequencing Control Forward Only modes will have no effect if applied to leaf activities. Besides, Sequencing Control Modes can be enabled simultaneously to show the various behaviors by creating combinations of control mode. To probe into improper sequencing composing results in Sequencing Trap, Table 1 lists all possible composing within Sequencing Control Mode.

Table 1. All possible elements composing within Sequencing Control Mode and the result of Sequencing Trap

Apply to	Parent Activity			Child(Active) Activity	Result
Case	Flow	Forward Only	Choice	Choice Exit	
1	*False*	*False*	*False*	*False*	*Sequencing Trap on Parent Activity*
2	*False*	*False*	*False*	*True*	*Sequencing Trap on Parent Activity*
3	*False*	*False*	*True*	*False*	*Sequencing Trap on Active Activity of Children*
4	*False*	*False*	*True*	*True*	*No Sequencing Trap*
5	*False*	*True*	*False*	*False*	*Sequencing Trap on Parent Activity*
6	*False*	*True*	*False*	*True*	*Sequencing Trap on Parent Activity*
7	*False*	*True*	*True*	*False*	*Sequencing Trap on Active Activity of Children*
8	*False*	*True*	*True*	*True*	*No Sequencing Trap*
9	*True*	*False*	*False*	*False*	*No Sequencing Trap*
10	*True*	*False*	*False*	*True*	*No Sequencing Trap*
11	*True*	*False*	*True*	*False*	*No Sequencing Trap*
12	*True*	*False*	*True*	*True*	*No Sequencing Trap*
13	*True*	*True*	*True*	*False*	*No Sequencing Trap*
14	*True*	*True*	*True*	*True*	*No Sequencing Trap*
15	*True*	*True*	*False*	*False*	*No Sequencing Trap*
16	*True*	*True*	*False*	*True*	*No Sequencing Trap*

Table 1 reveals two kinds of Sequencing Trap that learning attempt is stuck in Parent Activity and in Active Activity of Children. In the one hand, a Parent Activity has Sequencing Control Flow and Sequencing Control Choice set to false (shown in Figure 1), so LMS will disable the Continue and Previous navigation requests and all children will not be valid targets for learners to free choose neither. Learners will be stuck in Parent

Activity and Sequencing Trap occurs. Case 1, 2 5 and 6 in Table 1 can be illustrated by Figure 1(a).

In the other hand, learners will be stuck in active Activity of children when there are no ways to terminate itself or trigger other sibling. Once Parent Activity has Sequencing Control Choice set to truth but Sequencing Control Flow set to false, the child activities are all valid targets for learners to choose but LMS will not provide any mechanism for the learner to indicate their desire to "Continue" to the next activity or to go back to a "Previous" activity. When a learner's attempt is on Activity 2(shown in Figure 1(b)), the parent activity of Activity 2 has Sequencing Control Choice defined as true so every sibling of Activity 2 is a valid target for a Choice navigation request. Allow one of Activity 1 or Activity 3 to be identified for delivery would result in Activity 2 terminating, violating the intention of Choice Exit control. At the moment, the learner can neither trigger Flow nor Choice navigation requests, so there is no activity could be identified for delivery and Sequencing Trap occurs.

In [8], we had proposed a Sequence testing function that extends the recursive method from a single cluster to a whole cluster tree by using bottom-up method. However, this Sequence testing function is based on truth table as shown in Table 1. When Sequencing definition model implement completely, the truth table including all composing will grow exponentially. As a result, the Sequence testing function has limitation on Expansibility and Flexibility.

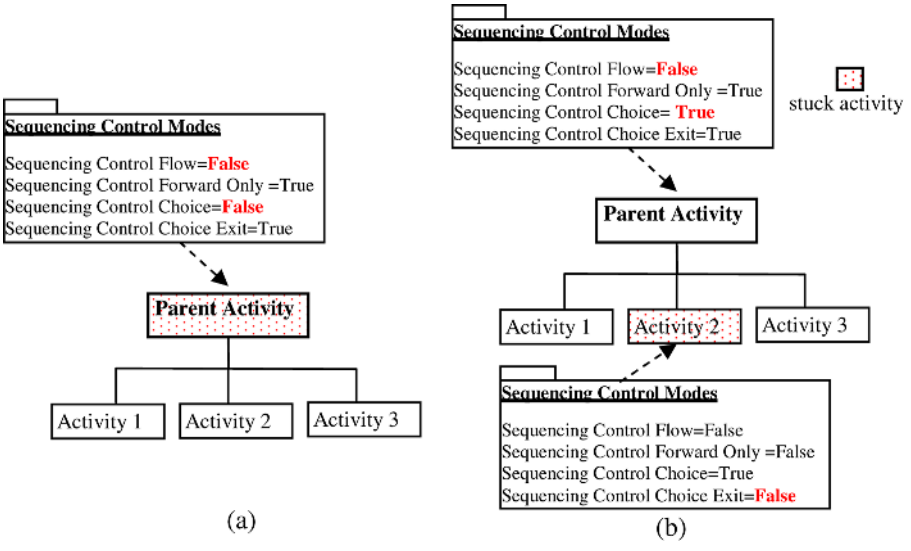


Fig. 1. An example of Sequencing Trap (a) on Parent Activity (b) on active Activity

3 Petri Net Model

Based on [6], the Petri Net model is extended to have the ability of preventing and detecting Sequencing Trap. There are three main causes why this paper adopts Petri net to mapping SCORM Sequence as below:

- **Graphic Characteristic:** It is too complex to trace any relative situation for content developer in the sequencing definition model. The graphic characteristic of Petri net supports the visible sequencing flow for designers and learners.
- **Formal Analysis:** Petri net is a famous process modeling technique with precise definition to support the dominant factor of workflow management in e-Learning environment.
- **Expansibility:** The Petri net model can extend to satisfy each application domain including learning education. Moreover, subnet oriented can combine sub-courses easily to decrease the complexity of course design

3.1 Petri Net Model

Proposed Petri net model is defined as a directed graph PN = (P, T, F, W, M_0); where

