James J. (Jong Hyuk) Park
Hamid R. Arabnia
Cheonshik Kim
Weisong Shi
Joon-Min Gil (Eds.)

LNCS 7861

Grid and Pervasive Computing

8th International Conference, GPC 2013 and Colocated Workshops
Seoul, Korea, May 2013, Proceedings

Springer

Lecture Notes in Computer Science 7861

James J. (Jong Hyuk) Park
Hamid R. Arabnia   Cheonshik Kim
Weisong Shi   Joon-Min Gil (Eds.)

Grid and Pervasive Computing

8th International Conference, GPC 2013
and Colocated Workshops
Seoul, Korea, May 9-11, 2013
Proceedings

Springer

Volume Editors

James J. (Jong Hyuk) Park
Seoul University of Science and Technology, Seoul, Korea
E-mail: parkjonghyuk1@hotmail.com

Hamid R. Arabnia
University of Georgia, Athens, USA
E-mail: hra@cs.uga.edu

Cheonshik Kim
Sejong University, Seoul, Korea
E-mail: mipsan@paran.com

Weisong Shi
Wayne State University, Detroit, MI, USA
E-mail: weisong@wayne.edu

Joon-Min Gil
Catholic University of Daegu, Gyeongsan-si, Gyeongbuk, Korea
E-mail: jmgil@cu.ac.kr

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-38026-6 e-ISBN 978-3-642-38027-3
DOI 10.1007/978-3-642-38027-3
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013936008

CR Subject Classification (1998): F.2, C.1, C.2, C.3, C.5, H.5, H.3, D.4,
K.6.5, K.4, B.2, B.8, C.4

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Welcome to the 8^{th} International Conference on Grid and Pervasive Computing (GPC 2013), held in Seoul, Korea, during May 9–11, 2013. GPC-13 was the most comprehensive conference focused on the various aspects of grid and pervasive computing. GPC 2006, GPC 2007, GPC 2008, GPC 2009, GPC 2010, and GPC 2011 took place in Taichung (Taiwan), Paris (France), Kunming (China), Geneva (Switzerland), Hualien (Taiwan), and Oulu (Finland), respectively, and GPC 2012 in Hong Kong, China.

The papers included in the proceedings cover the following topics: cloud, cluster and grid computing; grid and cloud computing economy and business models; security and privacy in grid, pervasive and cloud computing; embedded and pervasive computing; social network and services; machine to machine communications; service-oriented computing, mobile, peer-to-peer and pervasive computing. Accepted and presented papers highlight the new trends and challenges of grid and pervasive computing. The presenters showed how new research could lead to novel and innovative applications. GPC 2013 provided an opportunity for academic and industry professionals to discuss the latest issues and progress in the area of GPC. In addition, the conference published high-quality papers that are closely related to the various theories and practical applications in GPC. Furthermore, we expect that the conference and its publications will be a trigger for further related research and technology improvements in this important subject.

For GPC 2013, we received many papers submission from more than 12 countries. Out of these, after a rigorous peer-review process, we accepted 65 papers of high quality for the GPC 2013 proceedings, published by Springer. All submitted papers underwent blind reviews by at least two reviewers from the Technical Program Committee, comprising leading researchers from around the globe. Without their hard work, achieving such high-quality proceedings would not have been possible. We take this opportunity to thank them for their great support and cooperation. We thank the organizers of the International Workshop on Ubiquitous and Multimedia Application Systems (UMAS 2013), the International Workshop DATICS-GPC 2013: Design, Analysis and Tools for Integrated Circuits and Systems, the International Workshop on Future Science Technologies and Application (FSTA 2013), and the Workshop on Green and Human Information Technology (GHIT 2013). The goal of the workshops was to provide a forum for researchers to exchange and share new ideas, research results, and ongoing work on advanced topics in grid and pervasive computing. Finally, we would like to also thank all the authors, reviewers, and Organizing Committee Members.

May 2013

James J. (Jong Hyuk) Park
Hong Shen
Hamid R. Arabnia
Cheonshik Kim
Weisong Shi
HeonChang Yu

Organization

General Chairs

James J. (Jong Hyuk) Park	SeoulTech, Korea
Hong Shen	University of Adelaide, Australia
Hamid R. Arabnia	The University of Georgia, USA

General Vice-chairs

Martin Sang-Soo Yeo	Mokwon University, Korea
Young-Sik Jeong	Wonkwang University, Korea

Program Chairs

Cheonshik Kim	Sejong University, Korea (Leading Chair)
Weisong Shi	Wayne State University, USA
HeonChang Yu	Korea University, Korea
George Roussos	University of London, UK

Workshop Chairs

Joon-Min Gil	Catholic University of Daegu, Korea (Leading Chair)
Zhiyong Xu	Suffolk University, USA
Mohamed Gaber	University of Portsmouth, UK

Publication Chair

Hwa Young Jeong	Kyung Hee University, Korea

Steering Committee

Hai Jin	Huazhong University of Science and Technology, China (Chair)
Nabil Abdennadher	University of Applied Sciences, Switzerland
Christophe Cerin	University of Paris XIII, France
Sajal K. Das	The University of Texas at Arlington, USA
Jean-Luc Gaudiot	University of California - Irvine, USA
Kuan-Ching Li	Providence University, Taiwan
Cho-Li Wang	The University of Hong Kong, China
Chao-Tung Yang	Tunghai University, Taiwan

Publicity Co-chairs

Jaehwa Chung	Korea National Open University, Korea
Zili Shao	Hong Kong Polytechnic University, China
Chun-Cheng Lin	National Chiao Tung University, Taiwan
Julio Sahuquillo	Universidad Politecnica de Valencia, Spain
Akihiro Fujiwara	Kyushu Institute of Technology, Japan
Bong-Hwa Hong	Kyung Hee Cyber University, Korea

Program Committee

Alfredo Navarra	Università degli Studi di Perugia, Italy
Andrew L. Wendelborn	University of Adelaide, Australia
Beniamino Di Martino	Second University of Naples, Italy
Bin Guo	Institute TELECOM SudParis, France
Chao-Tung Yang	Tunghai University, Taiwan
Chen Liu	Clarkson University, USA
Chen Yu	Huazhong University of Science and Technology, China
Cheng-Chin Chiang	National Dong Hwa University, Taiwan
Chien-Min Wang	Academia Sinica, Taiwan
Chin-Feng Lai	National Ilan University, Taiwan
Ching-Hsien (Robert) Hsu	Chung Hua University, Taiwan
Christophe Cerin	Université de Paris XIII, France
Daewon Lee	Seokyeong University, Korea
Damon Shing-Min Liu	National Chung Cheng University, Taiwan
Dan Grigoras	University College Cork, Ireland
Dana Petcu	Western University of Timisoara, Romania
David De Roure	University of Southampton, UK
David H. C. Du	University of Minnesota, USA
David Hung-Chang Du	University of Minnesota, USA
David Laiymani	I.U.T. de Belfort-Montbéliard, Franace
Der-Jiunn Deng	National Changhua University of Education, Taiwan
El-ghazali Talbi	INRIA Lille - Nord Europe, France
Fahim Kawsar	Bell labs, University of Lancaster, UK
Fevzi Belli	Univ. Paderborn, Germany
Francis C.M. Lau	The University of Hong Kong, China
Guoying Zhao	University of Oulu, Finland
Hai Jiang	Arkansas State University, USA
Hamid R. Arabnia	University of Georgia, USA
Hedda Schmidtke	Carnegie Mellon University in Rwanda
Hong Tang	Aliyun Inc.
Hui-Huang Hsu	Tamkang University, Taiwan
Hung-Chang Hsiao	National Cheng Kung University, Taiwan

Hung-Chang Hsiao	National Cheng Kung University, Taiwan
Hwamin Lee	Soonchunhyang University, Korea
Incheon Paik	University of Aizu, Japan
Insik Shin	KAIST, Korea
Ioan Marius Bilasco	University of Science and Technology of Lille, France
Ioana Banicescu	Mississippi State University, USA
Ivan Stojmenovic	University of Ottawa, Canada
Jan-Jan Wu	Academia Sinica, Taiwan
Jemal Abawajy	Deakin University, Australia
Jenq Kuen Lee	National Tsing Hua University, Taiwan
Jerry Hsi-Ya Chang	NCHC, Taiwan
Jie Tang	Intel Research, Beijing, China
Jingling Xue	University of New South Wales, Australia
Jingyu Zhou	Shanghai Jiaotong University, China
Jose Fortes	University of Florida - Gainesville, USA
Junjie Peng	Shanghai University, China
Kaori Fujinami	Tokyo University of Agriculture and Technology, Japan
Kazunori Takashio	Keio University, Japan
Kewei Sha	Oklahoma City University, USA
Kuo-Chan Huang	National Taichung University, Taiwan
Luciana Arantes	LIP6, France
Marcin Paprzycki	Computer Science Institute, Poland
Martti Mäntylä	Helsinki Institute for Information Technology HIIT, Finland
Masayoshi Ohashi	ATR Media Information Science Laboratories, Japan
Meng-Yen Hsieh	Providence University, Taiwan
Michael Beigl	University of Braunschweig, Germany
Michael Hobbs	Deakin University, Australia
Michel Koskas	Amiens, France
Ming-Lu Li	Shanghai Jiang Tong University, China
Mitsuhisa Sato	Tsukuba University, Japan
Mohamed Jemni	ESSTT, Tunisia
Nabil Abdennadher	University of Applied Sciences, Switzerland
Niwat Thepvilojanapong	Mie University, Japan
Noel Crespi	Institut Telecom France, France
Nong Xiao	National University of Defense Technology, China
Osamu Tatebe	Tsukuba University, Japan
Pangfeng Liu	National Taiwan University, Taiwan
Pedro Medeiros	New University of Lisbon, Portugal
Pradip K. Srimani	Clemson University, USA
Putchong Uthayopas	Kasetsart University, Thailand
Raphael Couturier	LIFC, University of Franche Comte, France

Reen-Cheng Wang	National Taitung University, Taiwan
Rodrigo Mello	University of Sao Paulo, Brazil
Ronald H. Perrott	Queen's University Belfast, UK
Rui Zhang	Palo Alto Research Center, USA
Ruppa K. Thulasiram	University of Manitoba, Canada
Sabah Mohammed	Lakehead University, Canada
Sajid Hussain	Acadia University, Canada
Sasu Tarkoma	University of Helsinki, Finland
Satoshi Sekiguchi	AIST, Japan
Seungjong Park	Louisiana State University, USA
Sherali Zeadally	University of the District of Columbia, USA
Sun-Yuan Hsieh	National Cheng Kung University, Taiwan
Taegyu Lee	Korea Institute of Industrial Technology (KITECH), Korea
Taeweon Suh	Korea University, Korea
Takuro Yonezawa	Keio University, Japan
Tatsuo Nakajima	Waseda University, Japan
Ting-Wei Hou	National Cheng Kung University, Taiwan
Tomas Margalef Burrull	Universitat Autònoma de Barcelona, Spain
Tommi Mikkonen	Tampere University of Technology, Finland
Trung Q. Duong	Blekinge Institute of Technology, Sweden
Victor Malyshkin	Russian Academy of Sciences, Russia
Wang-Chien Lee	Penn State University, USA
Wasim Raad	King Fahd University of Petroleum and Minerals, Saudi Arabia
Weijun Xiao	Virginia Commonwealth University, USA
Weili Han	Fudan University, China
Weng Fai Wong	National University of Singapore, Singapore
Wenguang Chen	Tsinghua University, China
Won Woo Ro	Yonsei University, Korea
Xiangjian He	University of Technology Sydney, Australia
Xiaowu Chen	Beihang University, China
Yanmin Zhu	Imperial College London, UK
Yeh-Ching Chung	National Tsing Hua University, Taiwan
Yong-Kee Jun	Gyeongsang National University, Korea
Yuanfang Chen	Dalian University of Technology, China
Yuezhi Zhou	Tsinghua University, China
Yuhong Yan	Concordia University, Canada
Yulei Wu	University of Bradford, UK
Zhifeng Yun	Louisiana State University, USA
Zhiwen Yu	Northwestern Polytechnical University, China

Message from the UMAS 2013 Chair

Welcome to the proceedings of the 2013 International Workshop on Ubiquitous and Multimedia Application Systems (UMAS 2013), jointly held with GPC 2013 in Seoul, Korea, during May 9–11, 2013.

The fast developments in the electronics industry and the emerging convergence of the triple (video, voice, and data) signal services have allowed media communications and computing to increase ubiquitously. Meanwhile, the embedded systems, i.e., computers inside products, have been widely adopted in many domains including multimedia communications, traditional control systems, medical instruments, wired and wireless communication devices, aerospace equipment, human–computer interfaces, and sensor networks. These services create our consumer and brand environment and have been contributing extensively and more closely to our life experience, especially the applications in mobile and other embedded devices. With the increasing number of customers who would like to own a ubiquitous multimedia service because of the convenience, the requirements for this kind of service from customers are increasing, such as the quality, speed, and electric consumption. Therefore, the UMAS technologies have become state-of-the-art research topics and are expected to have an important role in the future.

UMAS 2013 aimed to advance ubiquitous multimedia techniques and embedded software and systems research, development, and design competence, and to enhance international communication and collaboration. The workshop covers traditional core areas of media and embedded systems in architecture, software, hardware, real-time computing, and testing and verification, as well as new areas of special emphasis: pervasive/ubiquitous computing and sensor networks, HW/SW co-, wireless communications, power-aware computing, security and data protection, and multimedia.

UMAS 2013 was supported by many people and organizations. We would also like to express our appreciation to the organizers of GPC 2013, especially James J. Park, for their constant support and kind help in the related items of UMAS 2013. Thanks to all the Program Committee members for their valuable time and effort in reviewing the papers. Without their help and advice, this program would not have been successful.

Ching-Nung Yang

UMAS 2013 Organization

General Chair

Ching-Nung Yang	National Dong Hwa University, Taiwan

Program Chairs

Cheonshik Kim	Sejong University, Korea
Eun-Jun Yoon	Kyungil University, Korea
Zhang Xinpeng	Shanghai University, China

Session Chairs

You-Sik Hong	Sangji University, Korea
Jungyeon Shim	Kangnam University, Korea
Woontack Woo	Kwangju Institute of Science and Technology, Korea
Minho Lee	Kyungpook National University, Korea
Kyu-Dae Lee	Kongju National University, Korea
Raylin Tso	National Chengchi University, Taiwan

Steering Committee

Injung Park	Dankuk University, Korea
Yong-Soo Choi	Korea University, Korea
Suk-Hwan Lee	Tongmyung University, Korea
Seongah Chin	Sungkyul University, Korea
Do Hyeun Kim	Jeju National University

Publicity Chairs

Beongku An	Hongik University, Korea
Hyoung-Joong Kim	Korea University, Korea
Kwang-Chun Ho	Hansung University, Korea
Seungcheon Kim	Hansung University, Korea
Hoojin Lee	Hansung University, Korea

Program Committee

Klaus Meißner	Technische Universitat Dresden, Germany
Jinsuk Baek	Winston Salem State University, USA
Xiao Dong Wang	Science & Technology, UK

Wei-Jen Wang	National Central University, Taiwan
Da-Zhi Sun	Tianjin University, China
Jun Jo	Griffith University, Australia
Wayne Pullan	Griffith University, Australia
Vallipuram Muthukkumarasamy	Griffith University, Australia
Junhu Wang	Griffith University, Australia
Alan Liew	Griffith University, Australia
Bela Stantic	Griffith University, Australia
Anne Nguyen	Griffith University, Australia
Soo-Hyun Park	Kookmin University, Korea
Moon-Wan Kim	Tokyo University of Information Sciences, Japan
Tzung-Shi Chen	National University of Tainan, Taiwan
Wei-Chen Cheng	Academia Sinica, Taiwan
Alex M.H. Kuo	University of Victoria, Canada
Truong Hai Bang	University of Information Technology, Vietnam
Ronald L. Hartung	Franklin University, USA
Gyei-Kark Park	National Maritime University, Korea
HyungJun Kim	Hansei University, Korea
Jae Gi Son	KETI (Korea Electronics Technology Institute), Korea
Dongkyoo Shin	Sejong University, Korea
Dongil Shin	Sejong University, Korea
Moon-Goo Lee	Kimpo College, Korea
Myoung Nam Kim	Kyungpook National University, Korea
Young-Sun Im	Kookje College, Korea
Shi Xue Dou	Wollongong University, Australia
Phan Trung Huy	Hanoi University, Vietnam
Byung-Tae Chun	Hankyong University, Korea
Kwang-Baek Kim	Silla University, Korea
Jang-Geun Ki	Kongju National University, Korea
Keehong Um	Hansei University, Korea
Wonil Kim	Sejong University, Korea
Joon-Shik Park	KETI (Korea Electronics Technology Institute), Korea
Masaaki Fujiyoshi	Tokyo Metropolitan University, Japan
Hae-Kyung Seong	Hanyang Women's College, Korea
Young Huh	KEIT(Korea Evaluation Institute of Industrial Technology), Korea
Young-Ho Park	Sejong Cyber University, Korea
Choonsuk Oh	Sun Moon University, Korea
Elena Tsomko	Namseoul University, Korea
Sang-Woon Lee	Nameseoul University, Korea
In-Hwa Hong	KETI (Korea Electronics Technology Institute), Korea

DATICS-GPC 2013: Design, Analysis and Tools for Integrated Circuits and Systems

The International Workshop DATICS-GPC 2013: Design, Analysis and Tools for Integrated Circuits and Systems at the 8th International Conference on Grid and Pervasive Computing took place in Seoul, South Korea, May 9–11, 2013.

The DATICS workshops were initially created by a network of researchers and engineers both from academia and industry in the areas of design, analysis and tools for integrated circuits and systems. Recently, DATICS has been extended to the fields of communication, computer science, software engineering and information technology.

The main target of DATICS-GPC 2013 was to bring together software/ hardware engineering researchers, computer scientists, practitioners and people from industry to exchange theories, ideas, techniques and experiences related to all aspects of DATICS.

The International Program Committee (IPC) of DATICS-GPC 2013 consisted of about 150 experts in the related fields both from academia and industry. DATICS-GPC 2013 was partnered with CEOL: Centre for Efficiency-Oriented Languages (Ireland), Minteos (Italy), KATRI (Japan and Hong Kong), Distributed Thought (UK), ASIC LAB - Myongji University (South Korea), Baltic Institute of Advanced Technology - BPTI (Lithuania), Solari (Hong Kong), Transcend Epoch (Hong Kong) and Xi'an Jiaotong-Liverpool University — XJTLU (China — UK).

The DATICS-GPC 2013 technical program included seven papers that were organized into lecture sessions. On behalf of the IPC, we would like to welcome you to the proceedings of DATICS-GPC 2013.

Ka Lok Man
Nan Zhang

DATICS-GPC 2013 Organization

General Chairs

Ka Lok Man	Xi'an Jiaotong-Liverpool University, China
Nan Zhang	Xi'an Jiaotong-Liverpool University, China

Organizing Chairs

Michele Mercaldi	EnvEve, Switzerland
Chi Un Lei	University of Hong Kong
Tomas Krilavicius	Baltic Institute of Advanced Technologies and Vytautas Magnus University, Lithuania

Program Committee

Vladimir Hahanov	Kharkov National University of Radio Electronics, Ukraine
Paolo Prinetto	Politecnico di Torino, Italy
Massimo Poncino	Politecnico di Torino, Italy
Alberto Macii	Politecnico di Torino, Italy
Joongho Choi	University of Seoul, South Korea
Wei Li	Fudan University, China
Michel Schellekens	University College Cork, Ireland
Emanuel Popovici	University College Cork, Ireland
Jong-Kug Seon	LS Industrial Systems R&D Center, South Korea
Umberto Rossi	STMicroelectronics, Italy
Franco Fummi	University of Verona, Italy
Graziano Pravadelli	University of Verona, Italy
Yui Fai Lam	Hong Kong University of Science and Technology, Hong Kong
Jinfeng Huang	Philips and LiteOn Digital Solutions Netherlands, The Netherlands
Monica Donno	Minteos, Italy
Jun-Dong Cho	Sung Kyun Kwan University, South Korea
AHM Zahirul Alam	International Islamic University Malaysia, Malaysia
Gregory Provan	University College Cork, Ireland
Miroslav N. Velev	Aries Design Automation, USA
M. Nasir Uddin	Lakehead University, Canada
Dragan Bosnacki	Eindhoven University of Technology, The Netherlands

Milan Pastrnak	Atos IT Solutions and Services, Slovakia
John Herbert	University College Cork, Ireland
Zhe-Ming Lu	Sun Yat-Sen University, China
Jeng-Shyang Pan	National Kaohsiung University of Applied Sciences, Taiwan
Chin-Chen Chang	Feng Chia University, Taiwan
Mong-Fong Horng	Shu-Te University, Taiwan
Liang Chen	University of Northern British Columbia, Canada
Chee-Peng Lim	University of Science Malaysia, Malaysia
Salah Merniz	Mentouri University, Constantine, Algeria
Oscar Valero	University of Balearic Islands, Spain
Yang Yi	Sun Yat-Sen University, China
Franck Vedrine	CEA LIST, France
Bruno Monsuez	ENSTA, France
Kang Yen	Florida International University, USA
Takenobu Matsuura	Tokai University, Japan
R. Timothy Edwards	MultiGiG, Inc., USA
Olga Tveretina	Karlsruhe University, Germany
Maria Helena Fino	Universidade Nova De Lisboa, Portugal
Adrian Patrick O'Riordan	University College Cork, Ireland
Grzegorz Labiak	University of Zielona Gora, Poland
Jian Chang	Texas Instruments, Inc., USA
Yeh-Ching Chung	National Tsing-Hua University, Taiwan
Anna Derezinska	Warsaw University of Technology, Poland
Kyoung-Rok Cho	Chungbuk National University, South Korea
Yuanyuan Zeng	Wuhan University, China
D.P. Vasudevan	University College Cork, Ireland
Arkadiusz Bukowiec	University of Zielona Gora, Poland
Maziar Goudarzi	Sharif University of Technology, Iran
Jin Song Dong	National University of Singapore, Singapore
Dhamin Al-Khalili	Royal Military College of Canada, Canada
Zainalabedin Navabi	University of Tehran, Iran
Lyudmila Zinchenko	Bauman Moscow State Technical University, Russia
Muhammad Almas Anjum	National University of Sciences and Technology (NUST), Pakistan
Deepak Laxmi Narasimha	University of Malaya, Malaysia
Danny Hughes	Katholieke Universiteit Leuven, Belgium
Jun Wang	Fujitsu Laboratories of America, Inc., USA
A.P. Sathish Kumar	PSG Institute of Advanced Studies, India
N. Jaisankar	VIT University, India

Atif Mansoor	National University of Sciences and Technology (NUST), Pakistan
Steven Hollands	Synopsys, Ireland
Siamak Mohammadi	University of Tehran, Iran
Felipe Klein	State University of Campinas (UNICAMP), Brazil
Eng Gee Lim	Xi'an Jiaotong-Liverpool University, China
Kevin Lee	Murdoch University, Australia
Prabhat Mahanti	University of New Brunswick, Saint John, Canada
Kaiyu Wan	Xi'an Jiaotong-Liverpool University, China
Tammam Tillo	Xi'an Jiaotong-Liverpool University, China
Yanyan Wu	Xi'an Jiaotong-Liverpool University, China
Wen Chang Huang	Kun Shan University, Taiwan
Masahiro Sasaki	The University of Tokyo, Japan
Shishir K. Shandilya	NRI Institute of Information Science and Technology, India
J.P.M. Voeten	Eindhoven University of Technology, The Netherlands
Wichian Sittiprapaporn	Mahasarakham University, Thailand
Aseem Gupta	Freescale Semiconductor Inc., Austin, USA
Kevin Marquet	Verimag Laboratory, France
Matthieu Moy	Verimag Laboratory, France
Ramy Iskander	LIP6 Laboratory, France
Suryaprasad Jayadevappa	PES School of Engineering, India
Shanmugasundaram Hariharan	Pavendar Bharathidasan College of Engineering and Technology, India
Chung-Ho Chen	National Cheng-Kung University, Taiwan
Kyung Ki Kim	Daegu University, South Korea
Shiho Kim	Chungbuk National University, Korea
Hi Seok Kim	Cheongju University, Korea
Brian Logan	University of Nottingham, UK
Asoke Nath	St. Xavier's College (Autonomous), India
Tharwon Arunuphaptrairong	Chulalongkorn University, Thailand
Shin-Ya Takahasi	Fukuoka University, Japan
Cheng C. Liu	University of Wisconsin at Stout, USA
Farhan Siddiqui	Walden University, Minneapolis, USA
Katsumi Wasaki	Shinshu University, Japan
Pankaj Gupta	Microsoft Corporation, USA
Taikyeong Jeong	Myongji University, South Korea
Masoud Daneshtalab	University of Turku, Finland
Amit Chaudhry	Technology Panjab University, India
Bharat Bhushan Agarwal	I.F.T.M., University, India

Abhilash Goyal	Oracle (SunMicrosystems), USA
Yue Yang	EJITEC, China
Boguslaw Cyganek	AGH University of Science and Technology, Poland
Yeo Kiat Seng	Nanyang Technological University, Singapore
Youngmin Kim	UNIST Academy-Industry Research Corporation, South Korea
Tom English	Xlinx, Ireland
Nicolas Vallee	RATP, France
Mou Ling Dennis Wong	Swinburne University of Technology, Malaysia
Rajeev Narayanan	Cadence Design Systems, Austin, USA
Xuan Guan	Freescale Semiconductor, Austin, USA
Pradip Kumar Sadhu	Indian School of Mines, India
Fei Qiao	Tsinghua University, China
Chao Lu	Purdue University, USA
Ding-Yuan Cheng	National Chiao Tung University, Taiwan
Amir-Mohammad Rahmani	University of Turku, Finland
Shin-Il Lim	Seokyeong University, Seoul Korea

Message from the FSTA 2013 Symposium Chair

Welcome to the proceedings of the 2013 International Workshop on Future Science Technologies and Applications (FSTA 2013).

The Internet as well as cellular and wireless systems are now converging, thus giving birth to the future Internet. The International Workshop on FSTA 2013 brought together scientists, engineers, computer users, and students to exchange and share their experiences, new ideas, and research results on all aspects (theory, applications and tools) of computer and information science, and to discuss the practical challenges encountered and the solutions adopted. The Workshop on Future Science Technologies and Applications aims to serve as an international forum for researchers and practitioners willing to present their early research results and share experiences in the field.

FSTA 2013 contained high-quality research papers submitted by researchers from all over the world. Each submitted paper was peer-reviewed by reviewers who are experts in the subject area of the paper. Based on the review results, the Program Committee accepted 13 papers.

In organizing an international workshop, the support and help of many people is needed. We would like to thank all authors for their work and presentation, all members of the Program Committee and reviewers for their cooperation and time spent in the reviewing process. Particularly, we thank the founding Steering Chair of GPC 2013, James J. (Jong Hyuk) Park, Workshop Chair GPC 2013, Joon-Min Gil, and the Program Chair of GPC 2013. Finally, special thanks are extended to the staff of FSTA 2013, who contributed greatly to the success of the conference.

Namje Park

FSTA 2013 Organization

Steering Chair

James J. Park	SeoulTech, Korea

General Chairs

Stefanos Gritzalis	University of the Aegean, Greece
Young-Sik Jeong	Wonkwang University, Korea
Han-Chieh Chao	National Ilan University, Taiwan

Program Chair

Namje Park	Jeju National University, Korea

Publicity Chairs

Deqing Zou	Huazhong University of Science and Technology, China
Damien Sauveron	University of Limoges, France
Neil Y. Yen	The University of Aizu, Japan

Program Committee

Qiang Zhu	The University of Michigan, USA
Shao-Shin Hung	WuFeng University, Taiwan
Tat-Chee Wan	Universiti Sains Malaysia
Jun Xiao	East China Normal University, China
Young soo Kim	Electronics and Telecommunications Research Institute, Korea
Oliver Amft	Eindhoven University of Technology, The Netherlands
Changjing Shang	Aberystwyth University, UK
Wanqing Tu	Glyndwr University, UK
Kok-Seng Wong	Soongsil University, Korea
Narayanan Kulathuramaiyer	Universiti Malaysia Sarawak, Malaysia
Nik Bessis	University of Derby, UK
Qinghe Du	Xi’an Jiaotong University, China
Kyoil Cheong	Electronics and Telecommunications Research Institute, Korea

Rachid Anane	Coventry University, UK
Mehul Bhatt	University of Bremen, Germany
Zhou Su	Waseda Univ, Japan
Chee-Peng LIM	Deakin University, Australia
Rami Yared	Japan Advanced Institute of Science and Technology, Japan
M. Elena Renda	IIT-CNR, Italy
Dooho Choi	Electronics and Telecommunications Research Institute, Korea
Youngsook Lee	Howon University, Korea
Soohyun Oh	Hoseo University, Korea
Taesung Lee	KISA, Korea

Table of Contents

Cloud, Cluster and Grid I

Cloud, Cluster and Grid II

Middleware, Resource Management

Mobile, Peer-to-Peer and Pervasive Computing

Multi-core and High Performance Computing

Other GPC Related Topics

Parallel and Distributed Systems

Security and Privacy

Ubiquitous Communications, Sensor Networking, and RFID

Ubiquitous and Multimedia Application Systems

Design, Analysis and Tools for Integrated Circuits and Systems

Future Science Technologies and Applications

Green and Human Information Technology

Erratum

Transparency in Cloud Business: Cluster Analysis of Software as a Service Characteristics

Jonas Repschlaeger

Technical University of Berlin, Chair of Information and Communication Management,
Straße des 17. Juni 135, 10623 Berlin, Germany
j.repschlaeger@tu-berlin.de
www.ikm.tu-berlin.de

Abstract. Cloud Computing shapes the IS Outsourcing landscape and enables new flexible delivery models. It has become a fast growing and non-transparent market with many providers, including heterogeneous service portfolios and business models, especially for Software as a Service (SaaS). Many researchers focus exclusively on the technical aspects of Cloud Computing and ignore the business perspective. Unfortunately, the terms Cloud Computing and SaaS are not defined clearly and face customers with several challenges related to the decision-making process. This article explores the nature of SaaS from a business point of view and examines 100 providers in order to gain new insights about the transparency of their service offerings. A cluster analysis is conducted to examine dependencies between different provider information. The results indicate that only basic data like contact information, provider profile and service functionality are provided by all vendors, whereas pricing, support and security information are only covered by half of the providers.

Keywords: Cloud Computing, Vendor Evaluation, Software as a Service, Service Transparency, Cluster Analysis.

1 Introduction

Cloud Computing has emerged as a new IT paradigm that promises elastic and flexible deliverance of IT resources provided by pooled resources through a network [1]. Foster et al. (2008) add that "[…] Cloud Computing is a specialized distributed computing paradigm […]" where the physical infrastructure is normally distributed over virtual layers/multiple machines and/or data centers, and the customer does not know the exact data location [2].For many, it has the potential to change the way organizations and individuals use IT resources [3]. Yet, uncertainty about benefits and risks still prevent companies from making use of Cloud Computing [4]. Cloud Computing enables a shift of the software market and related business models towards mass-customized and on-demand services. Instead of purchasing licenses, the software is provided as a service over the Internet, owned and managed remotely by the vendor [5]. The Software as a Service (SaaS) model evolved from the application service providing (ASP) with a revenue worldwide of $22.1 billion in 2012[6]. This

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 1–10, 2013.

continuous growth within the enterprise application markets leads to an increased amount of SaaS vendors. Currently, the market of SaaS contains over 650 different small and large providers (see also market research in chapter 3). For future research, especially methodologies for assessing Cloud services and comparing offerings from different providers will become important [7].

This article examines the transparency of SaaS offerings and the access to relevant information. Section 2 starts with a definition of Cloud Computing, presents the state of art regarding Cloud provider evaluation and summarizes the SaaS evaluation dimensions used for this article. The next section describes the research approach used to evaluate the SaaS vendors. The results are presented in section 4 and close up with a discussion of implications in section 5.

2 Characteristics of Software as a Service

Despite being a relatively young paradigm, several definitions exist for Cloud Computing so far, varying in scope and precision. However, recently the definition provided by the American National Institute of Standards and Technology (NIST) [8] is accepted by many practitioners and researchers (e.g.[9]).

2.1 Characteristics of Cloud Computing

Cloud resources (e.g. networks, servers, storage, applications and services) are offered in a scalable way via the Internet without the need for any long-term capital expenditures and specific IT knowledge on the customer's side. It is possible to obtain complete software applications or the underlying IT infrastructure in the form of virtual machine images. Basically, Cloud Computing consists of three levels: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).The National Institute of Standards and Technology defines five essential characteristics of Cloud Computing, which are applicable to assess the Cloud capability of SaaS [8]:

- On-Demand Self-Service (computing capabilities, such as server time and network storage, can be booked automatically without requiring human interaction with the service provider)
- Broad Network Access (capabilities are available over the network and accessed through standard mechanisms)
- Resource Pooling (computing resources are pooled using a multi-tenant model with different physical and virtual resources)
- Rapid Elasticity (ability to increase or decrease computing resources at an unlimited scale)
- Measured Service (to automatically control and optimize resource-use by leveraging a metering capability)

2.2 Evaluation of Cloud Providers

Cloud Computing has become a fast growing and non-transparent market with many small and large providers, each of them having their specific service model. Unfortunately, this makes it difficult to compare the providers with each other as well as their service offerings. In the majority of cases the service portfolios are heterogeneous and complex. In current literature there are attempts to classify the characteristics of Cloud vendors and to evaluate them (e.g. [10], [11], [12], [13]).

Martens et al. (2011) define a maturity model for the quality assessment of Cloud Computing services and describe the relationships between Cloud services, SLAs, technical implementation and provider characteristics[13]. The evaluation criteria are limited, focused on the maturity level of the provider and do not cover relevant characteristics like pricing or provider reputation.

Kaisler et al. (2012) study the service migration into the Cloud Computing environment by examining security and integration issues associated with service implementation [12]. The presented framework addresses 15 decision categories divided equally into three groups: application architecture, system architecture and service architecture. Unfortunately, the decision categories are based on a literature review and are not evaluated. Nevertheless, the presented framework covers most of the general provider characteristics.

Hetzenecker et al. (2012) develop a model for assessing requirements of Cloud providers based on literature analysis and expert interviews [11]. The model consists of 41 requirements grouped by the categories information security, performance and usability, costs, support and cooperation, as well as transparency and organization of the provider. Most of the provider characteristics are covered but the model does not show the relationship to the Cloud service models (SaaS, PaaS and IaaS) and their relevance.

Mahesh et al. (2011) provide a framework to evaluate Cloud Computing [14] and to discuss cost savings, technology insurance and security risks. However, the article focuses on the general make-or-buy decision and does not provide any criteria to evaluate a Cloud provider.

Aparicio et al. (2012) present a methodology to compare and choose cloud services [15]. The provided categories describe the suitability, the economic value, the control mechanisms, the usability, the reliability and the security of the service, including a total of 29 criteria. The criteria cover the general provider characteristics but are not evaluated regarding their completeness.

Repschlaeger et al. (2012) present a Cloud requirement framework which concentrates on relevant requirements for adopting Cloud services, targeting all three service models (SaaS, PaaS, IaaS) [10]. The framework consists of six target dimensions (costs, scope & performance, IT security & compliance, flexibility, reliability & trustworthiness, Service & Cloud Management) to group and to structure provider characteristics. Each target dimension represents a general objective from a customer's point of view. The provider characteristics are summarized by 21 abstract requirements and 62 evaluation criteria which are assigned to the target dimensions.

2.3 Evaluation Criteria for SaaS

For this article the research framework by Repschlaeger et al. (2012) is used due to its maturity and extent. This chapter provides an overview of the six main evaluation categories. For further information see [10].

Evaluation Dimension: Flexibility

A common advantage of Cloud Computing, identified in science and industry, is the gain in flexibility compared to traditional solutions. Flexibility describes the ability to respond quickly to changing capacity requirements. Resources can be allocated and de-allocated as required and the provisioning time is shorter compared to traditional outsourcing such as ASP. Additionally, the contract duration with a Cloud vendor is shorter. This evaluation dimension contains operationalized criteria important for the NIST criteria “On Demand Self-Service”, “Broad Network Access” and “Rapid Elasticity”.

Evaluation Dimension: Costs

The decision to choose Cloud Computing and a particular provider is often guided by monetary considerations and linked with the slogan "pay-as-you-use". Customers who decide to use Cloud services mostly benefit by small capital commitment, low acquisition costs for required servers, licenses or necessary hardware space and reduced complexity of IT operations. However, the pricing and billing models often differentiate between each provider, making it difficult for comparison. This evaluation dimension contains operationalized criteria relevant for the NIST criterion “Measured Service”.

Evaluation Dimension: Scope and Performance

This target dimension describes the scope of services and the performance of a Cloud provider. In order to select the appropriate provider which meets the requirements best, knowledge about their service and performance is of crucial importance. The manageability (usability) of services and the degree of customization (to which extent the service can be adapted), especially in a distributed IT architecture, are essential features. This evaluation dimension contains operationalized criteria important for the NIST criterion “On Demand Self-Service”.

Evaluation Dimension: IT Security and Compliance.

The decision on selecting a provider in the Cloud is also influenced by company and government requirements in the areas of security, compliance and privacy. Customers must be assured that their data and applications, even operated in the Cloud, meet both compliance guidelines required and are adequately protected against unauthorized access. This evaluation dimension contains operationalized criteria important for the NIST criterion “Resource Pooling”.

Evaluation Dimension: Reliability and Trustworthiness.

This target dimension summarizes criteria regarding the availability and conditions of Cloud services, for instance, Service Level Agreements (SLAs). The liabilities given by the provider and the reliability to keep these conditions are important. In contrast to the commitment the trustworthiness describes the provider's infrastructural features, which may be the evidence of a high reliability. These include disaster recovery, redundant sites or certifications. This evaluation dimension contains operationalized criteria important for the NIST criteria “Broad Network Access” and “Resource Pooling”.

Evaluation Dimension: Service and Cloud Management

The service & Cloud management includes features of the provider that are substantial for appropriate Cloud service operations. These include the support offered by the provider, e.g. consulting services during the implementation phase or support during service operation. Additionally, the monitoring of Cloud services is covered by this dimension. This evaluation dimension contains operationalized criteria important for the NIST criteria “On Demand Self-Service” and “Measured Service”.

3 Research Approach

This article follows a behavioral research approach using a quantitative analysis. By means of market studies, business publications of the Cloud market and an extensive Internet search 651 providers for SaaS are detected. The providers are located mostly in the U.S. (44%) followed by Germany (23%) and the UK (13%). Based on the criteria from Repschlaeger et al. (2012) 100 providers are evaluated. Therefore, a gradual approach is chosen. The evaluation process starts with an evaluation of the information provided on the provider’s website. The websites are examined regarding the availability of information of the Cloud vendor and its services. Secondly, Cloud services from the providers are tested for several hours as long as there are free or trial-accounts available to gather further information. Finally, missing information are requested directly (via email) from the provider. All responses from the vendors are collected and evaluated for a period of two weeks.

The SaaS market offers a wide range of services for several business needs. Most popular SaaS types are for collaboration and personal productivity purposes (overlapping market share 30%, e.g. ClickMeeting or Podio), customer relationship management (23%, e.g. MaximizerCRM or SalesCloud), project management (20%, e.g. ProWorkflow or InfoFlo) and content management (20%, e.g. Curata or Backbase).[1] The detailed examination of 100 providers covers at least 10 of these SaaS types.

The data is analyzed using a clustering approach. A cluster analysis is a quantitative method of classification in order to group objects based on the characteristics they possess [16]. During the analysis of data sets, it is attempted to maximize the

[1] Based on the conducted market analysis (n=651).

homogeneity of objects within the clusters while maximizing the heterogeneity between the clusters [16]. Several researchers propose to use a combination of hierarchical and non-hierarchical clustering techniques in a two-stage procedure where a hierarchical algorithm is used to define the number of clusters and the results serve as the starting point for a subsequent non-hierarchical clustering [16]. Therefore, a hierarchical cluster analysis using the Ward's algorithm followed by the non-hierarchical clustering procedure of k-means is used.

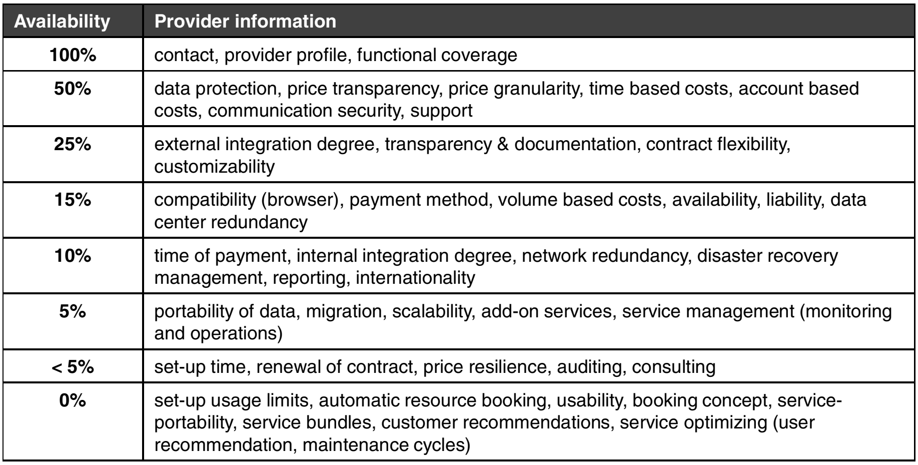
The information transparency is described by three levels. The first level of information represents unavailable data. The second level describes general but not detailed data, for instance marketing statements or press releases. Third level information are more detailed and provide the customer with sufficient data to evaluate one criterion, e.g. most pricing information are of third level type. Since the cluster analysis requires alpha-numeric values the information level is transformed into suitable values.

4 SaaS Business Transparency

4.1 First Evaluation Step: Information on Provider Website

In order to get information about a service, usually, the first step is to visit a provider's website. Depending on the complexity of the website, this process is more or less time consuming, but a fast way to get relevant information. Unfortunately, the results of the first evaluation step provide only information for 20% of the criteria, and 5% of this information are only second level type. Despite a high standardization degree of SaaS and the self-service principle the information on the website is scarce. The lack of crucial information makes it difficult for a customer to compare and to evaluate services and providers. Nevertheless, all providers contain data about their contact possibilities, their general company profile and their service functionality. These basic data enable customers to get in touch with the provider and get a first impression. Additionally, half

Table 1. Information provided by SaaS vendor's website



Availability	Provider information
100%	contact, provider profile, functional coverage
50%	data protection, price transparency, price granularity, time based costs, account based costs, communication security, support
25%	external integration degree, transparency & documentation, contract flexibility, customizability
15%	compatibility (browser), payment method, volume based costs, availability, liability, data center redundancy
10%	time of payment, internal integration degree, network redundancy, disaster recovery management, reporting, internationality
5%	portability of data, migration, scalability, add-on services, service management (monitoring and operations)
< 5%	set-up time, renewal of contract, price resilience, auditing, consulting
0%	set-up usage limits, automatic resource booking, usability, booking concept, service-portability, service bundles, customer recommendations, service optimizing (user recommendation, maintenance cycles)

of the evaluated SaaS vendors provide information about their pricing, service billing and support (see Table 1). Due to its relevance for the customer further information is given about the data protection mechanisms and communication security.

A correlation analysis is conducted to reveal information dependencies between the criteria. Correlations can be found between 18 criteria (see Table 2). Some correlations are not surprising and can be explained due to the similarity of the criteria. For instance, when a provider offers information about the contract flexibility, they also provide information about the renewal conditions. The same applies for the price transparency and the price granularity.

The costs for the usage of SaaS can be charged in different ways. The most popular one is a usage independent charging based on accounts. Alternatively, the services can be charged by the used volume or the time period. The correlation analysis shows that providers which offer a time based charging also provide the user with detailed pricing information. An account based model does not require very detailed pricing information due to its simplicity of charging whereas volume based or time based charging models are more complex and often not self-explanatory.

Table 2. Significant correlations between available information

Evaluation dimension	Correlation type	Service Criterion A	Service Criterion B
Flexibility	Positive, bilateral	internal integration degree	transparency & documentation
Flexibility	Positive, bilateral	contract flexibility	renewal of contract
Costs	Positive, bilateral	price transparency	price granularity
Costs	Positive, bilateral	time based costs	price transparency
Costs	Positive, bilateral	time based costs	price granularity
Scope & Performance	Positive, bilateral	customizability	add-on services
IT Security & Compliance	Positive, bilateral	data protection	communication security
Reliability & Trustworthiness	Positive, bilateral	network redundancy	disaster recovery management
Service & Cloud Management	Positive, bilateral	service management (operations)	consulting

4.2 Second Evaluation Step: Trial Account and Testing

The concept of SaaS is an easy to use and on-demand access to the service. There is no need to download a client and only a browser with common plug-ins for java or flash is required. The possibilities for a new customer are threefold and offer a service completely for free (18%), for a free trial period (42%) or provide only a demonstration on the website (40%). The second evaluation step is more time consuming and requires much more effort by the customer. However, this evaluation is necessary to get information about several criteria the vendor cannot provide. This way, especially information of the flexibility and scope&performance dimension is recorded. During the test period, information for the following criteria could be found: usability, compatibility, documentation, interoperability (internal and external integration), set up time, provisioning time, functionality, add-on services, and customizability.

4.3 Third Evaluation Step: Direct Contact Request

The last evaluation step involves a direct contact to the provider. Therefore, an email is sent to the provider requesting further information about criteria, which were not covered during previous evaluation steps.

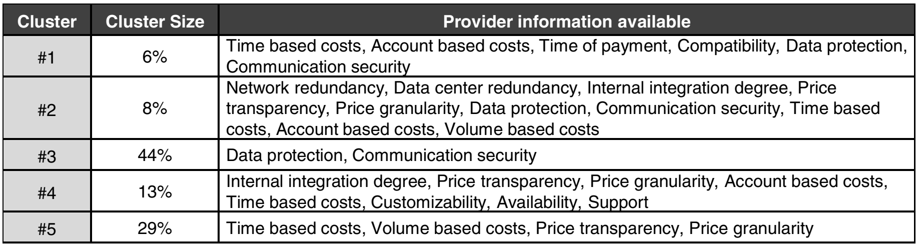
Only answers within a two week period are considered. The willingness to respond to the requests is low. Only 30% of the providers reply, and this without providing any relevant information. This low response rate can be explained by the principle of SaaS, which does not comprise a deep customer-provider relationship. This may be one elementary difference to IS outsourcing. The priority of SaaS providers lies on supporting their current customers and users instead of helping potential customers within their decision-making process. The author assumes that the willingness to communicate may be higher if the request comes from a large company.

4.4 Clustering of SaaS Providers

Based on the availability of information the providers are grouped by using a clustering procedure. The final cluster solution shows five clusters and their characteristic information (see Table 3). Each cluster provides information regarding functionality, provider profile and contact data. Cluster one, cluster two and cluster four provide the customer with the most relevant information, but represent only 27% of SaaS providers. The largest groups are cluster three and cluster five. These clusters provide information either related to the costs dimension or regarding the IT security and compliance dimension.

Most of the information available is related to costs and security issues. As long as a customer takes these two dimensions into account for his decision, the information level is sufficient. However, for more specific information requests, for instance related to service interoperability, much more effort is required, because this information is not available on the provider's website.

Table 3. Providers grouped by information availability



Cluster	Cluster Size	Provider information available
#1	6%	Time based costs, Account based costs, Time of payment, Compatibility, Data protection, Communication security
#2	8%	Network redundancy, Data center redundancy, Internal integration degree, Price transparency, Price granularity, Data protection, Communication security, Time based costs, Account based costs, Volume based costs
#3	44%	Data protection, Communication security
#4	13%	Internal integration degree, Price transparency, Price granularity, Account based costs, Time based costs, Customizability, Availability, Support
#5	29%	Time based costs, Volume based costs, Price transparency, Price granularity

5 Conclusion

The objective of this article is to obtain new findings about the transparency level of SaaS providers. Therefore, the information availability on the websites, via service tests and provider requests is examined. Especially the possibility to get up-front information directly from the provider is low. Five groups of providers are derived based on available information. The results show that basic data like contact information, provider profile and service functionality are provided by all vendors. However, much of the relevant information is not provided. For instance, information regarding interoperability, set-up time or contract conditions is scarce. Pricing and security information is covered by only half of the providers. This lack of transparency makes it challenging for customers to compare SaaS and to decide. An appropriate decision is possible as long as only costs and security aspects are considered.

As with any research, this study does have some limitations. First, to specify the level of detail for the information was challenging. To differentiate between helpful information and general marketing news was sometimes difficult. Furthermore, the response rate during the third evaluation was very low. The reason for that may be due to the fact that the email sent requested too much information. Especially the provider responses may be an interesting future research topic. In which way are responses from Cloud providers influenced?

SaaS has been one of the fastest growing markets and is characterized through many providers with differences in quality and transparency. Due to the self-service concept it will be important for providers to offer easy to use and transparent services as well. The author expects that providers which remain non-transparent for the customer will not succeed in this highly dynamic and customer-driven market. The transparency is not the only success factor but it is important to inspire trust and win over the customer to choose the service provided. Therefore, the author recommends further research in the fields concerning the influencing factors of trust in Cloud Computing or the relevance of provider information and the impact on the customer decision.

References

1. Koehler, P., Anandasivam, A., Ma, D., Weinhardt, C.: Customer Heterogeneity and Tariff Biases in Cloud Computing. In: International Conference on Information Systems (ICIS), Saint Louis, USA (2010)
2. Foster, I., Zhao, Y., Raicu, I., Lu, S.: Cloud Computing and Grid Computing 360-Degree Compared. In: Grid Computing Environments Workshop (GCE), pp. 1–10 (2008)
3. Leimeister, S., Riedl, C., Böhm, M., Krcmar, H.: The Business Perspective of Cloud Computing: Actors, Roles and Value Networks. In: 18th European Conference on Information Systems (ECIS), Pretoria, South Africa (2010)
4. Benlian, A.: A transaction cost theoretical analysis of software-as-a-service (SAAS)-based sourcing in SMBs and enterprises. In: 17th European Conference on Information Systems, pp. 25–36 (2009)
5. Xin, M., Levina, N.: Software-as-a Service Model: Elaborating Client-Side Adoption Factors. In: International Conference on Information Systems (ICIS), p. 86 (2008)
6. Pettey, C., van der Meulen, R.: Gartner Says Worldwide Software-as-a-Service Revenue to Reach $14.5 Billion in 2012 (2012)
7. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., Ghalsasi, A.: Cloud computing — The business perspective. Decision Support Systems 51, 176–189 (2011)
8. Mell, P., Grance, T.: The NIST Definition of Cloud Computing (2011)

9. Martens, B., Walterbusch, M., Teuteberg, F.: Costing of Cloud Computing Services: A Total Cost of Ownership Approach. In: 45th Hawaii International Conference on System Science, pp. 1563–1572 (2012)
10. Repschläger, J., Zarnekow, R., Wind, S., Turowski, K.: Cloud Requirement Framework: Requirements and Evalutation Criteria to Adopt Cloud Solutions. In: 20th European Conference on Information Systems (2012)
11. Hetzenecker, J., Kammerer, S., Amberg, M., Zeiler, V.: Anforderungen Cloud Computing Anbieter. In: Tagungsband der Multikonferenz Wirtschaftsinformatik (MKWI), Braunschweig, Germany (2012)
12. Kaisler, S., Money, W., Cohen, S.: A Decision Framework for Cloud Computing. In: 45th Hawaii International Conference on System Science, pp. 1553–1562 (2012)
13. Martens, B., Teuteberg, F., Gräuler, M.: Design and implementation of a community platform for the evaluation and selection of cloud computing services: a market analysis. In: 19th European Conference on Information Systems (2011)
14. Mahesh, S., Landry, B.J.L., Sridhar, T., Walsh, K.R.: A Decision Table for the Cloud Computing Decision in Small Business. Information Resources Management Journal 24, 9–25 (2011)
15. Aparicio, M., Costa, C.J., Reixa, M., Costa, C.: Cloud services evaluation framework. In: Proceedings of the Workshop on Open Source and Design of Communication, OSDOC 2012, pp. 61–69. ACM Press (2012)
16. Hair, J.F.: Multivariate data analysis. Pearson, Upper Saddle River (2010)

Distributed Accounting in Scope of Privacy Preserving

Marcus Hilbrich and René Jäkel

Center for Information Services and High Performance Computing (ZIH),
Technische Universität Dresden
marcus.hilbrich@tu-dresden.de

Abstract. Accounting is an essential part of distributed computing infrastructures, regardless whether these are more service-driven like Clouds or more computing oriented like traditional Grid Computing environments. Those infrastructures have evolved over more than the last decade and additional. beside the further development towards service-oriented architectures, the business aspect of especially Cloud Computing solutions becomes more and more relevant. In this paper we focus on user-centric aspects like privacy preserving methods to hide the users behaviour and to collect only necessary information for billing, under the assumption that an accounting system has to be integrated in the computing infrastructure and that a central interface is still desirable for billing and financial clearing.

1 Introduction

Nowadays, it is a quite common to pay just for metered services (pay-per-use) which are consumed for a specific task but having potentially a large resource share on hand. The payment is usually done by spending money, but could also realized by giving services in return or the promise that the work is relevant for a scientific community or a wider society. Meanwhile, different payment options in terms of pricing models have been developed, from simple flat rates (pay once and take what you need) to pay per request depending on the answer of the request and the number of requests.

As long as services of a single resource provider are used the payment has to be done straightforward. The provider logs what users or customers are using and tells them what they have to pay. If services have to be combined from different providers, or even service brokers, the accounting and billing issues are getting more complicated. It is obvious, that in such cases the billing of individual operators in a long service chain is not very comfortable, since in the common case direct contract between resource operators and users is needed.

Usually in an accounting and billing service in a distributed environment such as Grid, all accounting data are collected and transferred from all service providers to a central place. Based on this data pile bills are written and statistic information are created. The danger in this approach is that the operator of the accounting and billing service has a lot of sensitive information by hand, and it has to be guaranteed that privacy preserving issues for the users hold (usually done by contracts ore agreements).

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 11–22, 2013.

Our approach of an accounting concept is based on the prevention of such a central component to store all accounting information in a centralised manner. The accounting data are kept in the domain of the service operators and accumulate only coarsely granular data. In this way we realise billing and accounting without a need on transparent users.

2 Related Architectures

Accounting and billing is already done by various systems. The field of research we are working on are mainly covered by Grids and Clouds. Grid services are usually offered to a Virtual Organisation (VO) which allows its users to access services in its domain. Often the VO cares also of scheduling, billing and user support. The accounting systems tend to use centralised accounting databases, such as LUTS [1] or DGAS [2]. Accounting data are read e.g. from the batch system and are transferred to a central database. Afterwards the rights to access parts of the data are assigned to the users of the accounting system. SGAS [3] stores accounting data on VO servers to improve scalability. Common for all these systems is that accounting data are moved from the resource provider, which are the creator of the data, to a central component.

A quite new and emerging field of interest are federated Clouds, for which more advanced accounting concepts are needed in terms of privacy preserving. This kind of cooperating Cloud is not yet a way of Cloud usage, beside direct services or infrastructure utilization. In most cases, there are isolated Cloud provider [4], which can led to the widely discussed vendor-lock problem [5]. The manifold drawbacks (e.g. proprietary data formats which hinder exporting data and unexpected price changes or closing down of essential services) are already known [6] and different concepts were developed to overcome this limitations.

One initiative towards an Open Cloud is developed by the *Open Cloud Manifesto*[1], which is a loose group of companies and projects to communicate demands and solutions for an Open Cloud.

In recent years more and more Open Source Cloud middlewares evolved, e.g OpenNebula [7] or Eucalyptus [8], to name only some prominent projects. Those enterprise solutions also support interfaces to established commercial cloud service providers, such as the EC2 interface introduced by Amazon. On the other hand they also follow the recent Open Cloud Computing Interface (OCCI)[2], which represents a RESTful protocol and development API. The development of this interface is driven by community users and has some history in distributed computing and particular in grid computing. By introducing a flexible interface the interoperability between different cloud providers can be increased. Therefore, the migration of applications or services from one provider to a different one becomes relatively easy, which is a huge step avoiding the vendor locked-in problem on the way towards a common Cloud Computing standard.

[1] Details via the Open Cloud Manifesto: `http://www.opencloudmanifesto.org`

[2] Listed projects and details can be found on their website: `http://occi-wg.org/`

A Hybrid Cloud [9] combines different cloud resources (in most cases a local or private Cloud and a public Cloud). This allows to schedule the users requests (e.g. to run a job or to access a service) based on the job description and a set of rules (constraints where to run the job, available budget for external resources etc.) on one of the Cloud resources. The decision which cloud is used can be delegated to a Cloud broker [10]. In this case a user (e.g. a company or a scientific community) asks a so called broker, which is the best Cloud provider to run a specific job at a given time. Therefore, the broker gets the actual service description of different Cloud providers and ranks them according to the needs of the users. The user has to sign at least two contracts. The first with the broker and the second with the Cloud provider. If more then one Cloud provider is needed to complete a task (e.g. one for storing data and one for computing) additional contracts have to be closed. Additional conditions to drive such an architecture is to use compatible APIs to access the different Cloud providers and to offer similar services. This can be achieved by standardisation of services and their description. A wildly accepted framework to compare services is not yet established but there is already research (e.g. [11]) how service comparison can be realised. Also a standardisation of describing Cloud services and their performance has to be found and an automatic way of closing contracts has to be introduced. The service description could be given by Service Level Agreements (SLAs) which are automatically signed for using a service as described by [12].

A federated Cloud creates a market for resources and potentially deals accounting and billing issues. This means every user and resource provider has a contract with the federation instance for providing or utilising resources, but it is not necessarily needed that the user has a contract with a resource provider. This allows supplier which use services or resources of other providers to offer more complex products or to provide services independent of resources and to select different resources for a service depending on the kind of data (related to real persons, anonymized data or data publicly available) to process [13]. In such a system the billing and financial clearing has to be done by the federation and accounting data has to be recorded on the resource provider. The general demands on such an accounting system are presented by [14] and [15] the specifics of federated Clouds are covered by [16].

The specific concept of federated Clouds with a widely accepted use case are so called Government Clouds. A Government Cloud is a Cloud-based systems to handle the computing and storage needs of administrative agencies. The resources could be public Clouds, private Clouds driven by the government or a Cloud provider, or local data centres of agencies, which form a federation to share there resources with other agencies. The advantage of such a concept compared with the direct use off local resources at each agency is that local peak demands can be resolved by using resources of other agencies. This allows to reduce the overall amount of resources, which are needed to process the given governmental tasks. One of the challenges for a Government Cloud is to respect several juristic limitations. This limitations depend on the data which has to be processed, and therefore the according service requests have to be categorised, e.g. if specific data needed for a service execution are not allowed to be transferred to a different site. Such restrictions can be constraints on the security level, e.g. this is a common requirement of legislation in federal states like Germany or the European Union. This shall ensure that data handling stays in the same juristic domain and that the data douse not leave the domain of governmental controlling authorities. For instance the Japan Kasumigaseki Cloud[3] has to deal with this juristic limitation. Some data have to be processed at the district the agency is located. This demands that a computing centre has to be located at each district. To realise a compensation between the agencies an accounting system has to be established. The concept of our accounting system could be deployed to such an infrastructure. Similar to the Kasumigaseki Cloud a computing infrastructure could be deployed for India [17]. For both Clouds our accounting approach can be considered.

3 Data Minimisation and Privacy Preserving

Data minimisation and privacy preserving for users is not a major topic of common accounting systems on a technical level. Data minimisation is a concept to protect privacy of users by reducing data to a minimal level, which is essentially needed to realise the accounting service. This can be realised by deleting data, which is not longer needed or by storing data only in a non-personal way. This can be illustrate by the following examples:

- Someone prefers to by products of a special brand from an online seller, which could result in a handicap, if the particular person tries to apply for a job in a company of an competing brand.
- The information of that someone buy food that is considered unhealthy, or that this person buys medicine, could be used by an insurance company to tend to increase the insurance rate.
- To do overtime can be interpreted as health risk, or that persons are not very good in their particular job.
- Buying products or searching for keywords which categorise someone in your family as pregnant could influence the credit rating, or could turn into a handicap to apply for certain jobs.

All those information could be extracted from your daily behaviour by operators of third-party services. In most cases the information are not simply to spy users, since there is usually a trustworthy relation between the users and the information holder. But there are situations in which these collected information could eventually passed to a different authority, even without the knowledge of the users, either by simply selling them to other companies, or if a company is sold or goes out of business. In the later cases the originally trustworthy relation has ended, but the sensitive user data are still present.

The given example describes a complex problem with a few words. Users leave digital footprints, which are individually not meaningful, but by combining all these

[3] More information on Kasumigaseki Cloud:
`http://www.cloudbook.net/directories/gov-clouds/gov-program.php?id=100016`

single footprints valuable information might be eventually extracted with profiling techniques at a later stage. Furthermore, this profiling can even led to the categorisation of users to groups with similar behaviour by so called group profiling [18].

The categorisation could be even more problematic than to extract single information, because the ranking of individuals is therefore typically dependent on group parameters. In other words, the individual might get disadvantaged by sharing this group, whose parameters are based on specific algorithms but eventually effected by statistical fluctuations. Additionally, this process is not transparent to users of the system. In an extreme example, the credit-risk of a person for a contract could be based on those group characteristics for which the person is member of, such as its place of residence or age [19]. In a similar manner to the given example the accounting data of the daily work of users can be interpreted to get information about their behaviour, including daily work habits, e.g. how the work proceed or simply that overtime is needed each second weekend. If users are not informed about the profiling they have no chance to check the results and have to live with the consequences.

More generally, there is a need to deal with the right to informational self-determination in an appropriate manner. In short, informational self-determination is the right of an individual to control which personal information are used under which circumstances. This right was first formulated as a discrete right [20] by the German Bundesverfassungsgericht [21]. Nowadays, similar rights are established e.g. for the European Union and United States of America [22].

In case of scientific communities, there is no direct commercial interest of categorising people. E.g. the D-Grid (the German Grid community) uses resources of data centres of universities, which are in principal operated in the same way and therefore, the universities as resource providers have to respect the right to informational self-determination. This means there is the demand to avoid that detailed personal or project related information can be extracted out of the users behaviour, in particular if an external provider is used.

In the academic domain the user groups are rather manageable, limited in number and not highly dynamic. But there is also the trend to combine computing infrastructures over institutional boundaries (e.g. in Grids) or incorporate other service providers.

In this context, accounting data are again sensitive information and could potentially be used to analyse the users behaviour by third parties. To minimise the danger that users are traced, the accounting information have to be reduced to a minimal level, which is only needed to operate a billing service. This reduction mechanism is what we call data minimisation. We will show that data minimisation can be easily introduced for various systems (in the following we will show this for accounting systems).