1. $P = \{p_1, p_2,\ldots, p_m\} \cup \{cp_1, cp_2,\ldots,cp_r\}$ is a finite set of places that consist of two subset, ordinary place subset (circle) and control place subset (double circle), respectively. In the ordinary place subset, $\{p_1, p_2,\ldots, p_m\}$, each p_i with the form of $(p_i, \{p_j, p_k,\ldots, p_l\})$, p_i is the parent place and the set $\{p_j, p_k, \ldots, p_l\}$ represents the subnet (children) of p_i.
2. $T = \{t_1, t_2,\ldots, t_K\}$ is a finite set of transitions that draw by bars.
3. $F : \{P * T\} \cup \{T * P\} \rightarrow I$, $\{P * T\}$ is called input arc that defines directed arcs from places to output transitions, and $\{T * P\}$ is called output arc which defines directed arcs form input transitions to places, $I = \{1, 2,...\}$ representing set of nonnegative integers.
4. W: $F \rightarrow I$ is a weight function, $I = \{1, 2,... \}$ representing set of nonnegative integers. a k weight arc can be interpreted there are k parallel arcs.
5. M_0: $P \rightarrow \{ I_{C1}, I_{C2}, I_{C3},\ldots\}$ is the initial marking (dot) which assigns color tokens to each place in the net, I_C *is the* nonnegative integers *set* representing the number of color tokens.

In the ordinary place subset $\{p_1, p_2,\ldots, p_m\}$, if p_i with the form of (p_i, ϕ), it represents p_i is not only a child place in the cluster but also a leaf activity in whole Activity Tree. Such place may be considered as learning material including lessons, assessments or courses. In the contrast, if p_i with the form of $(p_i, \{p_j, p_k, \ldots, p_l\})$, it represents p_i is a parent place in the cluster. Although parent place stands for an abstract activity without any physical learning object mapping, it plays an important role to set sequencing rule and show whole Activity Tree well. The other type of place is control place that give assistance of model management. T is a finite set of transition which is event, computation step or state changing operator. Arc directs the information flow outgoing form input place either a point of departure or temporary pause. The dynamic behavior of model simulate by the firing rules. A transition will be fired if the token number of its input place is greater than the weight of its input arc. If the transition is fired, the token of input place will be moved to the output place according with the weight of output arc.

3.2 Applied Petri Net on Sequencing Control Mode Composing

Flow and Choice are atomic elements to control the sequencing behavior for a cluster. When a Parent Activity in a cluster has Sequencing Control Flow defined as true,

Figure 2(b) shows flow construction applying by Petri Net Model. Learners should start their learning path from parent place (abstract activity) when sequencing rule is firing and terminate learning activity at parent place (abstract activity) when the objective is satisfied.

When a Parent Activity has Sequencing Control Forward Only defined as true, the construction applying by PN is similar to Flow Construction. The difference is "Previous" activity is forbid among child-activities (shown in Figure 3(b)).

Choice represents the learner can select lessons in any order. To describe the learning path well, it is necessary to consider the composing of Sequencing Control Choice and Sequencing Control Choice Exit simultaneously.

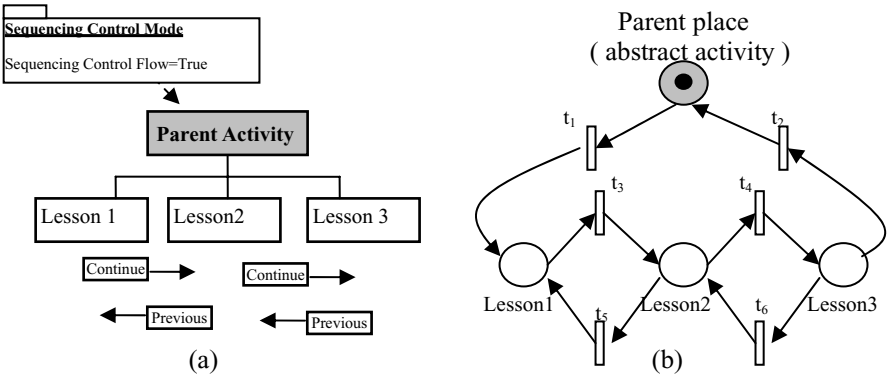


Fig. 2. Sequencing Control Flow Construction

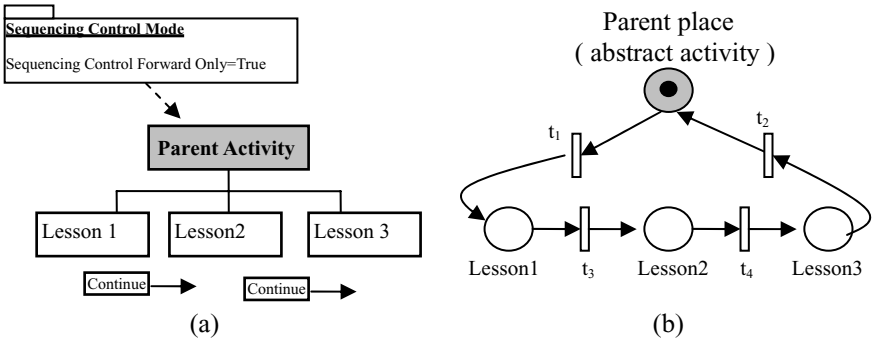


Fig. 3. Sequencing Control Forward Construction

In Figure 4, different value of Sequencing Control Choice Exit for active activities will decide if sibling of this active activity is a valid target of a Choice navigation request. The learner can choose "Lesson1", "Lesson 2" or "Lesson 3" by firing the transition 1, transition 2 or transition 3. If the learner wants to choose "Lesson 2", then t_2 is fired and token will be moved from parent place to "Lesson 2" place called active activity. Due to the

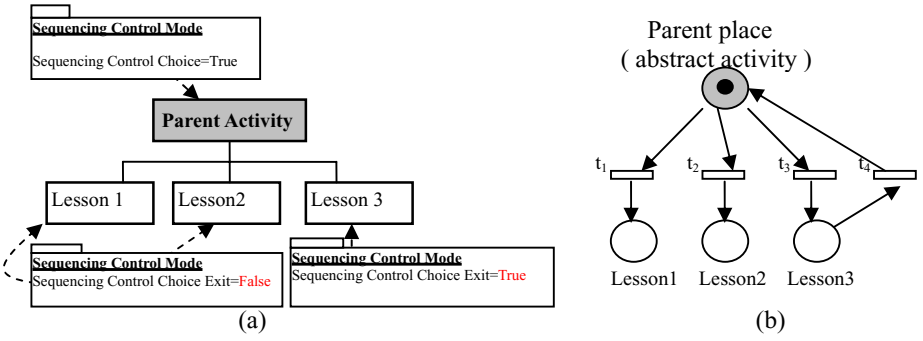


Fig. 4. Various Choice Construction Depends on Value of Sequencing Control Choice Exit