4 Accounting Using Data Minimisation and Anonymisation

In the following we consider an architecture which allows the federation of services across different providers. By combining services from different providers it is possible to create work-flows or high level services. To ease this process it is handy to introduce an abstraction high-level access layer, where all services are presented within a global address space.

In such a service infrastructure we still expect that the basic services are operated by distinct operators but have to be registered at a central point, usually available via service repositories (e.g. [23,24,25]). The basic model assumption is schematically visualised in fig. 1.

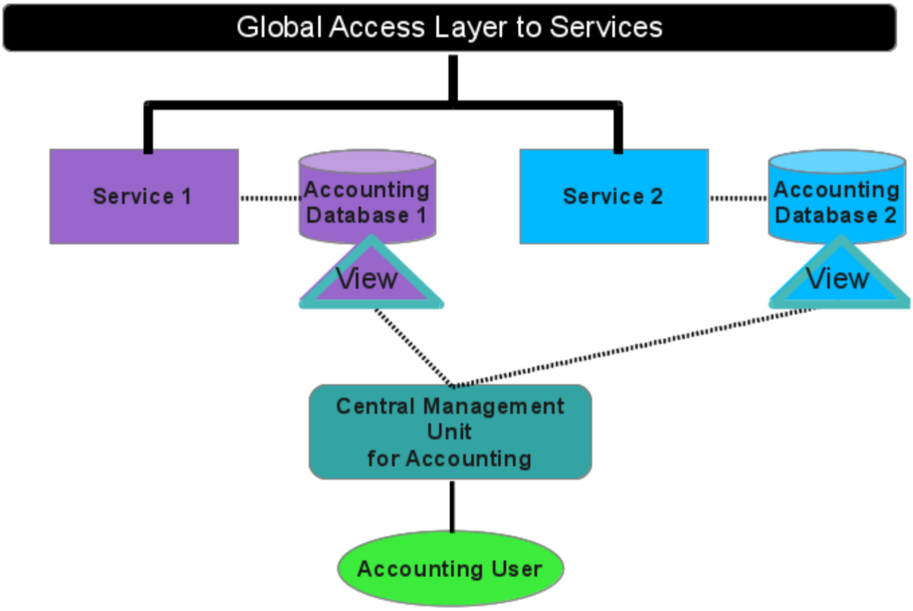


Fig. 1. General architecture of an accounting system. The solid lines show the access to data and services, while the dotted lines show the transfer of monitoring information.

In this paper we only consider accounting as a separate component which can be operated independently from any generalised global access layer, in our abstract scenario provided by the federation services. The global access layer (shown as separate component in fig. 1) can include a global name-space, management of user accounts and enabling Single Sign on (SSo) to all services.

Especially in federated systems, to bring together all relevant user data for necessary financial clearing, a central management component to access accounting data is demanded to realise an easy to use accounting and billing system. This component does not directly access the providers data store, but only indirectly via so called "views". Properties of these views are explained more in detail in section 4.2.

4.1 Aspects of a General Accounting Architecture

The main components of our approach are connectors to the local storage systems, in which the relevant accounting data are stored at each service provider. This way, the storage systems from each provider remain clearly separated and from the central management unit only the relevant aggregated information can be accessed. This ensures that the users behaviour is potentially only still gettable by the local providers. It is therefore not directly possible to combine the knowledge of multiple providers by a central unit to rank or profile users. This minimal management unit provides the central access point to regulate accounting and access reports for statistical analysis, billing or financial clearing by using an aggregation service (view), which is under control of the service providers. This architecture is shown in fig. 1.

The accounting management unit provides a user interface for easy configuration and adaptation of price and billing models. It also provides a central interface to get all data needed for billing and financial clearing based on the SLAs which signed by the resource providers and users, which might belong to companies or organisations.

As can be seen in fig. 1, the accounting system is not integrated in the global access layer. This means the transfer of the aggregated accounting data is triggered by the management unit. Also, the addresses to the views (provided by the service providers) and the corresponding logins have to be registered at the management unit.

In comparison to the accounting concepts of the systems mentioned in section 2, we introduce a concept of data minimisation. This is done by aggregation and anonymisation of the accounting data. The complete accounting data are only accessible for the local service provider. In a federated system it is of course not desired to make them visible for providers of other services[4] within the same environment or even the management unit. To restrict this direct access the management unit can not address the providers sensible accounting data as a whole. In order to realise a billing service, only the summed up information are transferred to the management unit, which are provided by the views.

4.2 Views

To transfer only relevant billing data, our approach to realise a data minimalistic access to this sensitive user data, is based on so called views. A view is a transformation of the accounting data to a report. This transformation only considers the needed minimal set of available user data to provide necessary information for the billing service.

The service provider is responsible for collecting the local accounting data. Accounting data for other services are completely out of scope for this provider, even if the operated service is part of a complex service orchestration. To be responsible for local accounting means to define which events and parameters for each service request have to be recorded. Therefore, the concrete realisation strongly depends on the provided service. E.g. a service for storing data probably needs to record the time stamp, the local account name of the user, the file size and whether the file was written or read, while a search service probably needs to record how much computing power was used to perform the request or how many data sets are read to give an result.

[4] Even if multiple service providers are needed to fulfill a single user request each provider has only access to the accounting data of the work processed on his service.

In the responsibility of the service provider is the safety and security of the accounting data, which are strongly related to individual persons. This also includes not to give information about users to other parties, or only in an anonymous way if necessary for billing purposes. A view can periodically be created (e.g. once a month) and contains the information which resources are used and how much has to be paid for this usage. The depth of detail which is required for such a report is low in most cases. To illustrate this let us give some examples:

1. The utilisation of a service can be calculated by knowing the number of requests to the service. It is not needed to know at which particular time or who triggered the request.
2. To bill a user, only the aggregated price information over the billing period is needed and not the individual services used.
3. A more detailed report is also possible (if demanded by the user). This could be the number of uses and the price individually for each service and each time slice with a special price. Such a report could contain the number of files stored during rush our (period with high price), stored during normal working time and during periods with low system utilisation (at night and weekend). The data are still aggregated and it is not reported at which exact time a service was used by the user.
4. Often it is not needed to bill single users. If users are part of a company or an organisation the report dos not contain the users identity. The report can be structured like in the examples above with the expect that only the summed up usage of all user of a group are presented. This results in an anonymity of individual users within the group.

These reports are based on SLAs between users, user groups or their representatives (e.g. VOs) and resource providers and describe the information. As already mentioned the accounting data are recorded by the resource providers and stored locally, e.g in a database (shown in fig. 1). Accessible by the central management unit are only the views. Technically a view provides a report which is an aggregation of accounting data. Depending on the particular aggregation process it can also anonymise by simply hiding information with directly link to individual users, like exemplified given by example 4. The view represents the instructions how the data are aggregated and how the price is calculated on the basis of this informations.

In the example 1 from above, selected are all records of the accounting data (the limitation to the reporting interval is automatically added by the system). Based on this view a report is created which hides the records itself and just contains the overall usage for the billing period.

Lets consider example 3, where a view for each price category is needed. One specific view selects all accounting data of the user for which the rush hour price has to be paid and calculates the price for a billing period (e.g. number of requests multiplied by the price per request). The views for the other pricing models are used in the very same way. Taking that example 1 and example 3 rely on the same price model like in example 1 only one view is sufficient for both cases. This way it is hidden whether the user performed many requests in a time period, for which a low price is active, or less during rush hour with a higher prize per service utilisation. In example 1 the view has to select all records of the user and the price calculation has to respect the individual price model for each record type. Thus, the complexity of the price calculation is slightly larger compared to example 3, but to the management component only the result of this calculation is reported.

If many users are combined in one view (example 4) the selection has to opt out the users e.g. by their account or group names. The price calculation is done similar to example 1. In this way the view combines the records of many users which results in anonymity within the group.

The views are created on the central management unit. The request for changing views are automatically transferred to the resource providers which have to check and implement the views. Once the view is active the central unit can get the reports. Altering or deleting a view is the same procedure like creating a new one. In this case it has to be ensured that deleted or overwritten views can still be checked by the resource providers. All requests from the central unit to alter a view are logged by the resource provider.

5 Reference Implementation

The accounting concept presented in this paper was developed for the knowledge infrastructure WisNetGrid[5], which offers a uniform access layer to data, information and knowledge. The access layer can connect sources from different providers using technically different storage and access systems and solutions for authentication and authorisation. By combining different sources of data, information and knowledge it is possible to use services for knowledge processing and knowledge generation.

The reference implementation of the presented accounting concept is part of a federation system and consists of components for user management, authentication and a web portal, which allows the use of services, such as searching and browsing of data, or tools for service management and workflow composition.

The accounting concept is implemented by following the concepts introduced in section 4.2. The accounting data are recorded by the operators of the potentially distributed resources. Within WisNetGrid we have realized a specialized federation entity to different types of data sources, such as databases, or Grid storage systems, which are necessary to create the common access layer. Each operator of a connected system stores the recorded accounting data in a separate accounting database. For this we use a H2 database[6] because this allows to drive the database as part of the resource federation entity, which is implemented using Java.

[5] The WisNetGrid Project is funded by the German Federal Ministry of Education and Research (BMBF), more information at: `http://wisnetgrid.org/`

[6] For more information about H2, see: `http://www.h2database.com`

The interface for billing is a central component within the WisNetGrid architecture. It offers different visualisation features to get an overview which price models had been used and how much has to be charged. A comparison to actual price models can also be made and visualised if desired. To use the results of this centralised accounting component in other programs (e.g. for the process of financial clearing) the data can be exported as CSV files. CSV is a common format and can be used by various programs for further processing.

The accounting component offers a restricted database access, which is realised by the views introduced in section 4.2. In this specific implementation the addresses and logins to the views are part of the configuration of the resource federation. This information is automatically transferred to the accounting component by registering a resource federation entity as part of the global access layer. If the resource provider offers accounting, the aggregated accounting information are automatically integrated to the central billing interface.

The management of the views is done in two steps. The accounting component offers a graphical user interface which can be accessed via a browser (by users authorised as accounting users) to delete, create or alter views. For this the selection and price calculation parts have to be specified. This is done by filling a form with SQL syntax. After submitting the form the accounting component extracts the information and stores them in a database at the resource federation instance. The second step is done manually by the operator of the resource federation or automatically by implementing a trigger on the database. Which mechanism is used depends on the configuration of the resource federation entity. Based on the request a SQL statement is build to create, alter or delete a database view. The “WHERE” clause is based on the selection part and the price column is based on the price field information from the filled form of the first step. Additionally, a “GROUP BY” clause over the reporting interval is added (e.g. “GROUP BY year, month” where year and month are fields of the accounting data). Afterwards the new view can be used by the portal to visualize accounting results according to the selected view.

6 Conclusion

We have presented a concept for accounting with privacy preserving for users, which is taking also data minimisation and anonymization into account. This was presented on a concrete implementation for the knowledge infrastructure WisNetGrid. This accounting concept allows to perform billing and financial clearing in a similar way compared to common centralized accounting systems, which are usually not designed with a strong focus on privacy preserving. A valuable field of application outside of the concrete implementation can be spotted for Grid and Cloud Computing, which was shortly discussed throughout this paper. Furthermore, we hope to inspire readers to further strengthen the user right of informational self-determination for all kinds of projects, which combine services from different partners or providers, where user data and behaviour are always sensitive information.

References

1. Sandholm, T.: Design Document: SweGrid Logging and Usage Tracking Service, LUTS (2003)
2. Piro Rosario, M., Andrea, G., Giuseppe, P., Albert, W.: Using historical accounting information to predict the resource usage of grid jobs. Future Generation Computer Systems 25(5), 499–510 (2009)
3. Elmroth, E., Gardfjäll, P., Mulmo, O., Sandgren, Å., Sandholm, T.: A Coordinated Accounting Solution for SweGrid Version: Draft 0.1.3 (October 7, 2003)
4. Mihailescu, M., Teo, Y.M.: Dynamic Resource Pricing on Federated Clouds. In: 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), pp. 513–517 (May 2010)
5. Parameswaran, A.V., Chaddha, A.: Cloud Interoperability and Standardization. SETLabs Briefings 7(7) (2009)
6. Lee, C.A.: A perspective on scientific cloud computing. In: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, HPDC 2010, pp. 451–459. ACM, New York (2010), `http://doi.acm.org/10.1145/1851476.1851542`
7. Sotomayor, B., Montero, R., Llorente, I., Foster, I.: Virtual Infrastructure Management in Private and Hybrid Clouds. IEEE Internet Computing 13(5), 14–22 (2009)
8. Nurmi, D., Wolski, R., Grzegorczyk, C., et al.: The Eucalyptus Open-Source Cloud-Computing System. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGRID 2009, pp. 124–131 (May 2009)
9. Rochwerger, B., Breitgand, D., Epstein, A., et al.: Reservoir - When One Cloud Is Not Enough. Computer 44(3), 44–51 (2011)
10. Buyya, R., Ranjan, R., Calheiros, R.N.: InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services. In: Hsu, C.-H., Yang, L.T., Park, J.H., Yeo, S.-S. (eds.) ICA3PP 2010, Part I. LNCS, vol. 6081, pp. 13–31. Springer, Heidelberg (2010), `http://dx.doi.org/10.1007/978-3-642-13119-6_2`
11. Repschlaeger, J., Wind, S., Zarnekow, R., Turowski, K.: A Reference Guide to Cloud Computing Dimensions: Infrastructure as a Service Classification Framework. In: Hawaii International Conference on System Sciences, pp. 2178–2188 (2012)
12. Bernsmed, K., Jaatun, M., Meland, P., Undheim, A.: Security SLAs for Federated Cloud Services. In: 2011 Sixth International Conference on Availability, Reliability and Security (ARES), pp. 202–209 (August 2011)
13. Badger, L., Bohn, R., Chu, S., et al.: NIST Special Publication 500-293, US Government Cloud Computing Technology Roadmap, Release 1.0 (Draft), Volume II Useful Information for Cloud Adopters
14. Sekar, V., Maniatis, P.: Verifiable resource accounting for cloud computing services. In: Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop, CCSW 2011, pp. 21–26. ACM, New York (2011), `http://doi.acm.org/10.1145/2046660.2046666`
15. Ruiz-Agundez, I., Penya, Y.K., Bringas, P.G.: A Flexible Accounting Model for Cloud Computing. In: Proceedings of the 2011 Annual SRII Global Conference, SRII 2011, pp. 277–284. IEEE Computer Society, Washington, DC (2011), `http://dx.doi.org/10.1109/SRII.2011.38`
16. Elmroth, E., Marquez, F., Henriksson, D., Ferrera, D.: Accounting and Billing for Federated Cloud Infrastructures. In: Eighth International Conference on Grid and Cooperative Computing, GCC 2009, pp. 268–275 (August 2009)

17. Chandra, D., Borah Malaya, D.: Problems & prospects of e-Governance in India. In: 2011 World Congress on Information and Communication Technologies (WICT), pp. 42–47 (December 2011)
18. Hildebrandt, M.: Profiling: From data to knowledge. Datenschutz und Datensicherheit - DuD 30, 548–552 (2006), http://dx.doi.org/10.1007/s11623-006-0140-3, doi:10.1007/s11623-006-0140-3
19. Metz, R.: Scoring: New Legislation in Germany. Journal of Consumer Policy 35, 297–305 (2012), http://dx.doi.org/10.1007/s10603-012-9191-z, doi:10.1007/s10603-012-9191-z
20. Hornung, G., Schnabel, C.: Data protection in Germany I: The population census decision and the right to informational self-determination. Computer Law & Security Review 25(1), 84–88 (2009), http://www.sciencedirect.com/science/article/pii/S0267364908001660
21. Bundesverfassungsgericht: BVerfGE 65, 1 – Volkszählung. Urteil des Ersten Senats vom 15. Dezember 1983 auf die mündliche Verhandlung vom 18. und 19. Oktober 1983 – 1 BvR 209, 269, 362, 420, 440, 484/83 in den Verfahren uber die Verfassungsbeschwerden (1983),
http://www.datenschutz-berlin.de/gesetze/sonstige/volksz.htm (access November 21, 2006)
22. Rehm, G.M.: Just Judicial Activism? Privacy and Informational Self-Determination in U.S. and German Constitutional Law (January 2000), available at SSRN http://ssrn.com/abstract=216348 or http://dx.doi.org/10.2139/ssrn.216348
23. Wu, Y., Yan, C., Ding, Z., et al.: A relational taxonomy of services for large scale service repositories. In: 2012 IEEE 19th International Conference on Web Services (ICWS), pp. 644–645 (June 2012)
24. Weiping, L., Weijie, C., Li, L., Fuliang, G.: A semantically enhanced service repository for service oriented application system development. In: World Conference on Services - II, SERVICES-2 2009, pp. 41–48 (September 2009)
25. Agarwal, S., Junghans, M., Jäkel, R.: Semantic modeling of services and workflows for german grid projects. In: Grid Workflow Workshop 2011 (2011)

Distributed Virtual Machine Monitor for Distributed Cloud Computing Nodes Integration

Li Ruan, Jinbing Peng, Limin Xiao, and Mingfa Zhu

State Key Laboratory of Software Development Environment, Beihang University,
Beijing 100191, China
ruanli@buaa.edu.cn

Abstract. Existing popular virtual machine monitors like Xen, VMware, etc. are mostly for virtualization of one single physical node. There are few researches on virtual machine monitor for distributed cross-node cloud computing resources integration. This paper introduces a novel distributed virtual machine monitor (CloudDVMM). We present its theoretical model, architecture and key technologies. Experiments and comparisons with existing researches show that our CloudDVMM achieves merits in architecture, extensibility, etc. and is promising for meeting the integration requirements of distributed virtual computing and cloud computing environments.

Keywords: MIPS, cloud computing, sever virtualization, memory virtualization.

1 Introduction

With the wide application of cloud computing, it becomes an urgent problem that how to integrate the distributed cross-nodes resources to improve the distributed resources' utility and reliability.

However, existing distributed nodes integration is traditionally based on non-virtualization technologies. Moreover, existing popular virtual machine monitors (VMM) like Xen, KVM, etc. focus more on single physical node virtualization. There are only few virtual machine based researches which focus on distributed cross-node resources integration for cloud computing, and practical VMMs are much fewer. This paper introduces a novel distributed virtual machine monitor (CloudDVMM) for distributed cloud computing nodes integration.

2 System Architecture

The system has three layers (Fig. 1): hardware layer, CloudDVMM layer and OS layer. CloudDVMM's running process is that (1) we create CloudDVMM above the SMP servers based on the hardware assisted virtualization support. CloudDVMM is constitute of VMMs distributed on each node, and each VMM is totally symmetric; (2) we

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 23–31, 2013.

run OS, which supports cc-NUMA, above CloudDVMM;(3)CloudDVMM percepts physical resources in distributed system, classifies, integrates, creates global physical resources information , virtualizes global physical resources, creates global virtual resources information and demonstrates it to OS; (4) OS creates, schedules and executes the processes, manages, assigns resources based on the virtual resources set. All these operations are transparent to CloudDVMM; (5)CloudDVMM hi-jacks and acts as an agent of OS to execute resources accessing operations, implement virtual resources to physical resources' mapping, operates physical resources and gains execution results, feedback execution results to OS.

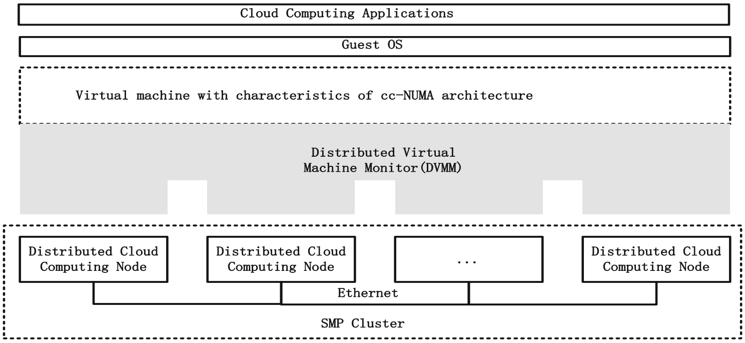

Cloud Computing Applications
Guest OS
Virtual machine with characteristics of cc-NUMA architecture
Distributed Virtual Machine Monitor(DVMM)
Distributed Cloud Computing Node
Distributed Cloud Computing Node
...
Distributed Cloud Computing Node
Ethernet
SMP Cluster

Fig. 1. Distributed Cloud Computing Nodes Integration based on CloudDVMM

3 System Modules

CloudDVMM has three layers (Fig.2): (1) The infrastructure layer is responsible for the provision of services to the above layers. It includes CloudDVMM startup module, eBIOS module, CloudDVMM communication module etc. (2)The middle layer

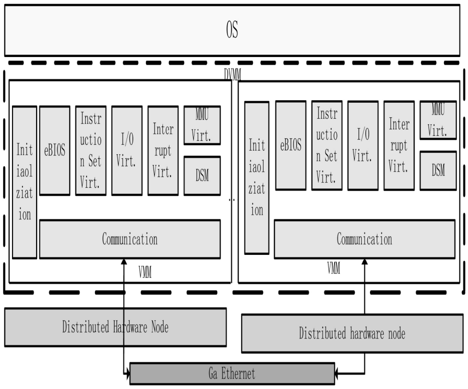


Fig. 2. Hierarchy and Modules of CloudDVMM

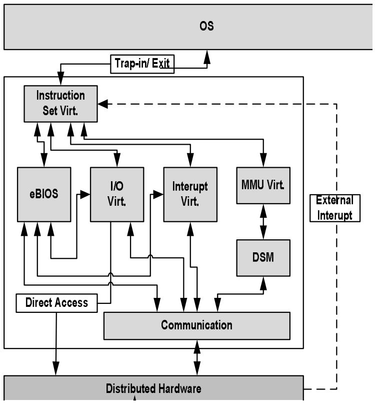


Fig. 3. Interaction Among Modules and System Running Process of CloudDVMM

will virtualize the resources, integrate and create global resources information based on the information based on the eBIOS module, virtualized resources. It includes processor virtualization module, storage virtualization module, software DSM module, interrupt virtualization module and I/O virtualization module. (3) The OS interface layer is responsible for the demonstration global virtual resources information to OS and interaction of CloudDVMM and OS. It includes vBIOS module and VMCS control module.

The interaction among modules and running process is shown in Fig. 3.The instruction set virtualization module is the entry point and the exit point of CloudDVMM.

4 Implementation and Experiments

4.1 System Implementation

CloudDVMM is implemented on Xen and includes four kernel modules of processors virtualization, memory virtualization, devices virtualization and communication modules and the extended modules of eBIOS, Xend, Qemu-dm and scheduler, etc..

4.2 Function Test

Test environment is as shown in Fig. 4. The two server nodes are connected through the high speed network. Each node's configuration is (1) CPU: AMD Opteron(tm)2350 Quad-Core Processor; (2) Memory: 4 X 1G DDR2 800;(3)Hard disk: 250G; (4) Network address configuration: 192.168.5.*. (5) CloudDVMM is installed on each node.

As is shown in Fig. 5 , the client OS which boots processor 1/16 eip 2000 shows that CloudDVMM is successfully started from node cpu. The information that a total of 2 processors are activated proves that two cpus are started up successfully. The processor information in / proc / cpuinfo after client OS started with processor: 0 and processor: 1 proves that current client OS started two cpus. The information on the customers OS shows that CloudDVMM successfully launched two perceived VCPUs.

As is shown in Fig. 5 , the client OS which boots processor 1/16 eip 2000 shows that CloudDVMM is successfully started from node cpu. The information that a total of 2 processors are activated proves that two cpus are started up successfully. The processor information in / proc / cpuinfo after client OS started with processor: 0 and processor: 1 proves that current client OS started two cpus. The information on the customers OS shows that CloudDVMM successfully launched two perceived VCPUs.

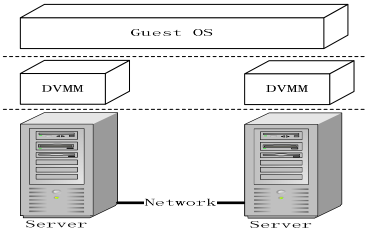
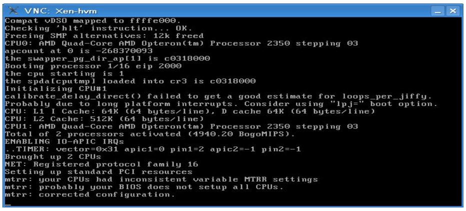
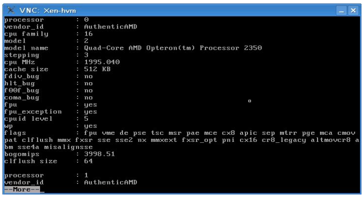


Fig. 4. Test environment



```
Compat vDSO mapped to ffffe000.
Checking 'hlt' instruction... OK.
Freeing SMP alternatives: 12k freed
CPU0: AMD Quad-Core AMD Opteron(tm) Processor 2350 stepping 03
apcount at 0 is -268370093
the swapper_pg_dir_ap[1] is c0318000
Booting processor 1/16 eip 2000
the cpu starting is 1
the spda[cputmp] loaded into cr3 is c0318000
Initializing CPU#1
calibrate_delay_direct() failed to get a good estimate for loops_per_jiffy.
Probably due to long platform interrupts. Consider using "lpj=" boot option.
CPU: L1 I Cache: 64K (64 bytes/line), D cache 64K (64 bytes/line)
CPU: L2 Cache: 512K (64 bytes/line)
CPU1: AMD Quad-Core AMD Opteron(tm) Processor 2350 stepping 03
Total of 2 processors activated (4940.20 BogoMIPS).
ENABLING IO-APIC IRQs
..TIMER: vector=0x31 apic1=0 pin1=2 apic2=-1 pin2=-1
Brought up 2 CPUs
NET: Registered protocol family 16
Setting up standard PCI resources
mtrr: your CPUs had inconsistent variable MTRR settings
mtrr: probably your BIOS does not setup all CPUs.
mtrr: corrected configuration.
_
```

Fig. 5. customer OS multiprocessor starting process



```
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 16
model           : 2
model name      : Quad-Core AMD Opteron(tm) Processor 2350
stepping        : 3
cpu MHz         : 1995.040
cache size      : 512 KB
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 5
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat clflush mmx fxsr sse sse2 nx mmxext fxsr_opt pni cx16 cr8_legacy altmovcr8 a
bm sse4a misalignsse
bogomips        : 3998.51
clflush size    : 64

processor       : 1
vendor_id       : AuthenticAMD
--More--_
```

Fig. 6. Processor information in / proc / cpuinfo after client OS start

4.3 Performance Test

4.3.1 The Unixbench Performance of CloudDVMM

Table 1 and Table 2 show the Unixbench-4.1.0's average performance data.

Table 1. The performance data with single-node

TEST	BASELINE	RESULT	INDEX
Dhrystone 2 using register variables	376783.7	18656301.6	495.1
Double-Precision Whetstone	83.1	1110.0	133.6
Execl Throughput	188.3	15697.9	833.7
File Copy 1024 bufsize 2000 maxblocks	2672.0	185840.0	695.5
File Copy 256 bufsize 500 maxblocks	1077.0	48055.0	446.2
File Read 4096 bufsize 8000 maxblocks	15382.0	2077483.0	1350.6
Pipe Throughput	111814.6	6234174.2	557.5
Pipe-based Context Switching	15448.6	341053.6	220.8
Process Creation	569.3	31673.6	556.4
Shell Scripts (8 concurrent)	44.8	2938.5	655.9
System Call Overhead	114433.5	11076033.0	967.9
FINAL SCORE	**533.9**		

Table 2. The performance with dual-nodes

TEST	BASELINE	RESULT	INDEX
Dhrystone 2 using register variables	116700.0	1087027.4	93.1
Execl Throughput	43.0	454.8	105.8
File Copy 1024 bufsize 2000 maxblocks	3960.0	188424.0	475.8
File Copy 256 bufsize 500 maxblocks	1655.0	187113.0	1130.6
File Read 4096 bufsize 8000 maxblocks	5800.0	43608.0	75.2
Pipe Throughput	12440.0	274099.1	220.3
Pipe-based Context Switching	4000.0	2960.2	7.4
Process Creation	126.0	1277.3	101.4
Shell Scripts (8 concurrent)	6.0	103.2	172.0
System Call Overhead	15000.0	307844.9	205.2
FINAL SCORE	**137.0**		

The experimental results show: (1) Unixbench benchmark program can operate normally on the prototype system; (2) in the same case containing two VCPU, the prototype system Unixbench scores below the virtual machine on the stand-alone.

4.3.2 The Ubench Performance of CloudDVMM

Table 3 and Table 4shows the average Ubench performance data on both platforms.

Table 3. The performance under single-node

Ubench CPU	181914
Ubench MEM	209674
Ubench AVG	195794

Table 4. The performance under dual-node

Ubench CPU	136954
Ubench MEM	81021
Ubench AVG	108987

The experimental results show that the: (1) Ubench benchmark program can run normally on CloudDVMM; (2) The scores from Ubench program with two VCPS is higher than that with only one VCPU virtual machine, which prove that the VCPU in two servers can work properly.

4.3.3 SPLASH-2 Test Performance of CloudDVMM

SPLASH-2 is used for evaluating the performance of shared memory systems which are mainly for the evaluation of the SMP, CC-NUMA, DSM shared storage architecture performance of the computer system. Table 5 and Table 6 show the test performance under a single node, Tables 7 and 8 show the performance under two-nodes.

The results show that: (1) Under the SPLASH-2, the test program can operate normally on the prototype system; (2) The performance of CloudDVMM containing two VCPUs is lower than that of the virtual machine on the stand-alone.

Table 5. The performance under single-node (sub-process statistics)

PROCESS STATISTICS			
Proc	Total Time	Multigrid Time	Multigrid Fraction
0	189448	71880	0.379

Table 6. The performance under single-node (phased Statistics)

Time	
Start time	405812431
Initialization finish time	405966330
Overall finish time	406155781
Total time with Initialization	343350
Total time without initialization	189451

Table 7. The performance under dual-node (sub-process statistics)

PROCESS STATISTICS			
Proc	Total Time	Multigrid Time	Multigrid Fraction
0	150002	60000	0.400

Table 8. The performance under dual-node (phased Statistics)

TIMING INFORMATION	
Start time	1379157799
Initialization finish time	1379387803
Overall finish time	1379537805
Total time with Initialization	380006
Total time without initialization	150002

4.3.5 The Linux Command Execution Performance

The average time performances of the linux commands like ls, make, gcc commands with 50 times each are as shown in Table 9 and Table 10.

Table 9. The command execution performance under single node and dual nodes

Name	**Description**	**Execution time (Physics)/s**	**Execution time (single node HVM)/s**	**Execution time (cross node HVM)/s**	**overhead ratio**
ls	Display bonnic++-1.03c directory	0.005	0.130	0.800	160
make install	Compile bonnic++-1.03c	0.020	0.050	1.050	52.5
gcc zx.c	Compile zx.c file	0.065	0.090	0.280	4.3

Table 10. The OS performance of Virtual Multiprocessor client

Name	Descripton	Execution time (physical)	Execution time (virtual)	Overhead ratio
ls	List information about hundreds of files	0.03	6.64	255
gcc	Compile a C program	0.14	0.98	6.81

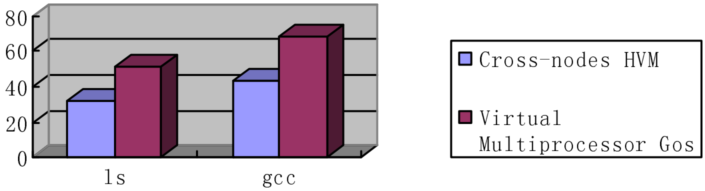


Fig. 7. The performance comparison between CloudDVMM and Virtual Multiprocessor

As can be seen from Fig.7, the CloudDVMM'ls and gcc overhead is superior to Virtual MultiProcessor.

4.4 Test Results Summary

From the test results, we have verified CloudDVMM ability to achieve a distributed nodes' SSI, the distributed resources integration, virtualization to a single resource space form to the OS, and that OS can use perceived cluster resource like single resource.

5 Related Work

Existing distributed cloud computing nodes integration technologies are traditionally based on non-virtualization technologies implemented on the hardware layer[1-3], those on system software layer like MOSIX[4-7],Sun Solaris-MC[8], SCO UnixWare NonStop Clusters[9], those on middleware lay like IVY[10], Mirage, etc. and those on application layer. Existing popular virtual machine monitors like Xen, VMware ESX Server, etc. are mostly for single physical node. There are only few researches based on virtualization except Virtual Multiprocessor and vNUMA which focus on distributed cross-node resources integration, and practical systems are much fewer [11-13]. By comparison results from implementation hierarchy (**Hier.**), Technology(Tech.) , Implememtation Difficulty(Diff.), Transparency(**Transp.**), Performance (**Perform.**), SMP nodes Supports(**SMP nodes Sup.**) and Architecture Supports (**Arich.**) in table 16, we can see that CloudDVMM is running above the hardware and beneath the OS , has a hierarchy and modular architecture and can provides a single system image for Cloud computing cluster. It also show that CloudDVMM achieve merits in architecture, extensibility, etc. and is promising for meeting the requirements of distributed virtual computing and cloud computing environments.

Table 11. Comparisons with related work

Re-searches	**Hier.**	**Tech.**	**Diff.**	**Transp.**	**Perf..**	**SMP Sup.**	**Arich.**
Multi-processor	**Application Lay**	**Para-virt.**	**High**	**Low**	**Low**	**N**	**IA-32**
vNUMA	**System Software Layer**	Pre-virtual.	**moderate**	**Good**	**Mod-erate**	**N**	**IA64**
Cloud DVMM	**System Software Layer**	**Hardware virt.**	**Low**	**Good**	**high**	**Y**	**IA-32**

6 Conclusions

In this paper, a novel distributed virtual machine monitor was introduced for distributed cross-node cloud computing resources integration. We are now trying to apply it to practical industrial applications and improve CloudDVMM's performance.

Acknowledgments. This work was supported by the Hi-tech Research and Development Program of China (863 Program) under Grant No. 2011AA01A205, National Natural Science Foundation of China under Grant No. 61003015, Beijing Natural Science Foundation under Grant No. 4122042, the fund of the State Key Laboratory of Software Development Environment under Grant No. SKLSDE-2012ZX-23.We are grateful to those students and colleagues who participated in our Cloud Servers project from Beihang University, Institute of Computing Technology (ICT), Chinese Academy of Sciences and Lenovo Company.

References

[1] IBM Enterprise X-Architecture Technology (OL), `ftp://ftp.software.ibm.com/systems/support/system_x_pdf/exabroc.pdf`

[2] Intel. Intel® 64 and IA-32 Architectures Software Developer's Manual. Vol. 1:Basic Architecture (2007)

[3] Kaneda, K., Oyama, Y., Yonezawa, A.: A virtual machine monitor for providing a single system image. In: Proceedings of the 17th IPSJ Computer System Symposium, pp. 3–12 (2005)

[4] Barak, A., Laden, O., Yarom, Y.: The NOW MOSIX and its Preemptive Process Migration Scheme. Bulletin of the IEEE Technical Commitee on Operating Systems and Application Environments 7(2), 5–11 (1995)

[5] Amar, L., Barak, A., Shiloh, A.: The MOSIX Direct File System Access Method for Supporting Scalable Cluster File Systems. Cluster Computing 7(2), 141–150 (2004)

[6] Haddad I. F., Paquin E. MOSIX: A Cluster Load-Balancing Solution for Linux. Linux Journal 2001(85es) (2001)

[7] Lottiaux, R., Gallard, P., Vallee, G., et al.: OpenMosix, OpenSSI and Kerrighed: a comparative study. In: Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2005), pp. 1016–1023. IEEE Computer Society, Washington (2005)

[8] Bernabeu, J.M., Khalidi, Y.A., Matena, V., et al.: Solaris MC: A Multi-Computer OS. Technical Report: TR-95-48. Sun Microsystems (1995)

[9] Walker, B., Steel, D.: Implementing a Full Single System Image UnixWare Cluster: Middleware vs Underware. In: International Conference on Parallel and Distributed Processing Techniques and Applications, vol. 6, pp. 2767–2773. Computer Science Research, Education, and Applications Press

[10] Li, K., Hudak, P.: Memory coherence in shared virtual memory systems. ACM Transactions on Computer Systems 7, 321–359 (1989)

[11] Kaneda, K., Oyama, Y., Yonezawa, A.: A Virtual Machine Monitor for Providing a Single System Image. Transactions of Information Processing Society of Japan 47, 27–39 (2006)

[12] Kaneda, K., Oyama, Y., Yonezawa, A.: A virtual machine monitor for utilizing non-dedicated clusters. In: Proceedings of the Twentieth ACM Symposium on Operating Systems Principles, pp. 1–11 (2005)

[13] Chapman, M., Heiser, G.: Implementing transparent shared memory on clusters using virtual machines. In: USENIX Annual Technical Conference, pp. 383–386. USENIX Association, Anaheim (2005)

Differentiated Policy Based Job Scheduling with Queue Model and Advanced Reservation Technique in a Private Cloud Environment

Shyamala Loganathan and Saswati Mukherjee

Dept of Information Science and Technology, College of Engineering, Anna University,
Guindy, Chennai-25, India
{L.Shyamala,SaswatiMukherjee}lshyamlabi@gmail.com,
msaswati@yahoo.com

Abstract. Cloud Computing can be viewed as a computing model containing a pool of resources and Internet based application services. Cloud makes on-demand delivery of these computational resources (data, software and infrastructure) among multiple services via a computer network. An infrastructure-as-a-service cloud system provides computational capacities to remote users. In present scenario, most of the Infrastructure as a Service (IaaS) Clouds use simple resource allocation policies like immediate and best effort. In private cloud, since the resources are limited, maximizing the utilization of resources and giving the guaranteed service for the user are the ultimate goal. Hence efficient scheduling is needed which is a major challenge in satisfying the user's requirement (QoS). In this paper, we propose an advanced reservation technique with backfilling in scheduling policy that aims at serving the user requests by satisfying the required QoS, achieving the guaranteed service for the request by making an efficient provisioning of cloud resources.

Keywords*:* Cloud computing, Job scheduling, Queue model, Reservation, CloudSim.

1 Introduction

The increasing demand of computational resources has led to new types of cooperative distributed systems, such as the grid [1] and cloud computing [2]. In IaaS cloud the resources (compute capacity and storage) are provided in the form of virtual machines to users. A scheduler can be used to decide when and where to place these virtual machines on a pool of resources. Scheduling jobs in a cloud environment is a difficult task because of its dynamic nature. Various researchers have dealt with the challenges in scheduling in a Cloud [3][4][5]. Perhaps the primary challenge of scheduling is the allocation of available resources efficiently. Therefore in cloud, job scheduling and resource management are related to the efficiency of the whole cloud computing facilities. Presently, most of the cloud providers rely on simple resource allocation policies like immediate and best effort [3]. Though advanced reservation technique is well studied in Grid environment and applied, due to the dynamic nature of incoming

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 32–39, 2013.

request in Cloud immediate and best effort provisioning is preferred so for in public clouds [3]. In general, usage pattern of cloud requests are not predictable because of its dynamic nature. Hence advanced reservation technique commonly used in grid is not appropriate for public cloud. However, for organizational cloud (private cloud) the usage pattern is predictable to an extent and can be defined in advance. Private Cloud is one which is owned by the organization and thus, maintained by same. The characteristics and scheduling challenges in a private cloud (Institutional Cloud) is discussed in [6]. Scheduling in private cloud environment poses a unique situation where job scheduling can benefit by taking advantage of policy based provisioning for different set of job request with different queues will lead to maximize utilization of resource and guarantees the service for a request. This technique avoids the fragmentation of resources when simply advanced reservation is used. The fragmented resources can be utilized by other policy of other queue of jobs. In this paper we exploit this factor and propose a technique in private cloud scheduling.

In Section 2, related works in this area are discussed. Section 3 analyzes the various system parameters used in the system model. Section 4 describes the proposed cloud architecture and the scheduling policy. Simulation model and performance evaluation in Section 5 brings forward the benefits of the research work. Finally, in Section 6, we conclude our work and discuss possible future work.

2 Related Work

Ningning Gao [7] has given a reservation algorithm with Multi-Parameters called MPRAR which consists of global queue to store reservation requests called FIFO and another queue named Heap which arranged in the order of weight to determine whether the reservation request would be accepted. But this suffers with fragmentation of resources. Algorithms proposed in [8] [9] are to schedule advance reservation with laxity considers non-preemptive tasks request in a grid environment. Preemption of job is not considered in these works. Sabitha Rani B.S [10] proposed a relaxed resource advance reservation policy (RARP) with trust factor to improve the utilization at both low and high reservation. Cao [11] a backfilling based gang scheduling mechanism is incorporated into the share based co-scheduling job (SCOJO) scheduler. The simulative results show that it can mitigate the negative effects of advance reservation. Kaushik [12] et al. proposes a flexible reservation window scheme. It concludes that when the size of the reservation window is equal to the average waiting time in the on-demand queue, the reservation rejection rate can be minimized close to zero but does not address the issue of low resource utilization rate by advance reservation. All the above mentioned works didn’t consider a differentiated policy for workload which will lead to maximization of resource utilization.

3 Problem Formulation

The problem of job scheduling in a cloud environment essentially consists of a dynamic set of j independent tasks to be scheduled on set of n computational nodes located in m datacentres (resources pool). In general the requests are handled by a resource manager which takes the request and sends to the dispatcher. The dispatcher dispatches the request in first come first serve mode for renting the capacities of a resource. To get a guaranteed service in a private IaaS Cloud where the capacity is limited we propose to include advanced reservation technique in this model. Here we have considered three different modes of renting the computing capacities as follows:

- Advance Reservation (AR- mode): Resources are reserved in advance. They must be made available at the specified time.
- Immediate (I-mode): When a client submits a request, based on the resource availability, either the required resources are provisioned immediately, or the request is rejected.
- Best effort (BE-mode): Jobs are kept in a queue and resources are provided when available. It can be batch jobs also.

The best-effort jobs are preemptable and they do not have any time constraints. Immediate and advance reservation jobs are non-preemptable and have time constraints, such as start time and end time. It will preempt best-effort mode whenever the resources are required for advance reservation or immediate mode. There is no guarantee that a submitted best-effort mode will get resources for completion within a certain time limit. We assumed that best-effort jobs are splittable, all jobs are independent and the scheduling algorithm assumes that there is no communication among jobs.

To identify the mode of the job, request is described as (JobID, UserID, N, M, D, B, timestamp, ST, FT),where N is number of CPUs, M is memory in megabytes, D is disk space in megabytes and B is the network bandwidth in megabytes per second, timestamp contains date, month, year, time. ST is start time and FT is finish time. For I-mode the request has information about current date, time, how long the execution lasts, but not the start time of execution. For BE-mode the request has information about how long the execution lasts, but not the start time of execution and date. For AR mode the request includes date, start time, finish time. The proposed architecture is shown in Fig. 1. Incoming jobs are placed in 3 different queues based on their arrival pattern by the CMS (Cloud Management System) and sent to the different datacentre. For an AR- mode the required capacity of the service request (say i mips) is calculated aperiori from their request and datacentre for that is assigned by the CMS Hence the immediate request is sent to the datacentre of ((m* pj) -i) mips by the CMS where p_j is the processing capacity of a datacentre. This assures the guaranteed service for the AR- mode jobs.

CMS continuously monitors the capacity utilized by the datacentres and keep checking for any rejection on the AR-mode. If any job is not taken at prescribed time of AR- mode the capacity allotted for that is taken for the current I- mode jobs. Local schedulers know the current status of VMs in their own datacentre. These schedulers communicate with CMS and pass the message regarding the processing of jobs and availed resources. CMS calculates the remaining capacity and based on this, further

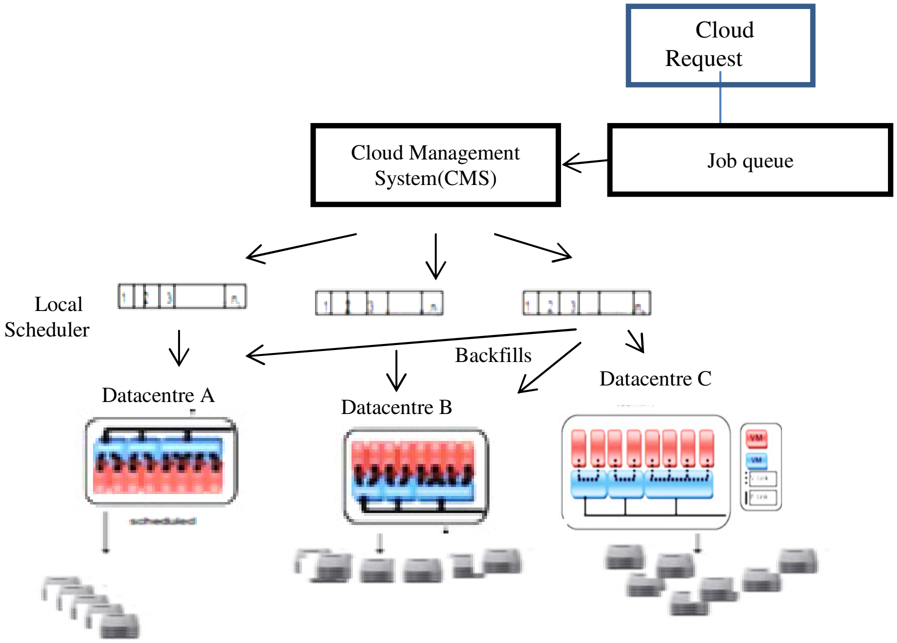


Fig. 1. Overview of the Scheduling Process

admits or rejects incoming I-mode request. Jobs with BE-mode are kept in a separate queue to process later or during the idle time of the resources.

4 System Model

We propose a queuing model namely ADQ model. In this model we considered different queue groups of service-requests in the cloud computing environment as M/G/S queue, and all the queues together make a queuing network, thus applying multiple server queuing system [13] for this model. Each resource in a data centre is

Table 1. Parameters considered for the queuing system model

Model Parameter	Definition
Ci	Capacity of datacentre (Data Centre) Expressed as $\sum P_j$ of hosts in single datacentre.
Ts	Service time taken by a single request.
TT_s	Total Service time taken by the request
λ	Mean arrival rate of the request.
μ_i	Total service time taken by the request in each queue+ waiting time in the queue.

characterized by processing capacity pj and processing availability aj. Both pj and aj are related to the resource capacity as regards its current availability (i.e. service time, waiting time) which are sufficient to process the job in each datacentre. Table I shows the parameters used in this model. We define three queues as QAD- Queue with AR-mode request, QIM- Queue with I-mode request, and QB- Queue with BE-mode Jobs to keep job requests.

We define the ADQ model as follows:

a. Every user must submit a request to the cloud management system. Broker in the cloud manager breaks the jobs into three queues based on the request pattern.
b. Requests arrival pattern: The user‘s request arrivals occur randomly according to a Poisson distribution with λ arrivals per unit time.
c. Queue behavior: Request is selected from one of the three queues based on the available capacity of each datacentre and total estimated time to process a request.

$$Di= (DLi - CTi) \text{ -- } (1)$$

Where DLi =Deadline given by ith request, CTi =Current time. Di=Delay threshold for request.

$$Ci=\sum_{1}^{m}\sum_{i=1}^{n} pi \quad \text{ ------------------------------------ } (2)$$

Where m is the no.of datacentre, n is the number of host in a single datacentre.

$$TTs= \sum_{i=1}^{N} Ts \quad \text{ -------------------------------------- } (3)$$

Where TTs is total service time required by requests in each queue.

To allocate BE-mode and I-mode request CMS calculates the service time Ts required completing the request and the availability of resources. For these job requests, if delay threshold is tolerable then it is kept in the queue otherwise rejected if the resources are not free. TTs are calculated to backfill the BE-mode jobs to allocate resources during its idle time. Di is used to allocate the resource on reserved datacentre for other mode request when it is not utilized by the advanced reservation request. If Di greater other than of Immediate mode request‘s service time then that IM request is allotted in reserved datacentre to execute before AD request.Otherwise not allocated in reserved datacentre.

5 Simulation and Results

In our model, we used CloudSim [14] to simulate our proposed technique. Cloudsim used to perform this as a single simulation where all jobs are submitted as cloudlets to the broker. In general CloudSim toolkit supports First Come First Serve (FCFS) for scheduling jobs with single queue. We used this as our baseline to compare our proposed model. We extended the CloudSim to support our proposed model as having a single cloud with group of 6 datacentres. We fixed the number of processors elements to 2, the number of virtual machines with 2, the number of Cloudlet with 4 per user and we varied the number of the users from 5 to 15 per step of 5 of each mode of job request. As CMS cannot have a control over the resources at a datacentre and the full set of jobs submitted to the resources, we implemented a low-level local scheduler to perform efficient job scheduling in cloud environment. In CloudSim, Datacentre Broker component randomly selects the datacentre irrespective of their heterogeneity in hardware; we have proposed a CMS that selects the datacentre, based on user defined QoS specifications given as (JobID, UserID, N, M, D, B, timestamp, ST, FT) and splits that job request as any one of the queue as QAD,QIM,QB. Broker component is used to identify the request into three groups by the date timestamp introduced in the request for advanced reservation. CIS (Cloud Information Service) is used to get available resource information, resource utilization and used to make the decision of request execution.

Fig. 2 shows that in the proposed algorithm, the reserved jobs have success rate almost 99.9%, which shows the QoS of guaranteed service. Fig. 3 shows the comparison of traditional algorithm FCFS and proposed algorithm. In FCFS success rate decreases as number of jobs increases. This indicates no guaranteed service for needy jobs and all requests are consider as same and put on a single queue and no guarantee is assigned to any job.

If the system is flooded with lots of advance reservation and immediate jobs then best effort jobs will not have enough resources to run on. In order to avoid this there will be admission control through (UserID, JobID) is provided to CMS for the user.

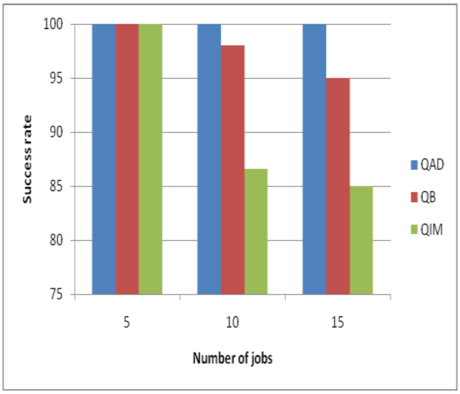


Fig. 2. Success rate of ADQ algorithm

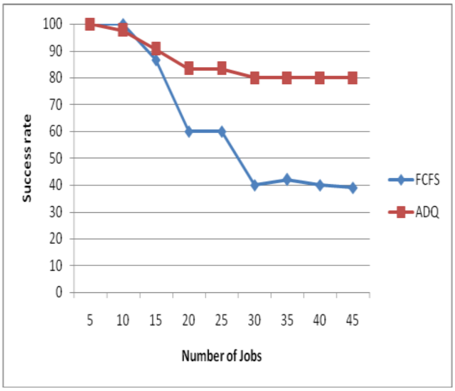


Fig. 3. Success rate of ADQ algorithm and FCFS

6 Conclusion and Future Work

The recent efforts to design and develop cloud technologies focuses on defining novel methods, policies and mechanisms for efficiently managing cloud infrastructures.

We have used advanced reservation technique by keeping different queues for jobs arriving in a private cloud. We studied the performance of this proposed approach from a point of view of enhancing the QoS by giving guaranteed service. This approach with the proposed cloud architecture has achieved very high (99%) service completion rate with guaranteed QoS for the reserved jobs over the traditional scheduling policy which does not consider any priority [FCFS] for incoming jobs. An algorithm can be developed to enhance the response time of best-effort jobs. The backfilling algorithm proposed is not implemented and tested and it can be considered as a part of future work. We are planning to extend this model where resources can be hired from other clouds (public, private) when need arises (at the times of peak load). This will help us to attain100% guaranteed service to all requests by employing multi agent system to negotiate between the clouds.

Acknowledgment. This research is supported by Anna Centenary Research Fellowship from Anna University, Chennai, India.

References

1. Hamscher, V., Schwiegelshohn, U., Streit, A., Yahyapour, R.: Evaluation of Job-Scheduling Strategies for Grid Computing. In: Buyya, R., Baker, M. (eds.) GRID 2000. LNCS, vol. 1971, pp. 191–202. Springer, Heidelberg (2000)
2. Buyya, R., Yeo, C.S., Venugopal, S.: Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering, IT Services as Computing Utilities. In: The 10th IEEE International Conference on High Performance Computing and Communications, pp. 5–13. IEEE Computer Society (2008)
3. Sotomayor, B., Montro, R., Llorente, I., Foster, I.: An open Source Solution for Virtual Infrastructure management in Private and Hybrid Clouds. IEEE Internet Computing, 5–8 (2009)
4. Sotiriadis, S., Bessis, N., Antonopoulos, N.: Towards inter-cloud schedulers: A survey of meta-scheduling approaches. In: International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp. 59–66 (2011)
5. Doelitzscher, F., Sulistio, A., Reich, C., Kuijs, H., Wolf, D.: Private Cloud for Collaboration and e-Learning Services: from IaaS to SaaS, vol. 91(1), pp. 1–20. Springer (2011)
6. Shyamala, L., Mukherjee, S.: EduCloud: An institutional cloud with optimal scheduling policies. In: Krishna, P.V., Babu, M.R., Ariwa, E. (eds.) ObCom 2011, Part I. CCIS, vol. 269, pp. 114–123. Springer, Heidelberg (2012)
7. Gao, N., Hong, J.: A Resource Reservation Algorithm with Muti-parameters. In: CHINAGRID 2011, pp. 211–214. IEEE Computer Society, Washington (2011)
8. Farooq, U., Majumdar, S., Parsons, E.W.: Efficiently Scheduling Advanced Reservations in Grids. Techical report, SCE-0514, Dept. of Systems & Computer Engineering, Carleton University, Ottawa, pp. 1–9 (2005)
9. Smith, W., Foster, I., Taylor, V.: Schedulling with Advanced Reservations. In: IEEE/ACM Proceedings of CCGrid 2000 (2000)
10. Sabitha Rani, B.S., Venkatesan, R., Ramalakshmi, R.: Resource reservation in grid computing environments: Design issues. In: ICECT 2011, pp. 66–70 (2011)

11. Cao, J., Zimmermann, F.: Queue scheduling and advance reservations with COSY. In: Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS 2004), p. 63a (2004)
12. Kaushik, N.R., Figueira, S.M., Chiappari, S.A.: Flexible time-windows for advance reservation scheduling. In: Proceedings of International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2006), pp. 218–225. IEEE Computer Society (2006)
13. Lu, C.: Queue Theory, Beijing YouDian Publish House (2007)
14. Buyya, R., Ranjan, R., Calheiros, R.N.: Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities. Keynote, High Performance Computing and Simulation (HPCS 2009) Conference, pp. 1–11 (2009)

Scaling Out Recommender System for Digital Libraries with MapReduce

Lun-Chi Chen*, Ping-Jen Kuo, I-En Liao, and Jyun-Yao Huang

Department of Computer Science and Engineering
National Chung-Hsing University, Taichung, Taiwan
{lunchi0124,allen501pc}@gmail.com,
{s99056001,ieliao}@nchu.edu.tw

Abstract. Recommender system can help users to effectively identify interested items from a potentially overwhelming huge collection of items, and it has been shown to be very useful in many e-commerce applications. Collaborative filtering (CF), which assumes that similar users may have similar tastes, is one of the most widely used Recommender system techniques. However, one of the major weaknesses for the CF mechanism is the computational cost in computing pairwise similarity of users. This paper attempts to tackle the computational problem of all pairs similarity using the MapReduce technique in the Hadoop framework. We give an overview of our development on using a parallel filtering algorithm to improve the performance of a personal ontology based recommender system for digital library. The experimental results show that the proposed algorithm can indeed scale out the recommender systems for all pairs search.

1 Introduction

Libraries collect a large volume of media including books, films, newspapers, and so on. Traditionally, libraries codify all their collections hierarchically, and try to help users query and find the physical locations of books based on codifications. Users need to be well-trained to submit correct keywords for cross-discipline or multi-dimensional surveys.

Recently, libraries have begun to provide recommender services to improve user satisfaction. Most library recommender services use a collaborative filtering technology to suggest books to users. It assumes that those who preferred something in the past tend to prefer the same thing again in the future [1]. That is, when a user gives feedback, the recommender service suggests items for the user that like-minded users preferred in the past. Collaborative filtering requires explicit information to describe user profiles. Like-minded users will have similar profiles, and their previous rating for each book will be used to compute the rate of suggested items [2]. Unfortunately, it is difficult to collect such explicit information for library systems. Users are usually not interested in giving ratings when they loan books and are asked to rate books, so the collaborative filtering recommender system is not easy to implement in digital libraries.

* Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 40–47, 2013.

During the last few years, a personal ontology-based recommender system has been applied in many diverse application domains [3]. Liao et al. incorporate collaborative filtering techniques with a personal ontology model for digital libraries to recommend English sources and solve the problem of making effective recommendations for users [4]. Also, they propose that implicit ratings can be inferred from the loan records because keywords extracted from the user's loan records indicate the preferences of user. This methodology proposed by Liao et al. has been implemented and this system called the personal ontology recommender (PORE) system [5]. This kind of recommendation method is effective for extracting potential preferences but it has a long runtime for each recommendation phase. To solve the time-consumption problem of the digital library recommendation system, much research adopts parallel computing such as MapReduce. MapReduce, designed in a parallel computing model, has been proposed by Google to process massive amounts of data. It has been proven that MapReduce is highly scalable, efficient, and reliable in big data computing [6]. However, these studies only focus on how deal with parallel programming; they don't integrate the characteristics of parallel computing and the data of a personal ontology.

To improve performance in computing personal ontology similarity in a library recommender system, we use a parallel filtering algorithm based on the characteristics of parallel computing and a personal ontology. The workflow of a parallel filtering algorithm is a two-phase MapReduce job to find no like-minded users and compute the similarity between like-minded users. To demonstrate the feasibility of the proposed method, we implement a library recommender system based on PORE. This research provides an alternative solution to create a more efficient library recommender system.

In section 2, this paper will introduce a related recommender system and MapReduce application. Section 3 describes our design concepts and the system. Section 4 shows the results of the implementation of our methodology. Finally, section 5 presents conclusions and future research directions.

2 Related Work

MapReduce is a programming model and an associated implement for processing and generating large datasets. MapReduce provides an abstraction that involves the programmer defining a mapper and a reducer function. A brute force approach is usually used in large collections with MapReduce [7]. A MapReduce implementation of the inverted index approach was presented by Elsayed et al. [8]. The proposed algorithm consists of two consecutive MapReduce jobs. The first job is to group the keywords as key, and a value consisting of the document ID and the term weight. The second job is to pair documents on the basis of a keyword in the map function and compute similarity in the reduce function , as follows:

Indexing. Given a document d_i, for each term, the mapper emits the term as the key, and a tuple consisting of the document ID and weight as the value. MapReduce groups these tuples in the shuffle phase, and then passes these inverted index lists to the reducers. The reducer accepts them and writes out to disk.

$$\texttt{map} : \langle i, d_i \rangle \rightarrow [\langle t, \langle i, d_i[t] \rangle \rangle \mid d_i[t] > 0]$$

$$\texttt{reduce} : \langle t, [\langle i, d_i[t] \rangle, \langle j, d_j[t] \rangle, \ldots] \rangle \rightarrow [\langle t, [\langle i, d_i[t] \rangle, \langle j, d_j[t] \rangle, \ldots] \rangle]$$

Similarity. given the inverted list of term t, the mapper generates key tuples corresponding to every pair of document IDs and produces the contribution $\omega = d_i[t] \cdot d_j[t]$ as its value. For any document pair, the shuffle phase provides a reducer with the contributions list W from the various terms, which only need to be summed up.

$$\texttt{map} : \langle t, [\langle i, d_i[t] \rangle, \langle j, d_j[t] \rangle, \ldots] \rangle \rightarrow [\langle \langle i, j \rangle, \omega = d_i[t] \cdot d_j[t] \rangle \mid i < j]$$

$$\texttt{reduce} : \langle \langle i, j \rangle, W = [\omega_0, \ldots, \omega_{|W|}] \rangle \rightarrow [\langle \langle i, j \rangle, \sigma(d_i, d_j) = \sum_{\omega \in W} \omega \rangle]$$

3 Computing Personal Ontology Similarity Using a Parallel Filtering Algorithm

In the following, we describe two approaches used to scale out the pairwise similarity comparison for all users' personal ontologies using the MapReduce technique in the PORE. The first approach, called Brute Force, is an intuitive sense of how the pairwise similarity comparison works in PORE. For brute force, we implement the pairwise ontology similarity algorithm of PORE using MapReduce framework. The second approach, called parallel filtering, is a parallel filtering algorithm that exploits an inverted index.

3.1 Personal Ontology Recommender System

The PORE system uses reference ontology to build a personal ontology for each user by mining the user's loan records. Dewey Decimal Classification (DDC) is used as the reference ontology to recommend English collections, and the Classification Scheme for Chinese Libraries (CCL) is used to recommend Chinese collections. Interested categories and keywords are two major impact factors as the personal preferences, and are used to build the personal ontology.

Interested Categories. The PORE system identifies the favorite categories by analyzing a user's loan records and loan times. The favorite value of category *i* for a user, denoted as *Fi*, is as follows:

$$F_i = \sum_{j=1}^{m} (a_{ij} \times (\frac{1}{2})^{(j-1)}) \tag{1}$$

Let a_{ij} denote the frequency of loaned items in category *i*.

Interested Keywords. After building the personal ontology of a user, interested keywords can be found in each favorite category. The interest level of keyword j in the category i for a user can be estimated by Equation (2), where b_{ij} denotes the frequency of keyword j that appears in the category i of the loan records of a user. The distinctness level of keyword j in this specific category, denoted as W_{ij}, considers that each keyword should be given different weights in different categories. Details of the formula W_{ij} are given in [5].

$$I_{ij} = b_{ij} \times W_{ij} \tag{2}$$

After building the personal ontology, the PORE system can compute the cosine-based similarity between users and then select the most similar users as the like-minded users. The similarity between user A and B, sim(A, B), is measured with Equation (3), where $\mathrm{ksim_i}$ and $\mathrm{ssim_i}$ are the keyword similarity and structural similarity, respectively, in the category i between them. Let $O_A$and $O_B$denote the ontology of user A and B, respectively. The union of the personal ontology for user A and B is denoted as C.

$$\mathrm{sim}(O_A, O_B) = \sum_{i \in C}((1-\alpha)ksim_i(A,B) + \alpha\ ssim_i(A,B)) \tag{3}$$

The keyword similarity and structural similarity between user A and B are measured with Equation (4) and Equation (5), respectively, where K is the union of the keywords in the category i between A and B, and N_i is the union of the specific category i and its sub categories.

$$\mathrm{ksim_i}(A,B) = \frac{\sum_{j \in K} I_{ij}(A) \times I_{ij}(B)}{\sqrt{\sum_{j \in K} I_{ij}(A)^2} \times \sqrt{\sum_{j \in K} I_{ij}(B)^2}} \tag{4}$$

$$\mathrm{ssim_i}(A,B) = \frac{\sum_{j \in N_i} I_{ij}(A) \times I_{ij}(B)}{\sqrt{\sum_{j \in N_i} I_{ij}(A)^2} \times \sqrt{\sum_{j \in N_i} I_{ij}(B)^2}} \tag{5}$$

For finding like-minded users, the pairwise similarity comparison for all users is a large scale problem. This paper uses a parallel filtering algorithm based on the characteristics of parallel computing and a personal ontology in the Hadoop.

3.2 MapReduce Brute Force

Measuring similarities between users' personal ontologies is an all pairs problem. The pair set is like a N×N symmetric matrix where N is the number of users. To improve the large-scale computing problem of the brute force approach, we only compute the upper-triangular part of the all-pairs matrix and write out both symmetric pairs. The map function emits the personal ontology similarity of every pair. The reduce function sorts the similarities to find out the top K of like-minded users for a particular user.

Each map function picks candidate pairs based on the upper-triangular part and computes the similarity score of candidate pairs with a particular user. The map function emits the symmetric user pairs after completing similarity computing. The reduce function accepts all similarity scores associated with a particular user ID and emits the top K results as the output by implementing a priority queue to sort similarity scores. The function *ComputeSimilarity* of the map function consists of keyword similarity and structural similarity.

To optimize the brute force approach we tune the number of users for one map task. That is, every map task reads at least one user from the input data. We use this method in our experiment.

3.3 MapReduce Parallel Filtering

For pairwise similarity computations of PORE on the personal ontologies, we propose the parallel filtering algorithm for evaluating all pairs that have one more preference in common with the inverted index.

The parallel filtering algorithm can be expressed as two modes: keyword similarity and structural similarity. The process of keyword similarity is different from the process of structural similarity because the hierarchical structure of the reference ontology affects the computation in the structural similarity. In the following we describe both modes in detail.

3.3.1 Keyword Similarity

The basic idea is to use an inverted index to filter the pairs without any interested keyword in common. The pseudocode for building an inverted list using MapReduce is shown in Fig. 1.

```
procedure Map(*,$U_i$)
   C ← FetchCategorySet($U_i$)
   for all $t \in C$ do
     $K$ ← FetchKeywordSet($t$,$U_i$)
     for all $e \in K$ do
       $s$ ← FetchKeywordLevel($e$,$U_i$)
       $ck$ ← <$t$,$e$>
       Emit($ck$,<$U_i$,$s$>)
procedure Reduce($ck$,[<$U_1$,$s_1$>,<$U_2$,$s_2$>,…])
   Define ArrayList $G$
   for all <$U_i$,$s_i$> $\in$ [<$U_1$,$s_1$>,<$U_2$,$s_2$>,…] do
```

Fig. 1. Pseudocode of building an inverted list for the keyword similarity of the PORE using MapReduce

In the process of keyword similarity this can be expressed as two separate MapReduce jobs, the first to build an inverted index and the second to compute similarities. For each interested keyword in the personal ontology, the mapper emits the set of category and keyword as the key, and a tuple consisting of the user ID and interested level of the keyword as the value. i.e. <U_i, s>. These tuples are grouped by the set in the shuffle phase. The reducer writes them to generate the inverted lists.

3.3.2 Structural Similarity

For the structural similarity of PORE, the relationship between a category and its subcategories is considered to be the level of structural similarity.

According to Equation 5, whether the category nodes of both personal ontologies are the same affects their structural similarity, because PORE adopts cosine-based similarity to compute structural similarity. Fig. 2 shows that we can first merge ontologies of two users by matching the reference ontology and then computing its similarity. For example, for category 312 of user A and category 312.6 of user B, category 312 and category 312.6 are of the parent-child relationship, although their IDs are different. Finally, the structural similarity between user A and user B is SS_1 + SS_2 when ontologies are merged.

Therefore, we use a hybrid method to scale out the structural similarity of PORE. The first step is to use an inverted index to filter out the pairs that are without any interested categories in common as shown in Fig. 3.

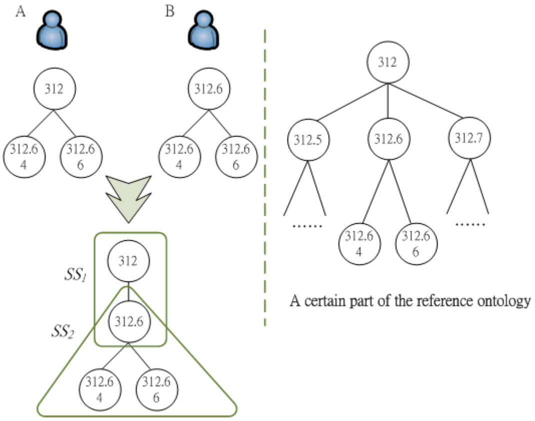


Fig. 2. Computing structural similarity by merging the ontologies of both user A and B in PORE

```
procedure Map(*,$U_i$)
  $C$ • FetchCategorySet($U_i$)
  for all $t \in C$ do
    Emit($t$,$U_i$)
procedure Reduce($t$,[$U_1$,$U_2$,…])
  for all $U_i \in [U_1, U_2, …]$ do
    Emit($t$,$U_i$)
```

Fig. 3. Pseudocode of the inverted index for the structural similarity of the PORE using MapReduce

The second step is to apply the third user-defined MapReduce job to all the pairs generated through parallel computing as shown in Fig. 4.

```
procedure Map($t$,$U_i$)
  for all $U_j$ ∈ [$U_1$,$U_2$,…] do
    if $U_i$ != $U_j$ then
        $P$ ← <$U_i$,$U_j$>
        Emit(P,NULL)
procedure Reduce([$P_1$,$P_2$,…],NULL)
  for all <$U_i$,$U_j$> ∈ $P$ do
    Emit(*,<$U_i$,$U_j$>)
```

Fig. 4. Pseudocode of pairwise users for the structural similarity of the PORE using MapReduce

4 Experimental Results

Experiments were run on a cluster with 16 machines. Each machine had one quad-core processor (2.4 GHz), 8GB memory, and two hard disks of 1.5TB as HDFS and about 640GB as MapReduce temporary. We improve the existing PORE system for pairwise personal ontology similarity. In June 2012 we collected 206,012 books with approximately 1,357,000 keywords, 51,454 user accounts, and 663,619 loan records from National Chung Hsing University (NCHU). We retrieved 10,000 user accounts from the dataset as experimental data to implement the proposed algorithm, and utilized the same dataset to compare against an equivalent run on a laptop with a dual-core 2.4GB processor, 4GB memory, and a 500GB hard disk.

The computation time of the parallel filtering algorithm consists of keyword similarity and structural similarity that can be run in parallel. Therefore, we chose the

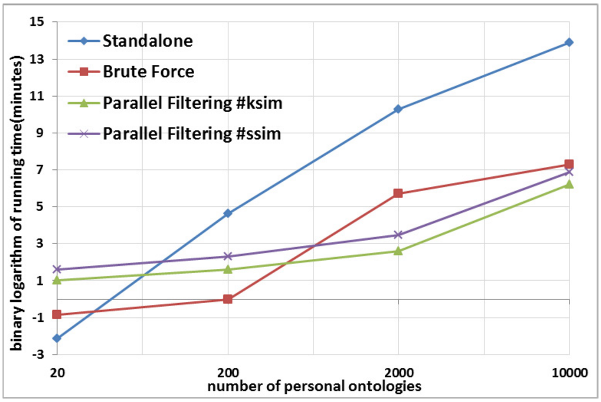


Fig. 5. Computation time of the algorithms for the PORE system

maximum computation time for these as the computation time of the parallel filtering algorithm. Fig. 5 shows that the process with standalone for 2000 users is completed in approximately 1260 minutes.

Computing similarity is very effective using MapReduce. This means that parallel computing is effective for the all pairs problem. The algorithm, parallel filtering, is more effective than the brute force approach when the number of users is 10,000. We need one more day to estimate the recommended information in the original library recommender system. However, the system can only take several hours for recommendation using our proposed algorithm.

5 Conclusions and Future Work

Finding like-minded users in a digital library is an all pairs problem and is also challenging for large collections of items. We investigate the problem of all pairs for personal ontology and introduction two MapReduce algorithms: brute force and parallel filtering. The parallel filtering algorithm, consisting of keyword similarity and structural similarity, is based on the inverted index approach using MapReduce. Also, we implement this algorithm in the PORE system.

Experimental results show that the parallel filtering algorithm is more effective, and solves the problems of finding like-minded users and the personal ontology comparison using MapReduce.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE TKDE, 734–749 (2005)
2. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295 (2001)
3. Avancini, H., Candela, L., Straccia, U.: Recommenders in a personalized, collaborative digital library environment. Journal of Intelligent Information Systems 28(3), 253–283(31) (2007)
4. Liao, I.-E., Liao, S.-C., Kao, K.-F., Harn, I.-F.: A Personal Ontology Model for Library Recommendation System. In: Sugimoto, S., Hunter, J., Rauber, A., Morishima, A. (eds.) ICADL 2006. LNCS, vol. 4312, pp. 173–182. Springer, Heidelberg (2006)
5. Liao, I.-E., Hsu, W.-C., Cheng, M.-S., Chen, L.-P.: A library recommender system based on a personal ontology model and collaborative filtering technique for English collections. The Electronic Library 28(3), 386–400 (2010)
6. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. ACM Communication 51(1), 107–113 (2008)
7. Lin, J.J.: Brute force and indexed approaches to pairwise document similarity comparisons with MapReduce. In: SIGIR 2009, pp. 155–162 (2009)
8. Elsayed, T., Lin, J., Oard, D.W.: Pairwise Document Similarity in Large Collections with MapReduce. In: Proc. HLT, pp. 265–268 (2008)

Layering of the Provenance Data for Cloud Computing

Muhammad Imran and Helmut Hlavacs

Research Group Entertainment Computing, University of Vienna, Wien, Austria
imran.mm7@gmail.com,
helmut.hlavacs@univie.ac.at

Abstract. With the recent advancements in distributed systems, Cloud computing has emerged as a model for enabling convenient, on-demand network access to a shared resource pool of configurable elements such as (networks, servers, storage, applications, and services). Various applications are developed and deployed into the Cloud following the layered architecture. The layered approach includes infrastructure, virtualization, application, platform and client tiers. Provenance (the meta-data), is the information that helps cloud providers and users to determine the derivation history of a data product, starting from its origin. Each layer in the Cloud has its own provenance data and generally, provenance data for each layer address different audience. For example, Cloud providers are interested in the infrastructure provenance data to verify the high utilization of resources through audit trials. Cloud users on the other hand are interested in the performance of the deployed application and the verification of experiments. In this paper, we present various queries regarding the provenance data for different layers of Cloud. Hereby, we integrate the provenance data from individual layers and highlight the importance of integrated provenance. We also outline the relationship between various layers of the Cloud by using the integrated provenance.

1 Introduction

Cloud computing is generally defined by its distributed model of utility computing which offers virtualization of resources (storage, computation, networking) and provisioned to users "on demand" and "pay as you go" basis. This new paradigm attracted the research community and businesses to host and execute their complex scientific applications [1] [1, 2]. In this model, applications are deployed and executed by using the type of service offered in the Cloud. These services reside on various layers or tiers of the Cloud architecture. For instance, Cloud providers are interested in the IaaS (Infrastructure as a Service) [2] layer of the Cloud, which supports virtualization of resources to enable computation, storage and communication. These resources are utilized by Cloud applications e.g., getting email [3] or for sharing documents, often termed as SaaS (Software as a Service). To fill the gap between IaaS and SaaS, the PaaS (Platform as a

[1] http://aws.amazon.com/swf/ [2] www.eucalyptus.com [3] www.gmail.com

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 48–58, 2013.

Service) layer is used by developers to customize and easily develop, deploy and manage Cloud-aware applications e.g. salseforce.com [4], WSO2 [5] and/or providing Enterprise Service Bus (ESB) [6] as a service.

Provenance is the metadata which describes the derivation history of an object. This data includes the source and intermediate datasets and processes involved to create the object [3]. In computing science, provenance is an important ingredient for the verification and reproduction [4, 5] of scientific experiments. The architecture of Cloud computing is divided into various components and these components are placed on top of each other [6]. IaaS, PaaS and SaaS are the types or layers which are mostly used in the Cloud environment. The development and execution of Cloud-aware applications follow this layered architecture and each layer contributes specific metadata (provenance) for the overall application. Subsequently each layer in the Cloud has its own provenance and specific importance to that particular layer. For example, the provenance data at the IaaS layer is important to the Cloud provider for resources utilization and fault tracking [7]. Cloud users (research community) are more interested in the execution of their deployed applications; the datasets which are produced/consumed, and the processes used for the production of the result.

Moreover, the provenance collection at individual layers e.g., for IaaS it can ensures the appropriate allocation and usage of the resources. Similarly, in case of faults and errors appropriate actions can be taken to resolve them accordingly by using the provenance [8]. When provenance is integrated from individual layers, it provides the in-depth details of the relationships which exist among various layers of the Cloud while executing a particular application. The integrated provenance data provides multiple views and enables Cloud providers to keep track of their resource usage, application and service collaboration for users and deployment/testing usage for developers.

For understanding the Cloud layered approach and the overall provenance data at each layer, in this paper we have developed a Content Relationship Management (CRM) application. With this particular CRM application, we differentiate between various layers of Cloud and their corresponding provenance data. We provide detail of the CRM application and its various parts from the users perspective, the Cloud provider and application developer. Following are the key contributions of this paper:

- to provide an overview of the Cloud layered technology and the presentation of provenance data for each layer.
- to present various queries and their visualization for the individual layers of the Cloud and for the collective provenance data.
- to highlight the importance of integrated provenance using an example.
- to evaluate the overhead for provenance collection at individual layers.

The rest of the paper is organized as follows. Section 2 provides the related work of provenance in the field of e-Science. Section 3 discusses the requirements for building a CRM application in the Cloud and its individual components.

[4] www.salesforce.com [5] www.wso2.com [6] http://www.mulesoft.org/

Section 4 provides a brief overview of layers in Cloud computing and Section 5 present the various quires for the provenance data on individual layers, their visualization and discusses the importance of integrated provenance data. Section 6 evaluate the collection of provenance data from different layers and section 7 concludes this paper.

2 Related Work

Application level provenance has been the major attraction in grid, distributed and workflow computing [5]. The techniques used in these environments are to capture provenance in Service Oriented Architecture(SOA) e.g., PASOA [9]. Recently, the research community focused on the usage of provenance for Cloud computing while describing and addressing the various challenges offered by this new paradigm [10, 11].

Previously [12], we proposed a framework which addresses the various challenges offered by Cloud technology and present the mechanism to incorporate the collection and storage of provenance for the Cloud IaaS. On the development layer of the Cloud, e.g., in mule ESB and WSO2 carbon platforms, various parts of application are integrated together that communicate based on different protocols and languages. Integration of provenance into the development layer will clearly identify the current status of any application, changes made by different developers of a group or team and information about the old and current version of the services and applications. There are other work which consider provenance at the layers like a web browser [13] and virtual machine [14].

To establish the importance of integrating provenance data from different layers, Muniswamy Reddy et. al. [15] discussed the layering of provenance data for workflow execution. However their work focused on combining the provenance data from a workflow engine, web browser and the python wrapper by extending the Provenance Aware Storage System (PASS) [16]. For Cloud environment, a short survey about various techniques from Grid and distributed computing are discussed to track the data in Cloud by using a layered architecture [17]. In this paper, we extend our provenance framework [18] for the platform and software layers of the Cloud.

3 Scenario Description (Components of CRM Application)

In this scenario the objective is to automate the installation of a sample CRM application. The sample CRM application consists of three main components which are: 1) the web server 2) the database server and 3) the client application. **Component 1** is the web server where we have two different web services. Web service 1 takes the data from the user and submit or sync it to the database server. This sync can be performed either for one particular item e.g., contacts or for over all data e.g., contacts, appointments and tasks. Web service 2 takes the data from the server and sync it to the client application. Again, the sync process is for one particular item or overall data.

Component 2 is the database server. It is mySQL database with various tables containing the information about an organization or a group e.g., contacts, appointments and tasks. The contacts table contains first name, last name, job title, group name etc. Appointments table contains information like place, time, appointment with, number of people attending, topic and location. Tasks table contains information like sender, receiver, title, subject, description etc.

Component 3 is the client application for a user to to see the tasks assigned to him/her and list the appointments. This also includes, who assigned the tasks, meetings and appointments time, members involved and location.

Summary: To link all these resources with each other, a script is required which deploys the application on Cloud and host the various components. Such a scrip will be passed to Cloud controller via user data. The end result would be a deployment of CRM system. Figure 1 presents the steps involved in deployment of such an application to the Cloud. We consider three resources to host web services, database server and client application. Each installed components requires some prerequisite and those need to be installed and configured. While deploying CRM, we observe and provide the details of the various components of Cloud and related provenance data.

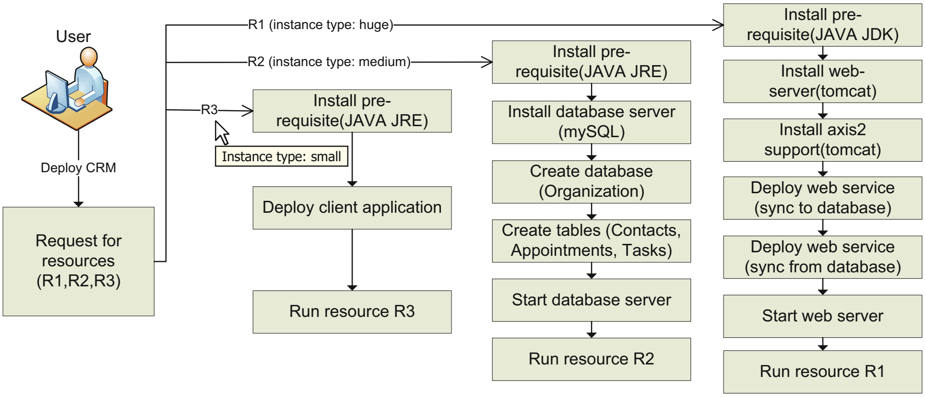

User
Deploy CRM
Request for resources (R1,R2,R3)
R1 (instance type: huge)
R2 (instance type: medium)
R3
Instance type: small
Install pre-requisite(JAVA JRE)
Deploy client application
Run resource R3
Install pre-requisite(JAVA JRE)
Install database server (mySQL)
Create database (Organization)
Create tables (Contacts, Appointments, Tasks)
Start database server
Run resource R2
Install pre-requisite(JAVA JDK)
Install web-server(tomcat)
Install axis2 support(tomcat)
Deploy web service (sync to database)
Deploy web service (sync from database)
Start web server
Run resource R1

Fig. 1. Steps involved in deployment of CRM application to Cloud

4 Cloud Layered Architecture

The Cloud architecture is perceived differently by various research community and businesses [19, 20]. Mainly we consider the following layers/components.

- Infrastructure layer: provides physical and virtual resources for storage, communication and computation e.g., Eucalyptus.
- Platform layer: provides tools and libraries to ease the development cost and effort for building Cloud aware application e.g., WSO2.

- Software layer: The applications which are provided by various organizations to vast number or users e.g. web application (gmail).

Figure 2 presents the main components in Cloud computing from the view point of a user, developer and Cloud resources provider. To connect the layers, there is always a middle-ware in between. We collect the provenance data on that middleware level. Previously we explained the mechanism to collect IaaS provenance data in [12] by using the interceptor mechanism and why such data is important. For this work, we extend the same mechanism, guideline and architecture of provenance towards PaaS and SaaS layers of Cloud computing.

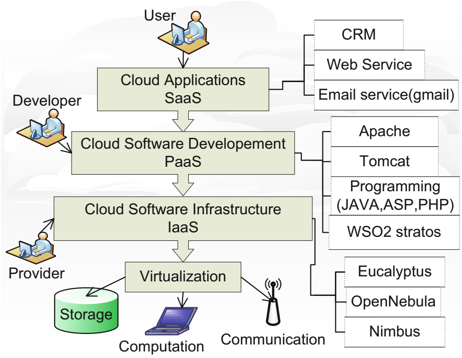


Fig. 2. Layered architecture of Cloud

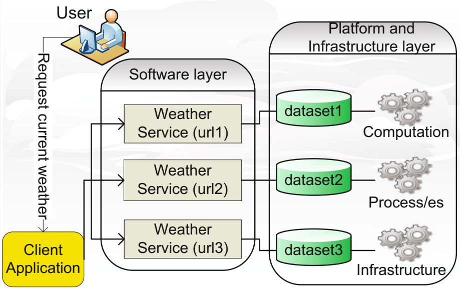


Fig. 3. Weather request from Cloud

4.1 Query and Visualization

The presentation of the collected provenance data is very important for users, administrators and application developers in Cloud and it depends on the submitted query. For a particular query, we may analyze the provenance data of one particular layer or integrated provenance. For example, when an administrator submit the following query:
Visualize the instance types from cluster1, requested by various users where the number of request are more than 100
This query requires the analysis of the infrastructure provenance. This provenance data is stored in a well defined xml file where the nodes represent the individual objects and the edges represent the relationship which exists in the provenance data. The numbers of relationship varies for the submitted query. For this particular query, the following relationships can be defined: i) request of instances types for cluster1 ii) relation of instances to various users iii) and relation of users to the cluster and instances. For analysis of this query and defining the relation, we apply pull based mechanism for the extraction of data from provenance. After the analysis of the submitted query and defining the relationships, we present the results in the graph form. These graph can be changed on run time and the results can be visualized in line, bar and pie forms.

5 Provenance and Cloud Layers

Following the layered architecture of Cloud, provenance is also divided into different layers. Each layer presents a different application domain for the usage of provenance data. The sections below investigate the provenance data and various queries which require individual and/or the integrated provenance.

5.1 IaaS Provenance and Queries

The infrastructure layer provides computational, storage and communication resources for the application deployment and services execution. Various parameters are considered when defining provenance data for IaaS Cloud e.g., 1) types of resources 2) types of instance 3) information about users 4) time taken by users for instances 5) data submitted by users before running a resource 6) information about cloud, clusters and node services. In the CRM application, these parameters maps to user (admin), resource types (R1, R2, R3), instance types (small, medium and huge), data (java jdk, tomcat, axis2 and mySQL versions).

Many applications can be defined depending on the granularity of the provenance data e.g., 1) to use the provenance data for auditing the usage of resources in Cloud. Provenance data can clearly mark the usage of resources from various clusters and nodes according to time, users, and resources types. 2) to find the similar requests in Cloud which are based on the instance types and user data. These similarities define patterns and are used for the efficient utilization of Cloud resources. The efficient utilization is achieved by reusing the existing running resources and predicting the upcoming requests. 3) the provenance data of various images is used for tracking malicious images uploaded in public Cloud and managing the access rights to various images [21]. Following are few example queries which can be generated for the Cloud infrastructure:

(i) visualize the instance types from last 24 hours (ii) visualize the standard instances from last 48 hours (iii) visualize the memory request from last week (iv) visualize the request for resources from most used to least used (v) list the prerequisite (user data) from R1 (vi) list the deployment time for resource R1, R2 and R3 (vii) validate the setup of CRM application.

The combination of the infrastructure provenance data with physical machine provenance e.g., memory, CPU utilization and the disk usage can further elaborate the provenance query:

– visualize the disk and memory usage for the most requested resources type.

These queries provides the administrator with an overview of the resources usage, instance types and data requested by users. Left side of the figure 4 present the memory requests for a particular cluster grouped for various users and right side of the figure 4 present the memory requests from various users in time by using the visualization module. Due to limited space, we will not present visualization of the provenance data for other layers.

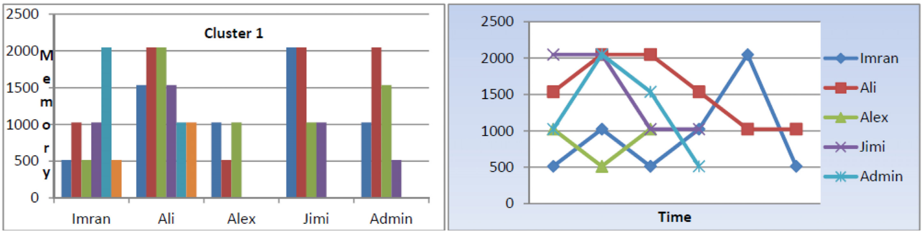


Fig. 4. Memory requests for a Cluster from various users

5.2 PaaS Provenance and Queries

Platform layer in Cloud provides the functionality for the development of new applications. This layer enables and manges the delivery of services that uses various communication protocol e.g. HTTP, XML and SOAP etc. The designer of these applications is responsible for assets availability and the management of application services pool. For example, Cloud is a favorable environment in stress testing, where a lot of resources are required just for a particular period of time. For this work, we consider WSO2 platform and it's various components.

WSO2 Carbon is the award winning PaaS and it provides many features to developers for building Cloud aware applications e.g., Enterprise Service Bus (ESB), Business Process management (BPM) and more. While developing applications using WSO2, the complex application is divided into various parts. Members of a team/s work on different components of a complex application. In the CRM application various parameters which are considered for the provenance data of platform layer are: 1) composition of the web services 2) the interaction mechanism between web services, database and web service engine, that is utilized by various protocols of communication 3) the interaction mechanism between web services and client application 4) composition of the database and corresponding tables and their structure. When one developer makes a change in a web service, other members will be able to find that particular change using the provenance data. Any change on platform layer e.g., uploading a new version of the web service will create a new node in the provenance data and hence the status will be updated. Some important queries on platform layer are following:

(i) visualize the components of the application where most bugs are found (ii) the identity of a person who made a change in CRM and the time when the change was made (iii) display the changes made to web services in last one month (CRM)

Consider a situation where the infrastructure is changed e.g, mySQL server is updated. The updated version does not support the existing communication mechanism with the deployed web services. This requires a change in the web services at the platform layer. This relation which exists between platform and infrastructure layer is exposed by integrating the provenance data from individual layers. The integrated provenance data and the corresponding relation will highlight the reason for any communication failure.

5.3 SaaS Provenance and Queries

SaaS is the application running on a Cloud platform. Various types of applications are deployed and executed on Cloud e.g., workflow, CRM and web applications. The provenance data of this layer depends on the type of application. In general, applications are deployed by using Service Oriented Architecture (SOA) in distributed computing. The important provenance parameters in SOA architecture and related queries are following:

(i) time taken by a particular application to generate the result (ii) time taken by individual services and components of the application (iii) tracking the dataset which are consumed and produced during the process (iv) information about the users who are invoking the services (v) the query about services or components taking part while executing the application e.g., services involved in executing a workflow (vi) input and output parameters passed to a particular service and/or method.

In the CRM application, the analysis of various events is an important aspect for organizations. The provenance data about users, time, and events is used for analysis and to get important informations like; the locations of the event, total time for the event, members who joined the event, the organizers of the event etc. There are other aspects of the provenance data for software layer e.g., trust, reliability and authenticity.

Considering a situation where changes are made to the web services on platform layer. This will require appropriate changes to the client application. If the client application is not updated, any sync process from database to the client application will result in failure. The provenance data from the client application will highlight the failure. User can use the provenance chart to find the failure, but it's reason is not clear until we layer the provenance data from platform to the software layer. Layering the provenance data will further explain the reason of failure and related data for changes made on platform layer.

5.4 Advantage of Integrated Provenance Data

In the above sections, we deployed the CRM application into the Cloud environment. We collected the provenance data on individual layers, provided various parameters, queries and the relations which exists between layers. Considering the fact that Clouds are abstract and the various layers are hidden from the user, we present the example in figure 3. Users request for the current weather information using a particular city and country. The client application randomly chooses one of the weather web services which are provided by different organizations. The selected service returns the current weather information. Since, these services use different datasets for the calculation of the weather, the result is not always the same. In scientific environment, it is important to know why the results differs from each other.

Without layering: Each layer of provenance gives valuable information. The software layers provides the provenance data for the selected web service and methods. The platform provenance provides the information regarding the datasets which are consumed and the algorithm which is used for the calculation. The infrastructure layer provenance data gives information regarding the Cloud provider and the location of the computation, storage, and communication devices.

With layering: The integrated provenance data from software, platform and infrastructure layer identifies the datasets which are used, the web service which is consumed and information about the Cloud provider. These information provides the relation between various layers and hence highlight the reason that why different results are not always the same.

6 Overhead Evaluation

The integrated solution of provenance into Cloud infrastructures particularly for e-Science applications causes extra overhead of calculation and storage. The calculation overhead is the extra time needed for the collection, parsing and storage of the provenance data. In our experiments, the overhead is calculated for the individual layers of Cloud using the CRM application. The calculation is performed for the various components of the CRM application which correspond to Cloud layers. At the infrastructure layer, we tested the eucalyptus Cloud with *node controller* and *cluster controller* services. Platform layer is tested with WSO2 *application service* and *enterprise service bus*. The software layer is evaluated for the web services *snyntodatabase* and *syncfromdatabase* in CRM.

Table 1 presents the performance overhead of provenance from various components in Cloud and CRM application. The maximum times are the exceptional cases and therefore average time was calculated from multiple runs (50) of components and layers. The average time presents the overhead for collection, parsing and storing of the provenance. Formula 1 is used to calculate the overall overhead by summation of individual overhead from software, platform and infrastructure layer.

Depending on the granularity and storage mechanism, time required for provenance may slightly vary. The very low overhead explains the utility of our provenance collection technique which follows the interceptor based approach for collection and link based approach to store the data [12]. Given the overall advantages of provenance, this extra overhead is negligible.

$$Total\,Overhead = \sum_{i=0}^{n}(\mathbf{S})i + \sum_{i=0}^{n}(\mathbf{P})i + \sum_{i=0}^{n}(\mathbf{I})i \tag{1}$$

7 Conclusions

In this paper we emphasis on the provenance data at the various layers of Cloud infrastructures for the applications deployed there in. To achieve this, first, we identified individual layers in Cloud computing and presented the related provenance data for each layer. Then various queries were explored that could be answered using the hierarchical architecture of Cloud and deployed applications.

Table 1. Calculation time overhead for provenance in milliseconds

Cloud layers	Max time(ms)	Min time(ms)	Avg time(ms)
Software (CRM application)	18	2	7
Platform (WSO2 AS)	22	1	3
Platform (WSO2 ESB)	12	1	2.5
Infrastructure (Eucalyptus NC)	15	2	4
Infrastructure (Eucalyptus CC)	20	7	12
Combined			26 ms

These queries utilized the provenance of individual layer or the integrated provenance data. Further, the identification of relations is provided which exists for one particular layer or in the integrated provenance data. By exploiting the Cloud architecture, we divided the provenance into various layers and presented the mechanism to query and visualize different requests from the perspectives of various stakeholders including users, developers and Cloud providers themselves.

References

[1] Deelman, E., Singh, G., Livny, M., Berriman, B., Good, J.: The cost of doing science on the cloud: The montage example (2008)

[2] Vöckler, J.S., Juve, G., Deelman, E., Rynge, M., Berriman, B.: Experiences using cloud computing for a scientific workflow application, pp. 15–24. ACM, USA (2011)

[3] Barga, R.S., Simmhan, Y.L., Chinthaka, E., Sahoo, S.S.: Jackson: Provenance for scientific workflows towards reproducible research. IEEE Data Eng. Bull. (2010)

[4] Bose, R., Frew, J.: Lineage retrieval for scientific data processing: a survey. ACM Comput. Surv. 37(1), 1–28 (2005)

[5] Simmhan, Y.L., Plale, B., Gannon, D.: A Survey of Data Provenance Techniques. Technical report, Computer Science Department, Indiana University (2005)

[6] Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee: Above the Clouds: A Berkeley View of Cloud Computing (2009)

[7] Imran, M., Hlavacs, H.: Applications of provenance data for cloud infrastructure. In: Eighth International Conference on Semantics, Knowledge and Grids (SKG), pp. 16–23 (2012)

[8] Crawl, D., Altintas, I.: A provenance-based fault tolerance mechanism for scientific workflows. In: Freire, J., Koop, D., Moreau, L. (eds.) IPAW 2008. LNCS, vol. 5272, pp. 152–159. Springer, Heidelberg (2008)

[9] Miles, S., Groth, P., Branco, M., Moreau, L.: The requirements of recording and using provenance in e-Science experiments. Technical report (2005)

[10] Muniswamy-Reddy, K.K., Seltzer, M.I.: Provenance as first class cloud data. Operating Systems Review 43(4), 11–16 (2009)

[11] Muniswamy-Reddy, K.K., Macko, P., Seltzer, M.: Provenance for the cloud. In: FAST 2010, pp. 197–210. USENIX Association (2010)

[12] Imran, M., Hlavacs, H.: Provenance in the cloud: Why and how? In: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, pp. 106–112 (2012)

[13] Margo, D.W., Seltzer, M.I.: The case for browser provenance. In: Workshop on the Theory and Practice of Provenance (2009)

[14] Macko, P., Chiarini, M., Seltzer, M.: Collecting provenance via the xen hypervisor. In: Workshop on the Theory and Practice of Provenance (2011)

[15] Muniswamy-Reddy, K.K., Braun, U., Holland, D.A., Macko, P.: Maclean: Layering in provenance systems. In: USENIX, USA (2009)

[16] Muniswamy-Reddy, K.K., Holland, D.A., Braun, U., Seltzer, M.I.: Provenance-aware storage systems. In: USENIX, pp. 43–56 (2006)

[17] Zhang, O.Q., Kirchberg, M., Ko, R.K.L., Lee, B.S.: How to track your data: The case for cloud computing provenance. In: CloudCom 2011, pp. 446–453 (2011)

[18] Imran, M., Hlavacs, H.: Provenance framework for the cloud environment (iaas). In: The Third International Conference on Cloud Computing, GRIDs, and Virtualization (2012)

[19] Youseff, L., Butrico, M., Da Silva, D.: Toward a Unified Ontology of Cloud Computing. In: Grid Computing Environments Workshop, GCE 2008, pp. 1–10 (2008)

[20] Rochwerger, B., Breitgand, D., Levy, E., Galis, A., Nagin, K.: The reservoir model and architecture for open federated cloud computing (2009)

[21] Wei, J., Zhang, X., Ammons, G., Bala, V., Ning, P.: Managing security of virtual machine images in a cloud environment. In: CCSW, pp. 91–96. ACM (2009)

JCL: An OpenCL Programming Toolkit for Heterogeneous Computing

Tyng-Yeu Liang and Yu-Jie Lin

Department of Electrical Engineering
National Kaohsiung University of Applied Sciences
lty@mail.ee.kuas.edu.tw,
jaredlin@hpds.ee.kuas.edu.tw

Abstract. In this paper, we propose a new OpenCL toolkit called JCL for heterogeneous clusters. Using this toolkit, users can make use of multiple remote heterogeneous processors including CPUs and GPUs for the execution of their OpenCL programs. Since load balance is an important issue for the performance of the user programs executed by heterogeneous processors, the proposed toolkit provides users with a set of load-balancing functions to automatically adjust the amount of data assigned to each processor according to processor's computation power. We have evaluated the performance of the proposed toolkit in this paper. Our experimental result shows that the proposed toolkit really can enable the test programs to effectively exploit heterogeneous processors for enhancing their execution performance.

Keywords: heterogeneous cluster computing, GPU, OpenCL, load balance.

1 Introduction

Recently, NVidia and AMD have proposed their own general-purpose graphic processing unit (GPGPU, simply called GPU [1] later) for scientific computing. Since GPU has a high density of computational cores and consumes less energy per instruction, it is more powerful for data computation and is more helpful for energy saving and carbon reduction than CPU. Accordingly, more and more cloud-service providers such as Amazon and Google start to provide not only CPU but also GPU resources for users to resolve data-intensive or massive-computation problems.

However, the proposed GPU programming toolkits such as CUDA [2] and Brook [3] are very different from OpenMP [4] or Pthread [5], which are popularly used for multi-core CPUs. This problem is a big barrier for simultaneously exploiting CPU and GPU to resolve the same problem since programmers have to learn different programming interfaces, and use them in the same program. Fortunately, Khronos Group has proposed a standard called OpenCL[6] for resolving this problem. The OpenCL programs can be executed by different processor architectures including CPU, GPU, CELL [7], and FPGA [8] etc. Consequently, OpenCL successfully reduces the programming complexity of heterogeneous processors. However, most of the implementations of OpenCL do not support cluster computing. When users intend to make use

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 59–72, 2013.

of cluster resources, they still need to combine OpenCL with MPI [9] in their programs to distribute data over loosely-coupled resources for concurrent computation. This is not convenient for users to exploit the resources of heterogeneous clusters.

To resolve this problem, we propose an OpenCL programming toolkit called JCL for heterogeneous cluster computing in this paper. With the support of JCL, users can transparently make use of CPUs and GPUs available in computer networks for parallel computation while they don't know the location of resources. From the viewpoint of users, they feel that their programs are executed on a computer with many OpenCL-compatible devices since they don't have to use MPI for data distribution any more. Consequently, the programming of heterogeneous clusters can be effectively simplified. Moreover, JCL supports load balance. That is, user programs can automatically self-adjust their data partition by calling the load balancing functions of JCL to achieve load balance among heterogeneous processors, and thereby enhance their execution performance.

The rest of this paper is organized as follows. Section 2 is background of OpenCL. Section 3 and Section 4 briefly describe the framework and implementation of JCL, respectively. Section 5 discusses the performance of JCL. Section 6 compares JCL with related programming toolkits. Finally, section 7 gives a short conclusion for this paper and our future work.

2 Background

OpenCL is a standard proposed by Khronos for programming on heterogeneous processor architecture. The user applications developed by using OpenCL can be executed with CPU, GPU and processors of other types. As a consequence, users need not learn a particular programming toolkit dedicated for each type of processors. The interfaces of OpenCL basically can be classified into platform layer and runtime layer. The functions of the platform layer are used for platform control while the functions of the runtime layer are used for executing kernel functions on the target device. User programs usually use the functions listed in Table 1.

Table 1. OpenCL APIs

Query Platform	clGetPlatformIDs()
	clGetPlatformInfo()

Query Devices	clGetDeviceIDs()
	clGetDeviceInfo()

Contexts	clCreateContext()
Command Queues	clCreateCommandQueue()

Memory Objects	clCreateBuffer()
	clEnqueueReadBuffer()
	clEnqueueWriteBuffer()

Program Objects	clCreateProgramWithSource()
	clBuildProgram()

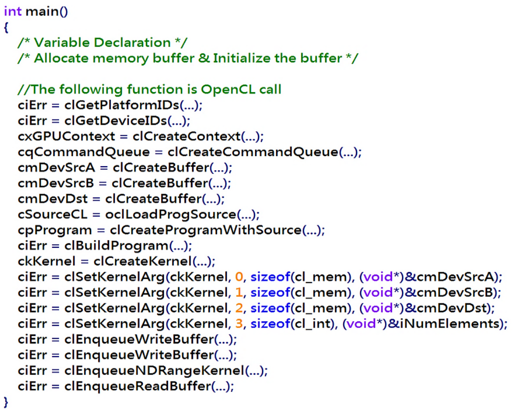
Table 1. *(Continued)*

Kernel Objects	clCreateKernel()
	clSetKernelArg()

Executing Kernels	clEnqueueNDRangeKernel()

Event Objects	clCreateUserEvent()
	clSetUserEventStatus()
	clWaitForEvents()

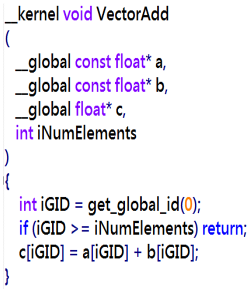
Here we give an example program of vector addition to briefly introduce the OpenCL functions as shown in Fig.1 and Fig.2. Basically, this program is partitioned into two parts: host and device. The host program is responsible for device allocation, creation and submission of kernel functions, and data communication between the host and the device. The device program is a kernel function executed by the device for processing problem data pending in the device memory.



```
int main()
{
   /* Variable Declaration */
   /* Allocate memory buffer & Initialize the buffer */

   //The following function is OpenCL call
   ciErr = clGetPlatformIDs(...);
   ciErr = clGetDeviceIDs(...);
   cxGPUContext = clCreateContext(...);
   cqCommandQueue = clCreateCommandQueue(...);
   cmDevSrcA = clCreateBuffer(...);
   cmDevSrcB = clCreateBuffer(...);
   cmDevDst = clCreateBuffer(...);
   cSourceCL = oclLoadProgSource(...);
   cpProgram = clCreateProgramWithSource(...);
   ciErr = clBuildProgram(...);
   ckKernel = clCreateKernel(...);
   ciErr = clSetKernelArg(ckKernel, 0, sizeof(cl_mem), (void*)&cmDevSrcA);
   ciErr = clSetKernelArg(ckKernel, 1, sizeof(cl_mem), (void*)&cmDevSrcB);
   ciErr = clSetKernelArg(ckKernel, 2, sizeof(cl_mem), (void*)&cmDevDst);
   ciErr = clSetKernelArg(ckKernel, 3, sizeof(cl_int), (void*)&iNumElements);
   ciErr = clEnqueueWriteBuffer(...);
   ciErr = clEnqueueWriteBuffer(...);
   ciErr = clEnqueueNDRangeKernel(...);
   ciErr = clEnqueueReadBuffer(...);
}
```

Fig. 1. Host Code of VectorAdd



```
__kernel void VectorAdd
(
  __global const float* a,
  __global const float* b,
  __global float* c,
  int iNumElements
)
{
   int iGID = get_global_id(0);
   if (iGID >= iNumElements) return;
   c[iGID] = a[iGID] + b[iGID];
}
```

Fig. 2. Kernel Code of VectorAdd

The execution of this program is described as follows. First, the host program queries the platform information by clGetPlatformIDs(), and then obtains a device from the platform by clGetDeviceIDs(). Second, it creates a program context for executing a kernel function with the device, and builds a command queue for sending commands to the device. Third, it calls clCreateBuffer() to allocate device memory for data communication between the host and the device, and invokes clLoadProgSource(), clCreateProgramWithSource() and clBuildProgram() to create the kernel function. Forth, it calls clSetKernelArg() for setting the arguments of the Kernel function, and then it copies data from the host memory to the allocated device memory by clEnqueueWriteBuffer(), and launches the kernel function into the device for processing the problem data by clEnqueueNDRangeKernel(). Finally, it copies the execution result from the device memory to the host memory by clEnqueueReadBuffer() after the execution of the kernel function is finished. When users intend to simultaneously exploit multiple OpenCL-compatible devices within a computer for data computation, they have to distribute data over a group of threads and make threads dispatch their assigned data and the same kernel function to different devices for concurrent computation.

3 JCL

JCL is compatible for the OpenCL standards 1.0. When users intend to execute their OpenCL on a heterogeneous cluster, the only thing they have to do is to recompile their programs and link with the JCL runtime library instead of the original OpenCL one. When the threads of a user program execute, the JCL client will transparently redirect the OpenCL functions invoked by the program threads to remote JCL servers for concurrent execution. As a result, users feel their programs are executed on a stand-alone machine with many devices. Moreover, JCL supports dynamic load balance for iteration applications. User programs can easily adjust the amount of data assigned to processors to achieve load balance, and thereby increase their execution performance.

The framework of JCL is based on a client-and-server model as shown in Fig. 3. The JCL client is responsible to catch the OpenCL functions issued by program threads, and then redirect the functions to remote JCL servers for execution via TCP/IP sockets [10]. In addition, it keeps track of track the execution time of each thread, and calculating the amount of data assigned to the thread based on its computational power whenever the thread calls the load-balancing function of JCL. On the other hand, the JCL server is responsible to manage OpenCL-compatible devices within a computer, and execute the OpenCL function calls coming from the JCL clients on the local devices. For transparent resource allocation, the JCL server gets the handles of all the devices within the local host by calling the OpenCL functions of getting device identifier, and registers its platform information and location to the resource broker in a computer cluster before it starts to serve the JCL clients. In the JCL framework, the client and the server can be executed on the same machine while the mapping between clients and servers is one-to-many. In other words, one client can be served by many servers while one server cannot serve multiple clients at the same time.

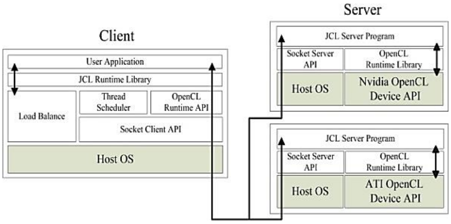


Fig. 3. Framework of JCL

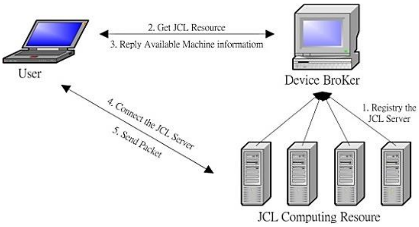
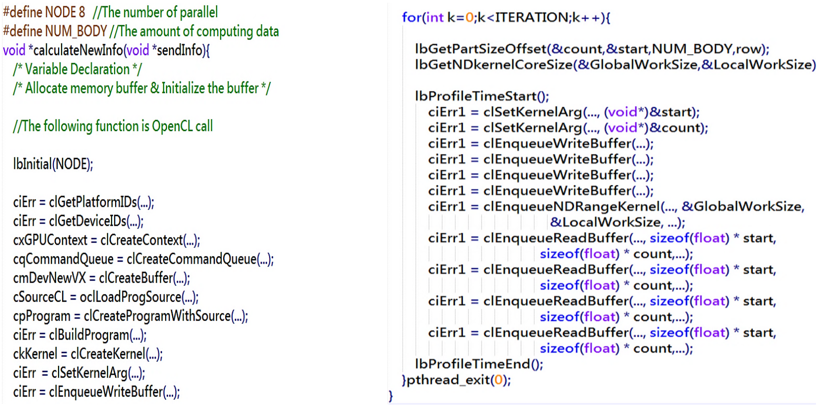

User
2. Get JCL Resource
3. Reply Available Machine informatiom
Device BroKer
1. Registry the JCL Server
4. Connect the JCL Server
5. Send Packet
JCL Computing Resoure

Fig. 4. Control flow of JCL

The scenario of program execution in the JCL framework is simply described as follows. When a user program starts to execute, the JCL client will ask the resource broker for allocating a group of free JCL servers according to user's resource requirement. After receiving the location and platform information of the servers allocated by the resource broker, the JCL client will ask the allocated servers for getting their device handles and will build a schedule table for mapping program threads to devices. Currently, the way of thread-to-device mapping is round robin. After resource allocation, the user program can continue to create a number of working threads according to the total device number, and evenly distributes data over the working threads for concurrent computation. When the program threads invoke OpenCL functions, the JCL client will look up the mapping table of threads to devices, and then redirect the invoked functions with the device handles to the mapped JCL servers for concurrent execution.

As previously described, JCL allows user programs to transparently exploit multiple heterogeneous resources including CPUs and GPUs at the same time. However, load balance is a big problem for the execution performance of user programs since the computational power of each processor is not identical, and the performance gap between CPU and GPU is very big. For resolving this problem, user programs can create more threads than devices, and then the JCL client dynamically allocate the threads onto the devices when idle JCL servers ask for more threads. As a result, the devices with higher computational power can get more threads than the others to achieve load balance. However, this method induces a significant overhead to affect the performance of high-speed GPUs since the JCL servers have to ask the JCL client for more pending threads many times. Therefore, JCL adapts data repartition instead of thread redistribution for load balance. To achieve this goal, JCL provides a set of load-balancing functions for users to partition their problems according to the computational power of processors. When user programs call the load-balancing functions, they can automatically adjust their data partition to achieve load balance among processors.



```
#define NODE 8  //The number of parallel
#define NUM_BODY //The amount of computing data
void *calculateNewInfo(void *sendInfo){
  /* Variable Declaration */
  /* Allocate memory buffer & Initialize the buffer */

  //The following function is OpenCL call

  lbInitial(NODE);

  ciErr = clGetPlatformIDs(...);
  ciErr = clGetDeviceIDs(...);
  cxGPUContext = clCreateContext(...);
  cqCommandQueue = clCreateCommandQueue(...);
  cmDevNewVX = clCreateBuffer(...);
  cSourceCL = oclLoadProgSource(...);
  cpProgram = clCreateProgramWithSource(...);
  ciErr = clBuildProgram(...);
  ckKernel = clCreateKernel(...);
  ciErr  = clSetKernelArg(...);
  ciErr = clEnqueueWriteBuffer(...);
```

```
for(int k=0;k<ITERATION;k++){

  lbGetPartSizeOffset(&count,&start,NUM_BODY,row);
  lbGetNDkernelCoreSize(&GlobalWorkSize,&LocalWorkSize)

  lbProfileTimeStart();
    ciErr1 = clSetKernelArg(..., (void*)&start);
    ciErr1 = clSetKernelArg(..., (void*)&count);
    ciErr1 = clEnqueueWriteBuffer(...);
    ciErr1 = clEnqueueWriteBuffer(...);
    ciErr1 = clEnqueueWriteBuffer(...);
    ciErr1 = clEnqueueWriteBuffer(...);
    ciErr1 = clEnqueueNDRangeKernel(..., &GlobalWorkSize,
                                     &LocalWorkSize, ...);
    ciErr1 = clEnqueueReadBuffer(..., sizeof(float) * start,
                                   sizeof(float) * count,...);
    ciErr1 = clEnqueueReadBuffer(..., sizeof(float) * start,
                                   sizeof(float) * count,...);
    ciErr1 = clEnqueueReadBuffer(..., sizeof(float) * start,
                                   sizeof(float) * count,...);
    ciErr1 = clEnqueueReadBuffer(..., sizeof(float) * start,
                                   sizeof(float) * count,...);
  lbProfileTimeEnd();
}pthread_exit(0);
}
```

Fig. 5. Nbody with using the load balancing functions of JCL

Here we give a working function of the threads in the N-body application to explain how to use the load-balancing functions of JCL as shown in Fig. 5. Nbody is an iteration application to simulate the motion of bodies under the effect of gravitation. At the beginning of each iteration, each thread calls lbGetPartSizeOffset() to obtain the amount of assigned data, and the address offset of the first one of the assigned data within a memory buffer. In addition, each thread calls lbProfileTimeStart() and lbProfileTimeEnd() to record the execution time of the current iteration for next-iteration data partition. Since the global working size must be dividable by the local working size based on the specification of OpenCL, each thread performs lbGetNDKernelCoreSize() to obtain the best size of the cores used to execute the kernel function for processing the assigned data.

4 Implementation

The implementation of JCL is based on Linux operation system and the TCP/IP protocol. We briefly describe how to implement the function redirection and load balance of JCL in this paper.

4.1 Redirection of Function Calls

Because of the length limit of the paper, here we describe the implementation of only three different OpenCL functions as follows.

cl_intclGetPlatformIDs(cl_uint num_entries,cl_platform_id *platforms, cl_uint* num_platforms).

This function is used to get the platform information. The *num_entries* argument is the number of platform. The platforms argument is the list of platform. The *num_platforms* argument is the number of platforms returned. The process of this function is shown in Fig. 6.

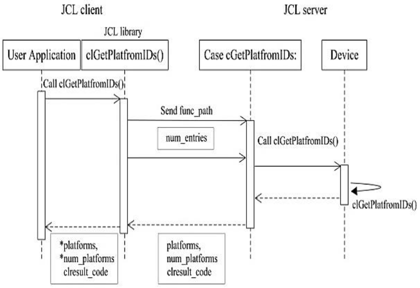


Fig. 6. Process of clGetPlatformIDs

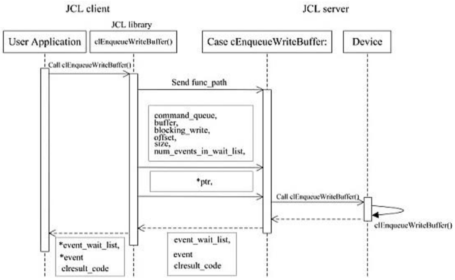


Fig. 7. Process of clEnqueueWriteBuffer()

When a user program calls this function, the JCL client will redirect this function to each of the JCL servers allocated to this program as shown in Fig.6. Each JCL server will generate a array which can store *N cl_platform_id* values, and a *cl_unit* variable called *num_platform* for calling clGetPlatformIDs(), and then return the *cl_platform_id* values and the *num_platform* value obtained by calling clGetPlatformIDs() to the JCL clinet. Finally, the JCL client will integrate all the *cl_platform_id* values coming from the JCL servers.

clEnqueueWriteBuffer (cl_command_queue command_queue, cl_mem buffer, cl_bool blocking_write, size_t offset,size_t size,const void *ptr,cl_uint num_events_in_wait_list, const cl_event *event_wait_list, cl_event *event).

This function is used for copying data from the host memory to the device memory. *Buffer* and *ptr* respectively denote the address of the host memory and the device memory. *Size* is the length of data. *Blocking_write* is a flag to specify the operation is blocking or non_blocking. The other augments are used to set waiting events. The process of this function is shown in Fig.7. When a program thread calls this function, the JCL client will send the values of *command_queue*, *blocking_write*, *offset*, *size*, and *number_events_in_wait_list* to the JCL server. The JCL server will allocate a buffer according to the value of *size* for receiving the data coming from the JCL client later. After sending the data to the JCL server, the JCL client will send the value of *ptr* to the JCL server. Next, the JCL server will copy the data from the buffer to the device memory from the *prt+offset* address to the *prt_offset+size-1* address, and will return clresult_code to the JCL client. Finally, the JCL client will transfer the received clresult_code to the calling thread.

cl_intclSetEventCallback (cl_event event, cl_int command_exec_callback_type, void (CL_CALLBACK *pfn_event_notify(cl_event event, cl_int event_command_exec_status, void * user_data), void *user_data).

This function is used for program threads to register a callback function for calling a given OpenCL function. When the execution status of the called OpenCL function matches the registered event, the registered callback function will be invoked to handle the event. Accordingly, the event argument is used to specify the registered event. *Command_exec_callback_type* is the condition of triggering the callback function. *Pfn_event_notify* is a function pointer, which points to the callback function. *User_data* is the argument of the callback function. The process of this function is shown in Fig. 8.

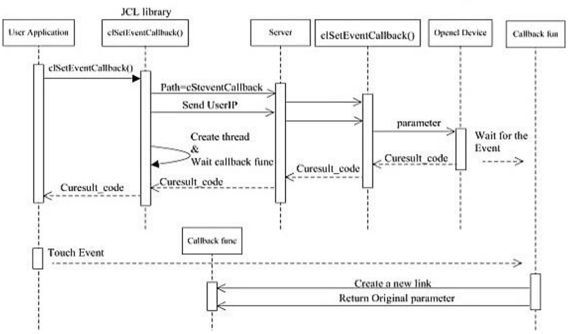


Fig. 8. Process of clSetEventCallback()

The JCL client will allocate a communication port for the callback function, and will create a thread to listen at the port in order for waiting a trigger signal from the JCL server. Next, the JCL client will sent the argument values and the port number to the JCL servers. The JCL server will create a dummy callback function for the specified event, and then will call clSetEventCallBack() with the received argument values, and the dummy callback function. When the execution status of a called OpenCL function matches the registered event, the dummy callback function will be invoked to send a signal to the JCL client for handling the event. The thread listening to the communication port created for the event will wake up to call the real callback function to handle the event.

4.2 Load Balancing Functions

void lbInitial(int thread_num).

This function is used for the initialization of dynamic load balance. The argument of thread_num is the number of program threads. When this function is called, the JCL client will allocate an time array, called *ttime* for storing the execution time of threads in each iteration, and will create a barrier for thread synchronization in lbGetPartSizeOffset().

void lbProfileTimeStart().

When a thread calls this function, the JCL client will record the current time into ttime[*id*] where *id* is the identifier of the thread.

voidlbProfiletimeEnd().

When a thread calls this function, the JCL client will record the current time and store the time interval between the current time and ttime[*id*] into ttime[*id*] for later use in lbGetPartSizeOffset().

void lbGetPartSizeOffset(size_t *getsize, size_t *get-offset, size_t problemsize, size_t rowsize).

The arguments of *getsize* and *getoffset* are the memory addresses of the variables used for storing the amount of data and the position offset of the first one of the data assigned to the calling thread. The arguments of *problemsize* and *rowsize* are the total amount of data pending for computation, and the number of data in a partition unit, respectively. When a thread calls this function, the JCL client will process this function call as follows.

First, the JCL client calculates the power factor of the calling thread as follows.

$$power_{ik} = \frac{datasize_{k}}{executiont\ ime_{k}} \tag{1}$$

Assume the identifier of the calling thread is *i*. In the above equation, the parameters of datasize$_k$ and executiontime$_k$ denote the amount of data computed by the calling thread, and the execution time of the thread at the *k*th iteration. Second, the JCL client calculates the average load factor of program threads, and then estimates the relative power factor of the calling thread at the *k*th iteration as follows.

$$average\_power_k = \sum_{i=1}^{N} power_{ik} / N, N = thread\ \ NO. \tag{2}$$

$$relative\_power_{ik} = power_{ik} / average\_power_k \, . \tag{3}$$

Finally, the JCL client calculates the amount of data assigned to the calling thread, and the offset of the first assigned data as follows.

$$datasize_{i(k+1)} = \frac{problemsize}{N} * relative\_power_{ik} \tag{4}$$

$$offset_{i(k+1)} = \sum_{i=0}^{i-1} datasize_{i(k+1)} \tag{5}$$

void lbGetNDkernelCoreSize(size_t*GlobalWorkSizesize_t *LocalWorkSize).

This function is used to get the maximal number of cores available in a working group. The value of the GlobalWorkSize argument usually is equal to the amount of data assigned to the calling thread. Since the global work size must be evenly dividable by the local work size according to the specification of OpenCL, the JCL client sets the LocalWorkSize as the maximal integer which is not bigger than CL_DEVICE_MAX_WORK_GROUP_SIZE, and can evenly divide GlobalWorkSize. The two arguments of GlobalWorkSize and LocalWorkSize are used for calling EnqueueNDkernel() later.

5 Performance Evaluation

We have implemented two OpenCL applications including Nbody and Matrix Multiplication for evaluating the performance of JCL. We compiled the test programs with linking the runtime library of JCL, and then executed them with a cluster of computers connected with 1Gbps Ethernet, as shown in Table 2. In this performance evaluation, we did three experiments for evaluating the performance of JCL. The first is to estimate the overhead of JCL. The second is to measure the speedup of JCL. The third is to evaluate the effectiveness of load balancing of JCL.

Table 2. Experimental environment

	Device type	Device & Memory
JCL servers	CPU	Intel Xeon 5500, 16GB RAM
	GPU	NVidia GT9800, 1GB VRAM
	GPU	NVidia 550Ti, 1GB VRAM
	GPU	ATI HD5570, 1GB VRAM
JCL client	CPU	Intel Q6600, 2GB RAM

5.1 Overhead of JCL

This experiment is aimed at comparing the cost of JCL runtime time library with that of the original OpenCL runtime library for several OpenCL functions, and estimating the communication overhead of exploiting a remote JCL server. The test program used in this experiment was the Nbody application or 100 iterations. The JCL server was executed on the host with the Xeon 5500 processor. The experimental result is shown in Fig. 9.

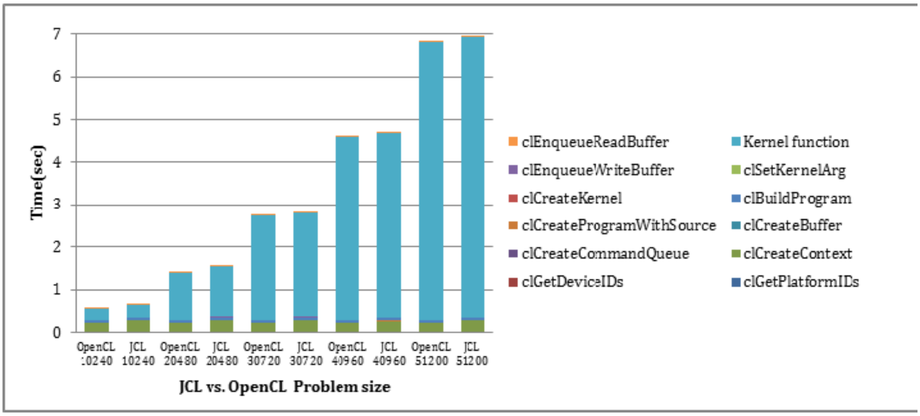


Fig. 9. Breakdown of the execution time of Nbody

It can be found that the overhead of the JCL library is low compared to the original OpenCL runtime library. The cost difference of the two libraries happens in the functions of clEnqueueWriteBuffer() and clEnqueueReadBuffer(). The main reason for this cost difference is that the JCL client exchange data with the JCL server through network when it executes the two functions. Although this communication cost increases as well as the amount of data, however it is necessary and unavoidable while it is reducible by higher speed networks. Fortunately, most of execution time is spent on the function of clEnqueueNDRangeKernel(), i.e., data computation. Consequently, the total execution time of the test program has no obvious difference although the program is linked with two different libraries.

5.2 Speedup

In this experiment, we intend to evaluate the effectiveness of JCL on the performance of the test programs, which are executed by multiple GPUs in a cluster. In order to control performance factors, we ran the test applications with using four NVidia 550Ti GPUs, and then estimated the speedup of the test programs by using the one-node case to be baseline. The speedups of the two test programs are shown in Fig. 10 and Figure 11, respectively. The experimental result shows that the speedups of the two test programs are effectively increased when the number of used GPUs increases in most of cases. However, the network speed is much slower than the computation speed of GPUs. Consequently, the speedup is not linearly increased with the number of used GPUs especially when the problem size is small. That is why the MM application with 1024x1024 float-point numbers cannot obtain a speedup because the communication cost of distributing data over GPUs cannot be compensated by the computation cost saved by parallel computation with the GPUs. Fortunately, higher speed networks can improve this problem. This problem will disappear and the speedup of the test programs will become more obvious when the problem size becomes large enough as shown in the experimental result.