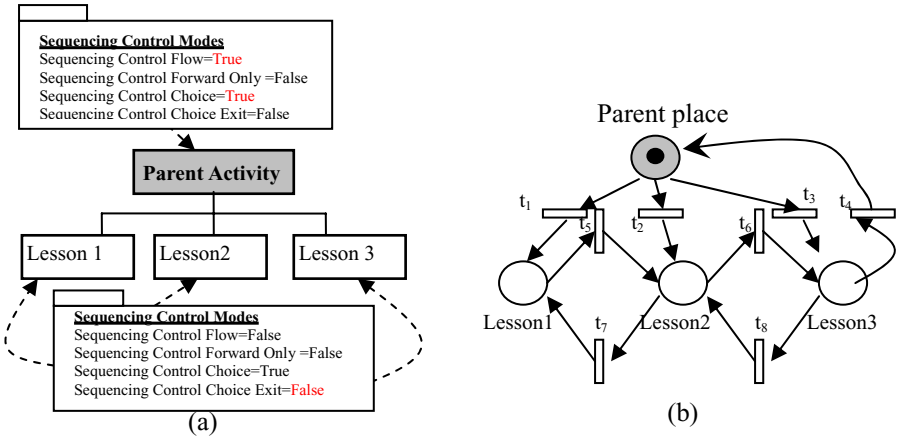


Fig. 5. An example of Composing Construction

value of Sequencing Control Choice Exit for active activity (Lesson 2) is False, the sibling is invalid target of a Choice navigation request for parent place.

Through our PN model, each cluster has its ability to illustrate the composing elements of SCORM Sequencing Definition Model to handle different learning behaviors. Figure 5(b) shows the composing by Petri Net model. Both Sequencing Control Flow and Sequencing Control Choice for Parent Activity set to truth, so LMS will enable the Continue and Previous navigation requests and all children will be valid targets for learners to free choose either. Although Sequencing Control Choice Exit for Parent Activity set to false, it will not affect the learning behaviors of descendents. Choice Exit only applies to active activities. However, all descendents has Sequencing Control Choice Exit set to false. While any descendent is active, neither sibling of this active activity is a valid target of a Choice navigation request. For example, the learner is currently firing t_2 and token will be moved from parent place to "Lesson 2" place. Although the parent of "Lesson 2" has Sequencing Control Choice defined as True,

neither sibling of "Lessons 2" is a valid target of a Choice navigation request. In this example, the only way allowing either "Lesson 1" or "Lesson 3" to be fired is due to the Sequencing Control Flow for Parent Activity as True.

4 Sequencing Trap Detecting by Petri Net Model

4.1 Detect by Petri Net Graph

To avoid the sequencing trap, we should have some a priori information on how each activity sequence will be set. According to the value of SCORM Sequencing Control Mode, a base of Sequencing Definition Model, Petri Net Graph can be built. While some composing fails to build the graph, Sequencing Trap on Parent Activity is occurred. Take an example, if Parent Activity has Sequencing Control Flow and Sequencing Control Choice set to false, it is impossible to draw any input arc or output arc. In other words, Sequencing Trap can be prevented depends on whether Petri Net Graph can be built or not.

4.2 Detect by Flow Matrix

If the sequence composing cannot be illustrated by Petri Net, then it is bound to result is Sequencing Trap. Crosswise, Petri Net existence doesn't imply avoidance of Sequencing Trap. In order to solve the Sequencing Trap problem, we devised a Detection-Algorithm.

From Figure 6(a), we cite Limit Condition Attempt Limit as an instance. Limit Condition describes condition under which an activity is not allowed to be delivered. Our model only focuses on the maximum number of attempts for the activity (Limit Condition Attempt Limit). There may be use cases where a content developer wants to limit the number of attempts that a learner is permitted on a given learning activity. To control the number of attempts, our Petri Net model adds control place to the input transitions of this limited learning activity. According to the number of tokens in the control place, we limit the times that the learning material can be read. For example, in Figure 6(a), P_3 has Limit Condition Attempt Limit defined as2. That is, the learner is permitted to learn P_3 only twice. As Figure 6(b), cp_1 is used which has two tokens to limit the firing time of input transition t_2, t_5, t_8. Initially, we construct a Flow Matrix, M, to keep track of whether exist a directed edge from place *i* to place *j,* where $i \neq j$, and *i* includes the control place. From Figure 6(b), the following Flow Matrix is constructed as 6(c).

Usually, the root activity of tree is encoded as number one and in turns for other activities. Within a token, P_1 is a start point and randomly pick up one transition to be fired. If the selected place has the Limit Condition, we further check whether the value of control place at the selected place is greater than one. If it is, we decrease the value of control place by one. If it is not, it means learner cannot exit from this limited place to other one. In other words, we are trapped. Repeat the above selecting process, until the selection is the root activity or we cannot find any place to advance the selected process. If the selection is the root activity we find a workable path otherwise we are trapped. Detection-Algorithm is shown as follows:

//Detection-Algorithm

1. Construct a Flow Matrix, M

$$m_{i,j} = \begin{cases} 1 & \text{if exist a direct edge from } i \text{ to } j, \\ 0 & \text{otherwise} \end{cases} \qquad 1 \le i, j \le n$$

   where n is the number of places, and add one row for control place by setting its value with available counts.
2. Consider a parent place as a starting place *i*
3. Randomly selecting a connected place *j* in M, where $m_{ij}=1$
   3.1 If place *j* is a place with Limit Condition, check the value of $m_{cp,j}$.
      i. If the value is greater than one, decrease $m_{cp,j}$ by one.
      ii. Else, we are trapped.
4. If place *j* selected in 3 is the root one, we have found a workable path. If we cannot find any place for making process, we are trapped. Otherwise, we repeat step 2 to 3.