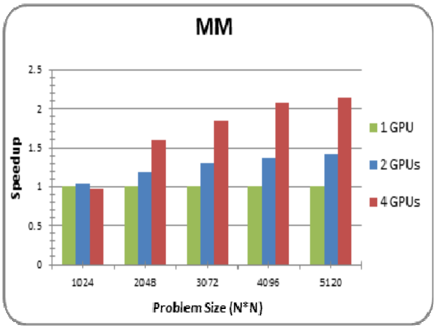


Fig. 10. Speedup of MM

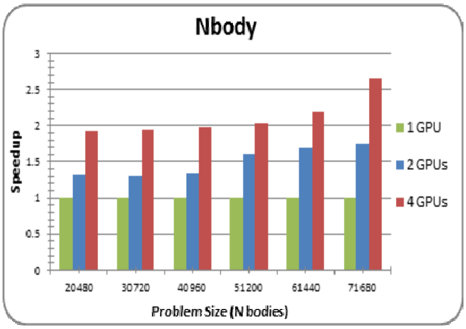


Fig. 11. Speedup of Nbody

5.3 The Effectiveness of Load Balance

In this experiment, we used four different-speed processors including Intel Xeon 5500, NVidia 9800, 550Ti and ATI 5570HD for executing the kernel functions of the test programs. We ran the test applications respectively with and without the load balancing functions of JCL, and evaluated the performance of the test applications in the two different cases. The experimental results are depicted in Fig. 12 and Fig. 13. It

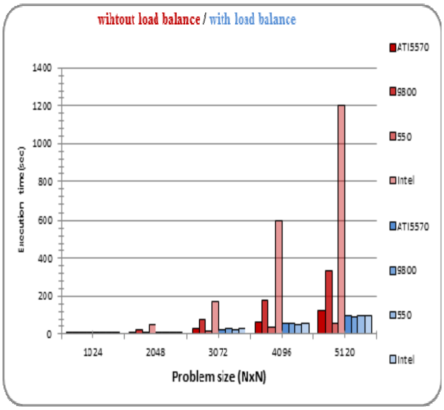


Fig. 12. Performance of MM with and without load balancing

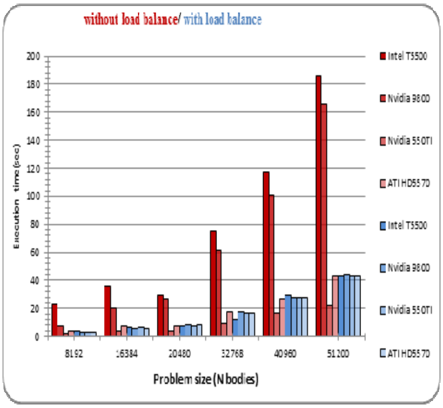


Fig. 13 Performance of Nbody with and without load balancing

can be found that when the test programs don't use the load balancing functions of JCL, the execution times of NVidia 550Ti and ATI 5570HD are much less those of Intel 5550 and NVidia 9800 since the former two is more powerful in data computation than the latter two. By contrast, all the execution times of the four processors become very average when the test programs use the load balancing functions of JCL. Consequently, the performance of the test programs is successfully improved due to load balance. As previously discussed, the load-balancing function of JCL is really effective for increasing the performance of user programs.

6 Related Work

Some programming toolkits like JCL had been proposed in past studies. For example, rCUDA [11] is aimed at resource sharing for minimizing the number of high-end GPU-compatible devices in clouds because of considering resource utilization and energy consumption. This toolkit is implemented also based on TCP/IP socket. With the rCUDA client, the CUDA functions issued by user programs can be redirected to remote CUDA-compatible GPU for execution. However, this toolkit supports only NVidia GPU, and does not support parallel computing. vCUDA [12] is a toolkit dedicated to enabling virtual machines to support user programs for accessing physical CUDA-compatible devices through the software boundary. This toolkit is implemented based on XML-RPC. As a result, the communication cost of vCUDA is higher than that of rCUDA. Hybrid OpenCL [13] is aimed at providing an abstraction of different implementations of OpenCL over network. With the support of this toolkit, user programs can connect multiple runtime systems of OpenCL over network. The goal of Hybrid OpenCL is as same as that of JCL while it does not support load balance and callback functions. Virtual OpenCL (VCL) [14] is a cluster platform, which allows OpenCL programs to make use of multiple GPU in a cluster. The framework and implementation of VCL is similar to JCL while it currently does not support load balance and provides only GPUs but CPUs for data computation. Compared to rCUDA and vCUDA, JCL is focused on parallel computing but not resource sharing or virtualization. Moreover, JCL supports not only NVidia GPU but also AMD GPU and x86 CPUs proposed by any vendors. Different to Hybrid OpenCL and VCL, JCL allows users to simultaneously exploit both of CPUs and GPUs in a cluster for resolving the same problem while they are not aware of resource location, and data communication between the local host and remote servers. In addition, JCL supports load balance for enhancing the performance of user programs.

7 Conclusion and Future Work

We have successfully developed an OpenCL programming called JCL for heterogeneous cluster computing in this paper. With the support of JCL, users can write programs by means of the same programming interface, i.e., OpenCL, and make use of multiple heterogeneous processors including GPUs and CPUs distributed in computer networks for the execution of their OpenCL programs while they are not aware of resource location, and data communication between the local host and the remote servers. Consequently, JCL successfully reduces the programming complexity of heterogeneous cluster computing. Our experimental result has shown that the overhead of the proposed toolkit is negligible. User programs indeed effectively exploit the computational power of processors with using the load balancing functions of JCL, and thereby enhance their performance.

In addition to CPU and GPU, the processors of embedded systems such as ARM and FPGA can support OpenCL. Therefore, we will extend the framework of JCL to aggregate FPGAs and ARMs with CPUs and GPUs together for minimizing the time of resolving data-intensive or massive-computation problems. In addition, we will build a cloud program development environment based on JCL for users to make use of heterogeneous resources in clouds for parallel computing [15].

Acknowledgement. We thank the grant support of National Science Council in Republic of China under the project numbered as NSC-101-2815-C-151-026-E.

References

1. Owens, J.D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A.E., Purcell, T.J.: A Survey of General-Purpose Computation on Graphics Hardware. Computer Graphics Forum, 80–113 (2007)
2. NVIDIA, NVIDIA CUDA Programming Guide (2011), `http://developer.download.nvidia.com/compute/DevZone/docs/html/C/doc/CUDA_C_Programming_Guide.pdf`
3. Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., Houston, M., Hanrahan, P.: Brook for GPUs: Stream Computing on Graphics Hardware. SIGGRAPH (2004)
4. Quinn, M.J.: Parallel Programming in C with MPI and OpenMP. McGraw-Hill (2004)
5. Buttlar, D., Farrell, J., Nichols, B.: PThreads Program Programing. O'Reilly Media (1996)
6. Khronos OpenCL Working Group, The OpenCL Specification (2011), `http://www.khronos.org/registry/cl/specs/opencl-1.0.29.pdf`
7. Kurzak, J., Buttari, A.: Introduction to Programming High Performance Applications on the CELL Broadband Engine. In: 15th IEEE Symposium on High-Performance Interconnects, p. 11 (2007)
8. Altera Corporation, Implementing FPGA Design with the OpenCL Standard (2011), `http://www.altera.com/literature/wp/wp-01173-opencl.pdf`
9. The MPI Forum, MPI: A Message Passing Interface. In: Proceedings of Super Computing, pp. 878–883 (1993)
10. Xue, M., Zhu, C.: The Socket Programming and Software Design for Communication Based on Client/Server. In: Proceedings of the 2009 Pacific-Asia Conference on Circuits, Communications and Systems, pp. 775–777 (2009)
11. Duato, J., Peña, A.J., Silla, F., Mayo, R., Quintana-Ort́, E.S.: Modeling the CUDA remote Virtualization Behaviors in High Performance Networks. In: First Workshop on Language, Compiler, and Architecture Support for GPGPU (2010)
12. Shi, L., Chen, H., Sun, J.: vCUDA: GPU accelerated high performance computing in virtual machines. In: International Parallel and Distributed Processing Symposium, pp. 1–11 (2009)

13. Aoki, R., Oikawa, S., Tsuchiyama, R., Nakamura, T.: Hybrid OpenCL: Connecting Different OpenCL Implementations over Network. In: 10th IEEE International Conference on Computer and Information Technology, pp. 2729–2735 (2010)
14. Barak, A., Shiloh, A.: The Virtual OpenCL (VCL) Cluster Platform. In: Proc. Intel European Research & Innovation Conf., Leixlip, p. 196 (October 2011)
15. Vecchiola, C., Pandey, S., Buyya, R.: High-Performance Cloud Computing: A View of Scientific Applications. In: 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN), pp. 4–16 (2009)

Network-Aware Multiway Join for MapReduce

Kenn Slagter[1], Ching-Hsien Hsu[2,*], Yeh-Ching Chung[1], and Jong Hyuk Park[3]

[1] Department of Computer Science, National Tsing Hua University
Hsinchu, Taiwan, R.O.C.
kennslagter@sslab.cs.nthu.edu.tw, ychung@cs.nthu.edu.tw
[2] Department of Computer Science, Chung Hua University
Hsinchu, Taiwan, R.O.C.
chh@chu.edu.tw
[3] Department of Computer Science and Engineering
Seoul National University of Science and Technology
Seoul, Korea
jhpark1@seoultech.ac.kr

Abstract. MapReduce is an effective tool for processing large amounts of data in parallel using a cluster of processors or computers. One common data processing task is the join operation, which combines two or more datasets based on values common to each. In this paper, we present a network aware multi-way join for MapReduce(NAMM) that improves performance by redistributing the workload amongst reducers. NAMM achieves this by redistributing tuples directly between reducers with an intelligent network aware algorithm. We show that our presented technique has significant potential to minimize the time required to join multiple datasets.

Keywords: MapReduce, Hadoop, Multiway Join, Workload Redistribution.

1 Introduction

MapReduce [1] is a flexible programming model proposed by Google for processing and creating data sets over a cluster of computers. The MapReduce model hides extraneous details inherent in distributed programming such as parallelization, fault tolerance, data distribution and load balancing within a library. This simplifies the process of writing distributed programs, which is an advantage MapReduce has over other distributed programming models such as MPI that requires the programmer to explicitly handle the data flow [2].

Programmers who use the MapReduce library need to write two functions a map function and a reduce function. The purpose of the map function is to take the input key/value pairs from an input source, process it and then outputs a set of intermediate key/value pairs. The intermediate key/value pairs it generates is then fed into a reduce function which processes the key/value pairs and then generates as output its own set of key/value pairs.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 73–80, 2013.

Parallelism is achieved by running multiple map and reduce functions on multiple processors or machines. The intermediate key/value pairs produced by each of the map functions are partitioned so that intermediate key/value pairs that share the same key are all sent to the same reduce function to be processed.

Since its conception by Google the MapReduce model has inspired others to adopt its paradigm. One of the most well known adopters was Yahoo, who developed an open source implementation known as Hadoop [3] which operates under the Apache license. Hadoop is a Java-based implementation and by default runs on its own distributed file system (HDFS). Because Hadoop is open source, well documented and easy to use, the tool has gained prominence in the distributed programming community. For this reason, we use Hadoop as our reference platform for MapReduce in this paper.

The MapReduce model is effective at processing large amounts of data or datasets. A dataset is essentially a set of tuples stored in a file. In this paper, we look at one of the most common data processing operations called a join, which combines two or more datasets together based on some common value. There are many possible ways to implement a join. The efficiency of a join implementation depends on how many data sets there are and how large the data sets are. A MapReduce join can be implemented as a map-side join or a reduce-side join and multiple datasets may be handled either as successive two-way joins known as a cascade of joins or with a multiway join [4].

Multiway joins have certain advantages and disadvantages over cascade joins. First, it avoids considerable overhead since it does not to setup multiple jobs. Second, it can save space on the network since it does not need to store intermediate results. However, there are some drawbacks to multiway joins. When a multiway join is performed it needs to buffer tuples. This can lead to memory problems, especially if the data is skewed. Therefore, the number of datasets and size of datasets are limited by the memory resources available.

The main idea of NAMM is to improve processing time of a multiway join by redistributing the workload between reducers. The main contributions of our work are as follows. First, we present a model to redistribute tuples amongst reducers on the MapReduce framework for a multiway join. Second, we show how the NAMM redistribution algorithm can reduce job response times for a multiway join by considering network distance and reducer workload. Third, we compare our method to an alternative method, which does not take into account these factors.

The rest of this paper is organized as follows. Section 2 explains our research model and presents the proposed techniques on multiway joins and tuple redistribution. In Section 3, the simulation results and performance analysis are given to weigh the pros and cons of the proposed method. In Section 4, we discuss related work. Finally, the conclusion and future work are presented in Section 5.

2 Research Model

2.1 Network Model

The research model for this study is presented in Figure 1, which shows a network environment consisting of switches, racks and nodes. The two-level tree topology shown in Fig 1(a) is a common network layout used by Hadoop. Each rack contains a set of servers (nodes) all interlinked by a switch. The racks themselves then uplink to a core switch or router. It is important to note that the total bandwidth between nodes on the same rack is much greater than that between nodes on different racks. The nodes are used to run map or reduce tasks as shown in Fig 1(b). In this paper, map tasks and reduce tasks are also referred to as mappers and reducers respectively.

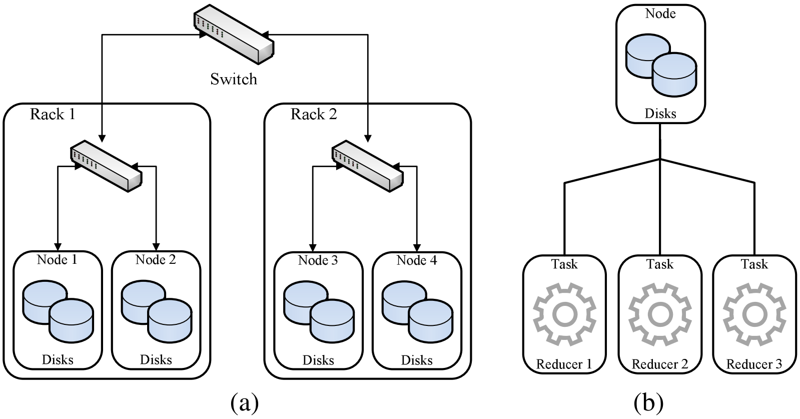


Fig. 1. Research model in this paper (a) a tree network consisting of racks and nodes (b) A node running a set of reduce tasks

2.2 Join Algorithms

Join algorithms have been studied extensively over the years, with many different variants existing for each type of algorithm. Many join algorithms in academia predate the invention of MapReduce, due to their ubiquitous use throughout the database community. The multiway join algorithm presented in this paper is a hybrid join based on pre-existing MapReduce join model for reduce-side joins and a hash-join which handles joins locally on the reducer.

2.2.1 Reduce-Side Join

Reduce-side joins are based on the MapReduce programming model which is composed of a map phase and a reduce phase. In the map phase, the datasets are read by a map function by each map task, one tuple at a time. The purpose of the map function is only to pre-process the tuples and sort them by the join key. Before tuples are partitioned based on their join key, they are tagged so that the reduce function can know which table the tuple originated from. The tuples are then sent to their respective reducer where they are to be joined. Each tuple is then joined by the reducer based on which table they came from.

2.2.2 Hash Join

A hash join is a traditional algorithm used by databases for joining two datasets together. A hash join consists of two distinct phases a ‘build’ phase and a ‘probe’ phase. In the build phase the smallest dataset is inserted into a in-memory hash table. In the probe phase the largest dataset is scanned and joined with the appropriate tuple(s) stored in the hash table.

2.2.3 NAMM Multiway Join

In this section we present our proposed multiway join. The purpose of a multiway join is to join multiple datasets together. Our proposed join improves performance of the multiway join by redistributing the workload amongst reducers. Unlike other schemes that redistribute the workload using a distributed queue [5], our methodology redistributes the workload directly between reducers with help of a mediator service, as shown in fig 2.

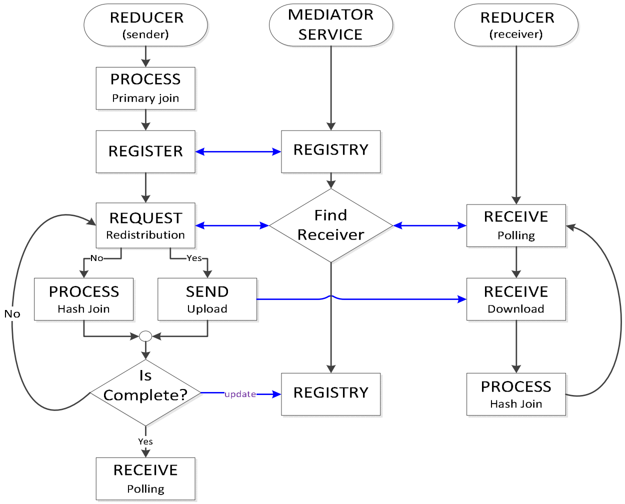


Fig. 2. Multiway Join Tuple Redistribution with NAMM

The reduce-side join and the hash join algorithms are both examples of a two-way join. Two-way joins are joins that involve only two tables. Multiway joins are joins involving more than two tables. The multiway join presented in this paper uses a reduce-side join to join the two largest datasets and a hash join on the reducer side to join that result with several smaller datasets. The two largest datasets are partitioned and sent to the various reducers using the typical reduce-side join mechanism.

Unlike the typical multiway join mechanism [6], the smaller datasets are sent to all the reducers. This can be done by duplicating tuples in the mapper phase so that a copy is sent to each reducer or in Hadoop by using its distributed cache mechanism. Therefore, this method is appropriate in situations where the aggregate size of the smaller datasets can fit in the memory of each node used to execute a reduce task.

Once the mappers have sent the reducers all the tuples, the reducers are able to process the tuples. The two largest datasets are then joined with a traditional reduce-side join. After completing the initial primary join, the reducer registers with the mediator. The mediator is provided details from the reducer about the number of tuples it has, and details on which node it resides.

From the list of reducers already registered, the reducer will attempt to send a batch of tuples to another reducer. For the sake of clarity in this paper, we define two types of reducers, senders and receivers. Senders are those reducers that still have tuples to join and receivers are idle reducers that already processed their workload. The sender makes a request to the mediator to find a receiver on the network. It then prepares a batch of tuples to send from the initial primary join. The size of the batch depends on memory size of reducers and number of tuples being processed. The mediator then finds the reducer closest on the network in terms of network distance. If the mediator is unable to find a suitable receiver, the sender hash joins the batch of tuples it prepared instead. Network distance in this study is based on the same concept used in Hadoop [3] and is calculated based on the number of switches that exist between two nodes. There are three different scenarios that may occur in a data center. Given a node *n1* on rack *r1* in data center *d1*. This can be represented as */d1/r1/n1*. Using this notation, here are the distances for the three scenarios:

- distance(/d1/r1/n1, /d1/r1/n1) = 0 (processes on the same node)
- distance(/d1/r1/n1, /d1/r1/n2) = 2 (different nodes on the same rack)
- distance(/d1/r1/n1, /d1/r2/n3) = 4 (nodes on different racks in the same data center)

Once a sender has sent all its tuples to a receiver it prepares another batch of tuples. If the sender has less tuples to process than other senders, it processse these tuples itself. Otherwise, the sender makes another redistribution request to the mediator.

3 Evaluation

3.1 Experiment Configuration

To evaluate the performance of the proposed technique, we implemented the NAMM multiway join method and tested its performance on a simulated MapReduce environment. We then evaluated its performance against a network unaware method. Overall, the network unaware method is the same as the NAMM method but sends its tuples to the first available receiver. We then tested both algorithms using various workloads using 200, 600 and 1000 million tuples. These workloads represent low loading (*LL*), medium loading (*ML*) and high loading (*HL*), respectively. For the sake of clarity, the experimental parameters used in this study are presented in Table 1.

Table 1. Experimental Parameters

Parameter	Definition	Description
LL	Low Loading	200M tuples
ML	Medium Loading	600M tuples
HL	High Loading	1000M tuples
n	*n* Loading	*n*M tuples

In order to test our proposed algorithm the MapReduce environment was setup to emulate a small cluster of computers. The cluster for our test environment consisted of two racks, with two nodes per rack and three reducers per node as shown in Fig 1. In total, 18 cases were used to test the performance of NAMM. These test cases are presented in Table 2.

Table 2. Test Cases

Rack		Rack 1						Rack 2					
Node		Node 1			Node 2			Node 3			Node 4		
Reducer		1	2	3	4	5	6	7	8	9	10	11	12
Test Case	1	ML	ML	ML	ML	ML	ML	ML	ML	ML	ML	ML	ML
	2	LL	HL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL
	3	ML	HL	ML	ML	ML	ML	ML	ML	ML	ML	ML	ML
	4	HL	LL	HL	LL	LL	LL	LL	LL	LL	LL	LL	LL
	5	LL	HL	LL	LL	HL	LL	LL	LL	LL	LL	LL	LL
	6	LL	HL	LL	LL	LL	LL	LL	LL	HL	LL	LL	LL
	7	HL	ML	HL	ML	ML	ML	ML	ML	ML	ML	ML	ML
	8	ML	HL	ML	ML	ML	ML	ML	ML	ML	ML	ML	ML
	9	ML	HL	ML	ML	ML	ML	ML	ML	HL	ML	ML	ML
	10	LL	LL	LL	HL	HL	HL	HL	HL	HL	HL	HL	HL
	11	LL	LL	LL	LL	LL	LL	HL	HL	HL	HL	HL	HL
	12	LL	LL	LL	LL	LL	LL	LL	LL	LL	HL	HL	HL
	13	ML	ML	ML	HL	HL	HL	HL	HL	HL	HL	HL	HL
	14	ML	ML	ML	ML	ML	ML	HL	HL	HL	HL	HL	HL
	15	ML	ML	ML	ML	ML	ML	ML	ML	ML	HL	HL	HL
	16	0	100	200	300	400	500	600	700	800	900	1000	1100
	17	1100	1000	900	800	700	600	500	400	300	200	100	0
	18	LL	HL	LL	HL	LL	HL	LL	HL	LL	HL	LL	HL

3.2 Experiment Results

In this paper, we compare five different redistribution methods using the test cases from Table 2. The redistribution methods considered in this study use a combination of the redistribution features shown in Table 3.

Table 3. Redistribution Features

Undistributed	no tuple redistribution
Network Unaware	network distance between reducers not considered
Network Aware	network distance between reducers considered
SingleSend	reducers redistribute tuples to only one receiver each hash join
Multisend	reducer with heaviest load attempts to redistribute tuples to multiple receivers before executing a hash join

The workload of a reducer is the total number of reduce-side joins and hash joins performed by that reducer. A MapReduce job does not complete until all mappers and reducers have completed their tasks. Therefore, the redistribution method that has the best performance is the redistribution method whose worst-case reducer has the least number of joins. We consider the worst-case reducer to be whichever reducer performed the most number of joins.

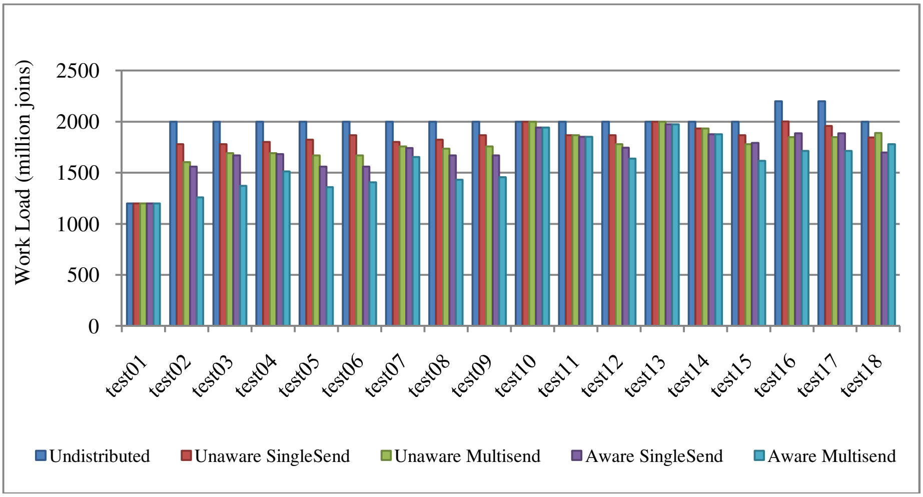


Fig. 3. The results for workload distribution on the worst-case reducer

In Fig 3, we compare the performance of the worst-case reducer for each redistribution method. In majority of test cases NAMM using network aware and multisend methods is able to significantly reduce the workload on each reducer, compared to other redistribution techniques covered in this study. The efficiency of NAMM is more apparent in test cases when the workload among reducers is less evenly distributed. In test case 1, all the reducers had exactly the same workload and consequently no redistribution takes place. In test case 13, and test case 14 the workload was distributed amongst reducer in such a fashion that there was no opportunity for multiple batch sends to occur. Test case 18 shows that in some instances multiple tuple redistributions from the same reducer can interfere with the overall time taken to complete a job. Overall, NAMM technique was able to improve workload redistribution by up to 12% when using the single send technique, and up to 21% improvement when compared to other network unaware methods assessed in this study.

4 Related Work

Joins have been studied in detail by many sources. Investigations and descriptions of the various joins have been collated by other works [6] [9]. One such work that discusses handling joins using a mediator over a network is presented by [7]. This system employs a balanced network utilization metric to optimize the use of all network paths in a global-scale database federation. It uses a metric that allows algorithms to exploit excess capacity in the network, while avoiding narrow, long-haul paths. A work similar to our paper is presented by [5] uses a distributed queue [8] rather than using peerwise network connections to perform multiway joins and does not take into account network distance when redistributing tuples.

5 Conclusion and Future Work

In this paper, a network aware multiway MapReduce join (NAMM) technique is presented for redistributing workload for MapReduce. The simulation results show that NAMM can significantly improve tuple redistribution with Mutiway Joins for MapReduce applications. NAMM's has shown up to 21% improvement over other network unaware methods.

NAMM is designed for users who intend to use to perform a multiway join between two large datasets and several smaller datasets with MapReduce. In future work it would be desirable to explore how this system could be extended to handle more than two large datasets and how to improve its performance on different network topologies or hardware configurations. We leave these tasks for future work.

References

[1] Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Commun. ACM 51, 107–113 (2008)

[2] Hoefler, T., Lumsdaine, A., Dongarra, J.: Towards Efficient MapReduce Using MPI. In: Ropo, M., Westerholm, J., Dongarra, J. (eds.) PVM/MPI. LNCS, vol. 5759, pp. 240–249. Springer, Heidelberg (2009)

[3] White, T.: Hadoop the definitive guide, 2nd edn. O'Reilly, Sebastopol (2010)

[4] Afrati, F.N., Ullman, J.D.: Optimizing Multiway Joins in a Map-Reduce Environment. IEEE Transactions on Knowledge and Data Engineering 23, 1282–1298 (2011)

[5] Lynden, S., Tanimura, Y., Kojima, I., Matono, A.: Dynamic Data Redistribution for MapReduce Joins. In: 2011 IEEE Third International Conference on Cloud Computing Technology and Science (CloudCom), pp. 717–723 (2011)

[6] Chandar, J.: Join Algorithms using Map/Reduce, Master of Science, School of Informatics, University of Edinburgh (2010)

[7] Wang, X., Burns, R., Terzis, A., Deshpande, A.: Network-aware join processing in global-scale database federations. In: IEEE 24th International Conference on Data Engineering, ICDE 2008, pp. 586–595 (2008)

[8] Hunt, P., Konar, M., Junqueira, F.P., Reed, B.: ZooKeeper: Wait-free coordination for Internet-scale systems. In: USENIX ATC (2010)

[9] Palla, K.: A Comparative Analysis of Join Algorithms Using the Hadoop Map/Reduce Framework, Master of Science, School of Informatics, University of Edinburgh (2009)

Automatic Resource Scaling for Web Applications in the Cloud

Ching-Chi Lin[1,2], Jan-Jan Wu[1], Pangfeng Liu[2], Jeng-An Lin[2], and Li-Chung Song[2]

[1] Institute of Information Science Research Center for Information Technology Innovation
Academia Sinica, Taipei, Taiwan
{deathsimon,wuj}@iis.sinica.edu.tw
[2] Department of Computer Science and Information Engineering Graduate
Institute of Networking and Multimedia National Taiwan University, Taipei, Taiwan
{pangfeng,r99944038,r00922089}@csie.ntu.edu.tw

Abstract. Web applications play a major role in various enterprise and cloud services. With the popularity of social networks and with the speed at which information can be disseminate around the globe, online systems need to face ever-growing, unpredictable peak load events.

Auto-scaling technique provides on-demand resources according to workload in cloud computing system. However, most of the existing solutions are subject to some of the following constraints: (1) replying on user provided scaling metrics and threshold values, (2) employing the simple Majority Vote scaling algorithm, which is ineffective for scaling Web applications, and (3) lack of capability for predicting workload changes. In this work, we propose an effective auto-scaling strategy, called *Work-load Based* scaling algorithm, for Web applications. Our proposed scaling strategy is not subject to the aforementioned constraints, and can respond to fluctuated workload and sudden workload change in a short time without relying on over-provisioning of resources. We also propose a new method for analyzing the trend of workload changes. This trend analysis method provides useful information to the scaling algorithm to avoid unnecessary scaling actions, which in turn shortens the response time of requests. The experiment results show that the hybrid *Workload Based* and *trend analysis* method keeps response time within 2 seconds even when facing sudden workload change.

Keywords: Cloud Computing, Auto-Scaling, Web Applications, Resource Provisioning, Trend Analysis.

1 Introduction

Web applications play a major role in various enterprise and cloud services. Many Web applications, such as eBanking, eCommerce and online gaming, face fluctuating loads. Some of the loads are predictable, such as the workload around a holiday for eShopping services. However, with the popularity of social networks and with the speed at which information can disseminate around the globe, online systems need to face ever-growing, unpredictable peak load events.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 81–90, 2013.

Auto-scaling is a solution that not only maintains application service quality but also reduces wasted resources while facing fluctuating loads. The basic idea of *auto-scaling* is to estimate the load for short window of time, and then be able to up-scale or down-scale the resources when there is a need for it. Many cloud services, such as Amazon EC2 [1] and Google App Engine [2], have proposed auto-scaling service. Other software such as Scalr [3] and RightScale [4] provide auto-scaling mechanism that can apply to cloud environments. However, most of the existing solutions are subject to some of the following constraints: (1) replying on user-provided scaling metrics and threshold values, (2) employing the simple *Majority Vote* scaling algorithm, which we will show in this paper to be ineffective for scaling Web applications, and (3) lack of capability for predicting workload change, and thus may result in unnecessary scaling actions.

We develop an auto-scaling system, *WebScale*, which is not subject to the aforementioned constraints. *WebScale* monitors the behavior of the applications and the system, and based on the collected metrics, decides in real time whether the number of VMs for an application needs to be increased or decreased. Because of page limit, in this paper, we focus on the new algorithms we propose for scaling resources for Web applications.

The main contributions of this work are as follows. (1) We show that the incoming requests from the clients (instead of standard metrics) and the HTTP response time can characterize the workload/performance behavior of Web applications more accurately. Based on this observation, we devise an effective scaling algorithm called *Workload-Based* algorithm for Web applications. (2) We propose an algorithm to analyze the trend of workload change in a Web application. This trend analysis algorithm can significantly reduce the number of peaks (longer than 2 seconds) in response time caused by workload fluctuating. (3) Our experiment results with workload generated by *httperf* [5] demonstrate that our scaling strategy can keep the average response time of Web applications within 2 seconds even when facing sudden load change.

The rest of the paper is organized as follows. Section 2 presents the Workload-Based scaling algorithm and the Trend Analysis algorithm. Section 3 presents and analyzes experiment results. Section 4 describes related work. Finally, Section 5 gives some concluding remarks.

2 Scaling Algorithms

In this section, we first give a brief overview of the widely used scaling algorithm, *Majority Vote*. We then present our *Workload-Based* scaling algorithm. Finally, we present our *Trend Analysis* technique, which co-works with the scaling algorithm to achieve better performance.

2.1 Overview of Majority Vote

Majority Vote selects the choice with the most properties among all choices. There are three choices, **scale in**, **scale out**, and **no scale**, for a VM when making decision.

Each VM makes their choice according to their current loading, and a final scaling decision will be made by majority vote.

Each VM makes its choice according to a chosen metric, such as CPU load or memory usage. For each chosen metric, there are two thresholds, *threshold_H* and *threshold_L,* which represent the high threshold and low threshold respectively. If the chosen metric of a VM is greater than *threshold_H*, the VM will choose **scale out**. On the other hand, if the chosen metric is smaller than *threshold_L*, the VM will choose **scale in**. Otherwise, the choice will be **no scale**.

2.2 Workload-Based Algorithm

The *Workload-Based* Algorithm determines the number of running VMs needed for the business-logic tier and the number of database servers needed for the data-access tier, based on the incoming workload. The latter is the number of requests per second for the business-logic tier, and the number of SQL queries per second for the data-access tier. In real world web applications, requests sent by clients will be received by the front-end load balancer, and then distributed to the back-end VMs. Since each VM has a capacity limitation, e.g. maximum number of requests per second a running VM can handle simultaneously, we can calculate the number of VMs needed to process the current workload, and make scaling decisions based on the number of current running VMs. The same applies to the number of database servers.

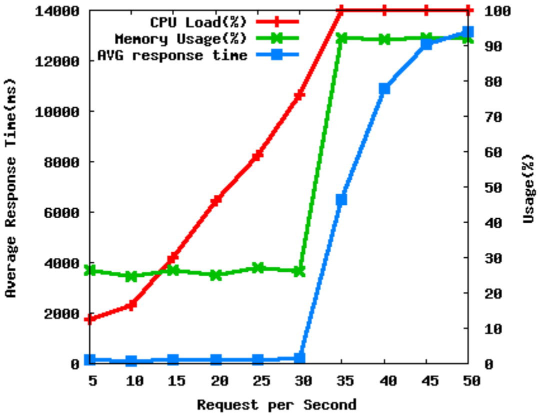


Fig. 1. Relationship between requests per second, average response time, CPU load, and memory usage

The reason that we choose requests per second instead of standard metrics to characterize workload of Web applications is that standard metrics fail to indicate "how busy" a VM is. Figure 1 shows the relationship between requests per second, two of the standard metrics, and the average response time. The average response time dramatically increases if the workload is over 30 requests per second. However, the standard metrics, CPU load and memory usage, in this example, remains constant when the number of requests per second is larger than 35. The standard metrics fail to characterize the workload, thus cannot provide accurate information to decide the number of VMs needed. On the other hand, using requests per second as the metric can avoid this problem. With this metric, we can calculate the number of new running VMs needed to make the average response time decrease to an acceptable range.

The assumption that a VM has a capacity limitation; that is, it can only process a fixed number of requests per second simultaneously, is reasonable because of the need to maintain QoS. It has been shown in many previous works [6, 7, 8] that the types of requests to a Web application are bounded by a small constant, and that the percentage of each kind of requests to a Web application can be estimated. With such information, we can determine the capacity limitation of a VM.

We propose a variation of *parameter selection* scheme [9] to determine the capacity limitation of a VM. Our parameter selection scheme collects the system and performance information of a VM under different workload. This information is stored in a list, sorted by workload in ascending order. We can choose the maximum workload value from the list such that the corresponding performance satisfies the QoS requirement. This workload value is used as the capacity limitation of the VM. The following is an example on how to decide the capacity limitation of a VM or database server.

Table 1. Average response time(ms) under different workload combination

req/s	100% Q1	75% Q1 + 25% Q2	50% Q1 + 50% Q2	100% Q2
15	140.6	237.6	326.7	503.5
20	159.0	251.4	367.2	18714.5
25	157.7	6964.0	14082.4	22805.6
30	8222.3	13860.2	17525.5	22552.0

We use Table 1 to illustrate how we determine the capacity limitation of a VM using the parameter selection scheme. We use MediaWiki [10] as the web application. We assume that there are two kinds of requests, Q1 and Q2. Q1 requests a static Wiki page and Q2 requests a dynamic page which lists the links of top 100 articles in the database, sorted in descending order. Q2 has longer processing time than Q1. This table shows that the average response time dramatically increases if the request per second grows beyond a certain value under different workload combinations. Furthermore, the capacity limitation decreases when the percentage of more expensive requests (i.e., Q2 in this example) increases.

$$S = \frac{W}{C} - R \tag{1}$$

Our *Workload-Based* algorithm works as follows. For every fixed time interval, or “monitor interval”, our WebScale scaling system collects the current workload information. By dividing the current workload W by the VM capacity C, we have the number of VMs needed. Then we subtract the current number of running VM R from this number, and get S, the amount of VM to be scaled. If S is positive, the decision will be **scale out**; on the other hand, if S is negative, the decision will be **scale in**. The same strategy applies to the scaling of database servers.

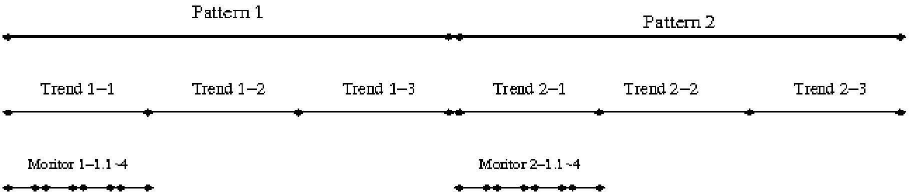


Fig. 2. Relation between different intervals

2.3 Trend Analysis

Some workloads exhibit periodic behaviors, e.g. stock market, enterprise applications, which we can take advantages while making scaling decisions. Periodic behavior means that similar behavior of the workload shows up every fixed length of time, thus we can predict the coming workload by historical data. Workload prediction has been studied by some previous works [11, 12]. Several different strategies have been proposed to predict application workload.

Instead of accurately predicting the workload value, for auto-scaling, it suffices to only predict the *trend* of workload change. *Trend* is the **direction** of workload changing in a fixed size of time, or “trend interval”. There are three possibilities, *up*, *down*, or *constant*. The workload *trend* of an application can be acquired from historical workload data. For most web applications that exhibit periodic behaviors, the pattern length is usually one day or one week. Given the pattern length, we can divide the historical data into pieces, each with length equals to the pattern length, and determine the trend of the pattern.

A pattern consists of several trend intervals, which can be further divided into monitor intervals. Figure 2 shows the relationship between the pattern of workload, trend interval, and monitor interval. Scaling decisions are made in every monitor interval and compared with the trend of the trend interval from previous pattern. For example, monitor interval 2-1.2 makes a decision “scale out”. This decision is then compared with the trend of trend interval 1-2 for confliction. Trend 2-1 will be updated after all the monitor intervals (2-1.1~4) make their decisions.

The trend analysis technique works as a helper to the scaling algorithms by providing workload trend information to the scaling algorithms to make more “correct” decision while handling workloads with periodic behaviors. If a *scale in* decision “conflicts” with the trend, i.e., the decision is *scale in* while the trend is *scale out*, then the decision will be canceled and *no scale* will be the new decision. The rationale is to avoid removing VMs during workload increasing.

3 Experiment Results

3.1 Experiment Setting

Our experiment environment consists of 24 physical servers, each with the following hardware specifications: quad-core X5460 CPU * 2 with hyper-threading, 16 GB memory, and 250 GB disk. The hypervisor is Xen 4.1 and the OS of domain 0 is

Gentoo. There are three kinds of VMs: the auto-scaling master (which manages the running VM cluster), the running VMs, and the data storages that runs MySQL servers. All the VMs use Gentoo OS. The configurations are as follow: Auto-scaling master: 2 core, 2G memory, and 4G disk space; Running VM: 4 core, 4G memory, and 4G disk space; Data storage: 1 core, 4G memory, 100G disk space.

MediaWiki [10] is used as the application benchmark in our experiments. MediaWiki is an open source wiki package originally for use on Wikipedia. We set up MediaWiki and create web pages to simulate a web application uploaded by a user. The contents are the dumps from Wikipedia. The web pages are set to read-only mode.

We use httperf [5] as our performance measuring tool. *Httperf* is a robust and well-known tool for measuring web server performance. It can generate various HTTP workloads, and measure the performance such as average response time.

The workload we used in the experiment is PREDICTABLE. PREDICTABLE is the workload from the log of *Judgegirl*, an online grading system for teaching purpose in department of CSIE, NTU. In the record, each data point represents the load in fifteen minutes. We shrink this length into thirty seconds and the result is shown in Figure 3. There is a pattern that appears four times in Figure 3, each of which is similar but is slightly different from the others. Also there are some sudden workload changes within a pattern.

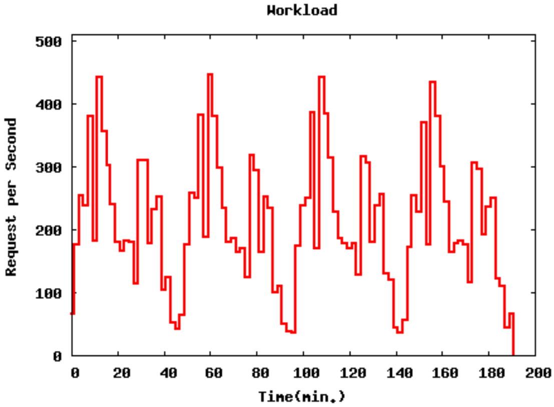
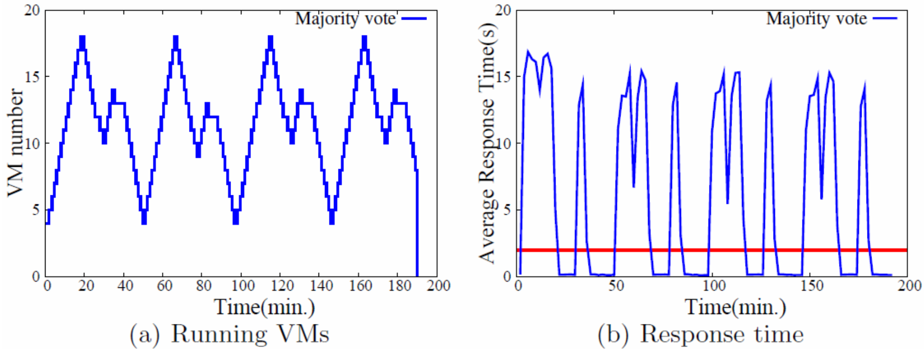


Fig. 3. PREDICTABLE workload

3.2 Comparison of Scaling Algorithms

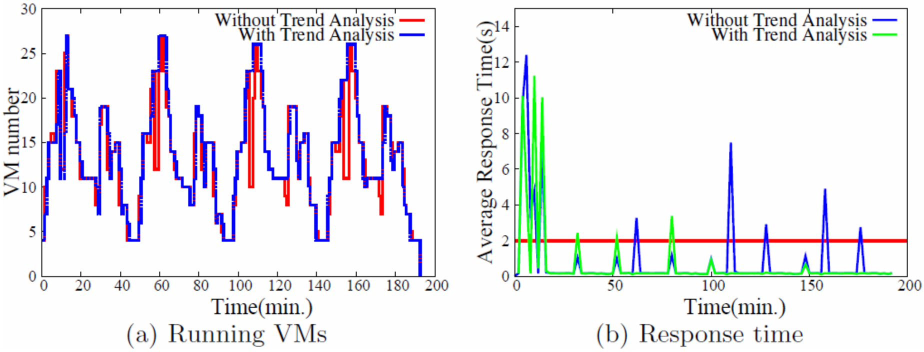
For some web applications, such as stock market or enterprise applications, there exist periodic behaviors. In this experiment, we use PREDICTABLE workload, which has repeated behavior patterns, to test our auto-scaling algorithm. Figure 3 shows the workload. The load interval is two minutes. We compare three scaling strategies: majority vote with scaling threshold (30, 70), workload-based, and workload-based with trend analysis. The monitor intervals for all three algorithms are one minute. The number of database servers is fixed to one in the experiment.



(a) Running VMs (b) Response time

Fig. 4. Majority vote under PREDICTABLE workload

Figure 4(a) and Figure 4(b) are the results of majority vote. Even though the number of running VMs changes with workload, the average response time suffers a lot. The results show that majority vote is not an effective scaling algorithm for web applications with frequently changing workloads.



(a) Running VMs (b) Response time

Fig. 5. Workload-based under PREDICTABLE workload

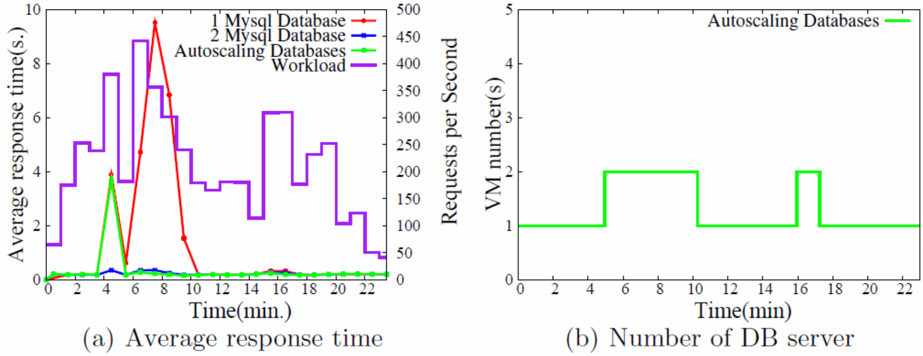
Figure 5(a) and Figure 5(b) depict the number of running VMs and average response time using workload-based as scaling algorithm. Workload-based outperforms majority vote. Furthermore, workload-based with trend analysis has better performance than without trend analysis. In Figure 5(b), there are five peaks (longer than 2 seconds) in the response time without trend analysis. These increasings are caused by sudden large workload changes. For example, in time 108, the workload drops, and the number of running VM decreases with it. However, in time 110, the load suddenly increases. Even if workload-based can respond to sudden workload increase, it still takes time to balance the load to these newly added VMs. Thus the average response time increases for a short period of time until the load is balanced.

On the other hand, workload-based with trend analysis takes the historical trend information into consideration while making scaling decisions. When there is workload fluctuating, it will not invoke scaling action. Therefore, it results in only two peaks in the response time (at time 50 and 80).

In summary, for workloads with periodic behavior, using workload-based algorithm with trend analysis performs the best among all three strategies. Slight sudden workload change will not affect workload-based algorithm with trend analysis. However, the cost of wrong analysis may be high.

3.3 Scaling Data Access Tier

In this section, we study the effect of auto-scaling on the data access tier. We add a backend VM with Round-Robin DNS. This VM also monitors the queries per second to the database servers, and make database scaling decisions using workload-based algorithm. The auto-scaling algorithm for running VMs is workload-based. Other settings are the same as previous section.



(a) Average response time (b) Number of DB server

Fig. 6. Database tier

Figure 6(a) shows the performance result. The purple line is the workload. The red line shows the average response time of using only one database server. The response time drastically increases while the workload increases. On the other hand, the average response times of using two database servers always remain short no matter how workload changes.

As can be seen from Figure 6(a), auto-scaling with workload-based algorithm can always maintain low average response time. When the response time increases to almost 4 seconds at the fourth minute, the large amount of workloads trigger the auto-scaling system and a new database server is added to share the workload. The average response time decreases and remains short after then.

Figure 6(b) shows the number of database servers used. It is clear that the number of database servers is dynamically adjusted according to the workload. The experiment result shows that by applying auto-scaling to the data access tier, we can maintain low average response time while using the right amount of active database servers. By keeping fewer number of active database servers, energy consumption can be reduced.

4 Related Work

The auto-scaling feature has been provided in several cloud service providers and cloud computing systems, such as Amazon EC2 [13] and Google App Engine [2].

Auto scaling in Amazon EC2 is enabled by Amazon CloudWatch, which monitors the resource usage on user instances. Google App Engine [2] provides a very simple auto-scaling strategy. If the volume of incoming requests exceeds the capacity of the instances currently available, they will have to wait in the Pending Queue. When the number of pending requests exceeds a threshold value, a new instance will be created to share the workload.

Many softwares such as Scalr [3] and RightScale [4] provide auto-scaling mechanism that can apply to cloud environments like Eucalyptus [14] or Amazon EC2. Both Scalr [3] and RightScale [4] scales the number of VMs based on the workload on each back-end server. The scaling algorithms are presumed to be majority vote. In this paper, our empirical study has shown that majority vote is not effective for scaling workloads with periodic behaviors.

All of the above existing solutions do not address the issue of workload trend prediction. Most of them use majority vote to deal with workloads even if the workload is predictable. In contrast, our auto scaling system provides trend analysis algorithm and can scale out quickly.

The scaling decision algorithm plays a critical role in an auto scaling system. Chieu et al. [15] proposed an architecture for scaling based on predefined thresholds for Web applications. The algorithm scales out when all VM session numbers exceed the threshold. This approach is simple but insensitive to workload change. Our algorithm is more responsive to workload change since the decision is based on requests per second and HTTP response time and thus can accurately characterize Web application behavior. Mao et al. [16] presented a scaling approach to deal with batch jobs. According to the deadline of each job, it decides whether using current number of VMs is sufficient to meet the deadline. Since Mao's algorithm only considers batch jobs, it is not applicable to Web applications.

Another way to make scaling decision is by workload prediction. Caron et al. [11] used KMP algorithm to find patterns from history data based on N previous time interval. Gmach et al. [12] used an ARMA scheme to find periodgram function. The two prediction algorithms aim to predict the precise workload. However, their results show that it is difficult to make accurate prediction of precise workload. In contrast, our analysis algorithm only predicts the trend of workload change for two reasons. First, predicting workload trend requires much less time complexity than predicting precise workload. Second, our experiments demonstrate that workload-trend guided scaling is very effective.

5 Conclusion

Auto-scaling technique provides on-demand resources according to workload in cloud computing system. In this work, we propose an effective scaling algorithm, *Workload Based algorithm*, for 3-tier web applications. We compare the effectiveness of two scaling algorithms – *majority vote* and *workload-based. Majority vote*, a simple scaling strategy used in most existing systems, makes scaling decisions according to the load of each running VM, while *workload-based* uses the incoming workload, which is requests per second in our work, as the criteria for making scaling decisions.

A helper algorithm, *Trend prediction*, is devised to deal with workloads that exhibit periodical behaviors.

We conduct experiments to evaluate the performance of different scaling algorithms. We compared the performance of these algorithms under actual workload with periodical behavior. The results show that for workloads with periodical behavior, using workload-based algorithm with trend analysis performs the best among all three strategies. Slight sudden workload change will not affect workload-based algorithm with trend analysis. We also show that applying auto-scaling to data access tier can reduce the total database server used while maintaining the performance.

References

1. Amazon elastic compute cloud, `http://aws.amazon.com/ec2/`
2. Google app engine, `https://developers.google.com/appengine/`
3. Scalr, `http://www.scalr.net/`
4. Rightscale, `http://www.rightscale.com/`
5. Mosberger, D., Jin, T.: httperf - a tool for measuring web server performance. SIGMETRICS Perform. Eval. Rev. 26(3), 31–37 (1998)
6. Urdaneta, G., Pierre, G., van Steen, M.: Wikipedia workload analysis for decentralized hosting. Comput. Netw. 53(11), 1830–1845 (2009)
7. Arlitt, M., Krishnamurthy, D., Rolia, J.: Characterizing the scalability of a large web-based shopping system. ACM Trans. Internet Technol. 1(1), 44–69 (2001)
8. Davison, B.D.: Learning web request patterns (2004)
9. Wang, H., Li, B.: Shrinking tuning parameter selection with a diverging number of parameters. Journal of the Royal Statistical Society 71(3), 671–683 (2009)
10. Mediawiki, `http://www.mediawiki.org/`
11. Caron, E., Desprez, F., Muresan, A.: Forecasting for grid and cloud computing on-demand resources based on pattern matching. In: Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science (CLOUDCOM 2010), pp. 456–463 (2010)
12. Gmach, D., Rolia, J., Cherkasova, L., Kemper, A.: Workload analysis and demand prediction of enterprise data center applications. In: Proceedings of the 2007 IEEE 10th International Symposium on Workload Characterization (IISWC 2007), pp. 171–180 (2007)
13. Amazon auto scaling, `http://aws.amazon.com/autoscaling/`
14. Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., Zagorodnov, D.: The eucalyptus open-source cloud-computing system. In: Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID 2009), pp. 124–131 (2009)
15. Chieu, T., Mohindra, A., Karve, A., Segal, A.: Dynamic scaling of web applications in a virtualized cloud computing environment. In: Proceedings of the 2009 IEEE International Conference on e-Business Engineering (ICEBE 2009), pp. 281–286 (2009)
16. Mao, M., Li, J., Humphrey, M.: Cloud auto-scaling with deadline and budget constraints. In: Proceedings of the 11th IEEE/ACM International Conference on Grid Computing (GRID 2010), pp. 41–48 (2010)

Implementation of Cloud-RAID: A Secure and Reliable Storage above the Clouds

Maxim Schnjakin and Christoph Meinel

Hasso Plattner Institute, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
maxim.schnjakin, meinel@hpi.uni-potsdam.de

Abstract. Cloud Computing as a service-on-demand architecture has grown in importance over the previous few years. One driver of its growth is the ever increasing amount of data which is supposed to outpace the growth of storage capacity. In this way public cloud storage services enable organizations to manage their data with low operational expenses. However, the benefits of cloud computing come along with challenges and open issues such as security, reliability and the risk to become dependent on a provider for its service. In general, a switch of a storage provider is associated with high costs of adapting new APIs and additional charges for inbound and outbound bandwidth and requests. In this paper, we describe the design, architecture and implementation of Cloud-RAID, a system that improves availability, confidentiality and integrity of data stored in the cloud. To achieve this objective, we encrypt user's data and make use of the RAID-technology principle to manage data distribution across cloud storage providers. The data distribution is based on users' expectations regarding providers geographic location, quality of service, providers reputation, and budget preferences. We also discuss the security functionality and reveal our observations on the utility and users benefits from using our system. Our approach allows users to avoid vendor lock-in, and reduce significantly the cost of switching providers.

1 Introduction

Cloud Computing is a concept of utilizing computing as an on-demand service. It fosters operating and economic efficiencies and promises to cause a significant change in business. Using computing resources as pay-as-you-go model enables service users to convert fixed IT cost into a variable cost based on actual consumption. Therefore, numerous authors argue for the benefits of cloud computing focusing on the economic value [11], [6].

Among available cloud offerings, storage services reveal an increasing level of market competition. According to iSuppli [9] global cloud storage revenue is set to rise to $5 billion in 2013, up from $1.6 billion in 2009. One reason is the ever increasing amount of data which is supposed to outpace the growth of storage capacity. Currently, it is very difficult to estimate the actual future volume of data but there are different estimates being published. According to IDC review [14], the amount of digital information created and replicated is estimated to surpass 3 zettabytes by the

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 91–102, 2013.

end of 2012. This amount is supposed to more than double in the next two years. In addition, the authors estimate that today there is 9 times more information available than was available five years ago.

However, for a customer (service) to depend on solely one cloud storage provider (in the following provider) has its limitations and risks. In general, vendors do not provide far reaching security guarantees regarding the data retention. Users have to rely on effectiveness and experience of vendors in dealing with security and intrusion detection systems. For missing guarantees service users are merely advised to encrypt sensitive content before storing it on the cloud. Placement of data in the cloud removes many of direct physical controls that a data owner has over data. So there is a risk that service provider might share corporate data with a marketing company or use the data in a way the client never intended. Further, customers of a particular provider might experience vendor lock-in. In the context of cloud computing, it is a risk for a customer to become dependent on a provider for its services. Common pricing schemes foresee charging for inbound and outbound transfer and requests in addition to hosting the actual data. Changes in features or pricing scheme might motivate a switch from one storage service to another. However, because of the data inertia, customers may not be free to select the optimal vendor due to immense costs associated with a switch of one provider to another. The obvious solution is to make the switching and data placement decisions at a finer granularity then all-or-nothing. This could be achieved by replicating corporate data to multiple storage providers. Such an approach implies significant higher storage and bandwidth costs without taking into account the security concerns regarding the retention of data.

A more economical approach which is presented in this paper is to separate data into unrecognizable slices, which are distributed to providers - whereby only a subset of the nodes needs to be available in order to reconstruct the original data. This is indeed very similar to what has been done for years at the level of file systems and disks. In our work we use RAID-like (Redundant Array of Independent Disks) techniques to overcome the mentioned limitations of cloud storage in the following way:

1. **Security.** The provider might be trustworthy, but malicious insiders represent a well known security problem. This is a serious threat for critical data such as medical records, as cloud provider staff has physical access to the hosted data. One solution might be to encrypt data before the transmission to providers and to decrypt data when receiving those. This requires users to handle the distribution of cryptographic keys when the data needs to be accessed by different users. For each potential customer, it is both expensive and time consuming to handle these security and usability concerns. We tackle the aforementioned problem by encrypting and encoding the original data and later by distributing the fragments transparently across multiple providers. This way, none of the storage vendors is in an absolute possession of the client's data. Moreover, the usage of enhanced erasure algorithms enables us to improve the storage efficiency and thus also to reduce the total costs of the solution.
2. **Service Availability.** Management of computing resources as a service by a single company implies the risk of a single point of failure. This failure depends on many

factors such as financial difficulties (bankruptcy), software or network failure, etc. However, even if the vendor runs data centers in various geographic regions using different network providers, it may have the same software infrastructure. Therefore, a failure in the software in one center will affect all the other centers, hence affecting the service availability. In July 2008, for instance, Amazon storage service S3 was down for 8 hours because of a single bit error [25]. Our solution addresses this issue by storing the data on several clouds - whereby no single entire copy of the data resides in one location, and only a subset of providers needs to be available in order to reconstruct the data.

3. **Reliability.** Any technology can fail. According to a study conducted by Kroll Ontrack[1] 65 percent of businesses and other organizations have frequently lost data from a virtual environment. A number that is up by 140 percent from just last year. Admittedly, in the recent times, no spectacular outages were observed. Nevertheless failures do occur. For example, in October 2009 a subsidiary of Microsoft, Danger Inc., lost the contracts, notes, photos, etc. of a large number of users of the Sidekick service [20]. Most of the data could be recovered within a few weeks, but the users of Ma.gnolia[2] were not so lucky in February of the same year, when the company lost half a terabyte of data [17]. We deal with the problem by using erasure algorithms to separate data into packages, thus enabling the application to retrieve data correctly even if some of the providers corrupt or lose the entrusted data.
4. **Data lock-in.** By today there are no standards for APIs for data import and export in cloud computing. This limits the portability of data and applications between providers. For the customer this means that he cannot seamlessly move the service to another provider if he becomes dissatisfied with the current provider. This could be the case if a vendor increases the fees, goes out of business, or degrades the quality of the provided services. As stated above, our solution does not depend on a single service provider. The data is balanced among several providers taking into account user expectations regarding the price and availability of the hosted content. Moreover, with erasure codes we store only a fraction of the total amount of data on each cloud provider. In this way, switching one provider for another costs merely a fraction of what it would be otherwise.

The main contribution of this paper is: we present a design of an application that can be used to overcome the limitations of individual clouds by using encryption, erasure codes and by integrating various cloud storage providers.

2 Architecture Overview

The ground of our approach is to find a balance between benefiting from the cloud's nature of pay-per-use and ensuring the security of the company's data. The goal is to achieve such a balance by distributing corporate data among multiple storage

[1] http://www.krollontrack.com/resource-library/case-studies/

[2] http://gnolia.com/

providers, automizing big part of the selection process of a cloud provider, and removing the auditing and administrating responsibility from the customer's side. As mentioned above, the basic idea is not to depend on solely one storage provider but to spread the data across multiple providers using redundancy to tolerate possible failures. The approach is similar to a service-oriented version of RAID. While RAID manages sector redundancy dynamically across hard-drives, our approach manages file distribution across cloud storage providers. RAID 5, for example, stripes data across an array of disks and maintains parity data that can be used to restore the data in the event of disk failure. We carry the principle of the RAID-technology to cloud infrastructure. In order to achieve our goal we foster the usage of erasure coding technics (see 3.3). This enables us to tolerate the loss of one or more storage providers without suffering any loss of content [26], [13]. Our architecture includes the following main components:

- **User Interface Module.** The interface presents the user a cohesive view on the data and available features. Here users can manage their data and specify requirements regarding the data retention (quality of service parameters).
- **Resource Management Module.** This system component is responsible for an intelligent deployment of data based on the user's requirements.
- **Data Management Module.** This component handles data management on behalf of the resource management module.

Interested readers will find more background information in our previous work [24],[21]. The system has a number of core components that contain the logic and management layers required to encapsulate the functionality of different storage providers. The next section gives an overview on the implementation of our system on a more detailed level.

3 Design

Any application needs a model of storage, a model of computation and a model of communication. In this section we describe how we achieve the goal of the consistent, unified view on the data management system to the end-user. The web portal is developed using Grails, JNI and C technologies, with a MySQL back-end to store user accounts, current deployments, meta data, and the capabilities and pricing of cloud storage providers. Keeping the meta data locally ensures that no individual provider will have access to stored data. In this way, only users that have authorization to access the data will be granted access to the shares of (at least) k different clouds and will be able to reconstruct the data. Further, our implementation makes use of AES for symmetric encryption, SHA-1 and MD5 for cryptographic hashes and an improved version of Jerasure library [18] for using the Cauchy-Reed-Solomon and Liberation erasure codes. Our system communicates with providers via ”storage connectors”, which are discussed further in this section.

3.1 Service Interface

The graphical user interface provides two major functionalities to an end-user: data administration and specification of requirements regarding the data storage. Interested readers are directed to our previous work [22] which gives a more detailed background on the identification of suitable cloud providers in our approach. In short, the user interface enables users to specify their requirements (regarding the placement and storage of user's data) manually in form of options, for example:

- **budget-oriented** content deployment (based on the price model of available providers)
- data placement based on **quality of service parameters** (for example availability, throughput or average response time)
- storage of data based on **geographical regions** of the user's choice. The restriction of data storage to specific geographic areas can be reasonable in the case of legal restrictions.

3.2 Storage Repositories

Cloud Storage Providers. Cloud storage providers are modeled as a storage entity that supports six basic operations, shown in table 1. We need storage services to support not more than the aforementioned operations. Further, the individual providers are not trusted. This means that the entrusted data can be corrupted, deleted or leaked to unauthorized parties [16]. This fault model encompasses both malicious attacks on a provider and arbitrary data corruption like the Sidekick case (section 1). The protocols require n = k + m storage clouds, at most m of which can be faulty. Present-day, our prototypical implementation supports the following storage repositories: Amazons S3 (in all available regions: US west and east coast, Ireland, Singapore and Tokyo), Box, Rackspace Cloud Files, Azure, Google Cloud Storage and Nirvanix SND. Further providers can be easily added.

Service Repository. At the present time, the capabilities of storage providers are created semi-automatically based on an analysis of corresponding SLAs which are usually written in a plain natural language [5]. Until now the claims stated in SLAs need to be translated into WSLA statements and updated manually (interested readers will find more background information in our previous work [22]). Subsequently the formalized information is imported into a database of the system component named service repository. The database tracks logistical details regarding the capabilities of storage services such as their actual pricing, SLA offered, and physical locations. With this, the service repository represents a pool with available storage services.