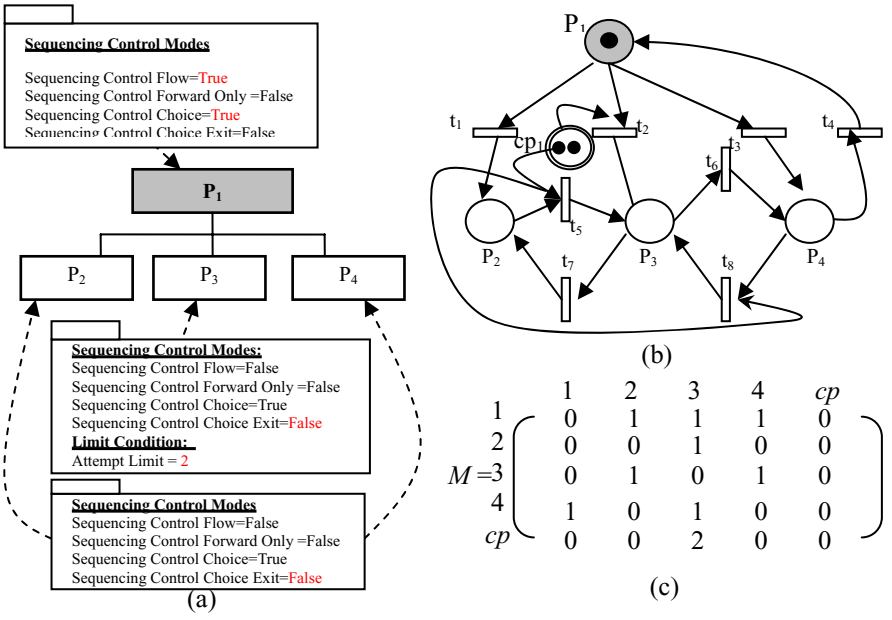


Fig. 6. An Example of Limit Condition Attempt Limit

An entire activity tree may be replaced by a single place or transition for modeling at a more abstract level or places and transitions may be replaced subnets to provide more detailed modeling. We can extend the above Detection-Algorithm to a more complex tree structure to detect whether there is a workable path. We use the recursion way to the complex tree. We show the checking algorithm as follows:

//Checking Algorithm

1. Starting from a sub-tree T_{sub}, with the form of (p_t,{ p_i, p_{i+1},…, p_j}) and the height of T_{sub} is 2, where p_t is the root of this sub-tree and p_i, p_{i+1},…, p_j are all leaves children.
2. If we can find a workable path in Detection-Algorithm of T_{sub}, repeat step 1 until we examine all sub-trees or p_t in step 1 is the root of the whole tree. Else, we concluded that we cannot find a workable path of the whole tree.

Our strategy is bottom-up by starting from a sub-tree with all leaves children and the height of this sub-tree is two. We then check this sub-tree by using the Detection-Algorithm, if we can find a workable path we repeat the checking process until we examined all sub-trees or we arrive at the root node in certain stage, if we are trapped in any one of the sub-trees, we concluded that we will be trapped in the finally.

5 Conclusion

Sequencing is a key point for adaptive learning. When a designer design a course, the visibility can help the designers to make sure what it will be. So, Petri Net Model is applied not only can image the final course easily but can detect sequencing trap in advance. Both flexibility and expansibility can be considered.

References

1. ADL Technical Team: Sharable Content Object Reference Model (SCORM) 2004 Documentation 1st Edition, Advanced Distributed Learning (ADL), January 30 (2004)
2. IMS Simple Sequencing Specification: IMS Global Learning Consortium, Inc., March (2003) http://www.imsglobal.org
3. Reload Editor (Reload): Reload project (2004) http://www.reload.ac.uk
4. J.T.D. Yang, C.Y. Tsai, T.H. Wu: Visualized online simple sequencing authoring tool for SCORM-compliant content package, Proceedings of the 4th IEEE International Conference on Advanced Learning technologies (ICALT 2004), Finland, August (2004)
5. Timothy K. Shih, Wen-Chih Chang and Wen-Chieh Ko: SCORM Sequence and Template Authoring System, Information Resources Management Association International Conference, New Orleans, Louisiana, USA, May 23-26 (2004)
6. H. W. Lin, Timothy K. Shih, Wen-Chih Chang, and Chun-Chia Wang: A Petri Nets-based Approach to Modeling SCORM Sequence, in Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004), Taipei, Taiwan, June 27-30 (2004) 1247-1250
7. Jun-Ming Su, Shian-Shyong Tseng, Chia-Yu Chen, Jui-Feng Weng, and Wen-Nung Tsai: Constructing SCORM Compliant Course Based on High Level Petri Nets, International Journal Computer Standards & Interfaces (2005)
8. Timothy K. Shih, Hsuan-Pu Chang, Chun-Chia Wang, Te-Hua Wang and Kuen Han Jan: SCORM Sequencing Testing, International Conference on SCORM, Taipei, Taiwan, January 16-19 (2006) 36-39
9. ADL Technical Team: SCORM S&N Version 1.3, Advanced Distributed Learning, January 30 (2004)

Multi-agent Approach for Ubiquitous Group Decision Support Involving Emotions

Ricardo Santos[1,2], Goreti Marreiros[1,3], Carlos Ramos[1,3],
José Neves[4], and José Bulas-Cruz[5]

[1] GECAD – Knowledge Engineering and Decision Support Group Porto, Portugal
{goreti, csr}@dei.isep.ipp.pt
[2] College of Management and Technology– Polytechnic of Porto Felgueiras, Portugal
rjs@estgf.ipp.pt
[3] Institute of Engineering – Polytechnic of Porto Porto, Portugal
{goreti, csr}@dei.isep.ipp.pt
[4] University of Minho Braga, Portugal
jneves@di.uminho.pt
[5] University of Trás-os-Montes e Alto Douro Vila Real, Portugal
jcruz@utad.pt

Abstract. The present business environment is characterized by the use of groups, which work in distributed environments and have to deal with uncertainty and rapidly changing information. In this work we propose an architecture for a ubiquitous group decision support system able to support persons in group decision processes. The system considers the emotional factors of the intervenient participants, as well as the argumentation between them. Particular attention will be taken to one of components of this system: the multi-agent simulator, modelling the human participants, considering emotional characteristics, and that allowing the exchanges of hypothetic arguments among the participants.