Matching. The selection of storage services for the data distribution occurs based on user preferences set in the user interface. After matching user requirements and provider capabilities, we use the reputation of the providers to produce the final list of potential providers to host parts of the user's data. A provider's reputation holds the

Table 1. Storage connector functions

Function	Description
create(ContainerName)	creates a container for a new user
write(ContainerName, ObjectName)	writes a data object to a user container
read(ContainerName, ObjectName)	reads the specified data object
list(ContainerName)	list all data objects of the container
delete(ContainerName, ObjectName)	removes the data object from the container
getDigest(ContainerName, ObjectName)	returns the hash value of the specified data object

details of his historical performance plus his ratings in the service registries and is saved in a Reputation Object (introduced in our previous work [3], [2], [4]). By reading this object, we know a provider's reputation concerning each performance parameter (e.g. has high response time, low price). With this information the system creates a prioritized list of repositories for each user. In general, the number of storage repositories needed to ensure data striping depends on a user's cost expectations, availability and performance requirements. The total number of repositories is limited by the number of implemented storage connectors.

3.3 Data Management

Data Model. In compliance with [1] we mimic the data model of Amazon's S3 by the implementation of our encoding and distribution service. All data objects are stored in containers. A container can contain further containers. Each container represents a flat namespace containing keys associated with objects. An object can be of an arbitrary size, up to 5 gigabytes (limited by the supported file size of cloud providers). Objects must be uploaded entirely, as partial writes are not allowed as opposed to partial reads. Our system establishes a set of n repositories for each data object of the user. These represent different cloud storage repositories (see figure 1).

Encoding. Upon receiving a write request the system splits the incoming object into k data fragments of an equal size - called chunks. These k data packages hold the original data. In the next step the system adds m additional packages whose contents are calculated from the k chunks, whereby k and m are variable parameters [18]. This means, that the act of encoding takes the contents of k data packages and encodes them on m coding packages. In turn, the act of decoding takes some subset of the collection of n = k + m total packages and from them recalculates the original data. Any subset of k chunks is sufficient to reconstruct the original object of size s [19]. The total size of all data packets (after encoding) can be expressed with the following equation: $\left(\frac{s}{k}*k\right)+\left(\frac{s}{k}*m\right)= s+\left(\frac{s}{k}*m\right)=s*\left(1+\frac{m}{k}\right)$. With this, the usage of erasure codes increases the total storage by a factor of m k . Summarized, the overall overhead depends on the file size and the defined m and k parameters for the erasure configuration. Figure 2 visualizes the performance of our application using different erasure configurations. Competitive storage providers claim to have SLAs ranging from 99% to 100% uptime percentages for their services. Therefore choosing m = 1 to tolerate one provider outage or failure at time will be sufficient in the majority of cases. Thus, it makes sense to increase k and spread the packages across more providers to lower the overhead costs.

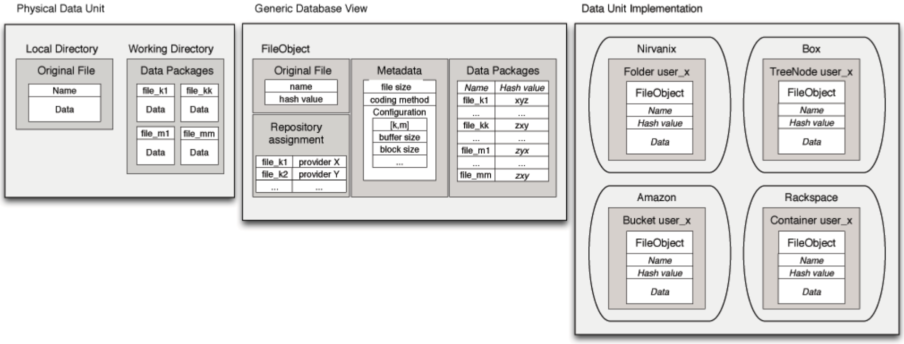


Fig. 1. Data unit model at different abstraction levels. At a physical layer (local directory) each data unit has a name (original file name) and the encoded k+m data packages. In the second level, Cloud-RAID perceives data objects as generic data units in abstract clouds. Data objects are represented as data units with the according meta information (original file name, cryptographic hash value, size, used coding configuration parameters m and k, word size etc.). The database table ”Repository Assignment” holds the information about particular data packages and their (physical) location in the cloud. In the third level, data objects are represented as containers in the cloud. Cloud-RAID supports various cloud specific constructions (buckets, treenodes, containers etc.).

In the next step, the distribution service makes sure that each encoded data package is sent to a different storage repository. In general, our system follows a model of one thread per provider per data package in such a way that the encryption, decryption, and provider accesses can be executed in parallel.

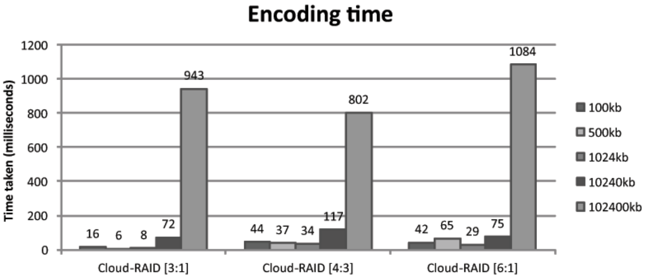


Fig. 2. The average performance of the erasure algorithm with data objects of varying sizes (100kB, 500kB, 1MB, 10MB and 100MB)

However, most erasure codes have further parameters as for example w, which is word size[3]. In addition, further parameters are required for reassembling the data (original file size, hash value, coding parameters, and the erasure algorithm used). This metadata is stored in a MySQL back-end database after performing a successful write request.

Data Distribution. Each storage service is integrated by the system by means of a storage-service-connector (in the following service-connector). These provide an intermediate layer for the communication between the resource management service (see section 3.4) and storage repositories hosted by storage vendors. This enables us to hide the complexity in dealing with proprietary APIs of each service provider. The basic connector functionality covers operations like creation, deletion or renaming of files and folders that are usually supported by every storage provider. Such a service-connector must be implemented for each storage service, as each provider offers a unique interface to its repository. As discussed earlier in this chapter all accesses to the cloud storage providers can be executed in parallel. Therefore, following the encoding, the system performs an initial encryption of the data packages based on one of the predefined algorithms (this feature is optional).

Reassembling the Data. When the service receives a read request, the service component fetches k from n data packages (according to the list with prioritized service providers which can be different from the prioritized write-list, as providers differ in upload and download throughput as well as in cost structure) and reassembles the data. This is due to the fact, that in the pay-per-use cloud models it is not economical to read all data packages from all clouds. Therefore, the service is supported by a load balancer component, which is responsible for retrieving the data units from the most appropriate repositories. Different policies for load balancing and data retrieving are conceivable as parts of user's data are distributed between multiple providers. A read request can be directed to a random data share or the physically closest service (latency-optimal approach). Another possible approach is to fetch data from service providers that meet certain performance criteria (e.g response time or throughput). Finally, there is a minimal-cost aware policy, which guides user requests to the cheapest sources (cost optimal approach). The latter strategy is implemented as a default configuration in our system. Other more sophisticated features as a mix of several complex criteria (e.g. faults and overall performance history) are under development at present. However, the read optimization has been implemented to save time and costs.

3.4 Resource Management Service

This component tracks each user's actual deployment and is responsible for various housekeeping tasks:

[3] The description of a code views each data package as having w bits worth of data.

1. The service is equipped with a MySQL back-end database to store crucial information needed for deploying and reassembling of users data.
5. Further, it audits and tracks the performance of the participated providers and ensures, that all current deployments meet the corresponding requirements specified by the user.
6. The management component is also responsible for scheduling of not time-critical tasks.

Further details can be found in our previous work [21].

4 Related Work

The main idea underlying our approach is to provide RAID technique at the cloud storage level. In [8] the authors introduce the HAIL (High-Availability Integrity Layer) system, which utilizes RAID-like methods to manage remote file integrity and availability across a collection of servers or independent storage services. The system makes use of challenge-responce protocols for retrievability (POR) [15] and proofs of data possession (PDP) [15] and unifies these two approaches. In comparison to our work, HAIL requires storage providers to run some code whereas our system deals with cloud storage repositories as they are. Further, HAIL does not provide confidentiality guarantees for stored data. In [12] Dabek et al. use RAID-like techniques to ensure the availability and durability of data in distributed systems. In contrast to the mentioned approaches our system focuses on the economic problems of cloud computing described in chapter 1.

Further, in [1] authors introduce RACS, a proxy that spreads the storage load over several providers. This approach is similar to our work as it also employs erasure code techniques to reduce overhead while still benefiting from higher availability and durability of RAID-like systems. Our concept goes beyond a simple distribution of users' content. RACS lacks the capabilities such as intelligent file placement based on users' requirements or automatic replication. In addition to it, the RACS system does not try to solve security issues of cloud storage, but focuses more on vendor lock-in. Therefore, the system is not able to detect any data corruption or confidentiality violations.

The future of distributed computing has been a subject of interest for various researchers in recent years. The authors in [10] propose an architecture for market-oriented allocation of resources within clouds. They discuss some existing cloud platforms from the market-oriented perspective and present a vision for creating a global cloud exchange for trading services. Further, our service acts as an abstraction layer between service vendors and service users automatising data placement processes. In fact, our approach enables cloud storage users to place their data on the cloud based on their security policies as well as quality of service expectations and budget preferences. Furthermore, the usage of erasure algorithms for data placement is more efficient than a native replication (in terms of storage and costs).

5 Conclusion

In this paper we outlined some general problems of cloud computing such as security, service availability and a general risk for a customer to become dependent on a service provider. In the course of the paper we demonstrated how our system deals with the mentioned concerns. In a nutshell, we stripe users' data across multiple providers while integrating with each storage provider via appropriate service-connectors. These connectors provide an abstraction layer to hide the complexity and differences in the usage of storage services.

We use erasure code techniques for striping data across multiple providers. The first experiments proved, that given the speed of current disks and CPUs, the libraries used are fast enough to provide good performance, reliable storage system. The average performance overhead caused by data encoding is less than 2% of the amount of time for data transfer to a cloud provider [23]. With this, encoding is dominated by the transmission times and can be neglected. Here, the storage overhead can be varied to achieve higher availability values depending on user requirements. It is up to each individual user to decide whether the additional cost caused by data encoding with higher availability due to determination of higher m parameter are justified. By spreading users data across multiple clouds our approach enables users to avoid the risk of data lock-in and provide a low-level protection even without using security functionality.

However, additional storage offerings are expected to become available in the next few years. Due to the flexible and adaptable nature of our approach, we are able to support any changes in existing storage services as well as incorporating support for new providers as they appear.

6 Future Work

In the last month, we deployed your application using seven commercial cloud storage repositories in different countries in order to conduct a comprehensive test of our system. This includes the predictability and sufficiency of response time and throughput, the overall performance as well as the validation of file consistency.

The results of the experiment are being analysed currently an will be addressed in our next publication. Whilst our system is still under development at present, we will have to use the results of the conducted experiment to improve the overall performance and reliability. This includes for instance the predictability and sufficiency of response time and throughput as well as the validation of file consistency.

In the next step in the development of our registry service we will have to look at ways in which we are able to verify that providers have retained data without retrieving it from the storage repositories and without having to access the entire data. Reading an entire archive, even periodically, is expensive in upload and download costs and limits the scalability of networks. Existing approaches as PDP [7] require service providers to run some code, which is not suitable with our solution.

In addition, we are also planning to implement more service connectors and thus to integrate additional storage services. Any extra storage resource improves the performance and responsiveness of our system for end-users.

References

1. Abu-Libdeh, H., Princehouse, L., Weatherspoon, H.: Racs: A case for cloud storage diversity. In: SoCC 2010 (June 2010)
2. Alnemr, R., Bross, J., Meinel, C.: Constructing a context-aware service-oriented reputation model using attention allocation points. In: Proceedings of the IEEE International Conference on Service Computing, SCC 2009 (2009)
3. Alnemr, R., Meinel, C.: Getting more from reputation systems: A context-aware reputation framework based on trust centers and agent lists. In: International Multi-Conference on Computing in the Global Information Technology (2008)
4. Alnemr, R., Schnjakin, M., Meinel, C.: Towards context-aware service-oriented semantic reputation framework. In: International Joint Conference of IEEE TrustCom/IEEE ICESS/FCST, pp. 362–372 (2011)
5. Amazon. Amazon ec2 service level agreement (2009) (online)
6. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: Above the clouds: A berkeley view of cloud computing. Technical Report UCB/EECS-2009, EECS Department, University of California, Berkeley (2009)
7. Ateniese, G., Burns, R., Curtmola, R., Herring, J., Kissner, L., Peterson, Z., Song, D.: Provable data possession at untrusted stores. Cryptology ePrint Archive, Report 2007/202 (2007)
8. Bowers, K.D., Juels, A., Oprea, A.: Hail: A high-availability and integrity layer for cloud storage. In: CCS 2009 (November 2009)
9. Burt, J.: Future for cloud computing looks good, report says (2009) (online)
10. Buyya, R., Yeo, C.S., Venugopal, S.: Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In: Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (August 2008)
11. Carr, N.: The Big Switch. Norton (2008)
12. Dabek, F., Kaashoek, M.F., Karger, D., Morris, R., Stoica, I.: Wide-area cooperative storage with cfs. In: ACM SOSP (October 2001)
13. Dingledine, R., Freedman, M.J., Molnar, D.: The free haven project: Distributed anonymous storage service. In: Federrath, H. (ed.) Anonymity 2000. LNCS, vol. 2009, pp. 67–95. Springer, Heidelberg (2001)
14. Gantz, J., Reinsel, D.: Extracting value from chaos (2009) (online)
15. Krawczyk, H.: LFSR-based hashing and authentication. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 129–139. Springer, Heidelberg (1994)
16. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Trans. Program. Lang. Syst. 4(3), 382–401 (1982)
17. Naone, E.: Are we safeguarding social data? (2009) (online)
18. Plank, J.S., Simmerman, S., Schuman, C.D.: Jerasure: A library in C/C++ facilitating erasure coding for storage applications - Version 1.2. Technical Report CS-08-627, University of Tennessee (August 2008)

19. Rhea, S., Wells, C., Eaton, P., Geels, D., Zhao, B., Weatherspoon, H., Kubiatowicz, J.: Maintenance free global storage in oceanstore. IEEE Internet Computing (September 2001)
20. Sarno, D.: Microsoft says lost sidekick data will be restored to users. Los Angeles Times (October 2009)
21. Schnjakin, M., Alnemr, R., Meinel, C.: A security and high-availability layer for cloud storage. In: Chiu, D.K.W., Bellatreche, L., Sasaki, H., Leung, H.-f., Cheung, S.-C., Hu, H., Shao, J. (eds.) WISE Workshops 2010. LNCS, vol. 6724, pp. 449–462. Springer, Heidelberg (2011)
22. Schnjakin, M., Alnemr, R., Meinel, C.: Contract-based cloud architecture. In: Proceedings of the Second International Workshop on Cloud Data Management, CloudDB 2010, pp. 33–40. ACM, New York (2010)
23. Schnjakin, M., Korsch, D., Schoenberg, M., Meinel, C.: Implementation of a secure and reliable storage above the untrusted clouds. In: Proceedings of 8th International Conference on Computer Science and Education, ICCSE 2013 (to appear in April 2013)
24. Schnjakin, M., Meinel, C.: Platform for a secure storage-infrastructure in the cloud. In: Proceedings of the 12th Deutscher IT-Sicherheitskongress, Sicherheit 2011 (2011)
25. The Amazon S3 Team. Amazon s3 availability event: July 20, 2008 (2008) (online)
26. Weatherspoon, H., Kubiatowicz, J.D.: Erasure coding vs. Replication: A quantitative comparison. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 328–337. Springer, Heidelberg (2002)

An Improved Min-Min Task Scheduling Algorithm in Grid Computing

Soheil Anousha[1,*] and Mahmoud Ahmadi[2]

[1] Department of Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran
Soheil.anousha@gmail.com
[2] Department of Computer Engineering, University of Razi, Kermanshah, Iran
M.ahmadi@razi.ac.ir

Abstract. Supercomputer prices on one hand and the need for vast computational resources on the other hand, led to the development of network computing resources were under name Grid. For optimal use of the capabilities of large distributed systems, the need for effective and efficient scheduling algorithms is necessary. For reduction of total completion time and improvement of load balancing, many algorithms have been implemented. In this paper, we propose new scheduling algorithm based on well known task scheduling algorithms, Min-Min. The proposed algorithm tries to use the advantages of this basic algorithm and avoids its drawbacks. To achieve this, the proposed algorithm firstly like Min-Min estimating of the completion time of the tasks on each of resources and then selects the appropriate resource for scheduling. The experimental results show that the proposed algorithm improved total completion time of scheduling in compared to Min-Min algorithm.

Keywords: Grid, resource, task scheduling algorithm, Min-Min, completion time.

1 Introduction

Reduction of Makespan is a fundamental objective of optimizing task scheduling algorithm in distributed systems. In this field, a lot of efforts have been made and huge projects such as Globus [1] and Condor [2] for the development of computational resources in computer networks is presented. The Grids use of resources of connected- computers to the network and using the outcome of these resources to easily do complex calculations. They do this with fragmenting of resources and allocation of them to a computer in the network. Resource allocation is done in two stages: Resource discovery and resource selection.

Stage 1 (Resource discovery): In this stage, List of all available resources is prepared. Actually, resource discovery generates a list of potential resources.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 103–113, 2013.

Stage 2 (Resource selection): this stage involves collecting information of resources and selecting the best set to match the application requirements. After this, the task is executed.

To make effective use of the huge capabilities of the computational grids, efficient task scheduling algorithms are required [9]. Many Grid task scheduling algorithms such as [9, 10] have some features in common, that are performed in multiple steps to solve the problem of matching application needs with resource availability and providing quality of service. Also we know that solving the matching problem to find the choice of the best pairs of jobs and resources is **NPcomplete** problem [17]. The well known example of algorithms is Min-min [17]. This algorithm estimate completion times of each of the tasks on each of the grid resources. Estimating the execution time of each task on different resources, the Min-min algorithm selects the task with minimum completion time and assigns it to the resource on which the minimum execution time is achieved. The algorithm applies a same procedure to the remaining tasks [8]. The Min-Min algorithm seems to do worse operation, whenever the number of small tasks is much more than the large ones. So, proposing a new algorithm to resolve the above mentioned problem is required.

This paper offers a new task scheduling algorithm to resolve this problem with applying the Min-Min or Max-Min algorithms to scheduling. To select the algorithm for first scheduling, we propose new Makespan. The most important of factor that can be improved by our algorithm is total completion time. The remainder of this paper is organized as follows. Related works are presented in section 2. In section 3, existing task scheduling algorithms is presented. In section 4, a new scheduling algorithm is proposed and the proposed the algorithm is depicted through an illustrative example. In section 5, the experimental results are presented and discussed. Finally, section 6 concludes the paper and presents future works.

2 Related Works

For optimal use of available resources in the network and getting the less execution time, needs to provide a new scheduling algorithm is crucial. These algorithms assign tasks to the resources and provide the best conditions of quality of services.

F. Dong et al. have proposed an algorithm called QoS priority grouping scheduling [8]. This algorithm, considers deadline and acceptation rate of the tasks and the makespan of the wholes system as important factors for task scheduling.

S. Parsa et al. also have proposed an algorithm called RASA [9]. RASA begins with Min-Min algorithm if the number of available resources is odd and starts with Max-Min algorithm if the number of available resources is even. The remaining tasks are assigned to their appropriate resources by one of the two strategies, alternatively.

K. Etminani et al. have proposed a new algorithm which uses Max-min and Min-min algorithms [10]. The algorithm determines to select one of these two algorithms, dependent on the standard deviation of the expected completion times of the tasks on each of the resources. These algorithms have some advantages and disadvantages.

For example in RASA [9], if number of available resources be odd, the Min-Min strategy is applied to assign the first task, otherwise the Max-Min strategy is applied. The remaining tasks are assigned to their appropriate resources by one of the two strategies, alternatively. Now, if we have odd resources and the Max-Min strategy have better situation than Min-Min, we should select Min-Min instead of Max-Min.

3 Existing Task Scheduling Algorithms

Generally, the scheduling algorithms are divided into two basic categories: **immediate mode** scheduling and **batch mode** scheduling. In Immediate mode task is mapped onto a resource as soon as it arrives at the scheduler. For this mode we can mentioned MET and MCT algorithms. The **MET** (minimum execution time) heuristic assigns each task to the machine that performs that task's computation in the least amount of execution time [17]. **MET** deployed in SmartNet [6] and have **O(R)** time complexity when we have **R** resources. The **MCT** (minimum completion time) heuristic assigns each task to the machine so that the task will have the earliest completion time [17]. Also **MCT** deployed in SmartNet [6] and like the **MET** have **O(R)** time complexity when we have **R** resources. In the batch mode, tasks are not mapped onto the resources as they arrive; instead they are collected into a set that is examined for mapping at prescheduled times called mapping events. The independent set of tasks which is considered for mapping at the mapping events is called a meta-task [14]. Min-Min, Max-Min and Sufferage Algorithm are examples of this type.

3.1 Min-Min Algorithm

Min-Min algorithm starts with a set of all unmapped tasks. The machine that has the minimum completion time for all jobs is selected. Then the job with the overall minimum completion time is selected and mapped to that resource. The ready time of the resource is updated. This process is repeated until all the unmapped tasks are assigned. Compared to **MCT** this algorithm considers all jobs at a time. So it produces a better makespan. Time complexity of Min-Min algorithm when we have **R** resources and **T** tasks is $\mathbf{O(T^2R)}$.

3.2 Max-Min Algorithm

Max-Min is very similar to Min-Min algorithm. Like the Min-Min, the machine that has the minimum completion time for all jobs is selected. Then unlike the Min-Min, the job with the overall maximum completion time is selected and mapped to that resource. The ready time of the resource is updated. This process is repeated until all the unmapped tasks are assigned. The idea of this algorithm is to reduce the wait time of the large jobs. This algorithm takes $\mathbf{O(T^2R)}$ time, when we have **R** resources and **T** tasks. The pseudo code of Min-Min and Max-Min algorithm is depicted in Fig.1.

```
for all tasks $t_i$ in $MT$
  for all machines $mj$
    $CT_{ij} = ET_{ij} + r_j$
   do until all tasks in $MT$ are mapped
    for each task $t_i$ in $MT$
      Find minimum $CT_{ij}$ and resource that obtains it.
      Find the task $t_k$ with the minimum $CT_{ij}$.
      Assign $t_k$ to resource $m_l$ that
      Delete $t_k$ from MT
  Update $r_l$
  Update $CT_{ij}$ for all i
  End do
```

Fig. 1. The pesudo code of Min-Min (Max-Min) algorithm

Note: In Max-Min algorithm replace the underlined word in Fig.1 *minimum* with *maximum.*

As shown in Fig.1, firstly it computes the amount of task completion time CT_{ij} for all tasks in MT on all resources from the following equation:

$$CT_{ij} = ET_{ij} + r_j \tag{1}$$

CT_{ij} is completion time and ET_{ij} is expected execution time of task *i*th on resource *j*th, and r_j is the ready time for resource *j*th ***(r_j is the ready time or availability time of resource j after completing previously assigned jobs)***. After that, the set of minimum expected completion time for each task in MT is found (**resource discovery**), then the task with the overall minimum expected completion time from MT is selected and assigned to the corresponding resource (**resource selection**).

4 The Proposed Algorithm

The Min-Min algorithm seems worse in the cases when the number of short tasks is much more than the long ones. For example, if there is only one long task, the Max-Min algorithm executes many short tasks concurrently with the long task. In this case, the makespan of the system is most likely determined by the execution time of the long task. However, since the Min-Min algorithm attempts to assign the short tasks before the long one, the makespan increases compared with the Max-Min. On the other hand, mapping the longest task to the fastest resource provides a better opportunity for concurrent execution of the small tasks on different resources. In this certain situation, the Max-min provides a better mapping which supports load balancing across the grid resources more than the Min-Min [7]. Our proposed scheduling algorithm is presented in Fig.2. Firstly all the tasks should be sorted ascending. It means tasks with minimum completion time are in the front of queue and tasks with maximum completion time are in the rear of queue. Secondly this algorithm like the Min-Min algorithm, computes minimum completion time of all tasks on available resources. After that, the resource according to the appropriate condition should be chosen. For choosing a task for scheduling, firstly we computes average of completion time and standard deviation of existing tasks. According to [16] average of completion time (ACT) and standard deviation (SD) of tasks can be calculated by using the following relations:

$$ACT = \frac{\sum_{i=1}^{r} CTij}{r} \tag{2}$$

$$SD = \sqrt{\frac{\sum_{i=1}^{r} (CTij+ACT)^2}{r}} \tag{3}$$

(Where ***r*** *denotes number of resources*).

After, the proposed algorithm compares values of ***ACT*** and ***SD***. By applying this heuristic, two cases might happen:

1. If ***ACT*** is less than ***SD***, it means the length of all tasks in *MT* is in a small range, so we will select from front of queue to assign the next task (line 13).
2. Otherwise, we will select from rear of queue to assign the next task (line 15).

4.1 Time Complexity of Proposed Algorithm

The order of this algorithm is depending to two for loop that mentioned in line (3) and (4) and also it's should be operable on all tasks (line (2)). In lines 3-5, two nested **for** loops takes *O(T.R)* time: internal **for** loop runs *R* times (number of resources) and external **for** loop runs ***T*** times (number of tasks). This process is done for all tasks in MT and runs ***R*** times. Therefore, lines 2-17 take ***O(T²R)*** time. So, this algorithm, likes the Min-Min and the Max-Min algorithm takes ***O(T²R)*** time, when we have ***R*** resources and ***T*** tasks.

```
(1) Sort all tasks in MT ascending // MT=Meta-Task
(2)  While there are tasks in MT
(3)    for all tasks t_i in MT
(4)      for all machines m_j
(5)        $CT_{ij}=ET_j+r_j$  // $r_j$ = Ready Time
(6)          for all tasks t_i in MT
(7)             Find the minimum CTij and resource m_j
(8)               if there is more than one resource that obtains it.
(9)                  Select resource with least usage so far // for load balancing
(10)                     Calculate average completion time & standard deviation of all tasks in MT.
(11)      If ACT > SD then // ACT = average of Completion Time, SD = standard deviation
(12)           Assign t_f to resource m_x that obtains $CT_{fx}$  // $t_f = t_{front}$
(13)      Else
(14)           Assign t_r to resource m_x that obtains $CT_{rx}$ // $t_r = t_{rear}$
(15)      End if
(16)             Delete assigned task from MT.
(17)  End While
```

Fig. 2. The pesudo code of proposed algorithm

4.2 An Illustrative Example

As a simple example, assume there is a grid environment with two resources. The completion time of the tasks are depicted in Table 1.

Table 1. Completion time of the tasks on the resources

	Resources	
Tas ks	$\mathbf{R_1}$	$\mathbf{R_2}$
$\mathbf{T_1}$	2	4
$\mathbf{T_2}$	6	10
$\mathbf{T_3}$	10	20
$\mathbf{T_4}$	45	90

Fig.3(a) includes one Gantt charts representing the results of applying Min-Min algorithms according to the values of completion time that described in Table 1. Also Fig.3(b) shows Gantt charts of our algorithm with the conditions of Table 1. Comparing the two figures shows that the proposed algorithm could obtains a better time unlike the Min-Min. Also the proposed algorithm uses Resource 2 and helps load balancing. As you see in Fig.3(a), Min-Min gives a makespan of 63, but in Fig.3(b) proposed algorithm gives a makespan of 45. Also, in proposed algorithm, two resources had been working throughout this assignment, but in the Min-Min algorithm, the resources R1 that obtains better completion time, is busy all the time but R2 is free. So here, proposed algorithm has better makespan and load balancing level than Min-Min algorithm.

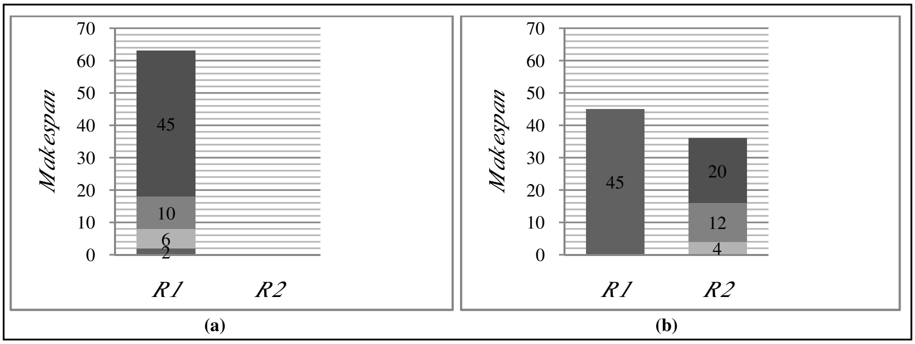


Fig. 3. Makespan of Min-Min algorithm and proposed algorithm

Also Fig.4 shows that how proposed algorithm selects tasks for scheduling, according to the values of completion time that described in Table 1.

Tasks	T_1	T_2	T_3	T_4
CT_{i1}	**2**	**6**	**10**	**45**
				*

ACT = 15.75 , SD = 17.12
ACT <= SD
(Select task from rear of queue) **(a)**

Tasks	T_1	T_2	T_3
CT_{i1}	**2**	**6**	**10**
	*		

ACT = 6 , SD = 2.83
ACT > SD
(Select task from front of queue) **(b)**

Tasks	T_2	T_3
CT_{i1}	**6**	**10**
	*	

ACT = 8 , SD = 2
ACT > SD
(Select task from front of queue) **(c)**

Tasks	T_3
CT_{i1}	**10**
	*

(Select last task of queue)
(d)

Fig. 4. Selection of tasks in proposed algorithm

In Fig.5.a, there exists one long task and three short tasks, the case where proposed algorithm unlike the Min-Min algorithm (that select T_1 for first step), select long task for scheduling (T_4). As it can be seen, the value of ***ACT*** is less than of ***SD***, so we should select task from the rear of queue.

5 Simulation and Experimental Results

To compare and evaluate the proposed algorithm with other algorithms such as Max-Min and Min-Min, a simulation environment known as *GridSim* toolkit [13] has been used. Our experimental testing performed in three assumptions:

1. Assumption I: A few short tasks along with many long tasks; i.e. the case where Min-Min outperforms Max-Min.

2. Assumption II: A few long tasks along with many short tasks; i.e. the case where Max-Min outperforms Min-Min.

3. Assumption III: With random tasks.

Number of resources is chosen to be 10. Three different numbers of tasks has been chosen: 1000 for light load and finally 5000 for heavy load. Result of this simulation as follows:

In Fig.5(a) and Fig.5(d), which Min-Min outperforms Max-Min, the proposed algorithm acts like Min-Min, and also In Fig.5(b) and Fig.5(e), which Max-Min outperforms Min-Min, the proposed algorithm acts like Max-Min. But in Fig.5(c) and Fig.5(f), with random tasks, we see that the proposed algorithm outperforms both Min-Min and Max-Min algorithm.

In Fig.6, which show the average resource utilization rate for 1000(Fig.6(a,b,c)) and 5000 (Fig.6(d,e,f)) tasks respectively, you can see that, again, the proposed algorithm perform like the best algorithm in each assumption. Even, in the third

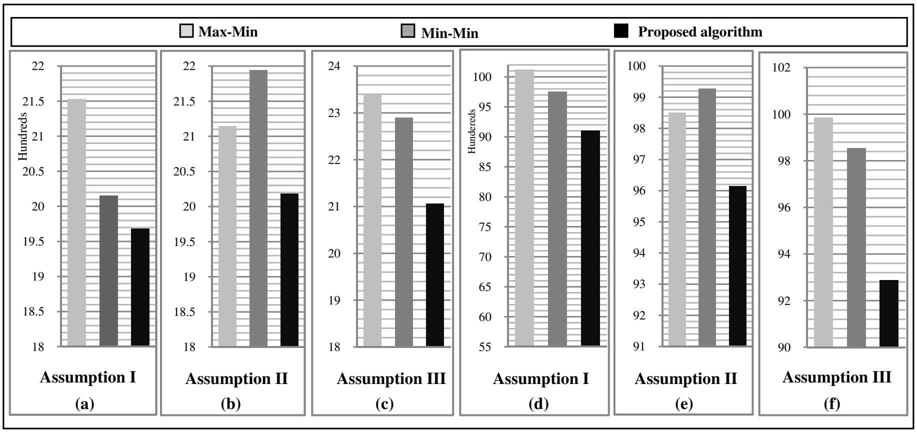


Fig. 5. Makespan for 1000 tasks (light load)

assumption, it acts better than both Min-Min and Max-Min algorithm. Average resource utilization rate is one of the metrics that is used in [16] and the most efficient is achieved if average resource utilization rate equals 1.

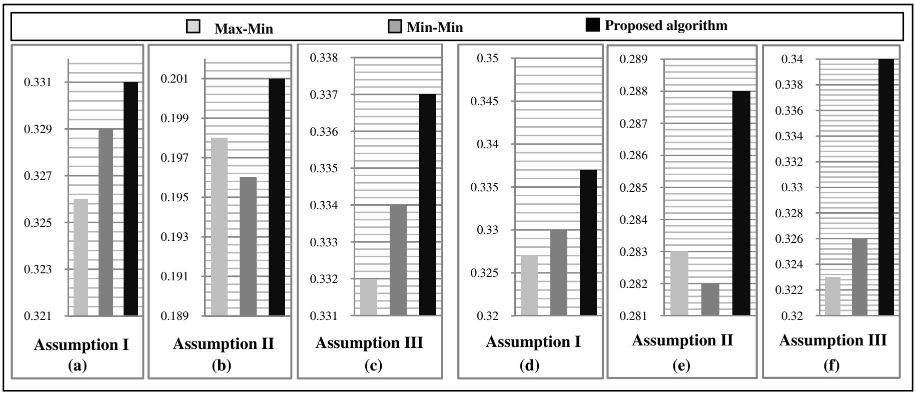


Fig. 6. Average resource utilization rate for 5000 tasks (heavy load)

For load balancing, in Fig.7, for 2000 tasks respectively, the proposed algorithm acts like the best algorithm. The best and most efficient load balancing level is achieved if load balancing equals 1. It is the other metric that is used in [16].

Here, in load balancing level metric, Max-Min has better load balancing level than Min-Min because, Min-Min assigns the task with the earliest completion time in each phase, results in some resources becoming busy all the time and others becoming free most of the time. Therefore, it has less load balancing level than Max-Min where it assigns the task with maximum completion time and lets other tasks executes along on the other resources, therefore have better load balancing level [10].

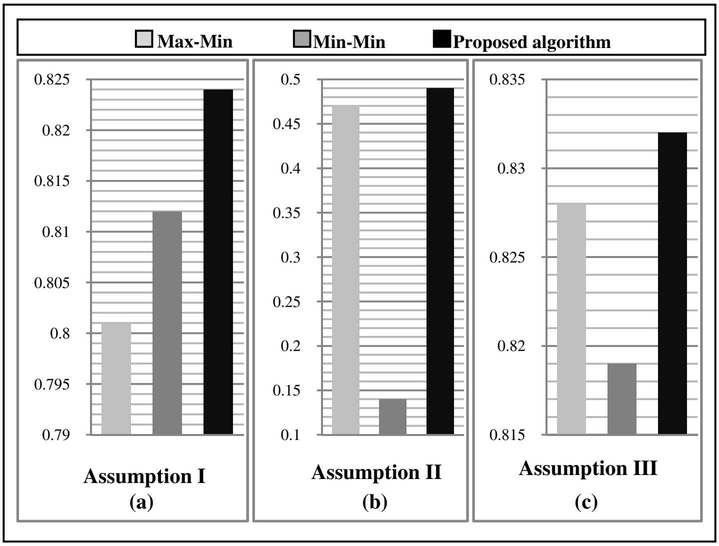


Fig. 7. Load balancing for 2000 tasks and makespan

Finally, in Fig.8, we compared makespan of Min-Min, Max-Min and proposed algorithm with 1000 tasks, when we have 2, 4, 6, 8 and 10 resources. As we see, the proposed algorithm outperforms both Min-Min and Max-Min algorithm and have minimum makespan.

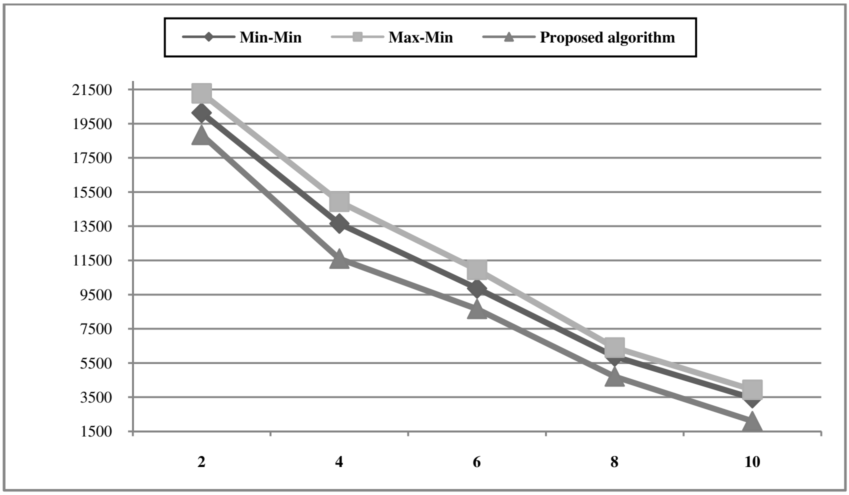


Fig. 8. Makespan of 1000 tasks for 2, 4, 6, 8 and 10 resources respectively

6 Conclusion and Future Works

To overcome the limitations of the Min-Min algorithm, in this paper an improved task scheduling algorithm based on well-known task scheduling algorithm, Min-Min was presented. This algorithm proposed new condition for selection of the task for scheduling. The proposed Algorithm uses the advantages of Min-Min algorithm and covers their disadvantages. The experimental results obtained by applying our algorithm within the GridSim simulator, shows that the proposed algorithm is outperforms better makespan than Min-Min and also helps load balancing . This study concerned task execution time and load balancing. For future works, we can apply other issues like deadlines on tasks and resources.

References

[1] Foster, I.: Globus Toolkit Version 4: Software for service-oriented systems. In: Jin, H., Reed, D., Jiang, W. (eds.) NPC 2005. LNCS, vol. 3779, pp. 2–13. Springer, Heidelberg (2005)

[2] Litzkow, M., Livny, M., Mutka, M.: Condor - A Hunter of Idle Workstations, In: Proceedings of the 8th International Conference of Distributed Computing Systems, pp. 104–111 (June 1988)

[3] Foster, I., Kesselman, C.: The Grid: Blueprint for a future computing Infrastructure. Morgan Kaufmann Publishers, USA (1999)

[4] Yagoubi, B., Slimani, Y.: Task Load Balancing Strategy for Grid Computing. Journal of Computer Science 3(3), 186–194 (2007)

[5] Maheswaran, M., Ali, S., Jay Siegel, H., Hensgen, D., Freund, R.F.: Dynamic Mapping of a Class of Independent Tasks onto Heterogeneous Computing Systems. Journal of Parallel and Distributed Computing 59, 107–131 (1999)

[6] Freund, R.F., Gherrity, M., Ambrosius, S., Campbell, M., Halderman, M., Hensgen, D., Keith, E., Kidd, T., Kussow, M., Lima, J.D., Mirabile, F., Moore, L., Rust, B., Siegel, H.J.: Scheduling Resource in Multi-User, Heterogeneous, Computing Environment with SmartNet. In: The Proceeding of the Seventh Heterogeneous Computing Workshop (1998)

[7] Braun, T.D., Jay Siegel, H., Beck, N., Boloni, L.L., Maheswaran, M., Reuther, A.I., Robertson, J.P., Theys, M.D., Yao, B.: A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems. Journal of Parallel and Distributed Computing 61, 810–837 (2001)

[8] Dong, F., Luo, J., Gao, L., Ge, L.: A Grid Task Scheduling Algorithm Based on QoS Priority Grouping. In: The Proceedings of the Fifth International Conference on Grid and Cooperative Computing (GCC 2006). IEEE (2006)

[9] Parsa, S., Entezari-Maleki, R.: RASA: A New Grid Task Scheduling Algorithm. International Journal of Digital Content Technology and its Applications 3(4) (December 2009)

[10] Etminani, K., Naghibzadeh, M.: A Min-min Max-min Selective Algorithm for Grid Task Scheduling. In: The Third IEEE/IFIP International Conference on Internet, Uzbekistan (2007)

[11] Afzal, A., Stephen McGough, A., Darlington, J.: Capacity planning and scheduling in Grid computing environment. Journal of Future Generation Computer Systems 24, 404–414 (2008)

[12] Brucker, P.: Scheduling Algorithms, 5th edn. Springer Press (2007)

[13] Buyya, R., Murshed, M.: GridSim: A toolkit for the odeling and simulation of distributed resource management and scheduling for grid computing. Journal of Concurrency and Computation Practice and Experience, 1175–1220 (2002)

[14] Benjamin Khoo, B.T., Veeravalli, B., Hung, T., Simon See, C.W.: A multi-dimensional scheduling scheme in a Grid computing environment. Journal of Parallel and Distributed Computing 67, 659–673 (2007)

[15] Czajkowski, K., Foster, I., Karonis, N., Kesselman, C., Martin, S., Smith, W., Tuecke, S.: A resource management architecture for metacomputing systems. In: Feitelson, D.G., Rudolph, L. (eds.) JSSPP 1998. LNCS, vol. 1459, pp. 62–82. Springer, Heidelberg (1998)

[16] Cao, J., Spooner, D.P., Jarvis, S.A., Nudd, G.R.: Grid Load Balancing Using Intelligent Agents. Future Generation Computer Systems 21(1), 135–149 (2005)

[17] He, X., Sun, X.-H., Laszewski, G.V.: QoS Guided Min-min Heuristic for Grid Task Scheduling. Journal of Computer Science and Technology 18, 442–451 (2003)

Heterogeneous Diskless Remote Booting System on Cloud Operating System

Jin-Neng Wu, Yao-Hsing Ko, Kuo-Ming Huang, and Mu-Kai Huang

Cloud Service Technology Center
Industrial Technology Research Institute, Tainan, Taiwan
{LeslieWu,sam_ko,huangkuoming,mkhuang}@itri.org.tw

Abstract. Nowadays, cloud computing has become one of the major issues on the progress of computer science. Applying Diskless Remote Booting (DRB) System to cloud computing has potential to reduce energy consumption and enhance Maintainability. Previous research has introduced homogeneous DRB system which consists of compute nodes with the same hardware and software configuration. However, in the homogeneous DRB system, adding a new node requires the same hardware and software configuration. In this paper, we propose a heterogeneous DRB system that allows compute nodes to have various hardware and software configurations. Moreover, the proposed scheme is equipped with hypervisor to each compute node, so that every compute node provides a virtual environment for its end-users. The experiment results show our approach can run a number of compute nodes with various hardware and software configurations concurrently. Furthermore, the proposed scheme has outstanding benefits to energy saving with negligible performance loss.

1 Introduction

Diskless Remote Booting (DRB) system consists of a number of compute nodes with no disks, and each node boots up its operating system by accessing local disks of sever over network. With diskless technique, it has potential to be adopted in cloud computing to meet the requirements of low maintenance cost and energy consumption. In other words, the power consumption of local disks on a compute node is eliminated. On the other hands, storing all data in a server helps system administrator to maintain system easily. Nowadays, DRB system has been used on real-world scenarios widespreadly, including distance education for off-campus students, system of computer classroom and cybercafe.

Virtualization offers a compute node to run a number of different and concurrent operating systems inside a diskless system. With multitudinous advantages, such as high flexibility, isolation, resource utilizing rate, easy management, power saving, etc., virtualization has become a well-known technique to offer various execution environments from cloud computing vendor [2]. In virtualization system, resource virtualization of hardware and concurrent execution virtual platform are operated by a software called *virtual machine monitor* (VMM) or *hypervisor* [9]. Typically, VMMs

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 114–123, 2013.

are categorized into four types: *Operating system-level virtualization*, *Full-virtualization*, *Para-virtualization* and *Hardware-assisted virtualization*.

Early research puts effort on the methods of building dislkess High Performance Computing (HPC) cluster and virtualization environment [6][11][12][14]. Much work has reported the performance of different types of virtualization, including execution time, kernel compiling time, memory bandwidth, I/O access, network traffic, context switch overhead and throughput [2][3][4][5][8][13]. And few researchers deal with applying diskless HPC cluster and virtualization in real-world scenarios, such as biomedical information and DNA issues, distance education environment and green computing [1][7][10].

Although the DRB system has been widely used on the diskless HPC cluster, it has much potential to be applied in cloud computing to improve energy efficiency and maintainability. The objective of this paper is to apply the DRB system in green cloud computing. By eliminating local disks from compute nodes, energy consumption can be reduced with less performance degradation. Previous research concerns the *Homogeneous* DRB system, i.e., every compute node has the same processor architecture, operating system and softwares [12]. However, it requires that a new added compute node needs the same hardware and software configurations. In this paper, we propose a heterogeneous DRB system, where compute nodes can use variety of different types of hardware, software and operating system. Furthermore, the proposed system encompasses a VMM to provide end-users a number of different virtual machines as a cloud computing platform.

The experiment results show that the proposed scheme runs compute nodes with different hardware and software configuration concurrently. Compared with *diskfull system* (system with local disks), the proposed scheme pays little performance loss for booting up. Furthermore, it reduces 10-25% energy consumption with less performance degradation.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 presents our heterogeneous diskless system. Section 4 provides the detail of implementation of our work. Section 5 shows our experiment results and Section 6 summarizes our conclusion.

2 Related Work

In recent years, studies have investigated methods to construct the DRB system which consists of a number of compute nodes without local disks [6][11][12][14]. T. Victoria and A. V. Nestor Waldyd [12] implemented a homogeneous diskless HPC cluster using Linux as operative system. K. Salah et al. [11] implemented a large-scale Infiniband-based diskless cluster which consists of 126 compute nodes with RedHat Enterprise Linux Server 5.3. C.-T. Yang and Y.-C. Chang [14] built an SMP-based PC cluster with a number of diskless slave nodes on Linux environment. J. H. Laros III and L. H. Ward [6] implemented a diskless cluster using the Network File System (NFS) that scales to thousands of compute nodes.

Virtualization allows compute nodes to run a certain number of different and concurrent operating system instances inside a system. Much research has demonstrated performance evaluation for various virtualization technologies, including operating system-level virtualization, para-virtualization and full- virtualization [2][3][4][5][8][13]. J. Che et al. [2] measured and analyzed the performance of operating system-level virtualization, para-virtualization and full-virtualization, and presented several comparison results, such as execution performance, kernel compiling time, memory bandwidth, I/O access, network traffic and context switch overhead. A. Chierici and R. Veraldi [3] presented the performance comparison of computing, network and I/O access between para-virtualization and full-virtualization. J. S. White and A. W. Pilbeam [13] analyzed the throughput of full-virtualization. M. Fenn et al. [5] showed the performance penalty of full-virtualization on different operating systems. D. Petrovic and A. Schiper [8] investigated the fault-tolerance issue on para-virtualization and full-virtualization. T. Deshane et al. [4] compare the performance isolation and scalability between para-virtualization and full-virtualization.

Several researchers have applied the DRB system and virtualization on real-world scenarios [1][7][10]. S. M. Sait et al. [10] focussed on biomedical information and DNA issues, and evaluated the Basic Local Alignment Search (BLAST) algorithms onto a large Infinibandbased diskless Cluster. L. Liu et al. [7] applied virtualization to reduce data center power consumption. B. R. Anderson et al. created a virtualized lab environment in distance education, and provided off-campus students to utilize the same environment as on-campus students [1].

3 Proposed Scheme

Virtual Machine Monitor (VMM) provides a virtual environment that allows multiple OS images to operate on a computer hardware concurrently. Applying the VMM to the DRB system can run a number of different OS images of end-users on a compute node to reduce the amount of hardware usage and energy consumption. Moreover, it provides an energy efficiency machine with no local disks on a compute node to meet the requirement of energy saving of green cloud computing.

Figure 1 illustrates the concept of the VM-based homogeneous DRB system in cloud computing. In the DRB system configuration, the clone node is a typical server with hardware and operating system, each compute node has the same hardware, which is nearly the same as the hardware of the clone node except with no local disks, and the server node is functioned as Dynamic Host Configuration Protocol (DHCP) server, Trivial File Transfer Protocol (TFTP) server and Network File System (NFS) server. The boot-up procedures are described as follows:

1. The server node acquires the booting configuration and the OS image from the clone node.
2. The server node offers these files to compute nodes so that a compute node can remotely boot up via network connection.
3. Each compute node maintains its own virtual machine structure and provides a cloud computing environment to end-users.

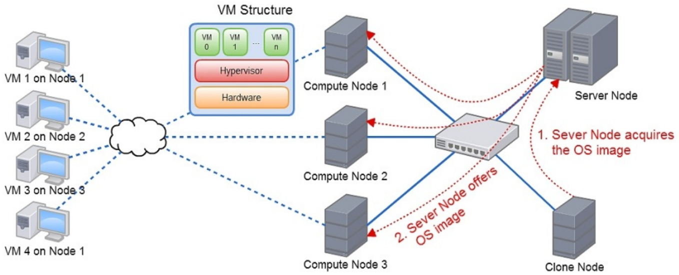


Fig. 1. The illustration of a homogeneous diskless system using the Xen

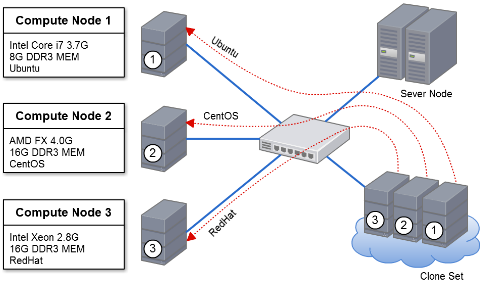


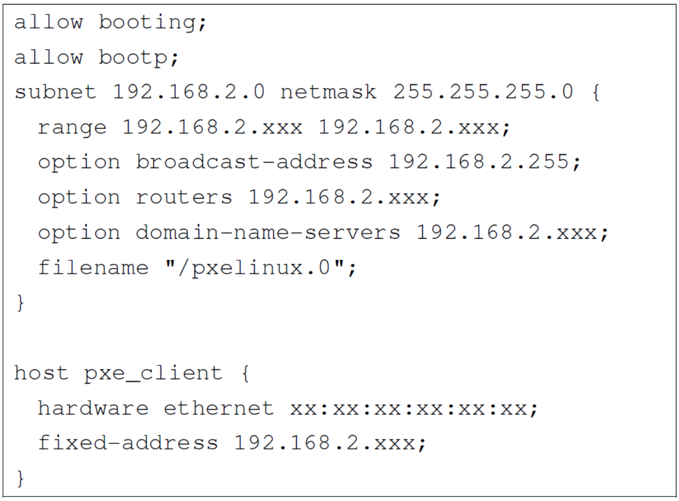
Fig. 2. The illustration of a heterogeneous diskless system

In homogeneous DRB system, each compute node retrieves the same configuration and OS image from the server node, which means each compute node requires the same hardware and software configurations (i.e., the configurations of clone node). It could constrain the hardware expandability and become a problem to system vendor. Hence, we introduce a heterogeneous DRB system that allows compute nodes to use variety of different types of hardware and software configurations. The major change of the heterogeneous technology is to use a set of clone nodes. It offers a number of different OS images and hardware configurations as an image pool. In the other words, compute nodes can retrieve various OS image and hardware configuration from image pool. With the heterogeneous technology, vendor can easily add new compute nodes without using the same configuration.

Figure 2 shows the illustration of the heterogeneous DRB system, where the clone set is composed of three types of clone nodes. As shown in this figure, compute node 1 retrieves the OS image and hardware configuration from the type 1 in the clone set, compute node 2 gets the files from the type 2 in the clone set, and so on. System vendor can add new compute node with different hardware or change hardware device of certain compute node flexibly. Moreover, system vendor can easily assign users into different compute nodes based on its priority (i.e., Gold Member, Silver Member or Free Member).

4 Implementation

To implement our approach, a PXE booting Environment (PXE) is required for compute nodes. In addition, certain tools have to be installed in the server node, including Dynamic Host Configuration Protocol (DHCP), Trivial File Transfer Protocol (TFTP) and Network File System (NFS). To provide cloud computing environment for end-users, Xen hypervisor is applied to each compute node as the VMM. Following describes the detail of implementation.



```
allow booting;
allow bootp;
subnet 192.168.2.0 netmask 255.255.255.0 {
  range 192.168.2.xxx 192.168.2.xxx;
  option broadcast-address 192.168.2.255;
  option routers 192.168.2.xxx;
  option domain-name-servers 192.168.2.xxx;
  filename "/pxelinux.0";
}

host pxe_client {
  hardware ethernet xx:xx:xx:xx:xx:xx;
  fixed-address 192.168.2.xxx;
}
```

Fig. 3. An example of DHCP configuration

PXE booting Environment (PXE) is a technology to boot up system from a network interface, it has been applied to many system architectures, such as Intel IA-64 and DEC Alpha. In our work, each compute node requires a PXE network interface controller to acquire the PXE configuration over network. In Linux system, the PXE configuration is defined in pxelinux.0 file.

Dynamic Host Configuration Protocol (DHCP) configures devices on network so that they can communicate with an IP. With the DHCP protocol, a device retrieves network information, such as IP address, default route and DNS server addresses from the DHCP server. In our work, the server node is configured as the DHCP server. And the configurations, such as network address range, router address, DNS address, and MAC address of compute nodes are defined in /etc/dhcp/dhcpd.conf file. As mention before, our heterogeneous DRB system creates a set of clone nodes, which means, the configuration of a compute node is from one of these clone nodes. Consequently, the DHCP configuration defines fixed IP address to each compute node so that the corresponding configuration and the OS image can be transferred to. Figure 3 gives an example of the DHCP configuration. While a compute node boots up, it sends a DHCP request to the DHCP server, and waits for DHCP response to get its IP address.

Trivial File Transfer Protocol (TFTP) is a technology that transfers files between network devices. Generally, it is widely used for transferring little files, such as configuration files or boot images. UNIX-like OS has initially installed typical ready-to-use TFTP tools, for example, tftpd, tftp-hpa and tftp-server. Hence, the server node can start up the TFTP service immediately with following command:

# chkconfig --level 345 xinetd on --level 345 tftp on During booting process, the compute node gets PXE configuration pxelinux.0, /pxelinux.cfg/default and OS image /kernel/initrd by tftp.

Network File System (NFS) allows a compute node to access files over network connection. After retrieving the PXE configuration and the OS image, the compute node decompresses its OS image, and mounts the kernel on the NFS server. Afterward, a compute node can operate as a diskless system. To support the NFS service for compute nodes, the administrator needs to create NFS share directories in the server node to share with. Then a compute node can access its files in the share directories as in a "virtual" disk. Figure 4 gives the procedures of diskless remote booting.

Xen hypervisor is a para-virtualization VMM that requires modification of virtual operating system to access privileged system calls. Figure 2 shows the VM structure of the Xen. The Xen runs the operating system of compute nodes in Domain 0 (D0), and maintains operating systems of different end-users in VM 1 to VM *n*. The Xen takes a full control on hardware resource and forbids each VM to execute sensitive privileged instructions. Instead, the Xen controls most device drivers in D0, and handles system calls from other VMs, such as CPU execution, memory allocation and I/O access. The Xen offers communication ways between the hypervisor and VMs, those are synchronous call by using hypercall and asynchronous event by using virtual interrupt [2]. In this work, the configurations of D0 are defined in /etc/xen/xend-config.sxp, and the configurations of other VMs are defined in /etc/xen/, such as VM kernel, virtual memory size, virtual CPU count, virtual network interface, etc. After setting VMs for end-users, the VMs can be started up by following command:

```
# xm create -c VM_NAME
```

5 Experiment Results

In this section, we first evaluate the performance of the proposed diskless RDB system, such as file transfer speed and boot-up time. Then, we compare runtime and energy consumption between the proposed DRB system and the system with local disks (following refer as *diskless system* and *diskfull system* respectively). All the experiments were executed on real cloud computing server with hardware configurations as shown in Table 1.

Figure 5(a) shows the comparison of file transfer speed of the diskless system with different compute node counts, where Type A, B and C are the hardware configurations listed in Table 1, and the OS image used in this comparison is CentOS 5.5.

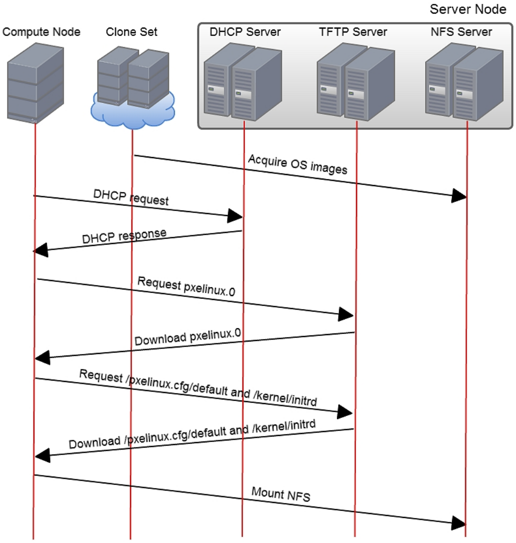


Fig. 4. The procedures of diskless remote booting

Table 1. The Hardware Configurations

Server	Intel Xeon E5620 2.4GHz 49G RAM
Client	A. Intel Core i5 3.2GHz 12G RAM
	B. Intel Core 2 Duo E6750 2.66GHz 4G RAM
	C. Intel Xeon E5620 2.4GHz 49G RAM

The experiment transferred files from the NFS server to compute nodes. As shown in Figure 5(a), with the increasing number of compute nodes, the transfer speed degrades progressively. The is due to the bandwidth contention of network, the more compute nodes, the slower transfer speed.

Figure 6(b) presents the boot-up comparison between the diskfull system and the diskless system. To enhance security, the proposed scheme is equipped with *Security-Enhanced Linux* (SELinux). The result shows that the booting time of the diskless system with SELinux is from 115*s* to 355*s* as compute nodes are increased from 1 node to 10 nodes. And the booting time of the diskless system without SELinux is from 110*s* to 347*s*. Compared with the diskless system without SELinux, the diskless system with SELinux enhances security with little overhead on booting time. The result shows the booting time of the diskfull system from 1 node to 10 nodes are about the same, this is due to each node loads the OS image from its local disks. Although the diskfull system avoids contention of bandwidth, a large number of local disks could cause considerable energy consumption.

Figure 6(a) presents the comparison of runtime between the diskfull system and the diskless system with modern benchmarks, including the boot-up testing, the

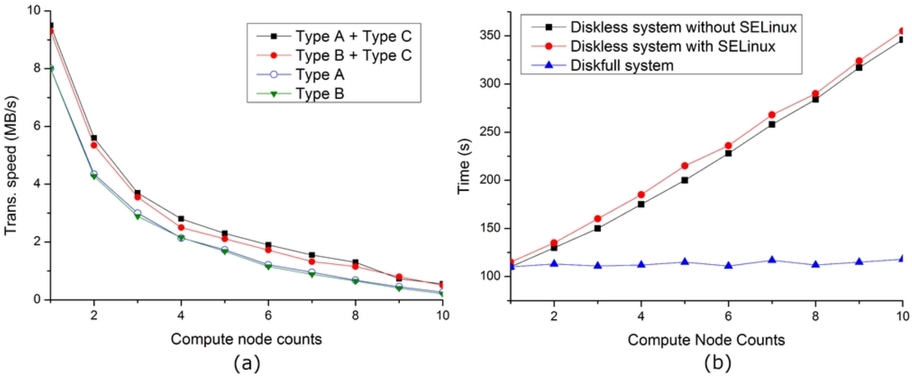


Fig. 5. The comparison of file transfer speed and boot-up

CPU-intensive testings (make and gcc) and I/O-intensive testing (dd). Also, we compared the proposed schemes with Che's work [2] named Gen in Fig. 6. In this comparison, the hardware configuration is type C. For the diskfull system, the OS image is CentOS 5.5. For the diskless system, the OS images are CentOS 5.5, Ubuntu 6.2 and Redhat 9. In the CPU-intensive testings, the runtime of the diskless system is less than that of the diskfull system. However, in I/O-intensive testing, the runtime is dominated by data access of hard drive. For instance, benchmark dd creates a 1.5GB image in hard drive, running it on the diskless system brings a longer runtime due to data is accessed over network. On the other hands, the diskfull system executes the I/O-intensive benchmark on its own local disk and eliminates the overhead of network data transferring. For the boot-up testing, the diskfull system takes about 79 seconds to boot up the system, the boot-up time of the diskless systems are 105 seconds for CentOS, 102 seconds for Ubuntu and 110 seconds for RedHat. The results show that diskless system takes less performance degradation in boot-up. Moreover, the results also show that our schemes have better performance on runtime than that of Gen.

Figure 6(b) demonstrates the comparison of energy consumption between the diskfull system and the diskless system. The results show that the diskless system performs less energy consumption in the CPU-intensive testings and boot-up testing. For I/O-intensive benchmark, the diskless systems have higher energy consumption. This is due to diskless systems take a long time for execution. On the other hand, the proposed schemes have less energy consumption than that of Gen.

In previous work [11], a compute node is equipped with two network interface adapters to handle (1) the communication with NFS server and (2) the communication with end-users. Using a network interface for the communication with the NFS server can avoid compute node losing its file structure in the NFS server while the internet is disconnected unexpectedly. However, it could cause the lack of IP address and increase hardware cost. Our work merges file structure of compute node into the OS image so that compute node stores its own file structure in RAM memory. In this way, compute node can avoid losing file structure without using two network interface adapters. Moreover, the usage of IP address and hardware cost can be reduced.

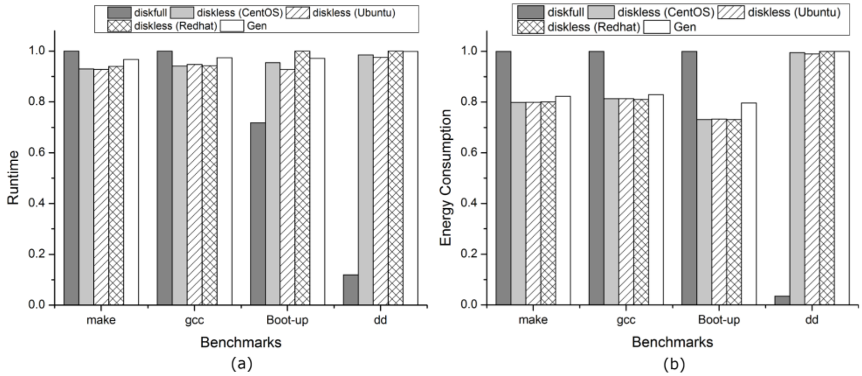


Fig. 6. The comparison of the diskfull system and the diskless systems

6 Conclusion

This work presents a heterogeneous *Diskless Remote Booting* (DRB) system which allows a number of compute nodes with different hardware and software configurations to run concurrently. With heterogeneous DRB system, a compute node with different hardware and software configurations can be joint flexibly. Also, system vender can add/update/remove devices inside a certain compute node without changing other compute nodes. The experiment results show that the proposed scheme reduces energy consumption and enhances system security with little performance degradation, which meets the requirement of green cloud computing. Moreover, comparing with previous work, the hardware cost and the usage of IP address are reduced.