1 Introduction

Despite the great variety of Decision Support Systems, most of them are individual tools developed to help a particular user involved in a specific decision process. Nowadays, groups are more and more used to perform decisions about some subject of interest for the organization/community in which they are involved. The scope of those decisions can be very diverse. It can be related to economic and political decisions, like for instance the acquisition of new military equipment. But it can also be a trivial decision like the choice of a vacancy destiny by a group of friends. Group decision support systems (GDSS) thus have emerged as a vital important area in several domains.

Generically we may say that GDSS aims to reduce the loss associated to group work (e.g. time consuming, high costs, improper use of group dynamics, etc.) and to maintain or improve the gains (e.g. groups are better to understand problems and in flaw detection, participants' different knowledge and processing skills allow results that could not be achieved individually). If the group members are dispersed in time and space, the need of coordination, informal and formal communication, and information share support will increase significantly.

J. Ma et al. (Eds.): UIC 2006, LNCS 4159, pp. 1174 – 1185, 2006.

In this work we propose an architecture to a ubiquitous group decision system able to support persons in group decision processes and considering the emotional factors of the intervenient participants as well as argumentation between them. Particular attention will be given to one specific component of this architecture: the agent based simulator for group decision that models the participants considering emotional characteristics, allowing the exchanges of arguments among the participants.

In our previous work we state that, the use of multi-agent systems seems to be very suitable to simulate the behaviour of groups of people working together and, in particular, to group decision making modelling, because it allows [1]: individual modelling, flexibility and data distribution. In classical decision theory proposals are chosen by individual decision makers in order to maximize the expected utility. However, if we transpose those choices to quotidian life, it is almost impossible to say that our decisions are not influenced by the emotion and moods that we are feeling. The addition of affect to group decision making processes combined with the possibility to simulate several possible scenarios will allow to understand better the implication of different argumentation alternatives among group participants.

The work described in this paper is included in ArgEmotionAgents project (POSC / EIA / 56259 / 2004 - Argumentative Agents with Emotional Behaviour Modelling for Participants' Support in Group Decision-Making Meetings), which is a project supported by FCT (Science & Technology Foundation – Portugal) envisaging the use of Multi-Agent Systems approach for simulating Group Decision-Making processes, where Argumentation and Emotion components are specially important.

This paper is organized as follows: Section 2 provides a general approach to Ubiquitous group decision. Section 3 introduces the system architecture that we are proposing and briefly presents the main modules. Section 4 presents the multi-agent model of our system. Section 5 describes with more detail one of the components of the system architecture we are proposing: Agent-based simulator for group decision. Section 6 presents conclusions, perspectives and ideas for further work.

2 Group Decision

Jonathan Grudin [15] classifies the digital technology to support the group interaction in three phases: the pre-ubiquitous, the proto-ubiquitous and the ubiquitous. In the pre-ubiquitous phase, that begin in the 70's, were supported face-to-face meetings. In the proto-ubiquitous phase distributed meetings were supported, this phase begun approximately at 1990. The ubiquitous phase is now getting under way and support meetings distributed in time and space. The system we are proposing will be built to support distributed and asynchronous decision meetings or social interactions.

2.1 Ubiquitous Group Decision Making

Ubiquitous computing was introduced by Mark Weiser in 1991 [2], and anticipates a digital world in which consists on many distributed devices that interact with users in a natural way. This vision was too far ahead for its time basically because of technology limitations, but since the appearance of this concept it has passed fifteen years and many of the hardware technologies that were necessary to the implementation of

Mark Weiser's vision is now commercially available to everyone. In an ambient intelligent environment, people are surrounded with networks of embedded intelligent devices providing ubiquitous information, communication and services. Intelligent devices are available whenever we need it, enabled by simple interactions and effortless interactions, attuned to all our senses, adaptive to users and context and autonomously acting. High quality information and content must be available to any user, anywhere, at any time, and on any device.

Today there is an increasing interest in the development of Group Decision Support Systems (GDSS) to support the "any time / any place" group decision making processes, instead of the "same place / same time". This interest appeared with the need of joining the best potential group of participants. With the globalization of the economy possible participants to the group, like specialist or experts in determined areas where located in different points of the planet and there was no way to join them in a decision room. Until some years ago the possible resolution for this scenario was to wait until all the participants meet together, actually there is a grow interest on developing systems to support these scenarios.

There are many areas where ubiquitous group decision making makes sense. One of the most cited areas in literature is Healthcare since patient's treatment involves various specialists, like doctors, nurses, laboratory assistants, radiologist, etc. These specialists could be distributed across departments, hospitals or even in different countries. The HERMES system, a web-based GDSS was tested inside this context [3]. There are other GDSS that support ubiquitous decision (GroupSystems software; *WebMeeting* [4]; VisionQuest software).

2.2 Mixed-Initiative Systems

In the last years researchers had pondered on the merits of a total automation of user necessities (via Intelligent Agents) versus a total user control of operations and decisions (via Graphical User Interfaces) [16].

There exist an interesting dualism between Artificial Intelligence (AI) and Human Computer Interaction (HCI). In AI, we try to model the human think way to create computer systems able to do intelligent actions. In HCI, we design computer interfaces to attract the user, supporting him to execute intelligent actions.

The link between these two fields appears as Mixed Initiative Interaction. This concept refers to a flexible interaction strategy, where each agent (human or computer) can contribute to problem resolution with the best he can do, at appropriated moment [17].

The concept of Mixed-Initiative Systems is quite adequate to the group decision field. It is certainly very useful for a participant in a meeting to be assisted by a system able to show the available information and knowledge, analyzing the meeting trends and suggesting arguments to be exchanged with other specific participants.

In our work we face Mixed Initiative Systems as a joint process between a user and a community of computer agents that work as a whole like a personal assistant to satisfy user needs. Agent-based systems are ideal to accomplish this purpose due to their autonomy and independence to execute repetitive, boring and time consuming tasks. Cesta and D'Aloisi [18] identified three characteristics needed in order to make the agents to participate in a mixed-initiative interaction with their users:

- Agents should adapt their behaviour to the user they are supporting, according to the philosophy of adaptive interfaces;
- Agents should move according to some principles of initiative shift between them, their users and other agents in the environment (when they exist);
- Agents should leave the user a level of 'super-control' in order to enhance his sense of trust towards them. The user must have the possibility of inspecting the agent and then he will be able to prevent any undesirable operations and failures.

3 System Architecture

Our aim is to present a ubiquitous system able to exhibit an intelligent and emotional aware behaviour in the interaction with individual persons and groups. This system supports persons in Group Decision making processes considering the emotional factors of the intervenient participants as well as argumentation between them.

Groups and Social systems will be modelled by intelligent agents that will be simulated considering emotional aspects, to have an idea of possible trends in social/group interactions. The paradigm of Mixed Initiative will be used to assist the participants in group decision. So the initiative of the ubiquitous system will be led to both the users and the system.

The main goals of the system are:

- To create a simplified model of Groups and Social Systems able to be used in Decision Making processes, balancing Emotional and Rational aspects in a correct way;
- To develop a decision making simulation system to support meeting participants, this system must involve the emotional component in the decision making process;
- To develop an argumentation support system, able to suggest arguments to be used by a meeting participant in the interactions with other participants;
- To define a mixed-initiative interface for the developed system;
- To be available in any place (meeting room, outside using a web based tool), in different devices (computers, notebooks, PDA, etc) and at different times (on-line meeting, asynchronous meetings, etc).

The system will consist of a suite of applications as depicted in the following figure:

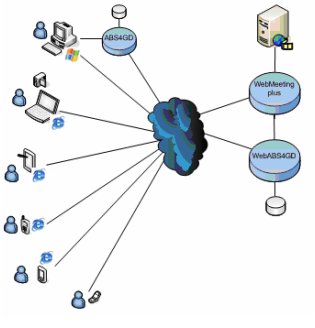


Fig. 1. System Architecture

The main blocks of the system are:

- **WebMeeting Plus** – this is an evolution of the *WebMeeting* project with extended features for audio and video streaming. In its initially version *WebMeeting* [4] was designed as a GDSS that supports distributed and asynchronous meetings through the Internet. The *WebMeeting* system is focused on multicriteria problems, where there are several alternatives that are evaluated by various decision criteria. Moreover the system is intended to provide support for the activities associated with the whole meeting life cycle, from the pre-meeting phase to the post-meeting phase. The system aims at supporting the activities of two distinct types of users: ordinary group "members" and the "facilitator". The system works by allowing participants to post arguments in favour/neutral/against the different alternatives being discussed for a problem. It will also be a window to the information repository for the current problem. This will be a web based application accessible by desktop and mobile browsersand eventually WML for WAP browsers;
- **ABS4GD** – this is the simulation tool resulting from the ArgEmotionAgents project. ABS4GD (Agent Based Simulation for Group Decision) is a multi-agent simulator system whose aim is to simulate group decision making processes, considering emotional and argumentative aspects. ABS4GD is composed by several agents, but the more relevant are the participant agents that simulate the human participants of a decision meeting (this decision making process is influenced by the emotional state of the agents and by the exchanged arguments). The user maintains a database of profiles and history with the model of the group; this model is built incrementally during the different interactions of the user in the system.
- **WebABS4GD** – this is a web version of the ABS4GD tool to be used by users with limited computational power (e.g., mobile phones) or users accessing the system through Internet. The data base of profiles and history will not be shared by all users, allowing for a user to securely store its database on the server application with guarantees that his/her model will just be available for him or her.

4 Multi-agent Model

As we state previously multi-agent systems seems to be quite suitable to simulate the behaviour of groups of people working together [1], as well as, to assist the participants presenting new arguments and feeding the simulation model of the group by observing the interaction and history of the meeting.

Each participant of the group decision has a community of agents to assist in future interaction with other participants. This community should be persistent because it is necessary to have information of previous group decisions about the credibility, reputation as well as past behaviours of other participants. The participant should have access to a Agent Based Simulation Tool for Group Decision (AGS4GD) resulted from the ArgEmotionAgents project developed in our previous work. This tool will improve the knowledge of the agent community in order to predict the behaviour of other participants and to advice him in the best way.

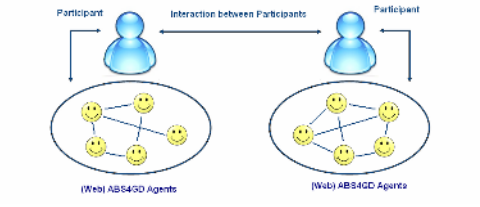


Fig. 2. Multi-agent Model

This support to the participant will be implemented using mixed-initiative interaction. According to this concept Intelligent Agent based System can propose solutions for the user, the user is able to change these proposed solutions and the Intelligent Agent based System learns with the user interaction, changing algorithms in order to be more close to user point of view in future interactions. In this way users start to be more confident with Intelligent Agent based System.

5 Agent Based Simulator for Group Decision (ABS4GD)

In ours previous work we identified the main agents involved in a simulation of a group decision meeting [5] and they are: Participant Agents; Facilitator Agent; Register Agent; Voting Agent and Information Agent. In the remain text of this section we will first present the architecture of participants agents, because they represent the main role in group decision making and then we will detail the components of this architecture.

5.1 Participant Agent Architecture

In figure 3 it is represented the architecture of participant agents [6]. This architecture contains three main layers: the knowledge layer, the reasoning layer and the communication layer.

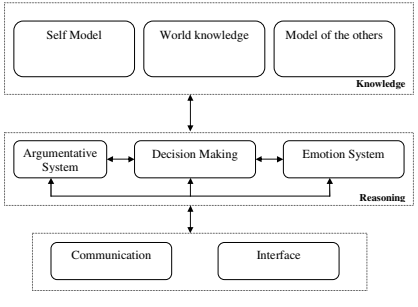


Fig. 3. Participant Agent Architecture [6]

In the knowledge layer the agent has information about the environment where he is situated, about the profile of the other participant's agents that compose the simulation group, and regarding its own preferences and goals (its own profile). The information

in the knowledge layer is dotted of uncertainty and will be accurate along the time through interactions done by the agent.

The communication layer is responsible for the communication with other agents and by the interface with the user of the group decision making simulator.

The reasoning layer contains three major modules:

- The argumentative system – that is responsible for the arguments generation. This component will generate explanatory arguments and persuasive arguments, which are more related with the internal agent emotional state and about what he, think of the others agents profile (including the emotional state);
- The decision making module – will support agents in the choice of the preferred alternative and will classify all the set of alternatives in three classes: preferred, indifferent and inadmissible;
- The emotional system – will generate emotions and moods, affecting the choice of the arguments to send to the others participants, the evaluation of the received arguments and the final decision.