References

1. Anderson, B.R., Joines, A.K., Daniels, T.E.: Xen Worlds: Leveraging Virtualization in Distance Education. In: Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education, pp. 293–297 (2009)
2. Che, J., Yu, Y., Shi, C., Lin, W.: A Synthetical Performance Evaluation of OpenVZ, Xen and KVM. In: Proceedings of the 2010 IEEE Asia-Pacific Services Computing Conference, pp. 587–594 (2010)
3. Chierici, A., Veraldi, R.: A quantitative comparison between xen and kvm. In: The 17th International Conference on Computing in High Energy and Nuclear Physics. Journal of Physics: Conference Series, vol. 219(4) (2010)
4. Deshane, T., Shepherd, Z., Matthews, J.N., Ben-Yehuda, M., Shah, A., Rao, B.: Quantitative Comparison of Xen and KVM. In: Proceedings of the Xen Summit (2008)
5. Fenn, M., Murphy, M.A., Martin, J., Goasguen, S.: An Evaluation of KVM for Use in Cloud Computing. In: Proceedings of the 2nd International Conference on the Virtual Computing Initiative (2008)
6. Laros III, J.H., Ward, L.H.: Implementing Scalable Disk-less Clusters using the Network File System (NFS). In: Proceedings of the 4th Symposium of the Los Alamos Computer Science Institute, pp. 27–29 (2003)

7. Liu, L., Wang, H., Liu, X., Jin, X., He, W., Wang, Q., Chen, Y.: GreenCloud: A New Architecture for Green Data Center. In: Proceedings of the 6th International Conference Industry Session on Autonomic Computing and Communications Industry Session, pp. 29–38 (2009)
8. Petrovic, D., Schiper, A.: Implementing Virtual Machine Replication: A Case Study using Xen and KVM. In: IEEE 26th International Conference on Advanced Information Networking and Applications, pp. 73–80 (2012)
9. Rosenblum, M., Garfinkel, T.: Virtual Machine Monitors: Current Technology and Future Trends. Computer, 39–47 (2005)
10. Sait, S.M., Al-Mulhem, M., Al-Shaikh, R.: Evaluating BLAST Runtime Using NAS-Based High Performance Clusters. In: The 3rd International Conference on Computational Intelligence, Modelling & Simulation, pp. 51–56 (2011)
11. Salah, K., Al-Shaikh, R., Sindi, M.: Towards Green Computing using Diskless High Performance Clusters. In: Proceedings of the 7th International Conference on Network and Services Management, pp. 456–459 (2011)
12. Victoria, T., Nestor Waldyd, A.V.: Diskless HPC cluster for parallel & Grid computing on fedora. In: IEEE Latin-American Conference on Communications, pp. 1–8 (2009)
13. White, J.S., Pilbeam, A.W.: A Survey of Virtualization Technologies With Performance Testing, ArXiv e-prints (2010)
14. Yang, C.-T., Chang, Y.-C.: A Linux PC Cluster with Diskless Slave Nodes for Parallel Computing. In: The 9th Workshop on Compiler Techniques for High-Performance Computing, pp. 81–90 (2003)

RTRM: A Response Time-Based Replica Management Strategy for Cloud Storage System

Xiaohu Bai, Hai Jin, Xiaofei Liao, Xuanhua Shi, and Zhiyuan Shao

Services Computing Technology and System Lab.
Cluster and Grid Computing Lab.
Huazhong University of Science and Technology, Wuhan, 430074, China
hjin@hust.edu.cn

Abstract. Replica management has become a hot research topic in storage systems. This paper presents a dynamic replica management strategy based on response time, named RTRM. RTRM strategy consists of replica creation, replica selection, and replica placement mechanisms. RTRM sets a threshold for response time, if the response time is longer than the threshold, RTRM will increase the number of replicas and create new replica. When a new request comes, RTRM will predict the bandwidth among the replica servers, and make the replica selection accordingly. The replica placement refers to search new replica placement location, and it is a NP-hard problem. Based on graph theory, this paper proposes a reduction algorithm to solve this problem. The simulation results show that RTRM strategy performs better than the five built-in replica management strategies in terms of network utilization and service response time.

Keywords: Dynamic replica management, Response time, OptorSim, Load balance.

1 Introduction

Since data replication has been widely used in storage systems [1-3], replica management has been a hot research topic [4-9]. As the storage environment changes dynamically, dynamic replica management gets more attention by researchers. Replica management includes replica creation, selection, and placement.

Most existing dynamic replica management strategies create new replica of the popular data based on the user access frequency, thus the replica creation always happens at the end of each time interval. But according to temporal locality and spatial locality, especially the pattern of user accesses, the distribution of the user accesses is uneven during the time interval. A file may have many concurrent requests during the time interval, and these concurrent requests will greatly increase the service response time of each single request. Two issues should be addressed: (1) when is the best time for replica creation of popular data to reduce the average service response time; (2) how many replicas can satisfy the response time requirement of a single request.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 124–133, 2013.

In this paper, we focus on the response time of a single request, and propose a response time-based replica management strategy, named RTRM, which includes three algorithms: replica creation, replica selection, and replica placement. Replica creation algorithm decides when and where to create replica based on the average response time. Replica selection method selects the best replica node for users based on response time prediction, while replica placement mechanism combines the number of replicas and the network transfer time. To evaluate the performance of RTRM, we run the strategies in OptorSim [10]. The evaluation results show that our replica management strategy performs better than the five built-in replica management strategies in OptorSim simulator in terms of service response time and network utilization.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 presents dynamic replica management strategy. The analysis and evaluation results are presented in section 4. In section 5, we give conclusions and possible future work.

2 Related Works

Replica management has been widely studied. Sun et al. [4] proposed a replica strategy based on the memory cache. Hou et al. [5] proposed a dynamic replica creation mechanism DynRM, which decides to create replicas according to the file access frequency. Chang et al. [6] set access-weights for each file, and choose hot file based on the value of access-weights. These replica strategies do not take the response time of a single request into consideration, while many requests have to be waiting for a long time.

Rahman et al. [7] proposed a replica placement algorithm used the p-median model to find the locations of p candidate nodes to place replicas, but the problem is how to determine an appropriate value of p. A model-driven replica strategy is proposed in [8]. This strategy first calculates the requisite number of replicas and selects the best set of nodes to host the replicas. However, as each node can only utilize partial information, this strategy may create too many replicas and result in prohibitive overhead. Li et al [9] proposed a DSRL replica location method in which each file has a home node to maintain the index of all the replicas. With the dynamic changes in the network, DSRL method would create too many replicas.

3 Design of RTRM

3.1 Replica Creation Method

In dynamic replica management strategy, replica creation decides which file is the popular data and when is the right time to create new replica of the popular data. Replica creation method first finds the best time to create new replica, an access recorder is assigned to each data node, which is used to store the number of concurrent user accesses to each file, including file name, number of concurrent access, file size, and so on. The service response time of single access can be calculated by the number of concurrent user accesses. Once the average service response time of a file is higher than a threshold, the file becomes popular data, and the creation of that file is started.

In our replica creation method, $T_{threshold}$ is set as the upper limit of the service response time of a single request. The average service response time of a file must be smaller than $T_{threshold}$.

Assume that data block b has n replicas, and distributed in n nodes. Let these n nodes be N_1, N_2, … N_n. To simplify the problem, for the user accesses of data block b, we have the denotations as follows:

The size of data block b is denoted as S_b.

The network transmission capability of node N_i is denoted as NTC_i.

The number of concurrent accesses of node N_i is denoted as Num_i.

The maximum service response time of single request of node S_i is denoted as $MSRT_i$. $MSRT_i$ can be computed by Equation (1).

$$MSRT_i = \frac{S_b}{NTC_i} \times Num_i (i = 1, 2, ..., n) \tag{1}$$

We define $MSRT_{MAX}$ as the maximum value of all $MSRT_i$, the average response time of all $MSRT_i$ is denoted as $MSRT_{average}$. Based on Equation (1), $MSRT_{MAX}$ and $MSRT_{average}$ can be computed by Equation (2).

$$\begin{cases} MSRT_{MAX} = \max(MSRT_1, MSRT_2, ..., MSRT_n) \\ MSRT_{average} = \frac{1}{n}\sum_{i=1}^{n} MSRT_i \end{cases} \tag{2}$$

Each time when a user access comes, we get the value of $MSRT_{MAX}$ and $MSRT_{average}$ through Equation (2). If the value of $MSRT_{average}$ is higher than $T_{threshold}$, file f is considered to be popular data, and new replica of file f will be created. If $MSRT_{average}$ is smaller than $T_{threshold}$, but $MSRT_{MAX}$ is higher than $T_{threshold}$, then the system would transfer some accesses from the relatively heavy load nodes to the relatively light load nodes.

3.2 Replica Selection Method

The goal of replica selection method is to select the best replica node of a file. In replica selection method, *LPC* is defined to represent the load process capability of a node. The metrics of *LPC* consists of three components: CPU process capability, network transmission capability, and I/O capability of disks, denoted by w_c, w_n, w_{io}, respectively. Given these metrics, *LPC* can be computed by Equation (3).

$$LPC = \alpha * w_c + \beta * w_n + \gamma * w_{io} \tag{3}$$

In Equation (3), α, β, γ are constants and can be determined according to service level. Replica selection method chooses the node with highest *LPC* to response the user request, the user then accesses the file from the node with highest *LPC*.

3.3 Replica Placement Mechanism

Replica placement has been proven to be NP-hard. We first give a model of replica placement, and then we propose a reduction algorithm to solve this problem.

Assume that the system has n storage nodes, let them be n_1, n_2, …,n_n. We want to get the minimal replicas of file *f*, and place these replicas to satisfy the requirement of a single request. To simplify the problem, the denotations are as follows:

(1) The replica number is denoted as *replicaDegree*, and the upper limit of the response time of a single request is set as T_{upper}.

(2) The response time that node n_i accesses file *f* is denoted as $responseTime_i$, it is the time that n_i accesses file *f* from the nearest node. If n_i contains file *f* or its replica, $responseTime_i$ is set to be 0.

(3) The total response time of the system is denoted as *TotalresponseTime*, and *TotalresponseTime* can be computed by Equation (4).

$$TotalresponseTime = \sum_{i=1}^{n} responseTime_i \tag{4}$$

The goal of our design is to make sure that the response time of a single request must be smaller than T_{upper}, and minimize the value of *replicaDegree* and the value of *TotalresponseTime*. Therefore, in this paper, we want to find an optimal replica scheme that can achieve the following goals:

(1) Minimize *replicaDegree*
(2) $responseTime_i <= T_{upper}$
(3) Minimize *TotalresponseTime.*

For goals (1) and (2), they can be described as a *Set Covering Problem* (SCP), which has been proven to be NP-hard. Based on greedy algorithm, by transforming the SCP into an equivalent graph, we design a reduction algorithm to figure out this model.

Based on the network topology and the network transfer time, we construct a graph *G=(V, E)*, this graph can be described as:

$V=\{n_1, n_2, \ldots, n_n\}$; $E=\{(n_i, n_j) \mid responseTime_{ji} <= T_{upper}\}$.

As an example, a network topology and the network transfer time is shown in Fig.1, and the value of T_{upper} in this example is 10s.

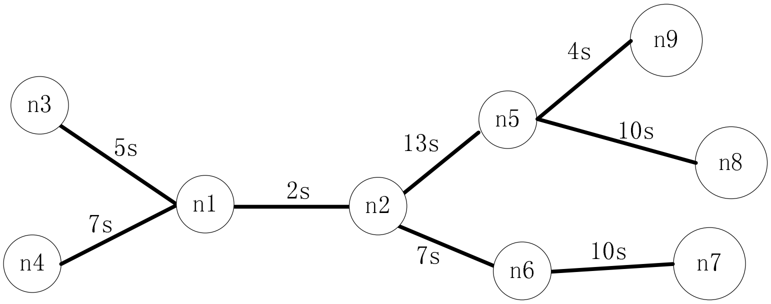


Fig. 1. Network topology and network transfer time

From the graph, we can get the value of V and E.

$V=\{n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9\}$; $E=\{(n_1, n_2), (n_1, n_3), (n_1, n_4), (n_1, n_6), (n_2, n_3), (n_2, n_4), (n_2, n_6), (n_5, n_9), (n_5, n_8), (n_6, n_7)\}$.

The goal is to find a subset V^*, which is a smallest subset of V, for each element v from V, there must have at least one element v^* from V^*, and (v, v^*) is an element in E. It means that for each node v in V, there must be at least one node v^* in V^*, and v can access file from v^* within T_{upper}.

Algorithm 1 shows the process of the reduction algorithm. We can place the replicas in the nodes from V^* to make sure that all the nodes can access file f within T_{upper}.

Algorithm 1. Reduction algorithm

```
INPUT: $G = (V, E)$; OUTPUT: $V^*$
// degree(v) gets the degree of v in G;
1.  Begin
2.      Initialize $V^*$ and $v^*$: $V^* = \emptyset$, $degree(v^*) = 0$;
3.      if ( $V == \emptyset$ ) {go to 18;}
4.      else {go to 5;}
5.      for ( each element v in V )
6.          if( $degree(v) > degree(v^*)$ ) { $v^* = v$;}
7.          push $v^*$ into $V^*$;
8.          delete all the edges incident to $v^*$ from V;
9.          delete $v^*$ from V;
10.     end for
11.     if ( $V == \emptyset$ ) {go to 18;}
12.     else {go to 13;}
13.     for ( each element v in V )
14.         if ( $(v^*, v) \subseteq E$ )
15.         {if(degree(v) == 0) { delete v from V;}}
16.     end for
17.     go to 3.
18.     return $V^*$;
19. End
```

4 Performance Evaluation

In this section, we first compare our replica placement mechanism with other four replica placement strategies, then compare RTRM strategy with the five built-in replica strategies in OptorSim. From the experiment results, RTRM strategy performs better in terms of network utilization, average response time, and total replica number.

4.1 Analysis of Replica Placement Mechanism

We will compare our replica placement mechanism with other four strategies: *Best Client*, *MinimizeExpectedUtil*, *MaximizeTimeDiffUtil*, and *MinimizeMaxRisk*.

The example in Fig. 1 is used in the analysis. The upper limit of the response time of a single request T_{upper} is set to 10s. We define *replicaDegree* to represent the number of replicas in the system, and use *TotalresponseTime* to represent the total response time of all nodes in the system. We perform two analyses. In the first analysis, we compare the value of *TotalresponseTime* of the five mechanisms with the same *replicaDegree*. In second analysis, we compare the smallest *replicaDegree* of the five mechanisms while making sure the response time of all requests is smaller than T_{upper}.

First Analysis

Because in general storage systems, the smallest replica degree is 3, we set the value of *replicaDegree* of all the five mechanisms 3, and access the file from each node, then compare the *TotalresponseTim*e of each mechanism. Result is in Table 1.

Table 1. Results of first analysis

Mechanism	*TotalresponseTime*	Nodes to host replica
RTRM	40	n_2, n_5, n_6
Best Client	77	n_2, n_3, n_4
MinimizeExpectedUtil	48	n_1, n_2, n_5
MaximizeTimeDiffUtil	52	n_1, n_2, n_9
MinimizeMaxRisk	69	n_2, n_3, n_7

From the first analysis, we can observe that with the same replicas, our replica placement mechanism performs best, and has the smallest *TotalresponseTime*.

Second Analysis

As smaller replica degree means less cost of management, we compare the smallest *replicaDegree* of each mechanism to make sure that the response time of a single request is smaller than T_{upper}. The result is shown in Table 2.

Table 2. Results of second analysis

Mechanism	*replicaDegree*	Nodes to host replica
RTRM	3	n_2, n_5, n_6
Best Client	4	n_2, n_3, n_4, n_5
MinimizeExpectedUtil	3	n_1, n_2, n_5
MaximizeTimeDiffUtil	4	n_1, n_2, n_6, n_9
MinimizeMaxRisk	4	n_2, n_3, n_5, n_7

From the second analysis, we can see that our replica placement mechanism has the smallest *replciaDgree*. *MinimizeExpectedUtil* also has smallest *replicaDegree*, but its *TotalresponseTime* is bigger.

4.2 Simulation of Dynamic Replica Management Strategy

OptorSim is a scalable, configurable and programmable simulation tool for grid. It has five built-in replica management strategies. We compare our RTRM strategy with the five built-in replica strategies in OptorSim, and give the performance analysis. The simulation grid topology is shown in Fig. 2.

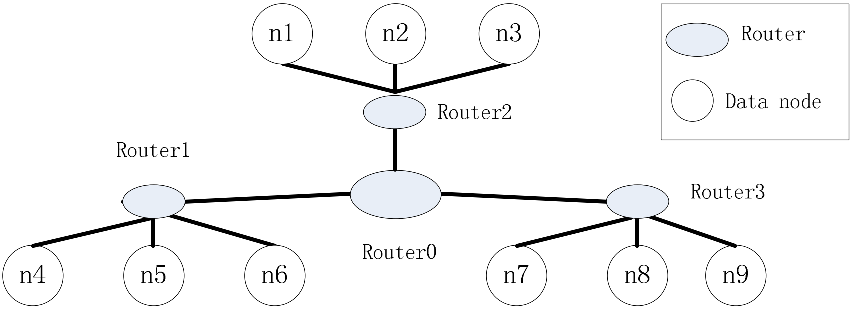


Fig. 2. The grid topology of simulation experiment

The simulation experiments are performed on a server machine, and the hardware and the software environment of the server machine is shown in Table 3.

Table 3. Environment of server machine

CPU	Quad-Core Intel Xeon 1.6GHz processors
Memory	4GB DDRII RAM
Hard Disk	320GB SATA II hard drive 7200RPM (ST3500418AS)
OS	64-bit CentOS 5.6 with Linux 2.6.18.8 kernel
OptorSim	OptorSim Release V 2.0.0

The simulation parameter configuration of the grid in our experiments is shown in Table 4.

Table 4. The configuration of simulation parameters

Parameters	value
Number of jobs	1000
Scheduler	File access cost + job queue access cost
optimizer	SimpleOptimiser LruOptimiser EcoModelOptimiserZipf DynamicOptimiser
Job delay	40000
Init file distribution	n_1, n_4, n_7
Max queue size	200

Fig. 3 shows the average job time of the six replica management strategies under three user access modes. In sequence mode, RTRM strategy is second best. In the random mode, RTRM strategy performs not so well. While in the Zipf distribution mode, our strategy performs best among all strategies.

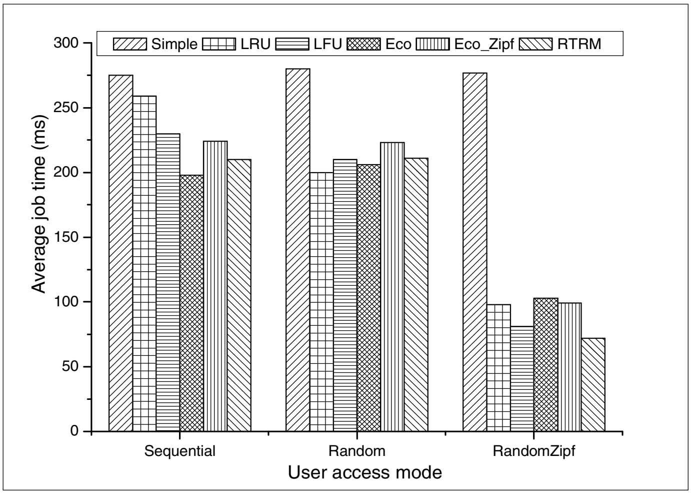


Fig. 3. Average job time

Fig. 4 shows the network utilization of the six replica management strategies under three user access modes. From the result, in any mode, RTRM strategy performs the best among the six strategies. This is because RTRM strategy takes the response time of a single request into consideration, making sure that the response time of any node smaller than T_{upper}.

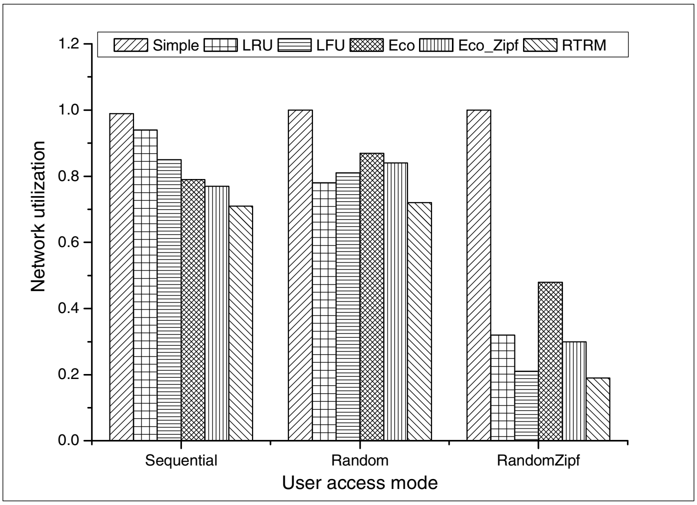


Fig. 4. Network utilization

Table 5 shows the number of total replicas of the six replica management strategies under three user access modes. Because the simple strategy has no replicas, the number of replicas of simple in Table 5 is always 0. From the table, we can see that the number of replica in RTRM strategy is far less than other five strategies in each access model. This is because we apply the reduction algorithm in the replica placement, and find the relatively better nodes to host the replicas for all the nodes in the system. Make sure the average service time is smaller than the threshold.

Table 5. Number of total replicas

	Sequential	Random	Random_Zipf
Simple	0	0	0
LRU	8851	6982	3583
LFU	6573	6751	3026
Eco	205	225	112
Eco_Zipf	425	512	374
RTRM	43	57	36

Through the analysis of simulation results, it can be deduced that RTRM strategy is very suitable for user access mode which follows Zipf distribution. The Zipf distribution means that user's access to file is coherent to time, which is very popular in the file sharing application of distributed storage system.

5 Conclusion and Future Work

Taking the response time of single request into consideration, we propose a response time-based replica management strategy referred to as RTRM, and it consists of replica creation method that can automatically increase the number of replicas based on the average response time. When a new request comes, RTRM will predict the bandwidth among the replica servers, and make the replica selection accordingly, and replica placement mechanism combing with the number of replicas and the network transfer time. In addition, we implement our dynamic replica management strategy in OptorSim. Through extensive simulations, we show that RTRM strategy behaves much better than the five built-in replica management strategies in OptorSim in terms of the network utilization and the service response time.

Finally, due to the limitation of OptorSim, the performance advantage of our replica selection method does not fully revealed in the simulation, but we believe that our replica selection method could achieve good performance and low response time, and provide rapid data download. In the future, we plan to apply our response time-based replica management strategy in HDFS [3], PVFS [11], pNFS [12], Gpfs [13], and LusterFS [14].

Acknowledgments. This paper is supported by National 863 High Technology Research and Development Program under grant No.2012AA010905 and National Science & Technology Pillar Program under grant No.2012BAH14F02.

References

1. Ghemawat, S., Gobioff, H., Leung, S.T.: The Google File System. In: Proceedings of 19th ACM Symposium on Operating Systems Principles, pp. 29–43. ACM Press, New York (2003)
2. Sage, A.W., Scott, A.B., Ethan, L.M., Darrell, D.E.L., Carlos, M.: Ceph: A Scalable, High-Performance Distributed File System. In: Proceedings of 7th Conference on Operating System Design and Implementation (OSDI 2006), pp. 307–320. USENIX Press, Seattle (2006)
3. The Apache Software Foundation, Hadoop, `http://hadoop.apache.org/`
4. Sun, H., Wang, X., Zhou, B., Jia, Y., Wang, H., Zou, P.: The Storage Alliance Based Double-Layer Dynamic Replica Creation Strategy-SADDRES. Chinese Journal of Electronics 33(7), 1222–1226 (2003)
5. Hou, M.S., Wang, X.B., Lu, X.L.: A Novel Dynamic Replication Management Mechanism. Compute Science 33(9), 50–52 (2006)
6. Chang, R.S., Chang, H.P.: A Dynamic Data Replication Strategy Using Access-Weights in Data Grids. Journal of Supercomputing 45, 277–295 (2008)
7. Rahman, R.M., Barker, K., Alhajj, R.: Replica Placement in Data Grid: Considering Utility and Risk. In: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC 2005), pp. 354–359. IEEE Press, Las Vegas (2005)
8. Ranganathan, K., Iamnitchi, A., Foster, I.: Improving Data Availability through Dynamic Model-Driven Replication in Large Peer-to-Peer Communities. In: Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 376–381. IEEE/ACM, Berlin, Germany (2002)
9. Li, D., Xiao, N., Lu, X., Wang, Y., Lu, K.: Dynamic self-adaptive replica location method in data grids. Journal of Computer Research and Development 40(12), 1775–1780 (2003)
10. Bell, W.H., Cameron, D.G., Millar, A.P., Capozza, L., Stockinger, K., Zini, F.: OptorSim-A Grid Simulator for Studying Dynamic Data Replication Strategies. International Journal of High Performance Computing Applications 17(4), 403–416 (2003)
11. Ross, R.B., Rajeev, T.: Pvfs: A parallel file system for linux clusters. In: Proceedings of the 4th Annual Linux Showcase and Conference, pp. 391–430. USENIX Press, Atlanta (2000)
12. Hildebrand, D., Ward, L., Honeyman, P.: Large files, small writes, and pnfs. In: Proceedings of the 20th ACM International Conference on Supercomputing, pp. 116–124. ACM Press, New York (2006)
13. Schmuck, F., Haskin, R.: Gpfs: A shared-disk file system for large computing clusters. In: Proceedings of the First USENIX Conference on File and Storage Technologies, pp. 231–244. USENIX Press, Berkeley (2002)
14. Lustre: A scalable, High-performance File System, `http://www.lustre.ort/docs/lustre.pdf`

Secure Hadoop with Encrypted HDFS

Seonyoung Park and Youngseok Lee

Chungnam National University, Daejeon, Republic of Korea, 305-764
{siraman,lee}@cnu.ac.kr

Abstract. As Hadoop becomes a popular distributed programming framework for processing large data on its distributed file system (HDFS), demands for secure computing and file storage grow quickly. However, the current Hadoop does not support encryption of storing HDFS blocks, which is a fundamental solution for secure Hadoop. Therefore, we propose a secure Hadoop architecture by adding encryption and decryption functions in HDFS. We have implemented secure HDFS by adding the AES encrypt/decrypt class to `CompressionCodec` in Hadoop. From experiments with a small Hadoop testbed, we have shown that the representative MapReduce job on encrypted HDFS generates affordable computation overhead less than 7%.

Keywords: Hadoop, HDFS, Security, Encryption, Decryption, Cryptography.

1 Introduction

Apache Hadoop [1], that originated from Google's MapReduce and GFS [2, 3], has been recently popularized due to its scalable distributed computing framework and file system, because it enables a big data processing platform for many data-intensive applications and analytics. Hadoop is an open-source distributed computing framework implemented in Java, and provides the MapReduce programming model and the Hadoop Distributed File System (HDFS). MapReduce allows users to harness thousands of commodity machines effectively in parallel for processing massive data in the distributed system by simply defining map and reduce functions.

Since Hadoop is usually used in a large cluster or a public cloud service such as Amazon Elastic MapReduce where multiple users run their jobs at the same time, it is essential to provide the security of user data on HDFS. However, the security service in the current Hadoop project is at the early design stage [4] that the simple file permission and access control mechanisms are employed. Particularly, encryption is the key means for the secure HDFS where many datanodes store files to HDFS and transfer user files among datanodes while executing MapReduce jobs. It is reported that a future Hadoop software release will include encryption [5]. For the secure HDFS, a few studies assume that encryption is applied to HDFS [6-9]. However, the native encryption modules for Hadoop have not been fully implemented and tested.

In general, encrypted file systems are widely deployed in various Operating Systems (OSes) such as MS Windows, Linux, MacOS and FreeBSD, and it is known that encrypted file system does not perform well because of the high CPU utilization of

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 134–141, 2013.

encryption or decryption processes. However, recent CPU architectures equipped with multi-cores and special encryption accelerators can perform better than expectation. On the other hand, due to the development of CUDA [10] and OpenCL, GPUs can run general-purpose programs by augmenting the computing power with many parallel processing units. Recent studies [11-15] begin to take advantage of GPUs for developing network systems such as routers and distributed file system by utilizing high degrees of parallelism of GPUs and saving CPU computing capacity. Yet, the current data center generally does not deploy the CUDA-capable GPUs to the cluster, because a cluster node is equipped with the popular hardware devices at the low cost. In addition, GPUs usually consume more energy than CPUs so that they cannot be adopted by a large-scale data center where the most serious problem is the energy-efficient computing infrastructure. This means that we cannot directly utilize the GPU capability for enhancing computing power in the commodity data center.

Therefore, in this paper, we propose a secure HDFS architecture that is compatible with the current Hadoop applications and show its performance results on the Hadoop cluster testbed. From the experiments with an AES encryption algorithm [16], we present that the secure HDFS causes the computation overhead only less than 7% for the representative MapReduce jobs.

The remainder of this paper is organized as follows. In Section 2, we describe the related work. Our proposal for the secure HDFS is explained in Section 3, and its experimental results are presented in Section 4. Finally Section 5 concludes this paper.

2 Related Work

As Hadoop becomes the main framework for the cloud computing service, a few studies [4, 6-9] have presented secure HDFS methods. In [4], a secure HDFS architecture has been proposed such that Kerberos over SSL is used for strong mutual authentication and access control to enhance HDFS's security. Tahoe [6], a prototype of using SSL and integrating an encrypted distributed file system with Hadoop, has been presented, but its write speed is 10 times slower and its read speed is about the same with the generic HDFS. In [7], an application-level encryption MapReduce, that assumes the pre-uploaded plaintext to HDFS, was proposed to support the file system. In [8], hybrid encryption of HDFS was proposed with HDFS-RSA and HDFS-Pairing. However, both read and write performance of encrypted HDFS is lower than those of the generic HDFS. In the write case, the encrypted HDFS is slower by 2 times. In [9], Hybrid cloud, where sensitive data is stored at private storage cluster and the remainder of data is transferred to public or partner storage cluster, was proposed. For more security, sensitive data can be encrypted using trusted platform module (TPM).

GPUs have been also applied to distributed storage systems and Hadoop. In [11], a GPU-based library that accelerates hashing-based primitives for distributed storage system has been presented. In [14], a framework for integrating GPU computing into storage systems has been proposed, and it has been prototyped in the Linux kernel. The AES cipher powered by GPUs is reported to achieve 4 GBps, whereas the results with CPUs are less than half of GPU's performance in [14]. Shredder [15] uses GPUs for incremental storage and computation in Hadoop by mitigating the CPU bottlenecks of content-based chunking. Generally, GPU-based approaches put emphasis on performance enhancement, but they do not reveal the energy efficiency and consider form factors that are currently deployed by commodity servers in a data center. Nowadays, dedicated encryption accelerators and CPUs with special encryption features such as Intel AES New Instructions (NI) [17] have been developed.

3 Secure Hadoop

3.1 Overview

HDFS consists of a master (namenode) and multiple slaves (datanodes). In HDFS, a file is chunked by a block with the fixed size (64 MB by default). The namenode manages the file system metadata and controls access to files from clients by maintaining the mapping between datanodes and blocks for a file. Each block belonging to a file is replicated three by default in HDFS. Hadoop provides a MapReduce programming framework that runs multiple tasks for a job. A MapReduce job divides its job into multiple maps or reduce tasks to process many HDFS blocks in parallel. HDFS is well suited with a write-once-read-many access model.

We assume that every file is encrypted and decrypted before it is written and read in the secure HDFS. In addition, we presume that each datanode is a commodity server that will encrypt and decrypt files with CPUs. Clients' requests to read or write a file in HDFS will trigger decryption or encryption functions to HDFS blocks at each datanode. We use 128-bit AES which is one of the most popular block cipher algorithm and suitable for handling HDFS blocks. There are a few modes of operation for AES: ECB, CBC, OFB, CFB, CTR and XTS [16]. We choose AES ECB, because its computation can be concurrently performed in a distributed computing environment. AES CBC is the most commonly used mode, but it is not suitable for HDFS cluster consisting of many nodes because HDFS blocks must be processed sequentially on one slave node.

3.2 Encrypting Files in HDFS

As shown in Fig. 1, following the same procedure of the file write operation in HDFS, a HDFS client splits a file by a fixed size, encrypts every block and saves it to HDFS. In HDFS, the encryption function itself can be easily implemented by writing an encryption Java class in the same way that the `CompressionCodec` is used for compressing and uncompressing files [5]. Based on `CompressionCodec`, we have devised an AES encryption module (AESCodec) that executes the encryption algorithm on the CPU. The HDFS client runs AESCodec class to perform encryption and to pass the encrypted HDFS blocks to a datanode. Then, the first datanode, that receives the encrypted HDFS blocks from the client, will stream the encrypted blocks to other datanodes for replication. In contrast to decryption, encryption of a file is performed at a HDFS client node, because the file write procedure in HDFS is sequentially carried out by a client. In this work, we have implemented only the AES function as the encryption algorithm. However, other encryption algorithms such as RSA or DES can be simply extended.

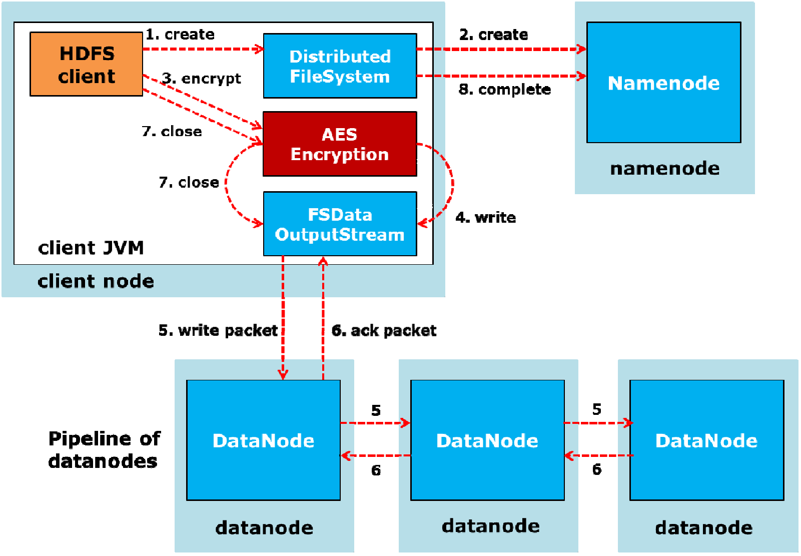


Fig. 1. Writing a file by adding an encryption step in HDFS

3.3 Decrypting Files in HDFS

With our own AESCodec, reading an encrypted file in HDFS is performed by multiple HDFS datanodes in parallel. That is, every block is processed by a map task at the HDFS datanode. Thus, assuming the same file read operation in HDFS, as shown in Fig. 2, we have added the decryption step with AESCodec when a task tracker launches a map task that reads a block. In general, multiple map tasks are executed by a Hadoop worker up to the number of available map task slots which is usually constrained by the number of CPU cores. Since HDFS assumes the write-once-read-many model, our concurrent decryption per-HDFS block architecture fits well for various MapReduce jobs.

4 Performance Evaluation

4.1 Experimental Environment

For the performance evaluation of encrypted HDFS, we have established a small Hadoop testbed consisting of a master node and three worker nodes. Each node has 8- core 2.83 GHz CPU, 4 GB memory, and 2 TB hard disk. All Hadoop nodes are connected with1 Gigabit Ethernet cards.

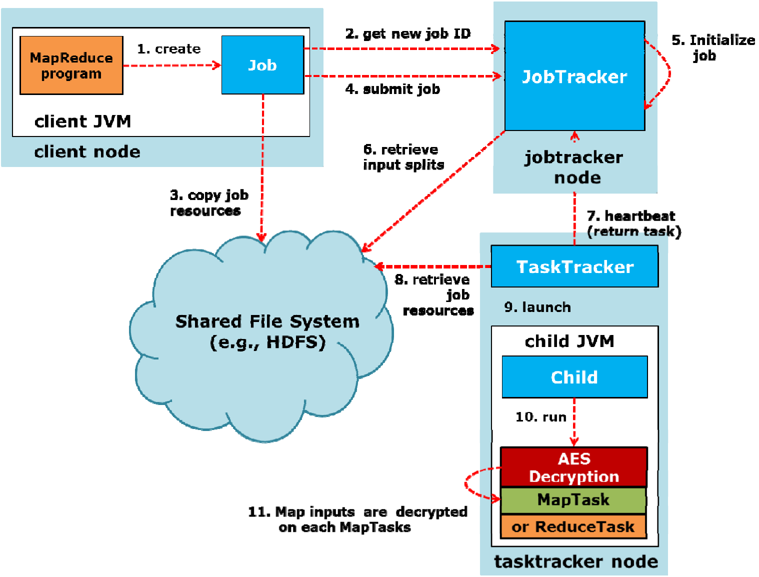


Fig. 2. A MapReduce job that read an encrypted file in HDFS

4.2 File Write: Generic HDFS vs. Encrypted HDFS

First, we have compared the file write time between generic HDFS and encrypted HDFS. As shown in Fig. 3, it took 430 minutes to write a 1 TB unencrypted file to HDFS, whereas 585 minutes to encrypt a file in HDFS, which is 36% performance degradation. The throughput of writing files in HDFS is 41 MB/s for the generic HDFS, while 30 MB/s for the AES encrypted HDFS. In the encryption phase, the HDFS client is in charge of encrypting the whole files, which is a bottleneck of uploading files to HDFS.

4.3 File Read and Computation of MapReduce Jobs: Generic HDFS vs. Encrypted HDFS

In order to evaluate the usefulness of encrypted HDFS, we have considered MapReduce jobs on multiple HDFS datanodes that read and compute encrypted files in HDFS. Thus, we have run a representative WordCount MapReduce job on the testbed that processes encrypted files on HDFS by decrypting files with AESCodec. Fig. 4 shows the performance of MapReduce jobs on unencrypted or encrypted HDFS. We can observe that 604 minutes was taken for running a WordCount MapReduce job for unencrypted HDFS for 1 TB file tests, while 635 minutes for the encrypted HDFS. In the read case, the overhead of decrypting files in HDFS is less than or equal to 5% for almost all cases except 7% for 128 GB. In contrast to the write case, the decrypt/read operation on encrypted files is executed in parallel, which mitigates the performance degradation, and it fits well for the write-once-read-many model.

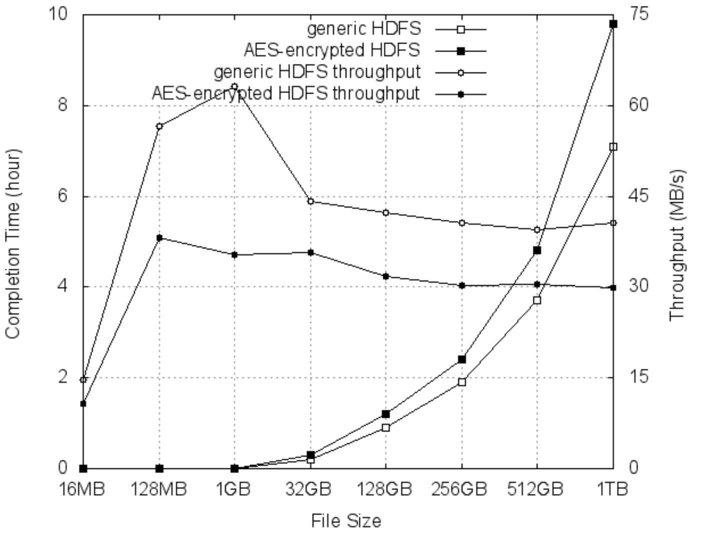


Fig. 3. File write performance under different file sizes: generic HDFS vs. AES-encrypted HDFS

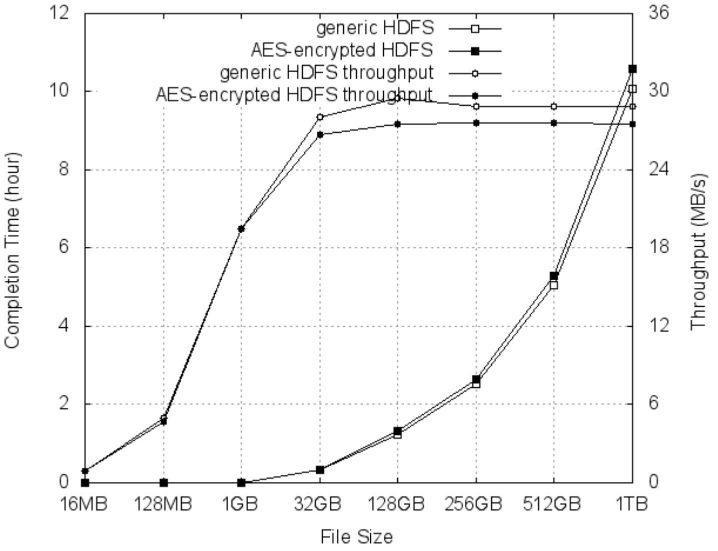


Fig. 4. Performance of MapReduce jobs under different file sizes: generic HDFS vs. AES-encrypted HDFS

5 Conclusion

Since generic Hadoop lacks in secure file management, it is necessary to be upgraded with encryption in HDFS. Though encryption is the essential file protection method, its real implementation has not been fully examined. In this paper, we presented a secure HDFS by adding encryption and decryption function as a built-in encryption/decryption class in Hadoop. Based on `CompressionCodec`, we have implemented AESCodec into Hadoop and shown that it is useful for securing MapReduce job in HDFS with marginal performance degradation less than 7%.

Acknowledgment. This research was supported by the Basic Science Research Program through the NRF, funded by the Ministry of Education, Science and Technology (NRF 2010-0021087). The corresponding author of this paper is Youngseok Lee.

References

1. Hadoop, `http://hadoop.apache.org/`
2. Dean, J., Ghemawat, S.: MapReduce: Simplified Data Processing on Large Cluster. In: OSDI (2004)
3. Ghemawat, S., Gobioff, H., Leung, S.: The Google File System. In: ACM Symposium on Operating Systems Principles (October 2003)
4. O'Malley, O., Zhang, K., Radia, S., Marti, R., Harrell, C.: Hadoop Security Design, Technical Report (October 2009)
5. White, T.: Hadoop: The Definitive Guide, 1st edn. O'Reilly Media (2009)
6. Cordova, A.: MapReduce over Tahoe. Hadoop World (2009)
7. Majors, J.H.: Secdoop: a confidentiality service on Hadoop clusters. Auburn University Master Thesis (May 2011)
8. Lin, H., Seh, S., Tzeng, W., Lin, B.P.: Toward Data Confidentiality via Integrating Hybrid Encryption Schemes and Hadoop Distributed FileSystem. In: IEEE AINA (2012)
9. Yang, Y., Wu, Z., Yang, X., Zhang, L., Yu, X., Lao, Z., Wang, D., Long, M.: SAPSC: Security Architecture of Private Storage Cloud Based on HDFS. In: Proceedings of 26th IEEE Workshops of International Conference on Advanced Information Networking and Applications (2012)
10. NVIDIA CUDA Programming Guide, `http://developer.download.nvidia.com/compute/DevZone/docs/html/C/doc/CUDA_C_ProgrammingGuide.pdf`
11. Al-Kiswany, S., Gharaibeh, A., Santos-Neto, E., Yuan, G., Ripeanu, M.: StoreGPU: exploiting graphics processing units to accelerate distributed storage systems. In: ACM HPDC (2008)
12. Han, S., Jang, K., Park, K., Moon, S.: PacketShader: A GPU accelerated Software Router. In: Proceedings of the ACM SIGCOMM (2010)
13. Jang, K., Han, S., Han, S., Moon, S., Park, K.: SSLShader: Cheap SSL Acceleration with Commodity Processors. In: Proceedings of NSDI (2011)
14. Sun, W., Ricci, R., Curry, M.L.: GPUstore: Harnessing GPU Computing for Storage Systems in the OS Kernel. In: ACM SYSTOR (June 2012)

15. Bhatotia, P., Rodrigues, R., Verma, A.: Shredder: GPU-Accelerated Incremental Storage and Computation. In: USENIX FAST (February 2012)
16. Advanced Encryption Standard, http://en.wikipedia.org/wiki/Advanced_Encryption_Standard
17. Intel, http://software.intel.com/en-us/articles/intel-advanced-encryption-standard-instructions-aes-ni

VM Migration for Fault Tolerance in Spot Instance Based Cloud Computing

Daeyong Jung[1], SungHo Chin[2], Kwang Sik Chung[3], and HeonChang Yu[1,*]

[1] Dept. of Computer Science Education, Korea University, Seoul, Korea
[2] Software Platform Laboratory, CTO Division, LG Electronics, Seoul, Korea
[3] Dept. of Computer Science, Korea National Open University, Seoul, Korea
{karat,yuhc}@korea.ac.kr,
sunghochin@gmail.com, kchung0825@knou.ac.kr

Abstract. The cloud computing is a computing paradigm that users can rent computing resources from service providers as much as they require. A spot instance in cloud computing helps a user to utilize resources with less expensive cost, even if it is unreliable. When a user performs tasks with unreliable spot instances, failures inevitably lead to the delay of task completion time and cause a seriously deterioration in the QoS of users. To solve the problem, we propose the VM migration scheme to reduce the job waiting time. And in this scheme we use our previously proposed checkpointing method. When a running instance occurs the out-of-bid situation (failure), the VM on the failed instance is to a new instance. Our proposed VM migration scheme reduces the rollback time and the task waiting time when an instance occur the out-of-bid situation. The simulation results show that our scheme achieves performance improvements in the task execution time of 68.94%, 68.61%, and 46.35% compared with the hour-boundary checkpointing scheme, the rising edge-driven checkpointing scheme, and our previously proposed checkpointing scheme., respectively Further, our scheme outperforms the existing schemes in terms of the reduction the total costs per spot instances for a user's bid.

Keywords: Cloud computing, Spot instances, VM migration, Price history, Fault tolerance.

1 Introduction

Recently, due to increased interests for cloud computing many cloud projects and commercial systems such as Amazon EC2 [1], GoGrid [2], FlexiScale [3], have been implemented. Cloud computing is a computing paradigm that constitutes an advanced computing environment that evolved from utility and grid computing. In addition, cloud computing involves a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provided and presented as one or more unified computing resources based on service level

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 142–151, 2013.

agreements established through negotiation between service providers and consumers [4]. Typically, Cloud computing provides high utilization and high flexibility for managing computing resources. And, cloud computing services provide a high level of scalability of computing resources combined with Internet technology to multiple customers [5]. In the most of these cloud services, the concept of an instance unit is used to provide users with resources in cost-efficient way. Generally instances are classified into two types: on-demand instances and spot instances. On-demand Instances allow the user to pay for computing capacity by the hour with no long-term commitments. This frees users from the costs and complexities of planning, purchasing, and maintaining hardware and transforms what are commonly large fixed costs into much smaller variable costs [1]. On the other hand, spot instances allow customers to bid on unused Amazon EC2 capacity and run those instances for as long as their bid exceeds the current spot price. The price for spot instance changes periodically based on supply and demand, and customers whose bids meet or exceed it gain access to the available spot instances. If you have time flexibility for executing applications, spot instances can significantly decrease your Amazon EC2 costs [6]. For task completion, therefore, spot instances may incur lower cost than on-demand instances.

Spot market-based cloud environment configures the spot instance. This environment changes spot prices depending on the user's supply and demand. The environment affects the successful completion or failure of tasks in accordance with the changing of spot prices. Spot price has market structure, law of demand and supply. Therefore, cloud service (Amazon EC2) can provide a spot instance when a user's bid is higher than current spot price. And, a running instance stops when a user's bid becomes less than or equal to the current spot price. After a running instance stops, the running instance restarts when a user's bid is greater than the current spot price [7, 8].

Therefore, we solve the problem that the performed task is failed according to the current spot price. In previous study, we propose VM migration scheme using checkpointing [9]. Our proposed checkpointing scheme basically performs a checkpointing operation based on two kinds of threshold: price and time. These two thresholds are extracted from the price history of spot instances and are used to determine the checkpointing time in the presence of failures of spot instances arising from price fluctuation in a cost-efficient way. Using this checkpointing scheme, cloud system is able to minimize loss of task and rollback time since rollback is shorter than that of existing checkpointing schemes. However, if spot price is higher than user's bid, an instance is suspended with checkpointing. And, the instance makes a task waiting time until a task restarts. As a consequence, in this paper, we propose the VM migration scheme using checkpointing to solve task waiting time problem. However, intuitively our scheme makes an additional VM migration time VM from current instance to new instance and has to reduce total execution time than without VM migration.

Lastly, we carry out simulations to demonstrate effectiveness of our scheme. Simulation results show that our scheme outperforms the existing schemes, such as hour-boundary checkpointing, rising edge-driven checkpointing, and our previous checkpointing, in terms of reduction of both total costs and total task execution time per spot instance for a user's bid.

The rest of this paper is organized as follows: Section 2 briefly describes related work on checkpoint and migration in cloud computing. Section 3 presents our system architecture and its components. Section 4 presents our checkpoint and VM migration algorithms based on the price history of spot instances. Section 5 presents performance evaluations with simulations. Lastly, Section 6 concludes the paper.

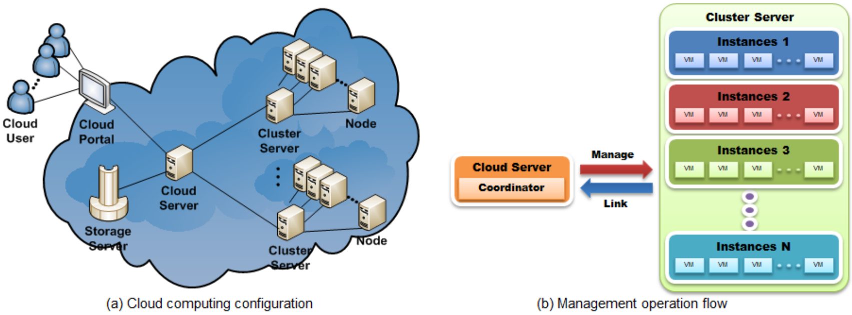
2 Related Work

Many researchers and companies have recently studied two different types of environment in cloud computing: reliable cloud computing environments, such as on-demand instances [7], and unreliable cloud computing environments, such as spot instances [8]. [7] has addressed acquiring on-demand or reserved instances. Focus of our research is the unreliable cloud computing environment. The cost of unreliable cloud computing environment (spot instances) is less than that in reliable cloud computing environment (on-demand instances) for task completion. However, a spot instance takes a longer task completion time than on-demand instance, because a running instance occurs out-of-bid situations (failure) when user's bid exceeds the spot price. Out-of-bid situations may make a task waiting time that is not task execution in instance. To solve this problem, existing researches have focused on studies on the resource allocation [10, 11] and fault tolerance [7, 8, 10].

Voorsluys et al. [10] proposed a resource allocation scheme and resource provisioning policy. Zhang et al. [11] introduced the question of how best to match customer demand in terms of both supply and price in order to maximize the provider's revenue and the customer's satisfaction in terms of VM scheduling delay. [10] and [11] focus on a resource allocation scheme to achieve higher revenues and a reduced task waiting time. There are various fault tolerance methods. [7, 8] introduce a checkpoint method to improve reliability of task. Based on the actual price history of EC2 spot instances, the authors compared several adaptive checkpointing schemes in terms of monetary costs and the improvement in job completion time. Other studies compared the performances of schemes based on fault tolerances in spot instances [8, 10]. Goiri et al. [7] evaluated three fault tolerances scheme, checkpointing, migration, and job duplication, assuming that the communication cost is fixed. [10] analyzed various types of schemes using spot instances. However, previous papers focus on reliability and do not consider a total task execution time to perform the entire operation. Only, papers focus on increment of reliability of task and reduction of total cost. Therefore, our paper focuses on decrement of a total task execution time and proposes the VM migration scheme using a checkpointing.

3 System Architecture

Fig. 1 shows the cloud computing environment assumed in this paper. Fig. 1(a) shows the cloud computing structure. This cloud computing structure basically consists of four entities: a cloud server, a storage server, cluster servers, and cloud users. The cloud server is connected to cluster servers and storage servers. The cluster server consists of a lot of nodes. The cloud users can access the cloud server via the cloud portal to utilize the nodes in the cluster servers as resources. Therefore, the cloud server takes responsibility for finding virtual resources to satisfy the user's requirements, such as SLA and QoS. The coordinator in the cloud server manages tasks and VM, and is responsible for the SLA management. Fig. 1(b) shows the management operation flow. A cluster server consists of nodes of multiple instance types. To configure a cluster server, each node creates VM depending on each instance type and manages a creation of VM. A cloud user accesses a cloud portal to select the type of spot instance of the cloud server. And they use a VM in a selected instance. In the cloud server, a coordinator manages history of multiple spot instances to meet the requirements of cloud users and monitors to migrate from a failed instance to a new instance. In addition, each VM node takes a checkpointing and determines the VM migration. We focus on the coordinator and the VM, which play an important role in our checkpointing and VM migration scheme.



(a) Cloud computing configuration (b) Management operation flow

Fig. 1. Cloud computing environment

3.1 VM Migration Scheme Using SLA-Based Checkpointing

In this section, we propose the VM migration scheme using SLA-Based checkpointing in the spot instances. We introduce our proposed scheme and then represent the proposed processing and algorithm using VM migration scheme.

Spot instance environment assume that VM can be performs on the same instance type until task completion. However, if the environment uses one instance type, a running VM stops when spot price is higher than the user’s bid. To solve the problem, we propose VM migration to continue job execution in spite of out-of-bid situations. And a running VM migrates from current instance type to new instance type. We use our previous proposed checkpointing scheme for VM migration scheme. Fig. 2 shows our proposed VM migration scheme to add previous proposed checkpointing scheme. Our VM migration scheme investigates an instance type to migrate the VM from current instance type to other instance type when execution time on current spot price meets time threshold. And, an available instance type selects to consider a user’s bid and a remaining execution time. The data of selected instance type use kind of two.

Frist, if spot price using an instance exceeds the user's bid, VM migrates a selected instance type. And the VM restarts from the last checkpoint. Second, if next spot price of an instance is lower than the user's bid, the coordinator deletes the stored information of before selected instance and obtains new information to select new instance.

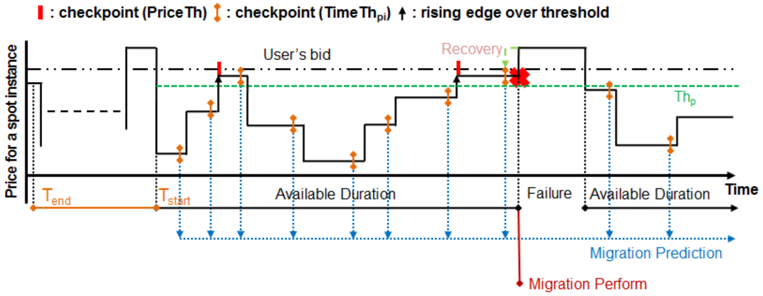


Fig. 2. Our proposed VM migration scheme

We explain our previous proposed checkpointing scheme. This scheme basically performs a checkpointing operation using two kinds of threshold, price and time, based on the expected execution time according to the price history. Now, let t_{start} and t_{end} denote, a start point and an end point, respectively, in the total of ETs. Based on t_{start} and t_{end}, we obtain price threshold ($PriceTh$) and time threshold ($TimeTh_{p_i}$), which are used as thresholds in our proposed checkpointing scheme.

The price threshold, $PriceTh$, can be calculated by eq. 1

$$PriceTh = \frac{PriceMin(t_{start}, t_{end}) + User_{bid}}{2} \tag{1}$$

where $User_{bid}$ represents the bid suggested by a user and $PriceMin(t_{start}, t_{end})$ represents an available minimum price in a period between t_{start} and t_{end}.

The time threshold of price P_i, $TimeTh_{p_i}$, can be calculated by eq. 2

$$TimeTh_{p_i} = AvgTime_{P_i}(t_{start}, t_{end}) \times (1 - F_{p_i}) \tag{2}$$

where F_{p_i} is a failure probability of price P_i and $AvgTime_{P_i}(t_{start}, t_{end})$ represents an average execution time of P_i in a period between t_{start} and t_{end}.

Using these two thresholds, our proposed checkpointing scheme performs checkpoint operations in two cases. The first case is that a checkpoint is taken when there is a rising edge between users bid and the price threshold; and the second case is based on the failure probability and average execution time of each price. A checkpoint is taken when the time threshold exceeds execution time of current price. And a migration prediction module is performed with taking a checkpoint when time threshold is calculated. The migration is performed when out-of-bid occurs in a running instance.

3.2 Efficient Checkpoint and VM Migration Scheme Algorithm

In this section, we introduce a proposed efficient checkpoint and VM migration algorithm. Fig. 3 and 4 show two kinds of checkpointing scheme and VM migration method. Two checkpointing points have price threshold and time threshold. Using these two thresholds, our proposed checkpointing scheme takes checkpoint in the two cases: first case is that a checkpoint is taken when there is a rising edge between a user's bid and price threshold. Second case is based on failure probability and an average execution time of each price. A checkpoint is taken when time threshold exceeds the execution time of current price. The migration method has a migration prediction, a migration execution, and predicted information of migration.

```
 1: Boolean F_flag = false          // a flag representing occurrence of a task failure
 2: Boolean M_flag = false     // a flag representing occurrence of the prediction
    information of a VM Migration position
 3: while (!task execution finishes) do
 4:     if (spot prices < User's bid ) then
 5:          if (F_flag) then
 6:                   VM Migration ( );
 7:               Recovery ( );
 8:               flag = false;
 9:          end if
10:        if (!F_flag) then
11:              if (rising edge && Price Threshold ≤ spot prices) then
12:                   Checkpoint ( );
13:              end if
14:              if (Time Threshold < execution time in current price) then
15:                   Checkpoint ( );
16:                        VM Migration Position Prediction ( );
17:             end if
18:        end if
19:     end if
20:     if (failure is occurred) then
21:        F_flag = true;
22:     end if
23: end while
```

Fig. 3. Checkpointing with VM migration and recovery algorithms

Fig. 3 shows the checkpointing and recovery algorithms with VM migration used in our proposed scheme. In these algorithms, the flag for representing an occurrence of a task failure is initially set to false. The checkpointing process repeats until all tasks are completed. Line 1 and line 2 show the flag information of task failure and prediction, respectively. When task execution is normal (i.e., the flag is false), the scheduler performs checkpoint process to provide against a job failure (lines 3-23). Recovery process is performed when the flag is true (lines 5-9). Two cases to take checkpoints are performed (lines 10-18). If a rising spot price is between user's bid and price threshold, the scheduler performs checkpointing operation (lines 11-13). If execution time is greater than time threshold, the scheduler also performs an operation of checkpointing and VM migration position prediction (lines 14-17). When a task failure event occurs, the flag is set to true to invoke the recovery function (lines 20-22).

```
 1: Function Checkpoint   ( )
 2:     take a checkpoint on the spot instance;
 3:     send the checkpoint to the storage;
 4: end Function
 5: Function Recovery ( )
 6:     retrieve  the checkpoint information from the storage;
 7:     restart the job execution;
 8: end Function
 9: Function VM Migration Position Prediction ( );
10:     if (M_flag) then
11:         delete the before prediction information of a VM migration Position;
12:     end if
13:     M_flag = true;
14:     calculate VM Position for the VM migration;
15: end Function
16: Function VM Migration ( );
17:     migrate the current VM position to the calculated VM position;
18:     M_flag = false;
19: end Function
```

Fig. 4. Algorithms for the operation of VM migration, checkpointing, and recovery

Fig. 4 shows the algorithms for the operation function of VM migration, checkpointing, and recovery. Lines 1-4 and 5-8 show detailed process of checkpointing and recovery, respectively. Lines 9-19 show the migration process. Line 9-15 and 16-19 show detailed process of VM migration position prediction and VM migration, respectively.

4 Performance Evaluation

In this section, we evaluate the performance of our checkpointing scheme with VM migration scheme using simulation and compare it with that of the other checkpointing schemes using VM migration. Our simulation are conducted using the history data obtained from the Amazon EC2's spot instances [12], which is accumulated during a period from 9-27-2010 to 10-4-2010. The history data before 10-01-2010 are used to extract expected execution time and failure occurrence probability for our checkpointing scheme. The applicability of our checkpointing scheme is tested using the history data after 10-1-2010, which are also used for hour-boundary checkpointing and rising edge-driven checkpointing schemes. In our simulations, one type of spot instances are applied to show an effect of analyses on the performance of three checkpointing schemes that is a user's bid.

Table 1 shows a various resource types used in amazon EC2. In this table, resource types show a number of different instance types and pot price information

(Max, Min, and Average) in each instance type. First, Standard Instances offer a basic resource type. Second, High-CPU Instances offer more compute units than other resources and can be used for compute-intensive applications. Finally, High-Memory Instances offer more memory capacity than other resources and can be used for high-throughput applications, including database and memory caching applications. Under the simulation environments, we compare the performance of our checkpointing scheme with that of two checkpointing schemes in terms of various analyses according to the user's bid. The information is measured during a period from 2009-11-30 to 2011-01-23. As the table result, if the user's bid sets lower, the High-Memory instance did not perform task. Our simulation set the user's bid from $0.31 to $0.34. This setting creates a test environment for migration. Table 2 shows the simulation parameters and values used for the analysis of computing type instances.

Table 1. Spot price (Max, Min, and Avg) information in nstance tpye

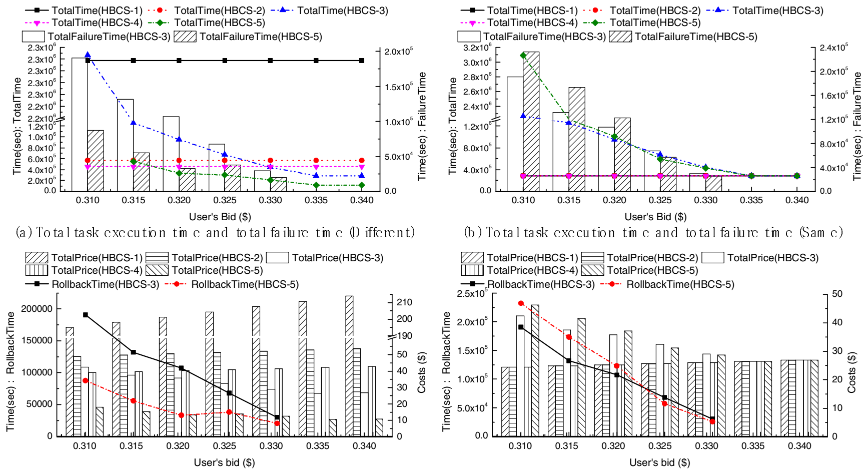
No	Instance type name	Compute unit	Spot price Max	Spot price Min	Spot price Avg
1	m1.small (Standard Instances)	1 EC2	$0.053	$0.038	$0.04
2	m1.large (Standard Instances)	4 EC2	$0.168	$0.152	$0.16
3	m1.xlarge (Standard Instances)	8 EC2	$0.336	$0.304	$0.32
4	c1.medium (High-CPU Instance)	5 EC2	$0.084	$0.076	$0.08
5	c1.xlarge (High-CPU Instance)	20 EC2	$1.52	$0.304	$0.323
6	m2.xlarge (High-Memory Instance)	6.5 EC2	$0.588	$0.532	$0.561
7	m2.2xlarge (High-Memory Instance)	13 EC2	$0.588	$0.532	$0.561
8	m2.4xlarge (High-Memory Instance)	26 EC2	$1.176	$1.064	$1.122

Table 2. Simulation parameters and values for instances

Simulation parameter	Users bid interval	Baseline	Task time	Migration time	Checkpoint time	Recovery time
Value	$0.005	m1.xlarge	259200(s)	300(s)	300(s)	300(s)

Fig. 5 shows the simulation results about hour boundary checkpointing scheme (HBCS). Same task time in our simulation means that we do not consider the performance condition of each instance (Same) and different task time means that we consider performance condition of each instance (Different). In fig. 6 the HBCS-Number is a type number of instance. In the case of same task time, the result of experiment shows that user's bid determines total execution time and total costs. Therefore, Rising edge-driven checkpointing scheme (RECS) and our previous checkpointing scheme (PCS) simulate a different task time to reflect the performance of each instance.

Fig. 6 shows the performance comparison of proposed scheme (PCS+M: PCS with VM migration scheme) with HBCS, RECS, and PCS. Various simulations set standard instances type (m1.xlarge) for performance comparison. However, P+M method uses all instance types. In fig. 6(a), PCS+M achieves performance improvements in an average task execution time of 69.94%, 69.61% and 46.35% over HBCS, RECS, and PCS, respectively. In fig. 6(c), PCS+M achieves performance improvements in terms of an average rollback time of 77.94% over HBCS and 74.76% over RECS and performance reduction in terms of the average rollback time of -7.93% over PCS. And, PCS+M reduces the cost by an average of 36.91%, 36.86%, and 1.71% over HBCS, RECS, and PCS, respectively.



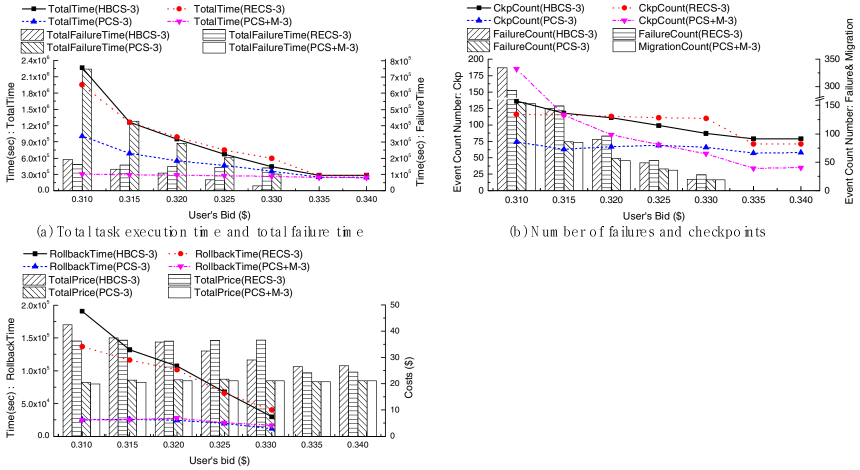
(a) Total task execution time and total failure time (Different)

(b) Total task execution time and total failure time (Same)

(c) Total costs and rollback time (Different)

(d) Total costs and rollback time (Same)

Fig. 5. Simulation result about HBCS



(a) Total task execution time and total failure time

(b) Number of failures and checkpoints

(c) Total costs and rollback time

Fig. 6. Comparison of PCS+M and checkpointing schemes (HBCS, RECS, and PCS)

5 Conclusion

In this paper, we proposed VM migration scheme with our previous proposed checkpointing scheme in order to reduce rollback time in unreliable cloud computing environment. Our previous proposed checkpoint scheme takes a checkpointing based on two kinds of thresholds: price and time. When a execution time is higher than the time threshold, VM migration predictor decides predicted migration time and checkpoint is taken. The VM migration is performed when out-of-bid occurs in a running instance. Our scheme removes a failure time and has additional migration time. The rollback time of our scheme can be lesser than that of the existing checkpointing schemes (HBCS, RECS, and PCS) because our scheme adaptively performs migration operation according to the time threshold of each spot price. The simulation results show that our scheme achieves performance improvements in the task execution time of 68.94%, 68.61%, and 46.35% compared with HBCS, RECS, and PCS. Further, our scheme reduces the cost by an average of 36.91%, 36.86%, and 1.71% over HBCS, RECS, and PCS, respectively. In the future, we plan to expand our environment with task scheduling and more efficient prediction method.

Acknowledgments. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea goverment (MEST) (No. 2012046684).

References

1. Elastic Compute Cloud, EC2 (2012), `http://aws.amazon.com/ec2`
2. GoGrid (2012), `http://www.gogrid.com`
3. FlexiScale (2012), `http://www.flexiscale.com`
4. Buyya, R., Chee Shin, Y., Venugopal, S.: Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. In: Proceeding of the 10th IEEE International Conference on HPCC, pp. 5–13 (2008)
5. Van, H.N., Tran, F.D., Menaud, J.-M.: SLA-Aware Virtual Resource Management for Cloud Infrastructures. In: Proceedings of the 2009 Ninth IEEE International Conference on Computer and Information Technology, vol. 2, pp. 357–362. IEEE Computer Society (2009)
6. Amazon EC2 spot Instances (2012), `http://aws.amazon.com/ec2/spot-instances/`
7. Singer, I., Livenson, M., Dumas, S.N.: Srirama, and U. Norbisrath.: owards a model for cloud computing cost estimation with reserved resources. In: CloudComp 2010, Barcelona, Spain. Springer (October 2010)
8. Yi, S., Kondo, D., Andrzejak, A.: Reducing Costs of Spot Instances via Checkpointing in the Amazon Elastic Compute Cloud. In: Proceedings of the 2010 IEEE 3rd International Conference on Cloud Computing, pp. 236–243. IEEE Computer Society (2010)
9. Jung, D., Chin, S., Chung, K., Yu, H., Gil, J.: An Efficient Checkpointing Scheme Using Price History of Spot Instances in Cloud Computing Environment. In: Altman, E., Shi, W. (eds.) NPC 2011. LNCS, vol. 6985, pp. 185–200. Springer, Heidelberg (2011)
10. Voorsluys, W., Buyya, R.: Reliable Provisioning of Spot Instances for Compute-intensive Applications. In: IEEE 26th International Conference on Advanced Information Networking and Applications (2012)
11. Zhang, Q., Gürses, E., Boutaba, R., Xiao, J.: Dynamic resource allocation for spot markets in clouds. In: Hot-ICE 2011, pp. 1–6 (2011)
12. Cloud exchange (2011), `http://cloudexchange.org`

A Cloud Based Natural Disaster Management System

Mansura Habiba and Shamim Akhter

American International University, Bangladesh (AIUB)
Dhaka, Bangladesh
mansura.habiba@gmail.com, shamimakhter@aiub.edu

Abstract. Natural disaster management needs to deal with large amount of data originated from various organizations and mass people. Therefore, a scalable environment provided with flexible information access, easy communication and real time collaboration from all types of computing devices, including mobile handheld devices, such as smart phones, PDAs and iPads are essential. It is mandatory that the system must be accessible, scalable, and transparent from location, migration and resources. In this paper a framework has been proposed in order to design a Cloud based workflow management system along with scheduler for natural disaster management system, where in Cloud environment, web service and EC2 technologies have been leveraged in order to design the Cloud based workflow model for disaster management system.

1 Introduction

The recent progress in virtualization technologies and the rapid growth of Cloud computing services have opened a new opportunity for complex scientific workflow such as Disaster Management System (DMS) [1, 3]. Cloud services such as Amazon EC2, Google Cloud services etc. can provide reliability, scalability and interactive platform to increase the performance [5].Therefore in this paper we have described a Cloud based implementation of Disaster management system [2]. Cloud can provide a large amount of computing power over short periods of time during a disaster - so several government agencies as well as NGOs and other associations like Red Cross, UNO etc. can respond more efficiently to anything in the world during disaster [4]. Moreover, in case of disaster people never knows when it might have a spike for a need in compute power or disk storage. In such case, Cloud platform allows resources to be used on an elastic basis. In addition, Cloud platform drives down costs by sharing resources and being more communal, it allows quicker communicating response to emergencies and disasters to be more agile. Furthermore, for massive information sharing among government and other agencies in such situation, Cloud computing environment can be the most helpful [3]. Finally, Cloud computing allows for rapid scaling when needed, it allows for significant flexibility and reduces cost tremendously. Therefore, most experts agree that when it comes to information technology, and especially a complex, uncertain and dynamic system likes disaster management, Cloud computing is the best way to go. In this paper, the design of a complete Cloud based DMS is described along with its different functionalities.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 152–161, 2013.

2 Related Work

Several ICT based DMS have been implemented already [1,10]. An ICT based DMS should be well designed to deal with all four stages of the life cycle of DMS [2] such as (1) planning, (2) emergency response, (3) recovery and (4) post-planning. However, most of existing DMS model mainly deal with first two stages. For example, Emergency response system [1] and warning system for mass people [10] are two significant example of existing DMS model which mainly design a warning system for people, vehicle and transport. On the contrary, the proposed system architecture of a DMS in this paper deals with all four stages along with following contributions

1. Improve the intelligent system of traditional DMS and make easier the decision making process
2. Increases the efficiency of the system through task distribution among different services
3. Simulate the performance and compare that with traditional system to evaluate the efficiency

3 Proposed Disaster Management Workflow Management System

In this paper, the implementation structure of a DMS has been proposed based on Cloud environment for following reasons

1. Huge data can be computed easily and quickly.
2. During natural disaster ICT infrastructure also get damaged, such as some servers of IEEE damages due to recent violent flood sandy [5]. However as the data will be stored in Cloud, those will have replicated back up.
3. The computation and decision making process for DMS is too complex and need apparently large amount of time. However, distributed environment in Cloud has expedite decision making process.
4. For data storage, Cloud is superior in providing security, easier sharing and migration, flexible access and rights management. if Cloud is used for implementing DMS data storage and management can be taken care of by Amazon S3 and Google BigTable [5].
5. Improve discovering different class and characterizes resources.

3.1 Components of Proposed Disaster Management System (DMS)

The proposed DMS consists of six major components such as (1) Web Portal, (2) Role Manager, (3) Workflow Engine, (4) Workflow Scheduler, (5) Workflow Monitor and (6) Notification depicted as in figure 1. Among these components, Workflow Engine (WE) and Workflow Scheduler (WS) are the most important components.

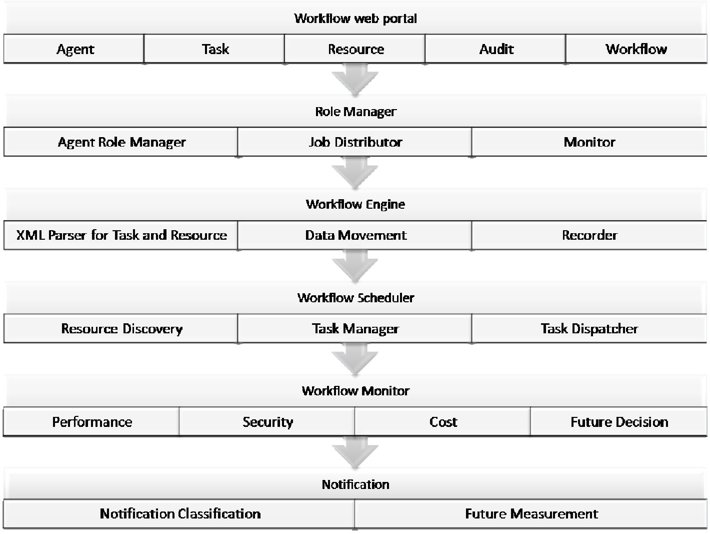


Fig. 1. Components of Proposed DMS Workflow Management System

Web Portal

Web portal provides a Graphical User Interface (GUI) that helps users to edit workflow. Mainly all kind of agent can access different pages of this web portal according to their role. All task management and workflow management can be done through this portal. Although tasks are prepared from collected data and assigned with number of required resources as well as priority by WE. However the portal also supports manual task configuration for dynamic and uncertain problem domain.

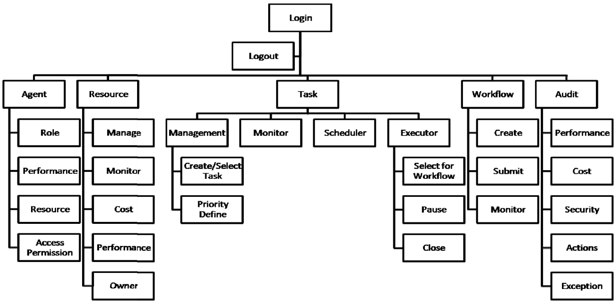
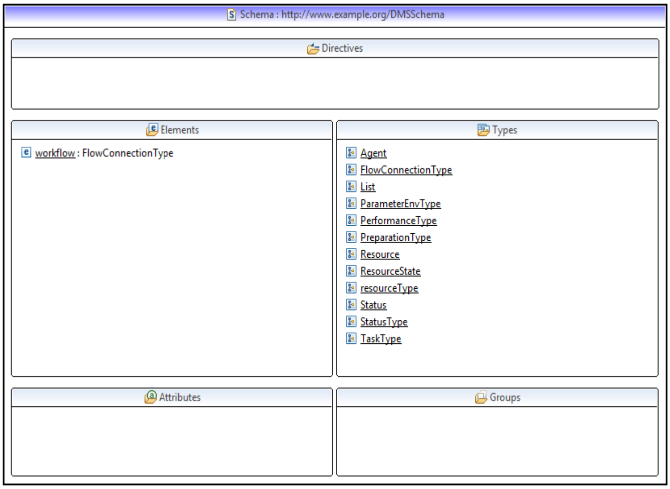


Fig. 2. Web Portal Navigation Design

Workflow Edit

In this work, Workflow is organized as DAG [2, 6], which has been converted to xml schema for implementation. Figure 3 represents the schema of a workflow in the proposed DMS. Each workflow consists of a number of parameters such as Agent, Flow connection, list of resources, environment parameters, performance, assigned resource status, overall workflow status, task status and task. In figure 4, different parameters of a workflow in this proposed DMS have been described.



Schema : http://www.example.org/DMSSchema

Directives

Elements

workflow : FlowConnectionType

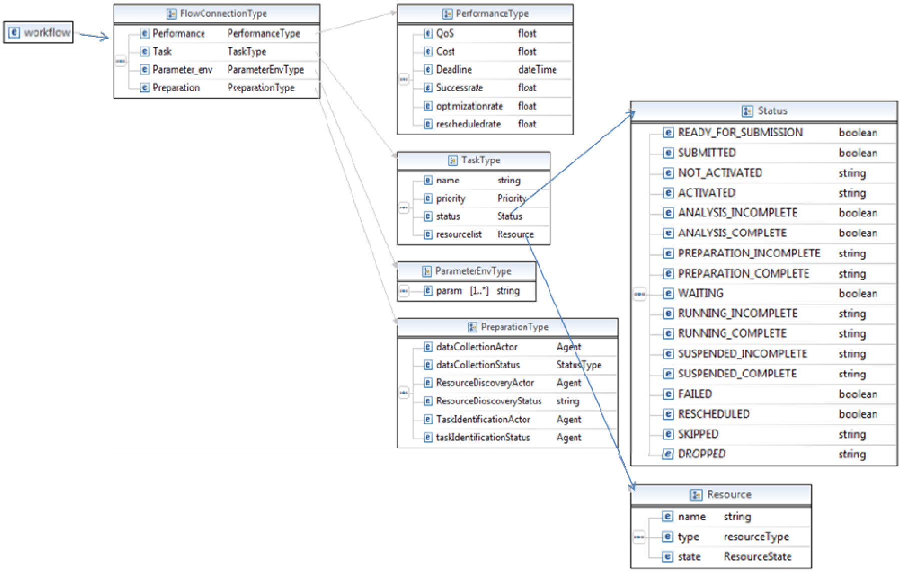
Types

Agent
FlowConnectionType
List
ParameterEnvType
PerformanceType
PreparationType
Resource
ResourceState
resourceType
Status
StatusType
TaskType

Attributes

Groups

Fig. 3. Workflow XML Schema



workflow

FlowConnectionType	
Performance	PerformanceType
Task	TaskType
Parameter_env	ParameterEnvType
Preparation	PreparationType

PerformanceType	
QoS	float
Cost	float
Deadline	dateTime
Successrate	float
optimizationrate	float
rescheduledrate	float

TaskType	
name	string
priority	Priority
status	Status
resourcelist	Resource

ParameterEnvType		
param	[1..*]	string

PreparationType	
dataCollectionActor	Agent
dataCollectionStatus	StatusType
ResourceDiscoveryActor	Agent
ResourceDioscoveryStatus	string
TaskIdentificationActor	Agent
taskIdentificationStatus	Agent

Status	
READY_FOR_SUBMISSION	boolean
SUBMITTED	boolean
NOT_ACTIVATED	string
ACTIVATED	string
ANALYSIS_INCOMPLETE	boolean
ANALYSIS_COMPLETE	boolean
PREPARATION_INCOMPLETE	string
PREPARATION_COMPLETE	string
WAITING	boolean
RUNNING_INCOMPLETE	string
RUNNING_COMPLETE	string
SUSPENDED_INCOMPLETE	string
SUSPENDED_COMPLETE	string
FAILED	boolean
RESCHEDULED	boolean
SKIPPED	string
DROPPED	string

Resource	
name	string
type	resourceType
state	ResourceState

Fig. 4. Extended Schema for Workflow

Role Manager

This module is consists of an Agent Role Manager (ARM) which is in charge of defining access permission for agents on different modules and section of DMS. ARM also differentiates tasks among all agents. The next sub module is Task Distributor which distributes the task to appropriate Agent as soon as a task is produced and defined by the Workflow Engine. In order to monitor the activities of different agents and prepare the Audit report, there is another sub module named Monitor.

Workflow Engine

This is the core component of this proposed DMS. Workflow Engine (WE) is consists of five basic components for managing such as resource, task, agent, data and audit activity as shown in Figure 5. Resource is the key element of this proposed system. two different sub modules Cost Monitor and Performance Monitor continuously monitor two most important attributes–cost and performance of the resources. Resource Monitor will investigate the status such as idle, out of work, running. Another important component of WE is Task. Four different sub modules Task Manager, Task Scheduler, Task Executor and Task Monitor are in charge of managing tasks. Besides secured data management another major challenge of this proposed system is data has to be shared beyond geographical boundary, among different countries, different NGOs, different organizations and take necessary decisions for future. All sub modules in Data component of WE have been integrated to attain the ultimate goal.

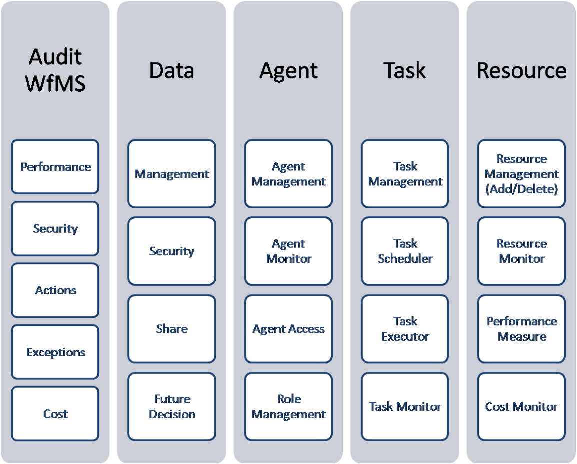


Fig. 5. Components of Workflow Engine

In order to distribute the roles among agent this MAS based system, it has a Role Manager. All activities of agents are monitored by Agent Monitor. The main functionality of Agent Access is to keep trace of the permissions allowed for different agents. Finally all internal communications among different agents as well as among different modules are managed by Agent Manager. Finally, all activities and actions taken by the proposed system are continuously audited by the Audit WfMS.

Workflow Scheduler

The MAS based workflow scheduling algorithm proposed by S. Akhter et al. [2] has been deployed in the workflow scheduler of the proposed DMS.

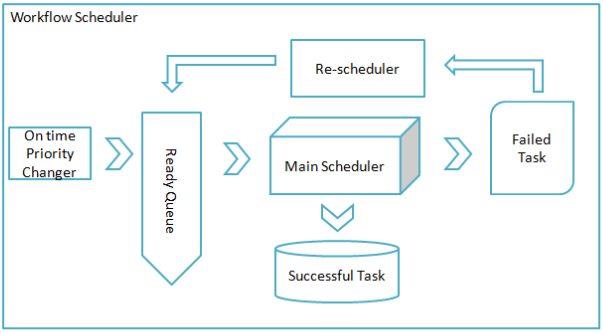


Fig. 6. Workflow Scheduler Structure

Figure 6 depicts that the workflow scheduler has six main components. At first all tasks of current workflow prepared by WE are put in to Ready Queue. The priority of tasks can be changed dynamically by On-time priority changer. After tasks are ordered according to their priority in Ready Queue, Scheduler runs the scheduling algorithms [2]. All successful jobs are stored in Successful Task list along with their report and failed tasks are stored in Failed Task. Failed task are rescheduled by Re-Scheduler and sent back to Ready Queue.

Workflow Monitor

All performance and status of currently running as well as previous workflow are generated by Workflow Monitor. Later all reports can be view from web portal based on performance parameters described in table 1.

Table 1. Performance parameters for Workflow Monitor

Name	Description
Number of Succeed Task per workflow	Defines the success rate
Number of failed task per workflow	Defines the failure rate
Number of used resources	Define the resource business
Number of total assigned resource	Define the actual resource capacity
Resource Status	Define the resource availability

Notification

This is an important component of the system. There are two different types of activities those have been performed by Notification. One is to implement Cloud to device messaging (C2DM) in order to broadcast notification to mass people. Another is to notify several components within the system.

3.2 Cloud Implementation

The most important contribution of this proposed DMS is cloud based implementation. Figure 7 describes the three layer of Cloud environment. PaaS is the preferred model over fully outsourced data processing and handling [3], presumably gaining support for having clear visibility, ownership and control over all the data. At the same time, system can quickly obtain the benefits of a fully-maintained software solution on a subscription basis. With PaaS system can get full control over data encryption and security. Therefore in this proposed system on PaaS, data related to all decisions taken for several past as well as current disasters for various locations for different type of incidents and tasks are store. This historical data are used as Heuristic data storage for further workflow scheduling. In addition in this layer, all record as a result of continuous audit performed by different agents, success and fail report for different workflow, status and performance evaluation of different resources, comparative analysis for different type of tasks in different workflow for different regional places are stored. The next layer (IaaS) is the most important layer. Amazon EC2 can be a suitable candidate as IaaS. The main components of this proposed DMS such as Workflow Engine (WE), Workflow Scheduler (WS), Monitor, Cloud web services as well as temporary data storages are put in this layer. The proposed web portal is established in SaaS layer.

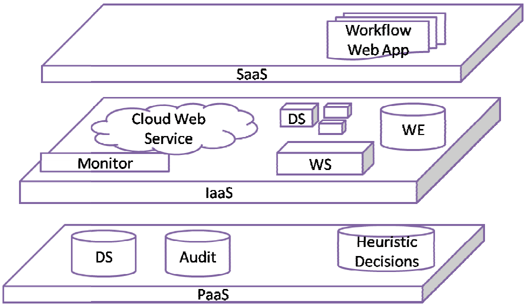


Fig. 7. Cloud Implementation for DMS

Cloud Service Implementation

GIS based emergency management system ArcGIS [9] is implemented in Cloud, however it uses GIS only to keep trace of location. However DMS is a complex and uncertain system. Therefore, along with location, it also has some other crucial and effective parameters such as weather, resources and data. In this proposed Cloud based system several RESTfull Cloud services as shown in figure 8have been designed whch are web services connected to core service and Cloud data storages. Firstly, user interacts with system through mobile devices, computers and web portal. In Cloud there are seven different web services connected to Core Computation Service (CCS) which is dependent on Cloud data storages for data. In this regard, *GIS Service* is necessary for tracing location which is indirectly connected to *Weather Service* to provide weather of particular location. *Emergency Response Service* is used for sending emergency notification to mass people. *Notification Service* is used for internal notification for the DMS. *Resource Management Service* as well as *Resource Discovery Service* both services deal with resource management and help other services for taking decision based on the availability of resources. *Data Record Service* is used for recording data and monitoring overall performance.

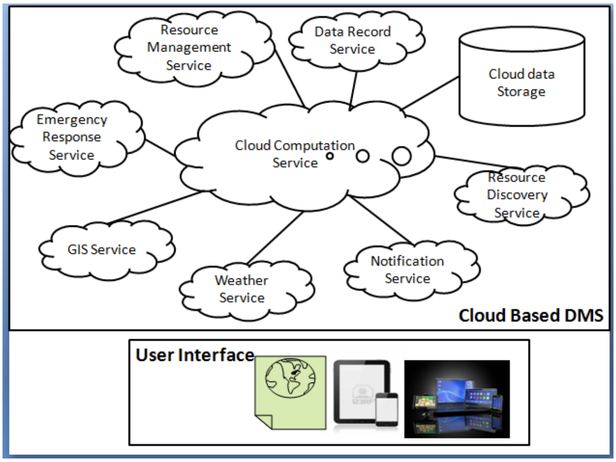


Fig. 8. Cloud Services for Proposed DMS

Proposed System Structure

In the proposed DMS, CCS performs the role to define workflow of DMS within WE with the help of other services and data storages. CCS communicates with GIS service in order to pull location based information. Similarly CCS gets weather related as well as available resource related information from weather service and resource management service. Finally CCS also gets data from storages and prepared the workflow of tasks those are needed to be performed in four different stages of Disaster Management lifecycle [2]. Once the workflow is prepared, WS schedules the workflow with the help of resource discovery service and resource management service. Therefore, all decision making tasks are performed in WE and workflow scheduling activities are performed in WS. Notification is responsible for messaging, alert, notification within the System or to other external system. As during natural disaster system internet connection or Wi-Fi connection could be damaged. In such situation C2DM can be a suitable solution. Therefore in this proposed DMS, along with web portal C2Dm based push notification service in mobile phone is implemented to send alert, notification or general information.

4 Experimental Evaluation

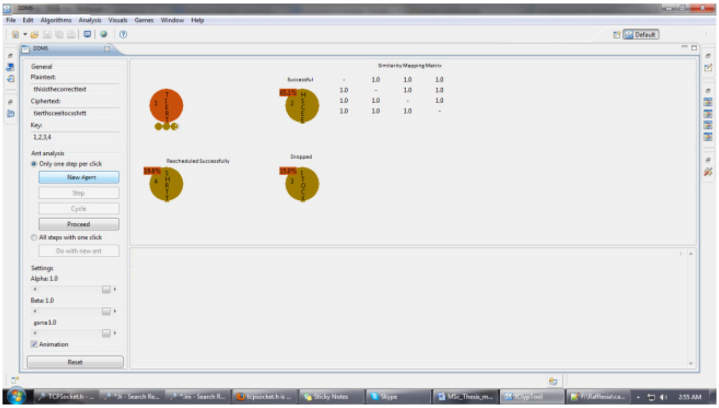
For simulation icanCloud [4] has been used in this proposed DMS system. Different types of damage for which we have collected data from two different data sources [7] and [8] are listed in table 2. The same data were used in [2].

Table 2. Query criteria for Ten Cases used in Simulation

	Data Source	Disaster Type	Location	Year
Case-1	DMSS	Tsunami	All Region	1974-1986
Case-2	DMSS	Flood	All Region	1990-2010
Case-3	DMSS	Epidemic	All Region	1990-2010
Case-4	DMSS	Flood + Epidemic	North and South coast	1990-2010
Case-5	DMSS	Forest Fire	North and South coast	1990-2010
Case-6	DMSS	Tornado	All Region	1990-2010
Case-7	DMSS	Strom	All Region	1990-2010
Case-8	NDDB	Earthquake	Asia Zone	NA
Case-9	NDDB	Cyclone + Flood	Asia Zone	NA
Case-10	NDDB	Tidal wave	Asia Zone	NA

4.1 Result and Observation

In this section we will describes the simulation result of the proposed DMS. Figure 9 describes that for this experiment with data from table [2], 71% of total tasks are successful. The proposed DMS also provides higher rescheduling success rate (83.31%) and comparatively lower dropout rate (9.66%). Moreover, the data migration time is comparatively less than other WfMS [6]. This is the most significant contribution of proposed DMS.



General
Plaintext:
thisisthecorrecttext
Ciphertext:
tierthsceeitocxshrtt
Key:
1,2,3,4
Ant analysis
● Only one step per click
○ All steps with one click
Settings
Alpha: 1.0
Beta: 1.0
gama 1.0
☑ Animation

Similarity Mapping Matrix

-	1.0	1.0	1.0
1.0	-	1.0	1.0
1.0	1.0	-	1.0
1.0	1.0	1.0	-

Fig. 9. Final Result (Successful, Dropped, Rescheduled)

5 Conclusion and Future Discussion

In this paper the implementation of this MAS model for DMS in real time system is described and designed. Moreover, the design and implementation plan of a web portal based on proposed DMS is described. The system architecture and components of the proposed DMS are also described in this paper. Following, but not limited to, are some contribution of this proposed work:

1. Web Portal Implementation for Automated workflow model for DMS
2. Real time implementation of DMS on Cloud
3. Eliminate dynamic on time dependency rather than providing proactive dependency calculation
4. In corporate more parameters to DMS such as time, cost, performance, Quality of Service etc
5. Redistribution of task among agents during idle time

Future work will focus on further analysis and validation of different stages of disaster management system life cycle, and on broadening the scope of this work to real-time operational, decision making and strategic management of DMS. Moreover, another suitable extensions of this proposed work can be implementation of Cloud based augmented reality to detect damaged area and possible easier transport route which can be a great contribution for recovery system in case of natural disaster.

References

1. Mendona, D., Wallace, W.A.: Studying organizationally-situated improvisation in response to extreme events. International Journal of Mass Emergencies and Disasters 22(2) (2004)
2. Habiba, M., Akhter, S.: MAS Workflow Model and Scheduling Algorithm for Disaster Management System. In: Proceedings of the 1st International Conference on Cloud Computing Technologies, Applications and Management, ICCCTAM 2012, Dubai, UAE (December 2012)
3. Kazusa, S.: Director for Disaster Management, Cabinet office, Government of Japan. Disaster Management of Japan (2011)
4. Nazrov, E.: Emergency Response management in Japan, Final Research report, ASIAN Disaster Reduction Center, FY2011A Program (2011)
5. Pandey, S., Karunamoorthy, D., Buyya, R.: Workflow Engine for Clouds. In: Buyya, R., Broberg, J., Goscinski, A. (eds.) Cloud Computing: Principles and Paradigms. Wiley Press, New York (2011) ISBN-13: 978-0470887998
6. Yoshizaki, M.: Disaster Management and Cloud Computing in Japan, Report from Ministry of International Affair and Communication (December 2011)
7. An Analytical Overview. Asian Disaster Reduction Center (March 2007)
8. Disaster Management System Srilanka, `http://www.desinventar.lk/` (last visited November 11, 2012)
9. ArcGIS as a System for Emergency/Disaster Management, `http://www.esri.com/industries/public-safety/emergency-disaster-management/arcgis-system` (last visited November 11, 2012)
10. Alazawi, Z., Altowaijri, S., Mehmood, R., Abdljabar, M.B.: Intelligent disaster management system based on Cloud-enabled vehicular networks. In: Proceedings of International Conference on ITS Telecommunications (ITST), August 23-25, pp. 361–368 (2011)

A Hybrid Grid/Cloud Distributed Platform: A Case Study

Mohamed Ben Belgacem[1], Haithem Hafsi[2], and Nabil Abdennadher[3]

[1] University of Geneva
Mohamed.Benbelgacem@unige.ch
[2] National School of Computer Science (ENSI), Tunisia
Haithem.Hafsi@gmail.com
[3] University of Applied Sciences, Western Switzerland, hepia Geneva
Nabil.abdennadher@hesge.ch

Abstract. The scene of the computational sciences has considerably changed during the last years. Today, new emerging Desktop grid and Cloud e-infrastructure have a considerable potential to be adopted and used in large scale to exploit thousands of CPUs power to run both scientific and commercial applications. This paper targets scientists and programmers who need to accelerate their scientific research by running their applications on distributed Grid/Cloud infrastructures. We present a hybrid Grid/Cloud platform used to deploy a phylogeny application called MetaPIGA. The aim is to combine the advantages of Grid and Cloud architectures in order to set up a robust, reliable and open platform. We propose two scenarios.

Keywords: distributed computation, Grid and Cloud computing, MetaPIGA.

1 Introduction

The concept of grid Computing was born in the mid of 1990s as an answer to the increased demand of high performance computing that required more computing power than a single cluster could provide [1]. According to [2], Grid Computing has three characteristics:

- decentralized resource control,
- non-guaranteed qualities of services : latency, throughput, and reliability,
- standardization: Grid middleware is based upon open and common protocols and interfaces.

Simultaneously with Grid Computing, a second alternative emerged. It consists of executing high performance applications on anonymous connected computers by using their available resources. This concept is called Volunteer Computing (VC). The most known systems are BOINC [3] and XtremWeb [4]. In the remainder of this paper, Grid will also include volunteer computing.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 162–169, 2013.

Despite the number of research projects carried out in the domain of Grid, these technologies were rarely commercialized. The development of Grid Computing and its standards was mainly driven by scientific communities.

For Cloud Computing, there is no established definition yet. According to [5], "a Cloud is a pool of virtualized computer resources". The same paper considers Clouds to complement Grid environments by supporting resources management. Clouds allow the dynamic scale-in and scale-out of applications by the provisioning and de-provisioning of resources. Many researchers and actors think that Cloud Computing is not a new paradigm. It draws on existing technologies and approaches, such as Utility Computing, Software-as-a-Service, distributed computing, and centralized data centers. What is new is that Cloud Computing combines and integrates these approaches, in particular, Utility Computing, represented by business models, pricing and SLAs.

This paper proposes a "hybrid" platform composed of a volunteer computing infrastructure, called XtremWeb-CH (XWCH: www.xtremwebch.net), and a Cloud infrastructure, used as provisioning system. The platform is used to develop, deploy and execute a high performance phylogenetic application called MetaPIGA [6]. As stated by [7] and [5], Clouds are a "useful utility that you can plug into your Grid". Our vision is to:

- combine the reliability of Cloud infrastructures and the "openness" of Grid environments,
- allow users deploying their applications on a reliable platform composed of a heterogeneous infrastructure: Grid, Cluster and Cloud.

This document is organized in 6 sections. After the introductory section 1, section 2 gives an overview of Grid vs. Cloud. Section 3 presents the Venus-C European project that aims at implementing a development environment for e-sciences applications on Cloud Infrastructure. The concepts proposed by Venus-C are used as guidelines in our research. Section 4 presents the hybrid solution developed in the framework of our research. Section 5 gives some experimental results carried out in order to evaluate the proposed solution. Finally, section 6 gives some perspectives of this research.

2 Grid vs. Cloud

This section compares Grid and Cloud [8] within 7 criteria detailed below:

1. *Resource localization*: while Grid Computing is defined by its geographically dispersed and decentralized resources, Cloud Computing seems to be a step back towards centralizing IT in data centers.
2. *Virtualization*: few research projects have integrated virtualization in grid projects. In Cloud, virtualization is one of the cornerstones; it allows the dynamic scale-in and scale-out of applications by the provisioning and de-provisioning of resources.
3. *Type of applications*: Contrarily to Grid, Clouds are not limited to e-sciences "batch" applications, but also support "interactive applications" such as Web and three-tier architectures.

4. *Development of applications*: the approach of how to develop applications is very different in Grids and Clouds. In Grids, the user typically needs to generate a binary for his application. This binary is then transferred to and executed on the remote resources in the Grid. Clouds allow a fundamentally different approach to software development. For instance, the Cloud provider offers "ready-to-use" components, the user can then dynamically assemble these existing functionalities to construct his Cloud-native application.
5. *Access & ease of use*: access to Grid resources is realized via a specific and often complex middleware. In contrast, interaction with resources in the Cloud is established via standard Web protocols, facilitating the access for the users. The lightweight accessibility and ease of use is one key factor that helped Cloud vendors succeed to convince non-academic customers to deploy their applications on their Cloud in a relative short period of time.
6. *Business model and SLAs*: as stated previously, business model, pricing and SLAs are one of the cornerstones of Cloud. These concepts are completely absent in Grid.
7. *Switching cost*: Through standardization, a Grid user can easily switch from the resources of one Grid provider to another. Due to the lack of standards, this is not possible in Cloud environment. Typically, Cloud providers have no interest in participating and implementing standards enabling potential customers to switch easily.

3 The Venus-C European Project

3.1 Project Overview

Venus-C [9] is a European project funded with the purpose to provide a new friendly-user Cloud solution for the scientific research domain in Europe. The target end-users are mainly individuals and researchers group that never have had access to high performance computing resources and are content with their desktop machines to run their applications. The objective of Venus-C project is to make it possible for researchers' community to run easily their applications on a large Cloud computing infrastructure in order to accelerate their scientific researches. Several scientific applications from several domains have been ported on the Venus-C platform.

Technically, the Venus-C is an interface between the Cloud providers and the end-users. It aims to provide a Platform as a Service (PaaS) with a set of tools and APIs to easily develop e-sciences applications and execute jobs that requires an execution coordination and a platform elasticity features.

One of the potential Cloud resources providers of the Venus-C project is the Microsoft Windows Azure infrastructure, which is based on Windows operating system. In what follows, we will interest on one of the programming model in Venus-C project: the "Generic Worker".

3.2 Generic Worker Concept

The Venus-C project comes up with the Generic Worker (GW) concept, an intermediate layer between the Azure platform and the end-users that shields them from technical complexity of the steps to use cloud computing resources. The main role of the GW is to facilitate the creation of the VM instances on the Cloud infrastructure and the application execution. Figure 1 depicts the GW component and its features.

In its simple form, the GW is composed of a .Net based package and an API used to start VM instances on the Azure platform. Several types of the VM are supported: small, medium large and extra-large. The type of a VM is mainly determined by the number of cpus and the memory size. To start VM instances through the Azure web portal, the user should upload the GW package with his XML configuration that mainly determines:

- the number of VM instances
- data access and certificate credentials.

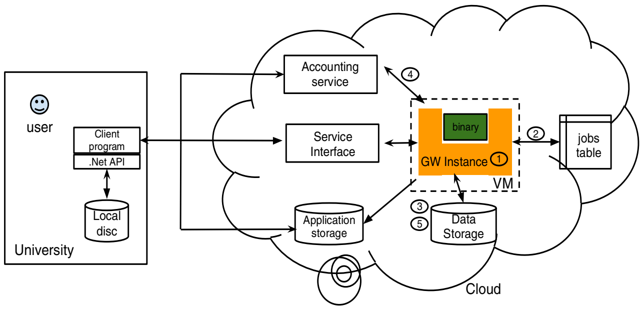


Fig. 1. Venus-C architecture overview

A Venus-C application can be composed of a workflow of jobs, where dependencies are based on input files: a job cannot be started unless its input files exist in the storage domain. Each running VM contains a GW instance that handles the execution of a given job (1). All the submitted jobs are stored in an Azure database table, and, then, scheduled by a service monitoring to the available GW instances. Periodically, each GW instance reads the database and retrieves its scheduled job (2). To execute the job, the GW instance should check the existence of its job's input file in the Cloud storage domain, then, loads them with the binary and any necessary libraries files to its local machine disc (3). Accordingly, the status of the job is tracked during its execution in the database (4). When the execution ends, the GW instance stages out the job result in the user storage domain (5). Besides, the GW provides a web service interface that allows the user through its client program to perform the scaling, notification and job management services. It worth noticing here that the number of the GW instances can be efficiently scaled on demand through the client program.

4 Hybrid Solution

The main idea behind the hybrid solution is to combine the XtremWeb-CH volunteer computing platform (XWCH: www.xtremwebch.net) with:

- Cloud infrastructures such as Amazon Elastic Cloud Compute (EC2) [10] and Azure,
- high performance oriented Cloud platforms such as Venus-C Generic Worker (GW) package.

The goal is to create a scalable and reliable large scale distributed platform used to deploy and execute the phylogenetic MetaPIGA application [6]. In what follows, we present, first, a brief description of the XWCH platform. Then, we describe how the hybrid solution is elaborated.

4.1 The XWCH Platform

The XWCH platform consists of three components: a coordinator, workers and warehouses. The coordinator schedules jobs and pre-assigns them to the workers. An XWCH worker is a small Java daemon that runs on a user or institute machine. Periodically, a worker reports itself to the coordinator, asks for a job, retrieves job's input files and stages out computation results in the warehouses. Since the workers could be fire-walled and could not communicate with each other to retrieve files for their jobs, the warehouses are used as file repositories to ensure file communication between the jobs within the same workflow. If the coordinator does not receive signal from a worker which is executing a job, it simply removes it from the workers list and assign its job to another available worker. A flexible API allows users to submit and monitor jobs according to their needs. Several applications have been ported on the XWCH platform [11].

4.2 Hybrid Platform

The "global" challenge behind bridging XWCH and Cloud is to scale up the XWCH infrastructure with Cloud resources. Let's remind here that XWCH workers are volunteer based, they belong to universities and/or individuals. Resources (CPU, number of cores, memory, software tools) which are available on these volunteer workers are similar to those available on "off-the-shelf" computers.

The main idea can be simply described as follow: when resources requested by the MetaPIGA application are not available on the volunteer XWCH infrastructure, the system creates its "private" resources on the Cloud according to the needs (processor performance, main memory, etc.) of the application. These resources are created "on the fly" on the Cloud, used by MetaPIGA jobs and released as soon as the execution ends. In this paper, we consider two scenarios for resources scaling.

In the first scenario (figure 2), MetaPIGA jobs are submitted to the XWCH coordinator (1). When the requested resources are not available on the Volunteer infrastructure, the XWCH-coordinator creates a "private" XWCH worker supporting these resources (2). This private worker will then execute the job for which it was created. In this scenario, Cloud resources are considered as part of the XWCH infrastructure. The user uses only one developing environment: XWCH API. This scenario was tested with two Cloud infrastructures: Amazon and Azure.

In the second scenario (figure 2), when XWCH infrastructure is unable to provide the necessary resources, the MetaPIGA application creates itself the requested resources (1) and submits directly its jobs to the Cloud (2). In this case, Cloud resources are not considered as part of the XWCH platform. MetaPIGA jobs use the Cloud storage to retrieve their input files. After execution, the job's results are stored in the Cloud storage in order to be retrieved by the metaPIGA application. The developer uses two different APIs to submit his jobs: XWCH API and Cloud API. He also manages data flow between jobs running on the Cloud and those running on XWCH platform. This scenario was tested with Venus-C GW platform.

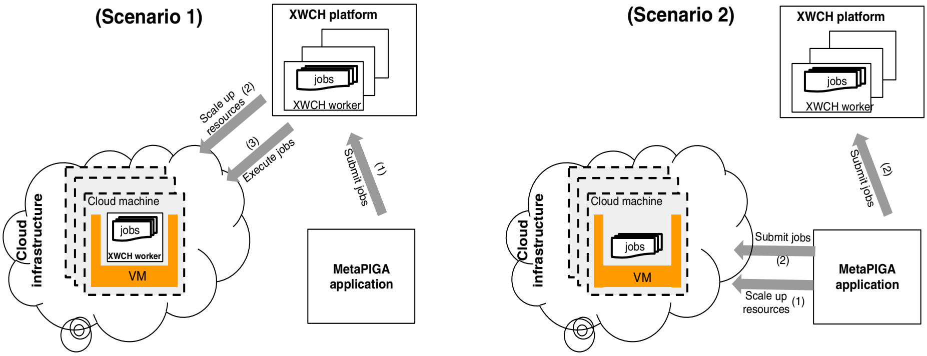


Fig. 2. Hybrid platform

5 Experiments

The two proposed scenarios are used to deploy and execute the MetaPIGA application, developed at the University of Geneva, over large hybrid computing infrastructure.

5.1 MetaPIGA Application

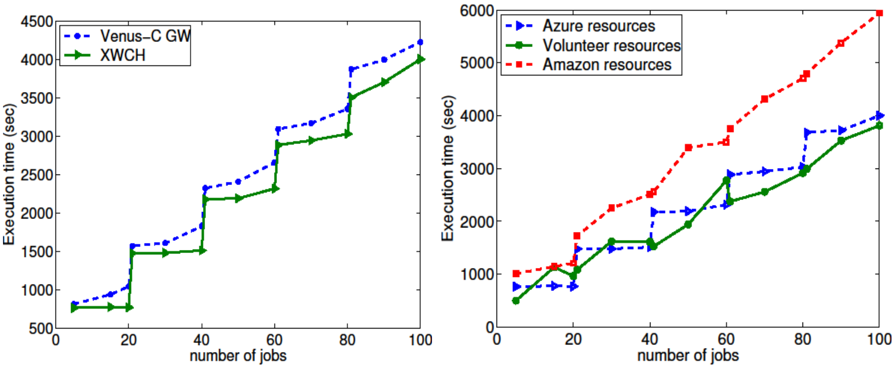
The Java based MetaPIGA [6,12] application consists of a robust implementation of several stochastic heuristics for large phylogeny inference (under maximum likelihood), including a simulated annealing algorithm, a classical genetic algorithm, and the metapopulation genetic algorithm (metaGA) together with complex substitution models, discrete Gamma rate heterogeneity, and the possibility to partition data. Heuristics and substitution models are highly customizable through manual batch files and command line processing.

MetaPIGA is a CPU time consuming application. For instance, one big dataset needs in general 500 CPU hours. Assuming that 200 analyses are launched every year, the total number of CPU hours needed per year is equal to 100'000.

MetaPIGA is well suited for parallelization since several populations can be run in parallel and can therefore be sent to different machines.

5.2 Measurements

Figure 3(a) compares the overhead generated by the two API: XWCH and Venus-C GW. Since the native Venus-C GW API is only implemented on Azure, we use this platform as a hardware infrastructure.



(a) on XWCH and Venus-C GW

(b) on workers deployed on Azure, Amazon and volunteer computer

Fig. 3. Time execution of MetaPIGA

The results show that the overhead generated by XWCH API is slightly inferior to Venus-C GW API. In this figure, the number of available XWCH workers (resp. GW instances) is equal to 20.

Figure 3(b) compares the performances of MetaPIGA when executed on an XWCH platform using three "types" of workers:

- volunteer, non dedicated, workers,
- workers deployed on Azure,
- workers deployed on Amazon.

For both Amazon and Azure workers, we have used small VM instances. An Amazon VM instance runs Ubuntu operating system and has a 1 ECU (*EC2 Compute Unit ≈ 1.0-1.2 GHz) of CPU speed and 1.7 GB of memory*}. An Azure VM instance runs Windows operating system and has a 1.6 GHz of CPU speed and 1.75 GB of memory. Regarding the XWCH workers (Linux and Windows), they are installed on student machines having each an average CPU speed of 2.2 GHz and 3 GB of system memory.

Results show that the execution time of the MetaPIGA application on Azure workers is slightly higher comparing to the Amazon ones. Besides, the non-stairs shape obtained in the XWCH curve can be explained by the volatility aspect of the XWCH platform, i.e that the number of connected XWCH workers on the platform can vary during execution.

6 Conclusion

This paper presents two scenarios to bridge the XWCH Grid platform with Cloud infrastructure. Cloud is used as a provisioning system which allows users to "rent" resources not supported by the Grid. Two scenarios were proposed: In the first case Cloud resources are not seen by the user, they are managed by the Grid itself. Contrarily to this approach, the second scenario assumes that the user submits himself the jobs to the Cloud.

It therefore obliges the developer to use two different developing environments. The two scenarios have been tested in the case of MetaPIGA application with Amazon, Azure and Venus-C Cloud platforms. The next step will be to generalize this approach to other applications and develop a "generic" toolkit environment that supports other Cloud infrastructures.

References

1. Kesselman, C., Foster, I.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers (November 1998)
2. Foster, I.: What is the Grid? - a three point checklist. GRIDtoday 1(6) (July 2002)
3. BOINC, `http://boinc.berkeley.edu/`
4. XtremWeb, `http://www.xtremweb.net/`
5. Boss, G., Malladi, P., Quan, S., Legregni, L., Hall, H.: IBM high performance on demand solutions. Technical report, IBM developerWorks (2007)
6. MetaPIGA 2 - Large phylogeny estimation, `http://www.metapiga.org/`
7. Gentzsch, W.: DEISA. Grids are Dead! Or are they? (2008), `http://www.hpcinthecloud.com/hpccloud/2008-06-16/grids_are_dead_or_are_they.html`
8. Weinhardt, C., Blau, B., Meinl, T., Sößter, J.: Cloud Computing - A Classification, Business Models, and Research Directions. Business & Information Systems Engineering Journal 1, 391–399 (2009)
9. VENUS-C European project, `http://www.venus-c.eu/`
10. Amazon Elastic Cloud Compute, `http://aws.amazon.com/ec2/`
11. Abdennhader, N., Ben Belgacem, M., Couturier, R., Laiymani, D., Miquée, S., Niinimaki, M., Sauget, M.: Gridification of a radiotherapy dose computation application with the xtremWeb-CH environment. In: Riekki, J., Ylianttila, M., Guo, M. (eds.) GPC 2011. LNCS, vol. 6646, pp. 188–197. Springer, Heidelberg (2011)
12. Belgacem, M.B., Abdennadher, N., Niinimaki, M.: The XtremWebCH Volunteer Computing Platform. In: Desktop Grid Computing, ch. 3, June 25. Numerical Analysis and Scientific Computing Series. Chapman and Hall/CRC (2012)

Comparison of Two Yield Management Strategies for Cloud Service Providers

Mohammad Mahdi Kashef[1,*], Azamat Uzbekov[1],
Jörn Altmann[1], and Matthias Hovestadt[2]

[1] Technology Management, Economics, and Policy Program
Department of Industrial Engineering
College of Engineering
Seoul National University
Seoul, South Korea
{mmkashef,batukasss}@temep.snu.ac.kr, jorn.altmann@acm.org
[2] Hanover University of Applied Sciences
Dept. of Computer Science
Hanover, Germany
matthias.hovestadt@hs-hannover.de

Abstract. Several Cloud computing business models have been developed and implemented, including dynamic pricing schemes. This paper extends the known concepts of revenue management to the specific case of Cloud computing from two perspectives. First, we propose system architecture for Cloud service providers for combining demand-based pricing and scheduling. Second, a comparison of two yield management methods for cloud computing has been compared: Limited Discount Period Algorithm and VM Reservation Level Algorithm. By taking advantage of demand estimation, the two algorithms find the optimum number of VMs that are sold at full price and the optimum time period before the allocation when the prices should change. Simulation results show that both yield management methods outperform static pricing models and the algorithms perform differently considering the deviation of demand.

Keywords: Cloud computing, revenue management, pricing strategy, autonomic resource management.

1 Introduction

Cloud service providers (CSPs) face challenges regarding performance and pricing. On the one hand, Cloud service consumers wish to minimize the execution time of their submitted tasks without exceeding a given budget, while, on the other hand, CSPs are keen on maximizing their revenue while keeping customers satisfaction [1]. A real-time view of the Cloud provider's business with respect to revenue and costs becomes essential. Such a system helps to respond in an economically efficient way. Solutions to these issues are provided through business economics [2].

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 170–180, 2013.

The current problem that CSPs face is that they have to reserve resources (e.g. a specific number of virtual machines) at a particular time upon a given user request. This reservation of resources basically follows the traditional first-come-first-served approach, so that late service requests have to be rejected if resources have already been reserved for earlier service requests, even if these late service requests have a higher value for the CSP. To address this problem, dynamic pricing methods have been successfully applied in many cases [3]. The reservation price can depend on the demand. For instance, an industry that has applied this pricing is the airline sector. Its solutions are based on yield management [4], maximizing revenue.

In the scope of this paper we apply two orthogonal yield management practices for maximizing the revenue. For both yield management practices, the full price and the discount price for VMs will be set. In addition, for the first practice we set the time L, which represents the time when price offerings switch from a discounted to a full price. For the second practice, we set the number of VMs (i.e., the protection level (PL) of VMs) that should be offered at full price. The research questions that will be addressed are how such a dynamic pricing model can be integrated into a scheduling architecture, and how those dynamic pricing models (if they perform differently) can be combined in the architecture through a smart switch.

For answering these research questions, this paper is structured as following: in the next section, an overview about related work is given. Chapter 3 introduces our proposed architecture as well as suggested pricing algorithms. The effectiveness of the work will be evaluated by simulation in Chapter 4 while Chapter 5 concludes the paper.

2 State-of-the-Art

2.1 Cloud Computing

It has recently become very popular as a new paradigm to shift IT resources and software from locally independent computers to a more collaborative level [4]. Cloud computing refers to not only “the applications delivered as services over the Internet” but also “the infrastructures and systems in the datacenters” [5]. In this work, we follow the definition of Cloud computing of NIST (National Institute of Standards and Technology) [6].

Though Cloud computing has a clear definition and features, there is no clear line of separation with other forms of distributed computing systems like Grid computing. It is no wonder because Cloud not only overlaps with Grid, it has indeed evolved out of Grid and relies on Grid as its backbone and infrastructure support [7]. In this work, the authors compare Cloud and Grids and despite the fact that they have similarity in their vision, architecture and technology, they significantly differ in security, programming model, level of abstraction, compute model, data model, applications, and business model. In case of our work, we have focused on business model differences between those systems. Han [8] believes that the Grid systems are scientific orientated, and are mainly supported by research communities; and compared to that, Cloud computing is profit-orientated and has a much broader user base, including non-IT companies and individuals.

2.2 Revenue Management

The idea of revenue management (RM) or yield management is to give the seller the right to set the optimum price. The techniques of RM have been firstly implemented in airline industry, which benefited at $1.4 billion over three years [9]. Other industries, such as hotels, restaurants, and car rental companies, also use RM as a tool for resource allocation and revenue maximization, and this has been studied by a large number of scientists. Following the definitions of yield management presented in [10, 11], the definition of RM that is most suitable for this paper is “a method that helps to sell the VMs to the right consumer, at the most suitable moment, and at the optimum price.”

Based on “expected marginal seat revenue” (EMSR) technique developed by Belobaba [12], airline companies make decisions about how many seats to sell for each price class they have. The same algorithm has been implemented in the hotel industry [13-15]. Hotels have different prices for the same quality of rooms. To separate two guest segments, the hotel introduces a protection level (PL) that divides the total capacity of rooms into two parts. The protected rooms will not be sold at a discount price because of the possibility that some customers might buy the same rooms at a full price later. Furthermore, the booking limit (BL) is the number of rooms that may be sold at the discount price.

In [10, 16, 17], the possibility of using RM in telecommunication industry and specifically in Grid was studied in detail. Arun Anandasivam et al. overviewed RM how this concepts can be deployed to Grid. He compared the Grid computing domain with other common areas for RM showing that even there are notable differences, RM is applicable on Grid. Anthony Sulistio et al. went deeper and presented the model using RM and simulation for two Virtual Organizations (VOs) with broker between users and Grid to determine pricing of reservations. Both of these works outlined the requirements for applying RM to the Grid and showed how the RM tools can be effectively exercised. But their architecture does not give consumer any possibility to use other allocation method that could be more applicable on consumer’s need.

2.3 Resource Management

Existing studies of the internet and media workloads indicate that client demands are highly variable (“peak-to-mean” ratios may be an order of magnitude or more), and it is not economical to overprovision the system using “peak” demands [18], [19]. Gmach has presented results that illustrate the peak-to-mean behavior for 139 enterprise application workloads. He has shown that an understanding of burstiness for enterprise workloads can help in choosing the right tradeoff between the application quality of service and the resource pool capacity requirements. The ability to plan and operate at the most cost-effective capacity is a critical competitive advantage [20].

Van et al. presented an autonomic resource management system, which has the ability to automate the dynamic provisioning and placement of VMs. For this, they have taken into account both the application-level service level agreements and resource exploitation costs with high-level handles for the administrator to specify trade-offs between the two [21].

3 System Architecture and Algorithms

For the sake of maximizing the revenue of a CSP, we propose the system architecture as shown in Fig. 1.

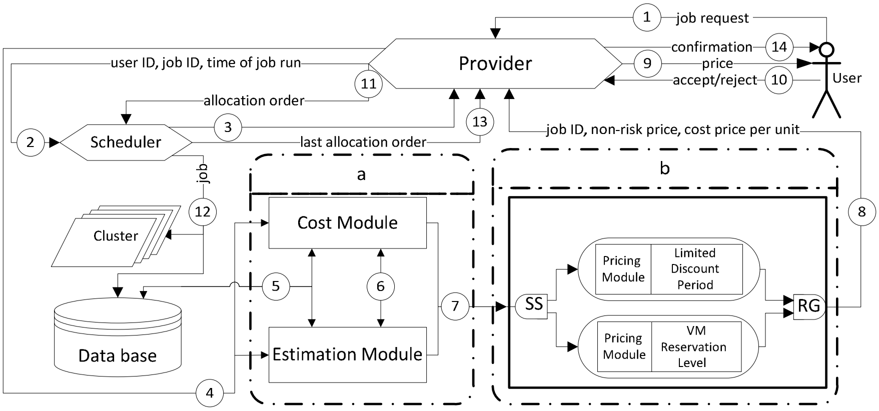


Fig. 1. Proposed architecture for pricing and resource allocation in a CSP

The proposed architecture for the CSP includes information about demand and the system cost [see a) in Fig. 1], the business support framework (BSF) [see b) in Fig. 1], the provider and the scheduler. The sequence is as explained here: In order to manage the requests, user interacts (1) with a component, namely, the provider. The provider manages requests based on First-Come-First-Served. It sends (2) job information to the scheduler to check the technical feasibility of the job (e.g. are sufficient resources of requested quality available). The scheduler will report (3) the feasibility to the provider. In case of being feasible, an order from provider will be sent (4) to block 'a' of Fig. 1, which means start demand estimation and cost calculation. Hence, the cost module and the estimation module request and get (5) needed data from data base to calculate the estimated demand by which (6) total cost is computed. The results are input (7) for BSF. The outcome of BSF processes will be sent (8) to provider. Based on the proposed data, the provider is able to negotiate with the user to finalize the deal (9, 10). Finally the provider orders (11) the scheduler to allocate proper VM on the requested time to the job. The scheduler will assign the VM to the job as per order (12), and send a confirmation notice back to the provider (13). Finally, the user will be informed of the confirmed deal (14). Yet, all information regarding the jobs goes to the database to keep the historical records. This data helps the CSP to set the prices that reflect the risk of losing opportunity cost and to estimate the near future demand.

3.1 Demand Estimation Module

Forecasting is often considered the most critical part of revenue management. The quality of decisions, such as pricing and capacity control, depends on an accurate forecast [22]. The data used in a demand estimation module is based on the historical data of requests and submitted jobs in the full price class. In designing the estimation module, such inputs should be available due to our modeling. Required inputs include historical data of demand (number of jobs requested at full price), application (VM) category, type of VM, e.g., in the case of Amazon: EC2, S3, etc. The module performs the analytical calculations and returns the estimated future trends, i.e., the demand for the full price. The module takes advantage of a heuristic method to estimate the demand of VM. More specifically, based on the past experience of the Cloud vendor, particular application categories, such as web server, game server, online-shopping server, e-learning server, etc., are required before going through the details of trend estimation since the module categorizes data based on its application category. This helps to obtain a better prediction for each category.

For simplicity, we assumed that the curves are of a monotonic function, modeling an overall upward or downward change in demand. We further assumed that all demand traces have a cyclic behavior. To perform the abovementioned prediction, four major processes are considered: extracting patterns, calculation of the pattern, deviation calculation and classification. Based on the estimation, two trends are generated, i.e., the demand for the full and the discount price to be used by the pricing module.

3.2 Cost Module

For any kind of service pricing, one should be aware of total cost of the service; so that price setting won't make any loss for the service provider. Hence in the proposed architecture all of the BSF processes are based on the calculated cost of the service. To calculate the overall costs of a VM, a detailed cost model has been proposed in [23] in which fixed and variable costs should be calculated. For fixed cost (FC) the proposed formula accounts all of the initial costs: server purchase, network device purchase, cost of software licenses, cost of facility space, the cost for cabling and preparation of data center. Then all variable cost factors should be extracted and calculated. Items as listed in [23] are cost of electricity, the cost for Internet usage and the cost of maintaining labor. Finally total cost is gained based on total fixed cost and total variable cost; which will be inputs for pricing module.

3.3 Business Support Framework

First step of is that the Smart Switch (SS) checks the demand situation and recommends algorithms performing superior for the respective time span. Then the selected algorithm(s) use provided information by step 7 for setting full and discount prices. The BSF is able to run various economic-based algorithms. By now we have proposed two algorithms namely, Limited Discount Period (L algorithm) and VM protection Level (PL algorithm) as shown in b) of Fig. 1. Both of the algorithms need proposed prices, so the pricing module is embedded in them. This module, being dependent on demand level, generates on-time prices for the allocation algorithms of the BSF. In the L algorithm, the prices of VMs depend on how many days in advance the request is made, while the PL algorithm sets prices according to the PL of resources. Then Report Generator (RG) will merge the results of algorithm(s).

Pricing Module. Since the approach of this work is YM method, the prices for the services of the CSP should be differentiated. The pricing module generates two prices: the full price (A) and the discount price (B), which are directly linked to demand. The key principle of the price strategy used by pricing module is that, if the probability of selling the product is very high then there is no need to offer a low price to sell it. Hence, the prices will be set using the full-cost method (or cost-plus)[24]. This method is very easy to explain, and the structure of the price is clear. The gross profit margin (GPM) in this method is set manually. To minimize human intervention in the price setting procedure in this work instead of the GPM other values that are directly tied to demand has been used as explained in equations 1 and 2.

$$A = \left(1 + \frac{q^{max}}{C_{full} + q^{max}}\right) * \frac{TC}{q_{avg}} \tag{1}$$

$$B = \left(1 + \frac{q^{min}}{C_{full} + q^{min}}\right) * \frac{TC}{q_{avg}} \tag{2}$$

In the proposed formula, where TC is the total cost, C_{full} means the total capacity of the CSP. Therefore, the full price is set based on the highest demand point (q^{max}), and the lowest demand point (q^{min}) determines the discount price. An average demand within the same accounting period is q_{avg}.