5.2 Decision Making

Participant agents use the decision making component to establish individual preferences; they can use a simple additive function or a more sophisticated method like for instance AHP (Analytical Hierarchical Process) [19]. After that participant agents divide the set of alternatives in three classes according to the initial preferences. These classes are: preferred, indifferent and inadmissible. To make this distribution the first two phases of NAI (Negotiable Alternatives Identifier) algorithm [7] will be used. The NAI algorithm proposes the classification more preferred, preferred and indifferent, and we will change this classification because in group decision scenarios some alternatives may be really inadmissible for a specific participant.

5.3 Argumentative System

This component will generate persuasive arguments based on the information that exists in the Participant agent knowledge base. We adopt the same ontology as the used in [8][9]. So, we have the following arguments: appeal to prevailing practice; a counter example; an appeal to past promise; an appeal to self-interest; a promise of future reward; a threat.

In [8] it is used an existent pre-order for the selection of arguments to send, and in [9] the selection is based on mixture of the alternatives utility and the trust in the interlocutor. In our model the selection of arguments is based on the agent emotional state, in the information detained about the interlocutor profile and in the quality of that information.

5.4 Emotional Module

The emotions that will be simulated in our system are those identified in the reviewed version of the OCC [10] model: joy, hope, relief, pride, gratitude, like, distress, fear, disappointment, remorse, anger and dislike.

An emotion in our system is characterized by the following properties: if it is positive or negative, moment in time (simulation time) when it was initiated, identification of the agent or event that cause the emotion and emotion intensity.

The simulation responsible will setup a set of rules to configure the emotion generation. The system is prepared to allow the configuration of all the set considered in the OCC model, but the responsible may opt just to configure a subset of that.

In figure 4 it is possible to visualize the main components of the emotional system.

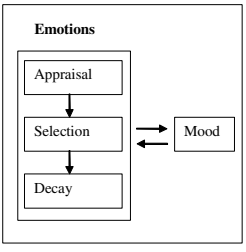


Fig. 4. Emotional Module [6]

The emotional module is composed by three main components: appraisal, selection and decay. The agent mood is calculated based on the emotions felt.

appraisal

The appraisal mechanism is based on OCC model, the simulator user defines the conditions for the emotion activation. An example may be:

$$Hope(AgPi,X):\text{-}Goal(AgP_i,X),\ Request\ (AgP_j,X).$$

In the previous example the emotion *Hope* is appraised if Agent AgP_i has the *goal(X)* and asks to agent AgP_j to perform the *X* then the emotion hope is generated.

To each condition for the emotion generation is settled a weight, in the interval [0,1]. The intensity of the emotion is calculated according the conditions weight.

A particular emotion could be or not expressed by the agent depending on the intensity of the others emotions.

selection

All the emotions defined in the simulator have a threshold activation, which can be influenced by the agent mood. The activation threshold is a value between 0 and 1. This component selects the dominant emotion.

$AgP_{i,Emo,t}$ is the set of all the emotions generated by the agent AgP_i and respective intensities and activations thresholds.

$$AgP_{i,Emo,t}=\{(Emo_1,Int_1,Act_1),\ldots(Emo_n,Int_n,Act_{n)}\}$$

The selected emotion in instant t, $AgPi_{ActEmo,t;}$ will be the one that have a higher difference between the intensity and the activation.

decay
Emotions have a short duration, but they do not go away instantaneously, they have a period of decay. There are several proposals for this calculation. In our model we consider three possibilities: linear, exponential and variant.

If the responsible for the simulator intends to distinguish between decays rates for positive and negative emotions, his choice must be the variant decay rate.

mood
The agent mood is calculated based on the emotions agents felt in the past and in what agent think about the moods of the remaining participants. In our approach only the process of mood contagion is being considered, we do handle the process of emotions contagion. We consider only three stages for mood: positive, negative and neutral. The mood of a specific participant is determined according the following:

$$K^+ = \sum_{i=t-n}^{t-1} I_i^+ \ , \ K^- = \sum_{i=t-n}^{t-1} I_i^-$$

K^+ and K^- are the sum of the positive/negative emotions felt in the last n periods, and n can be parameterized by the simulator user. Only emotions that are above the threshold activation are considered.

$$\begin{cases} if\ K^+ \geq K^- - l\ ,\ then\ positive\ mood \\ if\ K^- \geq K^+ - l\ ,\ then\ negative\ mood \\ if\ \left|K^+ - K^-\right| < l\ ,\ then\ neutral\ mood \end{cases}$$

The value of l varies according what a specific participant thinks about the mood of the group.

$$\begin{cases} l = 0.10\ ,\ if\ \ group\ mood\ is\ positive \\ l = 0.05\ ,\ if\ \ group\ mood\ is\ neutral \\ l = 0.01,\ \ if\ \ group\ mood\ is\ negative \end{cases}$$

Each participant agent has a model of the other agents, in particular has information about the other agent's mood. This model of the others considered incomplete information handling and the existence of explicit negation, following the approach described in Analide and Neves [11]. Some of the properties that characterize the agent model are: gratitude debts, benevolent, credibility [12], (un)preferred arguments.

Although the emotional component is based on the OCC model we think that with the inclusion of mood we can overcome one of the major critics that usually is pointed to this model, the fact that OCC model does not handle the treatment of past interactions and (in our case) past emotions.

5.5 Implementation

In this section we present some details of our implementation of the simulator previously described.

The system was developed in Open Agent Architecture (OAA) [14], Java and Prolog. OAA is structured in order: to minimize the effort involved in the creation of

new agents, that can be written in various languages and operating on various platforms; to encourage the reuse of existing agents; and to allow for dynamism and flexibility in the makeup of agent communities.

The OAA framework is composed by two distinct types of agents: the OAA facilitator agent and client agents. Usually there is only one facilitator by application, but it is also possible to have multiple facilitators. The OAA facilitator agent is responsible for tasks related to coordination and communication. The OAA Facilitator has, in somehow, behaviour similar to a router in the sense that it is responsible for distributing data and messages among its set of client agents.

OAA has an Inter-agent Communication Language (ICL) that is shared by all agents independently of the language in which they are programmed or the operating system of the machine where the agents reside. The ICL language is close to KQML. OAA imposes a common protocol for agents entering and registering at the group decision making simulator.

In figure 5 it is possible to see the emotional configuration process, in this particular case the responsible is configuring the emotion *Gratitude* that has an activation threshold of 0.6 and the decay rate used is linear.

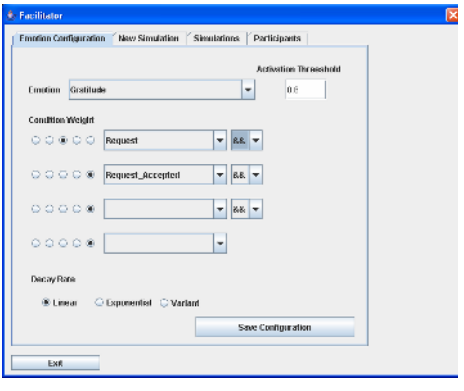


Fig. 5. Emotion configuration

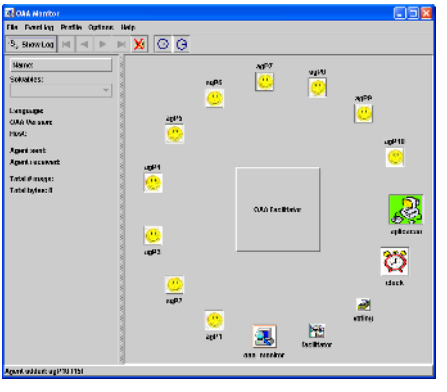


Fig. 6. Community of agents

Figure 6 shows all the agents that exist at a particular moment in the simulator: 10 participant agents, the facilitator agent (responsible for the follow-up of all simulations), the voting agent, the clock agent (OAA is not specially designed for simulation, for that reason it was necessary to introduce a clock agent to control the simulation) , the oaa_monitor (that is an agent that belongs to the OAA platform, and is used to trace, debug and profile communication events for an OAA agent community) and the application agent (responsible for the communication between the community of agents and the simulator interface).

6 Conclusions

This work proposes an architecture to a ubiquitous group decision system able to support distributed and asynchronous decision meetings. This system will support a

group of persons involved in a group decision, as was referred before the system should be available in any place (meeting room, outside using a web based tool), in different devices (computers, notebooks, PDA, etc) and at different times (on-line meeting, asynchronous meetings, etc). One of the components of this architecture is a multi-agent simulator of group decision making processes, where agents are designed with emotional characteristics, and are able to reason with incomplete information. The discussion process between group members is made through the exchange of persuasive arguments. Future work includes the refinement of this architecture, and the development of the other components. Particular attention should be give to the interaction between the simulator and the group members, which as we refereed will be done through the concept of mixed initiative systems.

Acknowledgements

The authors would like to acknowledge FCT, FEDER, POCTI, POSI, POCI and POSC for their support to R&D Projects and GECAD Unit.

References

1. Marreiros, G.; Santos, R.; Ramos, C. and Neves, J.: Agent Based Simulation for Group Formation. 19th European Simulation Multi-Conference, Latvia (2005) 521-526
2. Weiser, M: The Computer for the Twenty-First Century. Scientific American, (1991)
3. Marreiros, G.; Sousa, J.P. and Ramos, C: WebMeeting - A Group Decision Support System for Multi-criteria Decision Problems. International Conference on Knowledge Engineering and Decision Support, Porto, Portugal ICKEDS04 (2004) 63-70
4. Karacapilidis, N.; Papadias, D.: Computer supported argumentation and collaborative decision making: The Hermes system, Information Systems, Vol. 26 No. 4 (2001)259-277
5. Marreiros, G.; Ramos, C. and Neves, J.: Modelling group decision meeting participants with an Agent-based approach". Selected for publication in an upcoming special issue of the International Journal of Engineering Intelligent Systems, (2006)
6. Marreiros, G.; Ramos, C. and Neves J.: Dealing with Emotional Factors in Agent Based Ubiquitous Group Decision. Lecture Notes in Computer Science Embedded and Ubiquitous Computing: EUC 2005 Vol. 3823 (2005) 41-50
7. Yen, J.; Bui, T.: The negotiable alternatives identifier for group negotiation support. Applied Mathematics and Computation (1999) 259 – 276
8. Kraus, S. K. Sycara, and A. Evenchik. 'Reaching agreements through argumentation: A logical model and implementation'. Artificial Intelligence, Vol. 104 No. 12 (1998) 1–69
9. Ramchurn, S. D.; N. R. Jennings, and C. Sierra: Persuasive negotiation for autonomous agents: a rhetorical approach. In C. Reed, F. Grasso, and G. Carenini, editors, IJCAI Workshop on Computational Models of Natural Argument, AAAI Press, (2003) 9–17
10. Ortony, A.: On making believable emotional agents believable, In R. P. Trapple, P. (Ed.), Emotions in humans and artefacts. Cambridge: MIT Press, (2003)
11. Analide, C. and Neves, J.: Antropopatia em Entidades Virtuais. I Workshop de Teses e Dissertações em Inteligência Artificial (WTDIA'02), Porto Galinhas, Brazil (2002)

12. Andrade F., Neves J., Novais P., Machado J., Abelha A.: Legal Security and Credibility in Agent Based Virtual Enterprises, in Collaborative Networks and Their Breeding Environments, Camarinha-Matos L. Afsarmanesh H., Ortiz A., (Eds), Springer-Verlag, ISBN 0-387-28259-9 (2005) 501-512
13. Sabater J. and Sierra C.: Reputation and social network analysis in multi-agent systems. Proceedings of the first International Joint Conference on Autonomous Agents and Multi-agent systems, Bologna, Italy (2002) 475-482
14. OAA-URL: www.ai.sri.com/~oaa/.
15. Grudin, J.: Group Dynamics and Ubiquitous computing. Communications of the ACM, 45 (12) (2002)
16. Shneiderman, B. and Maes, P.: Direct Manipulation vs. Interface Agents. Interactions, 4(5), (1997)
17. Hearst, M. Trends & Controversies: Mixedinitiative interaction. IEEE Intelligent Systems, (1999)
18. Cesta, A. and D'Aloisi, D: Mixedinitiative-Initiative Issues in an Agent-Based Meeting Scheduler. User Modeling and User-Adapted Interaction:Vol. 9, No.1-2 (1999) 45 – 78
19. Saaty, T.L: The Analytic Hierarchy Process,New York, McGraw Hill (1980)

Author Index