The Limited Discount Period algorithm (L algorithm). The objective of the first algorithm is to define the time duration, L, before the "job-start-time" so that when it occurs, the price changes from discount price to full price [25]. The idea of starting from the discount price and then switching to the full price is selected for two reasons. First, a low price attracts more customers, and it reduces the risk of not covering the production cost. Second, the practice of buying services in advance contributes to forecasting and resource allocation planning. In this case, the discount price is seen as an incentive for customers.

Hence, we use a breakeven analysis as a simple and easily understandable method of examining the relationship between the fixed cost, the variable cost, the volume, and the price [26]. The breakeven sales quantity helps to assess the number of products that must be sold to generate a contribution equal to the total cost.

$$BEQ = \frac{FC}{p - VC(q)} \tag{3}$$

Where BEQ is the breakeven sales quantity, FC is the fixed cost, VC is the variable cost per unit, and p is the price per unit.

The calculation of point L starts with the breakeven analysis. Equation (3) helps to find the point where the sales revenue covers all the costs exactly, i.e., the profit is zero. Only when the next unit is sold, the CSP realizes any profit. Therefore, the CSP starts to receive requests taking into account the time of request, t_r, and the job-start-time, t_s. Having ascertained the BEQ and the outcome of the estimation module, the CSP can calculate how many days are needed, L, to cover the production cost.

Steps of the algorithm are: 1) First assess system's parameters C_i and p_i; 2) After receiving a request, set $j = t_s - t_r$ and $L = t$ where $t + 1$ until $\sum d_t = BEQ$; 3) If $j \geq L$, and $p_i = B$, let $C_i = C_i + 1$, otherwise, $p_i = A$ and let $C_i = C_i + 1$; 4) Calculate the total capacity. If $\sum C_i \geq C_{full}$, go to the last step, otherwise start from the beginning; 5) The calculation stops with summing the total revenue $\sum(p_i * C_i)$.

The VM reservation level algorithm (PL algorithm). Since the CSP must decide the quantity of VMs to sell at the full price, the CSP must determine the protection level (PL) that divides the CSP's total capacity into two parts: the protected VMs and the VMs at discount price[15].

To explain mathematically, we refer to the EMSR technique[12]. We define $F_i(d_i)$ to be the probability density function for the total number of reservations requests, d_i, for VMs in price class i. The number of VMs allocated to a particular price class, C_i, in case of rejection may not exceed the number of actual requests for that price class.

$$F_i(C_i) = \int_0^{C_i} F_i(d_i)\, \partial d_i \tag{4}$$

The optimal protection level, PL, for the business class is the value of C_i that satisfies the condition:

$$A * F_1(C_1) = B * F_2(C_2) \tag{5}$$

$$F_1(PL^*) = \frac{A - B}{A} \tag{6}$$

Steps of the protection level algorithm are as follows: 1) After assessing the system's parameters C_i and p_i, calculate Q^* and find from the table of cumulative probability the smallest cumulative value greater than or equal to PL^*; 2) Calculate $BL = C_{full} - PL$; 3) Receive a request and, if $C_i < BL$, accept the order and let $C_i = C_i + 1$ and $p_i = B$; 4) Otherwise switch the price so that $C_i = C_i + 1$ and $p_i = A$; 5) Then calculate the total capacity. If $\sum C_i \geq C_{full}$, go to last step, otherwise go to beginning; 6) Calculate the total revenue $\sum(p_i * C_i)$, then stop.

4 Simulation Experiments

4.1 Simulation Scenario

In our scenario, a CSP wants to identify optimum prices of VMs in order to maximize revenue. In presented simulations "the future time horizon of the simulation" is considered to be 30 days (but one may set it to other values). For simplicity, group requests are not considered hence one job means one VM for one hour. The Cloud vendor charges the customers two classes of prices: full price and discount price. There are three price-impacting inputs associated with the CSP's production cost: the fixed cost, the variable cost, and the total capacity. The fixed cost and variable cost of the Cloud provider are given as €20 and €1 accordingly. The total capacity of the CSP is assumed to be 100.

Each job request has two timing parameters: the job request time and the job start time. In this step of the work simulation is limited to solve the problem for a particular "job-start-time" (JST), i.e. to find the optimum L and the PL for a particular JST.

4.2 Data Generation

In this work, two types of simulations have been performed. First, demand was generated between 1 and 100 based on uniform random distribution. Second, the simulator generated demand based on normal random distributions, for which μ was set to be 25, 50 and 100, while σ was considered to be 1, 10, 20 and 30.

4.3 Simulation Results

Based on the generated random data in each simulation, full price and discount price have been calculated using formula 1 and 2. Then, the total revenue of each simulation is counted. Total revenue of both pricing strategies is shown in Fig. 2. The diagram shows a comparison of four pricing methods: the L algorithm, the PL algorithm, fixed high price (A) and fixed low price (B). It shows that the L algorithm often generates more revenue than the PL algorithm. But in some other cases PL algorithm is making more profit; and with this information one can't imply when and under which situation L or PL works better. So another round of simulation has been performed to find out the situation of better revenue by each algorithm.

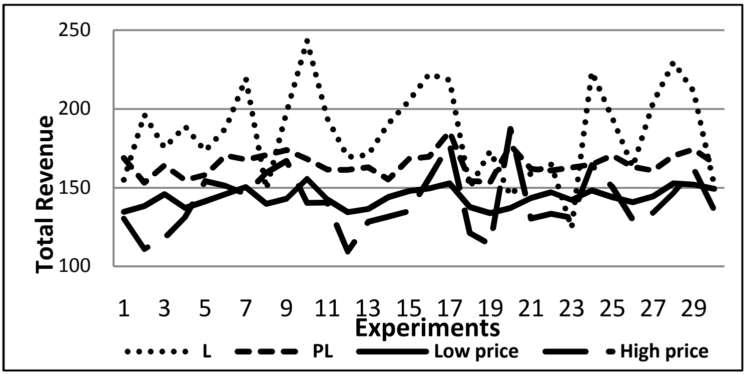


Fig. 2. Comparison of the results of the four pricing strategies in first round of simulation

Then a second round of simulation has been performed. The average results are shown in Table 1 (from 50 simulation runs). Based on those values, the full price and discount price are also calculated (Table 1).

Table 1. Average results for mean μ = 25, 50, 100 and standard deviation σ= 1, 10, 20, 30

St.Dev	**1**			**10**			**20**			**30**		
Mean	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**
Max demand	27	52	102	46	71	120	68	91	140	87	99	161
Min demand	23	48	98	5	30	80	1	8	57	1	2	39
Avg demand	25	50	100	25	50	100	35	49	99	44	51	100
Full price (A)	2.3	2.1	2.4	2.6	2.4	2.6	2.7	2.7	2.9	2.7	2.8	3.1
Disc price (B)	2.2	2.1	2.4	1.9	1.8	2.2	1.7	1.5	1.9	1	1.4	1.7

The revenues of the L and the PL algorithms have been calculated according to the 3.2.3.2 and 3.2.3.3. The generated revenue ratio of L and PL algorithms is compared in Fig. 3. Moreover, the average total revenue of 50 experiments is depicted in Fig. 4.

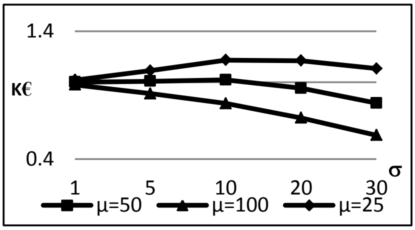


Fig. 3. Comparison of the total revenue ratio of L to PL in case of five σ for three different μ.

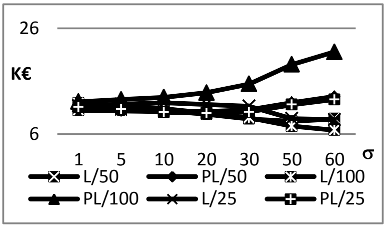


Fig. 4. Comparison of the total revenue in seven cases of σ and three means (25, 50 and 100; unit: K€)

4.4 Analysis and Discussion

The results of the first simulation are not giving meaningful conclusion about which algorithm (i.e., L or PL) is superior. To figure out the conditions, under which it is better to run the L algorithm, some additional simulations have been done. As diagram of Fig. 3 shows as the demand approaches the full capacity (here 100 VMs) revenue ratio is increasing for PL algorithm. The statement is true vice versa, i.e. for the μ=100 the more variation of demand the more revenue of PL algorithm. It can imply that if demand is very near to full capacity, the revenue is better when considering the protected level of VMs than L days prior to the job-start-time. While observing the cases in which the L algorithm made less revenue, it can be concluded that, if demand is low, the L strategy is more applicable. Nevertheless, both of the proposed pricing strategies are more effective than selling the VMs at any fixed price.

The conclusion is supported by the Fig. 4 showing in case of PL algorithm as far as getting farther from full capacity revenue decreases, but this is reverse for L algorithm.

In the real world, CSPs may take advantage of the two proposed models to gain greater revenue. A smart switch in BSF as drawn in Fig. 1, can select best method according to demand situation.

5 Conclusion

Within this paper, we described how providers can be supported in price setting decisions. We proposed a system architecture, which included a demand prediction module, a cost module, a pricing module, and a business support framework. The system modules are described in details. In particular, we introduced two different pricing strategies, which follow the yield management method, for selling a Cloud service.

Our simulation results show that the proposed architecture and algorithms can be helpful to set an optimum price that generates maximum revenue. Our future work aims at extending this research so that more complicated cases, which have greater applicability in the real world, are assessed.

Acknowledgements. This work has been funded by the Korea Institute for Advancement of Technology (KIAT) within the ITEA 2 Project 10014 EASI-CLOUDS.

References

1. Tsakalozos, K., Kllapi, H., Sitaridi, E., Roussopoulos, M., Paparas, D., Delis, A.: Flexible use of cloud resources through profit maximization and price discrimination. In: Proceedings of the 2011 IEEE 27th International Conference on Data Engineering, pp. 75–86. IEEE Computer Society (2011)
2. Altmann, J., Hovestadt, M., Kao, O.: Business support service platform for providers in open cloud computing markets. In: 2011 The 7th International Conference on Networked Computing (INC), pp. 149–154 (2011)
3. McGill, J.I., Van Ryzin, G.J.: Revenue Management: Research Overview and Prospects. Transportation Science 33, 233–256 (1999)
4. Hayes, B.: Cloud computing. Commun. ACM 51, 9–11 (2008)
5. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: Above the Clouds: A Berkeley View of Cloud Computing. University of California at Berkeley (2009)
6. Mell, P., Grance, T.: The NIST definition of cloud computing. National Institute of Standards and Technology 53, 50 (2009)
7. Team, M.S.R.: Cloud Computing takes off. BLUE PAPER (2011)
8. Han, L.: Market Acceptance of Cloud Computing - An Analysis of Market Structure, Price Models and Service Requirements. Information Systems Management 42 (2009)

9. Smith, B.C., Leimkuhler, J.F., Darrow, R.M.: Yield Management at American Airlines. Interface 22, 8–31 (1992)
10. Iallat, F., Ancarani, F.: Yield management, dynamic pricing and CRM in telecommunication. Journal of Services Marketing (2008)
11. Kimes, S.E.: The basics of yield management. The Cornell H.R.A Quarterly (1989)
12. Belobaba, P.P.: Application of a probabilistic decision model to airlines eat inventory control Operations Research 37,14 (1987)
13. Gayar, N.F.E., Saleh, M., Atiya, A., El-Shishiny, H., Zakhary, A.A.Y.F., Habib, H.A.A.M.: An integrated framework for advanced hotel revenue management Hospitality Management 23, 14 (2011)
14. Relihan, W.J.: The Yield-Management Approach to Hotel-Room Pricing
15. Netessine, S., Shumsky, R.: Yield Management (1999)
16. Sulistio, A., Kim, K.H., Buyya, R.: Using Revenue Management to Determine Pricing of Reservations. In: IEEE International Conference on e-Science and Grid Computing, Bangalore, pp. 396–405 (2007)
17. Anandasivam, A., Neumann, D.: Managing Revenue in Grids. System Sciences. In: 42nd Hawaii International Conference on HICSS 2009, Big Island, HI, pp. 1–10 (2009)
18. Arlitt, M.F., Williamson, C.L.: Web server workload characterization: The search for invariants. Performance Evaluation Review 24, 126–137 (1996)
19. Cherkasova, L., Gupta, M.: Analysis of enterprise media server workloads: Access patterns, locality, content evolution, and rates of change. IEEE/ACM Transactions on Networking 12, 781–794 (2004)
20. Gmach, D., Rolia, J., Cherkasova, L., Kemper, A.: Workload analysis and demand prediction of enterprise data center applications, pp. 171–180 (2007)
21. Van, H.N., Tran, F.D., Menaud, J.M.: SLA-aware virtual resource management for cloud infrastructures, pp. 357–362 (2009)
22. Chiang, W.-C., Chen, J.C.H., Xu, X.: An overview of research on revenue management: current issues and future research Int. J. Revenue Management 1 (2007)
23. Kashef, M.M., Altmann, J.: A cost model for hybrid clouds. In: Vanmechelen, K., Altmann, J., Rana, O.F. (eds.) GECON 2011. LNCS, vol. 7150, pp. 46–60. Springer, Heidelberg (2012)
24. Paleologo, G.A.: Price-at-Risk: A methodology for pricing utility computing services. Systems Journal 43 (2004)
25. Svrcek, T.: Modeling airline group passenger demand for revenue optimization. Massachusetts Institute of Technology, Flight Transportation Laboratory, Cambridge, Mass. (1991)
26. Monroe, K.B.: Pricing: Making profitable decisions, 2nd edn. McGraw-Hill Companies (1990)

Comparing Java Virtual Machines for Sensor Nodes

First Glance: Takatuka and Darjeeling

Oliver Maye[1] and Michael Maaser[2]

[1] IHP, Frankfurt (Oder), Germany
maye@ihp-microelectronics.com
[2] Anting/Shanghai, China
dr.michael.maaser@googlemail.com

Abstract. For comparing Java virtual machines targeting smart systems such as wireless sensor nodes, a list of qualitative and quantitative criterions is proposed. The open source JVMs Takatuka and Darjeeling are then compared by architecture and features. The JVM runtime properties are benchmarked on an MSP430-based test platform. Results show that Takatuka is the mature, feature-rich, multi-purpose JVM near J2ME with a 50% advantage in Java byte code size. Darjeeling fits well for tiny, focused applications and offers a runtime performance bonus of up to a factor of six.

Keywords: Java, JVM, Benchmark, Performance, Takatuka, Darjeeling.

1 Introduction

For wireless sensor network (WSN) nodes, energy efficiency directly translates into feature opulence, agility and computational power. Smart firmware development is one of the most complex and hence, time-consuming tasks during WSN development.

In heterogeneous environments, Java plays best its "platform-independency" card. So, specifically for WSNs, Java is an attractive alternative to well established programming languages, such as C/C++.

We aim at answering the question: *What are the similarities and differences of the Java virtual machines available for WSN nodes?* As the properties of a Java Virtual Machine (JVM) strongly depend on the underlying hardware platform as well as on the operating system (OS), we first narrow down the latter two.

To take advantage from synergies between related projects, the hardware platform was chosen to be TI's MSP430. For easily reproducing the results, the TI MSP430 experimenter board (MSP-EXP430F5438) defines the hardware test bed.

In order to produce the slimmest possible software stack, no operating system is used. Instead, the JVMs run "natively" or "barely" on the hardware. A few hardware-adapters to bridge the gap between JVM and hardware were added manually. This collection of hardware-adapters was named *ocapi* and was published as open source.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 181–188, 2013.

1.1 Comparison Parameters

The comparison parameters are features and attributes that characterize the JVM's behaviour during both, compile- or runtime. To relate results to earlier comparisons by Brouwers [2009] and Aslam [2010] and motivated by general requirements in software development, the following set is proposed.

- Standard Conformance (qualitative); Compatibility with the Java language specification by Gosling et al. [2005] and conformance to a Java Core API (J2ME, J2SE).
- Pointer Size (quantitative)
- Threads (qualitative)
- JNI Support (qualitative); How much of a Java native interface is supported.
- Tool Chain (qualitative); Complexity, availability, openness of the tool chain.
- Build Time (quantitative). Time necessary to build the JVM including the Java application from a clean project.
- Size in Memory (quantitative). Size of the final byte code and VM's native part.
- Runtime Performance (quantitative). Accomplishment of several reference-tasks.
- Power Consumption (quantitative). During runtime for a specific reference task.
- Energy Efficiency (qualitative). Use of idle and sleep modes.
- RAM Usage (quantitative). Peak requirement for volatile memory.

2 Field of Candidates

For the comparison, we seek suitable JVMs for the MSP430 microcontroller platform. They should be general enough to run different types of applications. Furthermore, they should be alive, i.e. actively supported and developed further. An extra bonus will be rewarded to open source JVM due to public availability and transparency. Anticipating the result of surveying the manifold of available JVMs, we select the following two for an in-depth comparison.

Takatuka is a matured JVM for sensor nodes by a research group with Faisal Aslam [2011]. The VM's hardware abstraction layer supported Atmel's AVR processors right from the beginning, while a port to MSP430 was added as a result of this work. Takatuka aims at providing J2ME CLDC. Depending on the application, it occupies less than 40KB of flash and about 4KB of RAM. Takatuka is open source and still developed further by an active community.

Darjeeling by Niels Brouwers [2009] and his team targets 16-bit microcontrollers like Atmel's ATmega128. It was implemented on different OS platforms, TinyOS and Contiki among them. The VM features a well-designed hardware abstraction allowing also "native" deployment on a suitable set of drivers. Darjeeling supports on-the-fly loading of Java modules. The memory requirements are very similar to those of Takatuka. The VM is an open source project with a moderately active community.

3 State of the Art – Earlier Comparisons

Darjeeling was examined by Brouwers [2009] on an ATmega128 at 8 MHz. Performance tests revealed an execution overhead of roughly two orders of magnitude when compared to a native C implementation. Thanks to elimination of string literals, the code size could be shrinked over jar files by a factor of two to six. A portability section compares memory usage of that same VM on different hardware platforms, Atmega128 and MSP430 among them.

Aslam [2010] et al. compared Takatuka, Sentilla and Darjeeling. Takatuka occupies less RAM than Sentilla on the JCreate (MSP430), and less than Darjeeling on the Mica2 (ATmega128) platform. Further, the impact of different byte code compaction algorithms on the runtime performance of Takatuka was analysed. Comparing the three JVMs compacted class files size shows that Takatuka in generally produces the smallest Java binary. Interestingly, the presented data suggest, that code produced for the MSP430 platform tends to be smaller than for the ATmega platform.

Concluding, there is only little comparison between Darjeeling and Takatuka on the same hardware. This is especially true for the MSP430 processor platform and parameters like runtime performance or energy efficiency.

4 Qualitative Comparison

This section discusses the JVM candidates subject to their qualitative features.

4.1 Standard Conformance and Features

Darjeeling provides neither floating point nor 64bit data types, while Takatuka does. Darjeeling can run multiple applications within separate infusions concurrently, but the current implementation provides insufficient control of this feature.

Both candidates do not fully comply with the J2ME CLDC specification. They do not support reflection mechanisms and lack a meaningful *java.lang.Class* implementation. However, Takatuka is a more complete subset of J2ME, than Darjeeling is.

4.2 Multi-threading

Both candidates support threading in which all threads share a single stack.

Unlike Takatuka, Darjeeling also supports multiple applications at a time. This is accomplished by a concept of static class file libraries called infusions which can use each other. Infusions render dead code removal as in Takatuka unfeasible (see Subsection 4.6). As the GC does not completely handle indirect references to unloaded infusions, a dedicated VM exception leaves it up to the application to cope with it.

Synchronization is supported by both JVMs. However, both appear to lack synchronized method calls. Fortunately this does not reduce the flexibility as such methods can easily be rewritten.

4.3 JNI Support and Tool Chain

Both JVMs do not comply with the JNI specification but instead, provide a proprietary interface for invoking native methods from Java. The native code is linked statically with the JVM binary.

Takatuka and Darjeeling both rely on Apache Ant as the build environment. They support at least one commonly accessible tool chain for each target system. Extending the Ant scripts to accommodate further compilers is a minor effort.

When targeting MSP430 hardware, Takatuka supports both, the TI tool chain as well as GCC. It optionally allows transferring the binary onto the node, but we preferred using the debugging software NoICE for this purpose, instead.

Darjeeling relies solely on GCC when targeted to the MSP430 platform.

4.4 Energy Efficiency

The MSP430 has five operating modes (LPM0…LPM4) each of them at a different power consumption level. Both JVMs do not take advantage of this technique, but always run the processor in the full-functional mode at the cost of the highest-possible power dissipation.

4.5 Garbage Collection and Memory Compaction

The Darjeeling approach is to minimize the GC and memory compaction effort with the introduction of a double ended stack by Brouwers [2009]. So it stacks reference types and non-reference types on either end of the stack. This renders runtime type analysis unnecessary, reducing the GC effort to O(n) and eliminating false positives. The costs are one byte for an additional stack pointer in each stack frame. It further requires the introduction of customized instructions, such that *pop* has to split into *apop* and *ipop* for references and non-reference types. Further the non-reference types are packed, that is, *byte* and *short* occupy only one or two bytes on the stack. Respectively, the *getfield* and *setfield* operations are replaced with typed versions for *byte*, *short*, *int* and *ref*.

Takatuka approaches to minimize running the GC at all by the introduction of an offline-GC. With a data flow analysis at compile time, Takatuka identifies objects that might still be reachable but are guaranteed not to be used again, as reported by Aslam [2011]. At those positions, customized instructions for explicit memory freeing are inserted. This increases the free RAM at runtime up to 66%.

4.6 Code Compaction

In order to achieve smaller class files, most JVMs for embedded devices apply a split VM architecture as originally introduced by Simon et al. [2006]. By transforming dynamic linking information into static linking of classes, the code size can be significantly reduced at the cost of losing Java reflection capabilities and, thus, dynamicity.

Darjeeling statically links classes within an infusion in a way that becomes merely a set of up to 255 methods, which are mutually called. An infusion header file keeps track of mapping of these methods to their names and classes at compile time. Using the header files, other infusions can correctly lookup the methods in this infusion.

Similarly, Takatuka statically links classes and methods into a monolithic tukfile. Besides stripping off class and method names from classes' constant pools, Aslam [2011] describes, how Takatuka globalizes the constant pools into a single one. By that, duplicated constants from different classes can be removed, saving further space.

Takatuka's dead-code removal mechanism eliminates all methods or classes that are never used in the flow path. So, a programmer can take advantage of a broad class and method library, e.g., almost complete J2ME CLDC and third party APIs.

As introduced by Aslam et al. [2010], Takatuka uses few of the 52 customizable Java byte code instructions for single instruction compaction (SIC) and multiple instructions compaction (MIC). The sorted, globalized constant pool allows for a constant pool access instruction with a single byte operand for the most frequently used constants. This reduces the code size in lots of places. With MIC, recurring sequences for instructions are combined into one instruction. The operands are just concatenated.

The Takatuka implementation uses a label-as-values approach, described by Ertl [2001] and Aslam [2011], which is more efficient than using a *switch* statement.

5 Quantitative Comparison

For quantitative comparison, both candidate JVMs where compiled to run "natively" on the MSP430 experimenter board. The CPU clock was configured at 16.7 MHz.

For the comparison to be most expressive, one application containing five characteristic test cases was written. *HelloWorld* is a well-known minimalistic case just printing the string "Hello World!" to the standard output. *IterativeSort* is to bubble-sort a 253 elements array to ascending order. The array is deterministically initialized with the values $\mathtt{i}^{17}$ `mod 253` (`i` being the field index). *MemoryGC* tests the memory allocation and garbage collection performance by allocating 50 byte arrays of size 100. In a second step, half of those are dropped and then newly allocated. *Arithmetics* is iteratively calculating the result of 69!, which is a rather computationally extensive task. Finally, *HanoiTowers* is a recursive implementation of the solution to the towers of Hanoi problem with 10 discs.

Numerical results of comparing the pointer size, VM build time and size in memory are given in **Table 1** while runtime performance, energy consumption and RAM usage are summarized in **Table 2** below.

Table 1. Pointer size, JVM build time and memory size for Takatuka and Darjeeling

Parameter	Takatuka	Darjeeling
Pointer size [bit]	8, 16, 24, 32	16
Build time: VM alone / optimizer [s]	33 / 8	17 / 16
Size in memory: Java / Native [byte]	8435 / 46076	19626 / 42336

Table 2. Runtime performance, energy consumption and dynamic memory consumption of Takatuka (TT) and Darjeeling (DJ) for different test cases

Test Case	Runtime [ms]		Energy [µJ]		RAM usage [byte]	
	TT	DJ	TT	DJ	TT	DJ
HelloWorld	21	73	242	836	1413	6446
IterativeSort	67766	11753	779822	134553	1725	1422
MemoryGC	11112	11004	127872	125978	7417	7912
Arithmetics	33	5	380	57	1337	846
HanoiTowers	14172	2252	163085	25782	2013	1142

5.1 Pointer and Stack Slot Sizes

Darjeeling defines a fixed 16-bit slot size and introduces 16-bit pendants for each 32-bit instruction. In an optimization step, this allows replacing low-range 32-bit instructions by corresponding 16-bit instructions reducing RAM usage and CPU cycles.

Takatuka introduced a variable slot size. At the programmers choice, it uses 32, 16 or even 8-bit slots, to waste as little RAM as possible. Obviously, there will be no savings unless the Java data types are shorter than 32 bits. Since the operations remain 32-bit, a smaller slot size likely comes at the cost of CPU cycles.

Also, Takatuka allows an adjustable reference/pointer size. Darjeeling references are always 16-bit. **Table 1** gives a comprised view on the supported pointer sizes.

5.2 Build Time

Build time was measured automatically by the build environment. Each measurement started from a clean project and extended to when the binary file was created. Instead of detailing certain fragments, investigations were restricted to only the optimizer tool and the total VM build time.

Averaged results over 10 runs for each JVM are given in **Table 1**.

The build time comparison is clearly advantageous for Darjeeling. For the VM alone, the advantage is 17 s versus 33 s for Takatuka and thus, about 50%. For a full build including the optimizer tool, Darjeeling needs 33 seconds, which is 8 seconds or 20% less time than Takatuka.

5.3 Size in Memory

Two characteristic measures comprise this attribute. The size of the VM's native part covers the lower-level part, including necessary hardware drivers. The byte code size covers all Java code, which is the high-level VM code plus the application code.

These static sizes were determined by the object file inspection utilities *size* and *objdump*. A comparison of results for both JVMs is given in **Table 1**.

It can be seen that Takatuka has a roughly 10% larger native part and produces about 50% less byte code. Obviously, the larger native part must be paid for more features and capabilities, but is more than compensated by the resulting byte code.

5.4 Runtime Performance

The *java.lang.System.currentTimeMillis()* function was deployed to measure runtime performance. Averaged results over four runs are given in **Table 2**.

For the one-liner *HelloWorld*, Takatuka is faster by roughly a factor of 4. With the memory-intensive task *MemoryGC*, both JVMs perform nearly the same. With all other tasks, Darjeeling is faster by a factor of about 6, demonstrating the pay-off of the simpler, feature-constrained JVM over a standard-like, multi-purpose JVM.

5.5 Power Consumption

The power consumption was deduced arithmetically from the product of supply voltage, the run of current drawn and the specific run time for each test case.

Supply voltage was measured independently of test cases using a digital multimeter. The average over 100 samples was 3.286 V with a standard deviation of 33 µV.

Current was measured and averaged over execution time using the same multimeter. For Takatuka, 5000 samples yield 3.502 mA with a standard deviation of 8.0 µA. Darjeeling's 2100 samples average to 3.484 mA with a standard deviation of 34.7 µA.

The resulting amount of energy, expressed in Micro Joule, is given in **Table 2**.

Except for the primitive *HelloWorld*, Takatuka consumes more energy than Darjeeling, sometimes by just 1.5%, in other cases by roughly a factor of 6. The dominant source for this effect is the advantageous runtime performance of Darjeeling.

5.6 RAM Usage

The peak RAM usage is an important indicator for dynamic memory requirements. Measurements were made by using the function *java.lang.Runtime.freeMemory()*, neglecting the distribution between VM and native, as well as for stack and heap. Results are given in **Table 2**.

Obviously, RAM usage depends very much on the type of application. While Takatuka is good at the extreme ends, i.e. one-liner and memory-intensive tasks, Darjeeling copes well with medium-complicated applications stressing indexing or looping.

6 Conclusion

Takatuka and Darjeeling were compared subject to qualitative and quantitative measures, relevant to WSN software development.

Takatuka makes points on the architectural side as it does not restrict the number of classes or methods, deals with various pointer sizes, introduces variable stack slot sizes and has a very efficient byte code compaction phase leading to a 50% reduction when compared to Darjeeling. Moreover, it supports 32bit floating point and 64bit integer data types and implements the J2ME CLDC specification more completely. Despite this fact, the size of the JVM's native part is still competitive.

Darjeeling convinces on the performance side of the competition. It has a 20% advantage in build-time and uses less RAM in most of the test cases. Subject to runtime performance and power consumption it displaces Takatuka by a factor of 6.

Both JVMs are well-suited for resource-constrained devices. They could significantly decrease power consumption by deploying power-saving run modes provided by the MSP430 hardware. We suggest tiny, performance critical applications to run on Darjeeling, while more complex, feature rich applications should prefer Takatuka.

Future work should relate upcoming JVMs with the given results. To increase expressiveness, the test suite should converge to standardized benchmarks, such as Ackermann, Dhrystone, Whetstone or Linpack. However, additional metrics will be needed to fairly assess missing features like floating point arithmetic.

Acknowledgement. Many thanks to Luminous Fennell and Faisal Aslam (Takatuka) as well as to Niels Brouwers (Darjeeling) for their cooperative support and on-going development of their respective JVM. We wish to thank Peter A. Bigot and John Hartman for responsively fixing the MSP430 GCC compiler and the NoICE debugger, respectively.

References

1. Aslam, F., Fennell, L., Schindelhauer, C., Thiemann, P., Ernst, G., Haussmann, E., Rührup, S., Uzmi, Z.A.: Optimized Java Binary and Virtual Machine for Tiny Motes. In: Rajaraman, R., Moscibroda, T., Dunkels, A., Scaglione, A. (eds.) DCOSS 2010. LNCS, vol. 6131, pp. 15–30. Springer, Heidelberg (2010)
2. Aslam, F.: Challenges and solutions in the design of a Java Virtual Machine for resource constrained microcontrollers. Dissertation for doctorate degree, Technical Faculty of the University of Freiburg, Germany (2011)
3. Brouwers, N.: A Java Compatible Virtual Machine for Wireless Sensor Networks. Master thesis, Faculty of Electrical Engineering, Mathematics and Computer Science of the Delft University of Technology, Netherlands (2009)
4. Ertl, M.A., Gregg, D.: The Behavior of *Efficient* Virtual Machine Interpreters on Modern Architectures. In: Sakellariou, R., Keane, J.A., Gurd, J.R., Freeman, L. (eds.) Euro-Par 2001. LNCS, vol. 2150, pp. 403–413. Springer, Heidelberg (2001)
5. Gosling, J., Joy, B., Steele, G., Bracha, G.: The Java^{TM} Language Specification, 3rd edn. Addison-Wesley Professional (2005)
6. Simon, D., Cifuentes, C., Cleal, D., Daniels, J., White, D.: Java^{TM} on the bare metal of wireless sensor devices: The squawk Java virtual machine. In: 2nd International Conference on Virtual Execution Environments, pp. 78–88. ACM, New York (2006)

Research on Opinion Formation of Microblog in the View of Multi-agent Simulation

Jianyong Zhang[1], Qihui Mi[2], Longji Hu[2], and Yue Tang[2]

[1] Beijing Institute of Technology, Beijing, CO 10081 China
zjy@baihc.com
[2] Huazhong University of Science& Technology, Wuhan, CO 430074 China
miqihui510@163.com

Abstract. In order to research the law of public opinion formation of microblog from the perspective of complex systems, agent's action rules are put up. Opinion updated equation is amended in accordance with affinity of participants of topics in microblog. Combined with the network topology of micro-blogging topics participants, the process of opinion formation is simulated. Disordered individual opinions emerged out of the system of ordering turns out to be the result, which is, forming public opinion. Examples are used to verify the validity of the model. It will make exploratory groundwork for further media dissemination and monitoring of public opinion online.

Keywords: Public Opinion of Micro-blogging, Multi-agent Simulation, Small World.

1 Introduction

Ever since 2010, microblog has become the most powerful online public media and the first choice for netizen to release information (Microblog Annual Report of China, 2010). According to the report released by DCCI(Data Center of China Internet), by the end of December 2012,the number of microblog users have reached 327 million in China,and more than 70% users tweet everyday. It can highlight the important position of microblog.The characteristics of microblog netizen and information dissemination have attracted researchers in different fields.

Microblog contains a great amount of information, and users are very complicated. It is difficult to use linear or macroscopic quantity theory to explore such a complex system. However, we can establish a simplified model based on microblog network society by using multi-agent model method. And then the emergence of macroscopic systems can be obtained through microscopic individual interactions. Generally, it is difficult for multi-agent modeling method to make accurate predictions of real system, but it can provide insight and understanding of the nature of system. From complex system perspective, this paper take the opinions formation in a particular topic of microblog as our study object with agent modeling method. We defines the interactive rules of individual's

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 189–198, 2013.

behavior combining with the topological structure of interpersonal network between participants of tweets. Thus we can derive the emergence of group opinion. Through investigation, the validity of the simulated opinion formation model of tweets discussed in microblog is verified. This study makes a foreshadowing for further transmission of microblog opinion and public opinion monitoring.

2 A Multi-agent Model of Microblog Opinions

2.1 Elements of the Model

This study introduces a social network topology between the participants of microblog tweets, combined with equation of opinion update and the affinity [1-3]. The formation of microblog opinion is simulated with agent modeling method. Before modeling, the elements related to the model and the relationship between them should be explained.

Explanation of Concepts. One of the study object is the tweets in microblog. The range of the microblog tweets is wide, containing topics which are discussed by interest groups in the micro-groups, as well as topics which cause the public discussion, such as "edible oil prices", " price war among e-commerce firms" and so on. We try to understand the formation of public opinion more extensively and directly by imitating it in these topics.

There is no authoritative and unified definition about microblog opinion.In this paper microblog is defined as the synthesis of opinions which are expressed by the majority participated in the microblog issues.

As to social network topology of microblog participants, the nature that network does not depend on specific location of the nodes and specific form of the edges is called topological properties of the network. The corresponding structure is called network topology [4]. The participants' social network topology refers to the specific real social relations among the members who participate in a specific topic. It is manifested as the relationship of "unilateral fan" and "bilateral fan" in microblog.

The relationship of "unilateral fan" which performs in topology structure refers to the concept of "in-degree". If individual A is an "unilateral fan" of individual B, meaning that A accept published tweets from B unilaterally, namely information flows from B to A. "bilateral fan" represents the relationship between users is two-way, namely A is B's fan and B is A's fan,too.

The model of dynamic opinion is mainly divided into two categories: discrete opinion and continuous opinion. These two models must meet following conditions that the system of dynamic opinion is closed and the participants are fixed during the evolution of opinion. Participants meet in some space. They exchange views, and then they update viewpoints according to certain rules.

As most of microblog users always have their own circles, the affinities change between different users. Based on the views of social impact model, this study believes that users' opinions are influenced by the affiliation between them, and they will make decisions depending on the affinities.

Agent in Microblog. Maes defined agent as "computing system that trying to achieve a set goal in complex dynamic environments" [5]. Generally speaking, agent should have the following attributes: autonomy, reactivity, initiative, sociability, evolutionary[6]. As opinions in microblog are complex, dynamic, relatively closed etc. , for simplicity, this paper only focuses on a particular microblog topic; ongoing interactions between the participants reflect the dynamic opinions on the topic. In addition, a specific number of participants will join in interaction. Agents have the attributes of autonomy, reactive, proactive and sociality.

In the model, evolution of public opinion is reflected by opinion updating equation which means that agents update their opinions based on the reinforcement learning.

2.2 Modeling of Microblog Opinion

Assumptions of the Model. For simplicity, we assume that the number of involved agents is a known constant. In the future, we will do further study considering dynamic number of agents.

We assume that the agents' social network is consistent with the small-world network topology, which means the shortest path length between each node is limited. But it is probably that two nodes are connected together through their own adjacent nodes, interpreting as limited path length and high degree of polymerization [4]. As a lot of researches have verified the characteristics of small-world in the virtual space, this assumption has sufficient scientific basis. Many of these researches are about the topological properties of social network in microblog specifically. The results show that the network has a high degree of polymerization and limited path length, in line with small-word properties and power-law distribution.

Figure 1 is a directed small-world graph with 200 nodes generated by simulation in Repast Simphony-2.0 where there are bidirectional connection nodes representing the relation of "bilateral fan" and unidirectional connection nodes representing the relation of "unilateral fan".

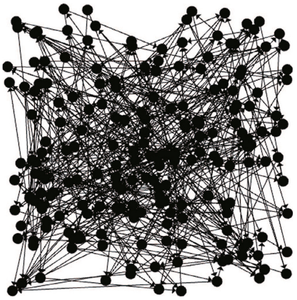


Fig. 1. Small-world graph generated by simulation

Action Rules of Agents. The rules here are established according to the opinion updating equation [7-9]. Considering that affinity changing between individuals in microblog topics need a long time, we assume that affinity does not change and then adjust the existing models.

When agents participate in a topic, their attitudes are not entirely clear. Therefore, we assume that the opinion of an individual on this topic is continuous. O_i represents agent i's opinion about a specific question. When O_i approximates 1, it implies that the majority of participants have negative opinions to this topic, otherwise, they hold a positive view. $\alpha_{ij} \in [0, 1]$ indicates the affinity between agent i and j. The larger the α_{ij} is, the closer the two agents are; In general, opinion formation can be completed in a relatively short time, and the affinity among agents is difficult to change within a short time. Thus, that α_{ij} is assumed as a constant.

At the beginning of the simulation, the number of agents in the system is k, and they form a small-world network topology. There is a threshold l_c. If agent A is agent B's fan, the affinity will be any value between $[0, l_c]$. If agent A and B are not directly connected, the affinity will be any value between $[0, l_c]$.

s agents are selected randomly as the initial agents, and their opinions are initialized randomly.

Every time step, a number of fans are selected from s agents' fans randomly, and their fans' opinions about this issue are initialized. The affinity between "fan agent" and "fancied agent" exist threshold f_c. If the affinity between the two agents exceeds the threshold f_c, "fan agent" will change its opinion based on the average value of the coupling point of view, otherwise opinions stay the same.

The opinion updating equation is below:

$$O_m^{t+1} = O_m^t + \mu \frac{\tanh(\zeta(\alpha_{mn} - \alpha_c)) + 1}{2}(O_n^t - O_m^t) \tag{1}$$

α_c is used to measure the affinity of opinions convergence, taking a constant value of 0. 5. μ is fixed to 0. 5, and its value does not affect the system's dynamical behavior, but affects the time to reach equilibrium. ζ is set to 1000 for converting the function tanh.

There is an additional case that agent m is a fan of two or more "fancied agents", and then agent m's opinion is decided by "fancied agent" who has minimum social distance. Select another agent n in all in-degree nodes of m; the rule of selecting n is to minimize the social distance which is influenced by opinion and affinity between n and m.

The equation of selecting n is below:

$$n = \arg[\min((1 - \alpha_{mj})|O_j^{t-1} - O_m^{t-1}|) + N(0, \sigma)], \forall j \in N : j \neq m \tag{2}$$

$N(0, \sigma)$ is a random element of normal distribution, generally called social temperature [10]. It is used to indicate the degree of randomness of the individual's behavior, also mean fluctuations of the group [11]. In addition, social temperature is also used to identify different social systems. For instance, people would not accept different opinions in a relatively conservative social system. They stick to their traditions and do not accept others' opinions. But in active or free social system, people are willing to accept different opinions, and they are more likely to change their opinions when affected by the views of others [12].

3 Multi-agent Simulation of Microblog Opinion

Supposing that the small-world network consists of fifty nodes, the simulation result is shown from Figure 2 to Figure 5.

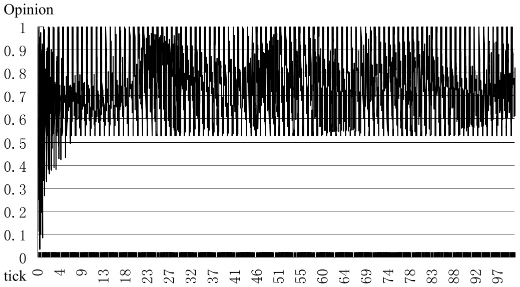


Fig. 2. The public opinion tends to be positive

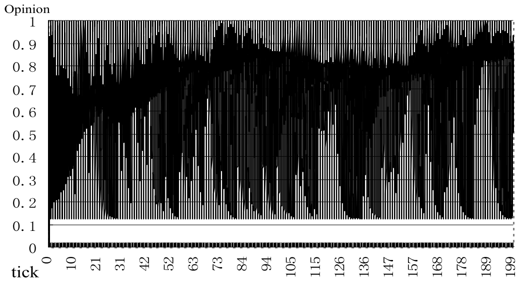


Fig. 3. The public opinion tends to be evenly distributed

At the beginning, we analyze the simulation results combined with the actual situation of public opinion in microblog. The opinions are evenly distributed at the start, then become steep and ultimately the polarization opinion arise. Initially, opinions distribute from 0 to 1 randomly, and neither side has a distinct advantage. As time goes by, the discussion in the group is deepening. Information is excavated continuously and opinions are constantly fused, and then divided. Both positive and negative opinions come into fierce conflict. In the competition, if one side is persuaded by the other side or one side comes into silence, the other side will hold the dominant position, emerging a consistent public opinion.

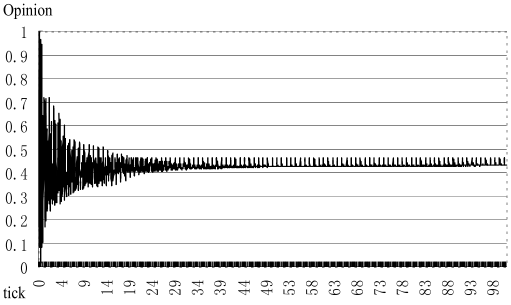


Fig. 4. The public opinion tends to be neutral

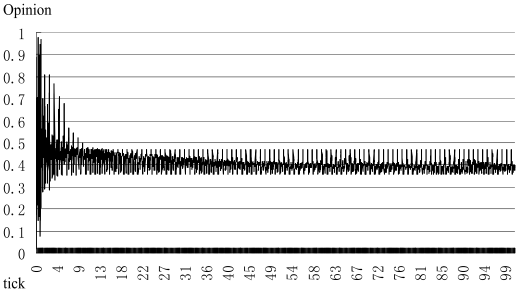


Fig. 5. The public opinion tends to be negative

Figure 3 shows the public opinion towards the certain issues is not formed. Generally speaking, the concentration goes toward intermediate point, but it does not reach the final unification. We can see that opinions of individuals are widely distributed. Figure 4 shows opinions are dispersed initially but ultimately tend to neutrality. In fact, topics in microblog are quite different. There are mainly three types of public opinion: message,concept and art [13]. As to the characteristics of message, “Firstly, people pass the message to each other in a short time. An opinion trend forms eventually because of people’s high interest in spreading some message. Secondly, people may not even become aware of the opinions tendency contained in the message. They merely want to tell others the facts they know. ” The characteristics of concept are manifested as “to varying degrees,the tendency of public opinion is directly expressed as agreement (sympathy), opposition (abhorrence), or indifferent (neutrality). ” Artistic forms are“ the tendency of public opinion is manifested through various genres such as literature, music, dance, painting and other art information. ” Generally speaking, most views of message are neutral. Views of concept are mixed, and mostly in criticism. Views of art are uncertain. Figure 4 shows the characteristic of message, while figure 2 and figure 5 show that of concept, and figure 3 shows that of art.

From the perspective of complex system, the simulation results represent that the system emerges orderly equilibrium from disordered initial state, showing an spontaneous order. Figures 2 to 5 represent the four results emerged from disordered initial state. The simulation result is neutral, neither good or bad. The simulation results confirm the explanation of emergence [14]. From the points of macro-level and the dynamic tendency, the trend of emergence is determined, which is the inevitable result of any system on the evolution of path dependence. And the specific pattern of emergence has nothing to do with the preset behavior. With the increase of complexity, the system's structure will not be identified completely before the actual emergence is completed. In addition, the emergence of system does not depend on initial state. When we regulate microblog rumor or public opinion which is not conducive to the development of a healthy society, the way to minimize the cost and maximum the utility is guide the public opinion before it forms.

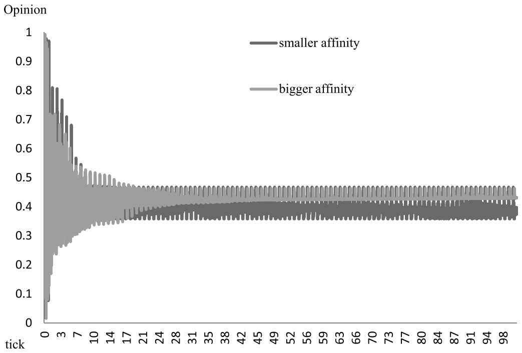


Fig. 6. Affinity's influence to public opinion dynamics

4 Empirical Analysis of the Multi-agent Model

We choose a sample from the list of hot topic of Sina Microblog. A topic can be heatedly discussed if an increasing number of netizens are involved. And then the topic becomes a hot one. We select a topic randomly in the hot topic list of Sina Microblog which is about that college students use performance art to protest unfairness in education which is caused by residency restrictions. The topic triggers a big discussion on the issue that household registration would lead to unfair education.

We sampled the opinions of participants randomly from 17:20 to 19:50 on October 8th, 2012 and gained 1065 samples in 150 minutes, including 880 effective samples published by 203 participants. Removing the 24 invalid participants (the content of discussion is not relevant to the topic, such as advertising, etc.), the effective number of participants is 179. The samples have the following features:

a). Groups update opinions continuously along with new participants joining in. And many participants have fan-relationship.

b). Opinions are widely exchanged; microblog users are persuaded by others or try to convince others. We use content analysis to encode 880 samples: 0. 1 means extreme opposition; 0. 2 means extreme questioned; 0. 3 means opposition; 0. 4 means questioned; 0. 5 means indifferent; 0. 6 means worried; 0. 7 means favor; 0. 8 means anger; 0. 9 means extreme anger. Statistical results are illustrated as follows.

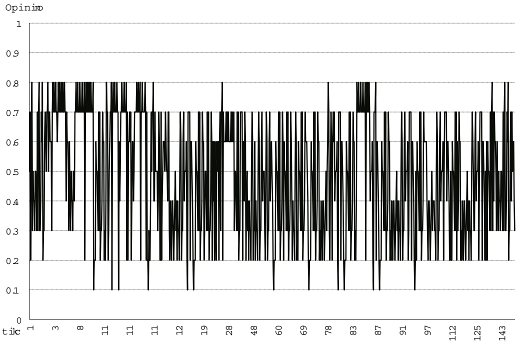


Fig. 7. Public opinion dynamics of empirical topics in Sina microblog

It can be seen in figure 7 that public opinion is not formed eventually. There are always different views.At the beginning of the discussion, more extreme opinions appear and the positive group takes dominant position. Negative views gradually gain the upper hand. They hold the view that the unfair education results from unbalanced economic development and government's imbalanced investment, not simply the household register system. However, with the involvement of some opinion leaders, remarks eventually tend to rational. Discussants prefer to think about deep-rooted reasons for education unfairness, such as unbalanced economic development and government's imbalanced investment, not simply the household register system.

Based on the participants and evolutionary time, we get figure 8 of opinion evolution.

Firstly, the model assumes that the initial state is random, so they distribute evenly from 0 to 1. Empirical evidence shows that public opinion is indeed at uncertain state in the beginning.

Secondly, the convergence rate of the model is faster than the empirical convergence rate. Both positive and negative opinions always exist in the entire simulation process, and they cannot come to an agreement in a short time, which is consistent with the empirical trend.

Thirdly, from the perspective of opinion evolution, there is a small number of extreme opinions in topics at the beginning. As individuals continue to interact, opinions update constantly, and they become more calm and rational.

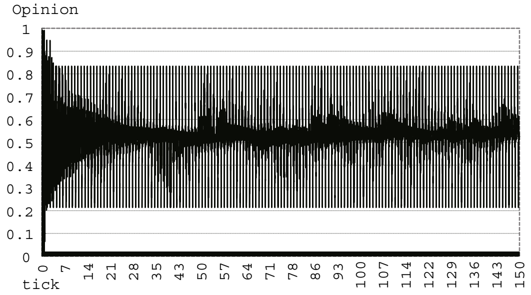


Fig. 8. Simulation of opinion dynamic in the topic of Sina microblog

Finally, public opinion converges toward middle ultimately. While there are always different voices, public opinion tends to the middle point of view, such as " “anxiety”, “approval”, etc.

5 Conclusions

We assume that social network in microblog topics has a small-world topology. People select the other to retweet according to the affinity between them. We set the opinion updating equation to update agents' opinions on a specific topic. Then, there is the emergence of public opinion formation. The examples show that the study of public opinion formation mechanism in microblog space combined with network topology is feasible at some extent. The simulation results show that the microblog issue is prone to polarization in the formation of public opinion. The main reason is the emotional or rational guides from opinion leaders leading to the silence spiral. The empirical result indicates that there exists the spiral of silence and it is consistent with the simulation results.

The intelligent agent simulation of microblog opinion combined with opinion update equation is a new perspective to study the formation of online public opinion. We can still get some realistic revelation from the simulation results and conclusions above.

Because virtual space has some specific characteristics, such as hide and virtuality, public opinion is easy to fall into chaos so that it goes out of control. When some topic attracts public attention in the early stage, the opinions are relatively dispersed. There are no opinion leaders and opinion tendency is not formed. It is the best opportunity to guide the public opinion by grasping the development direction at this moment and costs least. It is important to seize the opportunity to attack illegal activities such as spreading of rumors and false information, stirring of public scare.

In the first stage for hot topics, it is susceptible to incite the masses and make the extreme opinions occupy the high ground of public opinion. It is important to value the role of opinion leaders in microblog and with their help a rational and healthy online environment can be more easily established.

The future study will set various of agents in the model who have different natures and action rules. Besides, the opinion updating equation will be further improved. A more scientific and effective model will be used to simulate the formation and spread of public opinion in microblog.

Acknowledgement. Thanks to the support from the National Nature Science Fund of China.

References

1. Bagnoli, F., Carletti, T., Fanelli, D., Alessio, G., Guazzini, A.: Dynamical Affinity in Opinion Dynamics Modeling. Physics 2, 204 (2007)
2. Carletti, T., Fanelli, D., Guarino, A., Bagnoli, F., Guazzini, A.: Birth and Death in a Continuous Opinion Dynamics Model: the Consensus Case. Physics 1, 4062 (2008)
3. Grabowski, A.: Opinion Formation in a Social Network: the Role of Human Activity. Physica A 388, 961–966 (2009)
4. Huang, P., Zhang, Liu, X.J.G.: The Present Research Situation and Forecast of the Small World Network. Journal of Information 4, 66–68 (2007)
5. Liu, J.M.: Autonomous Agents and Multi-agent Systems: Explorations in Learning, Self-organization and Adaptive Computation, p. 8. World Scientific, Hong Kong (2001)
6. Singh, M.P.: Multiagent Systems. LNCS, vol. 799. Springer, Heidelberg (1994)
7. Carletti, T.: On the Evolution of a Social Network. Physics 1, 0689 (2010)
8. Weisbuch, G., Deffuant, G., Amblard, F., Nadal, J.P.: Meet, Discuss and Segregate! Complexity 7 (2002)
9. Righi, S., Carletti, T.: How Opinion Dynamics Shapes Social Networks Topology. In: Proceedings of AI*IA Conference, Italian (2009)
10. Radosz, A., Ostasiewicz, K., Hetman, P., Magnuszewski, P., Tyc, M.H.: Social Temperature Relation Between Binary Choice Model and Ising Model. In: An International Conference. AIP Conf. Proc.: Complexity, Metastability, and Nonextensivity, vol. 965, pp. 317–320 (2007)
11. Janusz, A.H., Kacperski, K., Schweitzer, F.: Phase Transitions in Social Impact Models of Opinion Formation. Physica A 285, 199–210 (2000)
12. Nowak, A., Szamrej, J., Latané, B.: From Private Attitude to Public Opinion: a Dynamic Theory of Social Impact. Psychological Review 9, 362–376 (1990)
13. Lidan, C.: Public opinion: the Study of the Guide of Public Opinion, pp. 98–100. China Broadcasting & TV Publishing House, Beijing (1999)
14. Shi, Y.: Disciplines Development of Emergence and Systematic Thinking. Journal of Systems Science 14, 58–63 (2006)

Implementation of Cloud IaaS for Virtualization with Live Migration

Chao-Tung Yang, Kuan-Lung Huang, William Cheng-Chung Chu,
Fang-Yi Leu, and Shao-Feng Wang

Department of Computer Science, Tunghai University, Taichung, 40704, Taiwan ROC
{Ctyang,cchu,leufy}@thu.edu.tw, peter760504@gmail.com

Abstract. Virtualization is a process of manifestation of the logical group or subset of computer resources in computer science. However, virtualization we mentioned in this paper is "platform virtualization". It is VM (virtualization machine) as we often named. Benefits of virtualization are numerous and it is considered to be a previous procedure to learn before adapting cloud technology for the enterprise. The constructed environment in this paper has implemented virtualization for further experiment. The main subject of this paper is how to construct virtualization in the cloud with integration of KVM and OpenNebula for users. It provides private cloud solutions for enterprises or organizations and focuses on IaaS of three services in cloud. This system can reduce the complexity of accessing the cloud resources through users' interface. That is to say it is easy to manage deployment of the VMs by the web-based users' interface. The paper contains of live migration data measurement, comparison of physical machines and virtual machines, and analysis results. In the experimental environment, we prove that the performance of full virtualization is closer to the physical machine as well.

Keywords: Cloud Computing, IaaS, Live migration, VM provision.

1 Introduction

With cloud computing is a highly important topic in the recent IT world, and there are many of the different services developed. In actually cloud computing is not a new technology; it is a new concept [1, 10, 19, 23, 21, 22, 24, 28]. The early stages of the laboratory started in the creation and development of gird computing cluster and other distributed computing technologies and related issues, for the vigorous development in recent years is also very interested in cloud computing [13, 15]. There are many companies currently offer a cloud of related services, like Google, Amazon, Yahoo! other companies, tens of thousands of servers used to construct a large-scale computing resources, and provides a variety of services previously not available such as: large storage space, a huge amount of computing power, no need to download the online features such as edit view, for their own local computing and storage resources are limited, users can access via the Internet computing resources they need.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 199–207, 2013.

This paper focused on the cloud computing infrastructure, particularly virtual machines and physical monitoring component and cloud computing infrastructure especially virtualization[3, 4, 5, 6, 7, 8, 9, 11, 12, 14, 18]. The goal is to implement a system which can manage and deploy VM for cloud system. The information which be supplied by system can be monitored include CPU utilization, disk usage, virtual machine space, memory usage. This system also uses a mechanism for Migration, when a problem occurs, the administrator can shift the user's virtual machine to another physical machine operation, and the user will not feel any abnormalities. Meanwhile, this paper also carried on the system performance test using the KVM [2, 20, 26, 27, 29, 30, 31, 32].

2 Background Review

2.1 Virtualization

Virtualization is simply the logical separation of requests for some services from the physical resources where the service is actually provided. In practical terms, virtualization allows applications, operating systems, or system services in a logically distinct system environment to run independently of a specific physical computer system. Obviously, all of these must run on a certain computer system at any given time, but virtualization provides a level of logical abstraction that liberates applications, system services, and even the operating system that supports them from being tied to a specific piece of hardware. Virtualization, focusing on logical operating environments, makes applications, services, and instances of an operating system portable across different physical computer systems. Virtualization can execute applications under many operating systems, manage IT more efficiently, and allot computing resources with other computers [2].

2.2 Virtualization Management

The virtual machine is not only available user interface, but it is the computer with actual loading. Management of virtual machines and management of physical systems are equally important. Virtualization Management includes a set of integrated management tools, can be minimize complexity and simplify the operation. It should centrally manage physical and virtual IT infrastructure, increased server utilization, but also across multiple virtualization platforms to optimize dynamic resources.

2.3 Live Migration

By adjusting the resources with virtual technology, to make provided services to closer to the actual needs of different users. Live migration of virtual machines is an important technology. The live-migration of VM can transfer VM to other physical servers without shutdown. It achieve the high HA ability with the non-stop services.

Live migration is the movement of a virtual machine from one physical host to another while continuously powered-up. When this process functions properly, it shows no noticeable effect from the end-user's point of view. Live migration allows an administrator to take a virtual machine offline for maintenance or upgrading without subjecting the system's users to downtime. When resources are virtualized, additional management of VMs is needed to create, terminate, clone or move VMs from host to host. Migration of VMs can be done off-line (the guest in the VM is powered off) or on-line (live migration of a running VM to another host).

One of the most significant advantages of live migration is that it facilitates proactive maintenance. If an imminent failure is suspected, the potential problem can be resolved before service disruption. Live migration can also be used for load balancing, in which work is shared among computers to optimize the usage of available CPU resources.

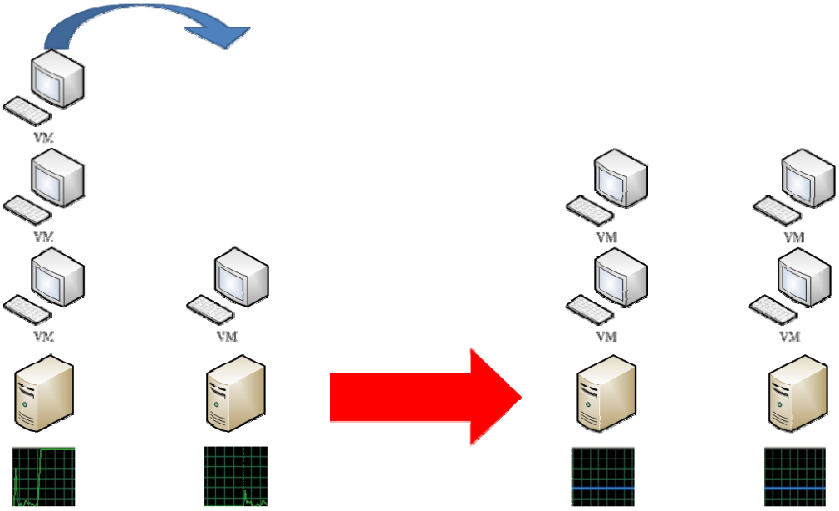


Fig. 1. The concept of Live Migration

2.4 Related Work

There are many researches about the live migration and some have been proposed to improve the down time of live migration. In this section, some related works will be presented briefly.

The most important paper about live migration is. Discussion in this paper is also the implementation of the live migration in Xen. To avoid difficulties of the migration in the process level and residual dependency, instead of the process, the VM with its application regarded as the migration unit. The main goal of live migration is to reduce the time of down time and total migration time.

- Down time: The time of shutdown during migration
- Total migration time: The VM can run in the target host without error and source host discarded the old VM.

The method of live migration in this paper is "Pre-Copy ". It's the same with KVM. Another important paper about live migration is "Post-Copy Live Migration of Virtual Machines". The method proposed in this paper is "Post-Copy". The purpose of pre-copy method is trying to reduce the down time. But the purpose of post-copy method is trying to reduce the total migration time.

3 System Implementation

3.1 System Architecture

Besides managing individual VMs' life cycle, we also designed the core to support services deployment; such services typically include a set of interrelated components (for example, a Web server and database back end) requiring several VMs. Thus, we can treat a group of related VMs as a first-class entity in OpenNebula. Besides managing the VMs as a unit, the core also handles the context information delivery (such as the Web server's IP address, digital certificates, and software licenses) to the VMs. In Figure 2, it shows the system architecture perspective. According to the previous works, we build a cluster system with OpenNebula and also provide a web interface to manage virtual machines and physical machine. Our cluster system was built up with three homogeneous computers; the hardware of these computers is equipped with Intel i7 CPU 2.8 GHz, four gigabytes memory, 500 gigabytes disk, CentOS operating system, and the network connected to a gigabit switch.

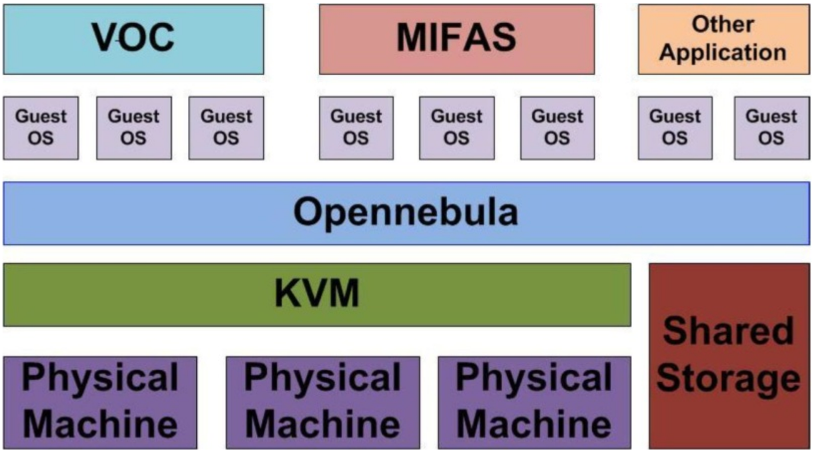
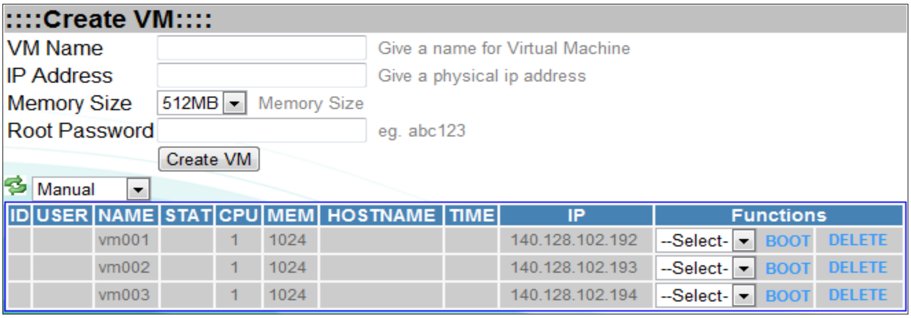


Fig. 2. System architecture

3.2 Management Interface

We design a useful web interface for end users and it is indeed the fastest and the friendliest to Implementation virtualization environment. The authorization mechanism, through the core of the web-based management tool, it controls and manages both physical machine and VM life-cycle. The entire web-based management tool includes physical machine management, virtual machine management and performance monitor. In Figure 3, it sets the VM attributes such as memory size, IP address, root password and VM name, etc. It includes the life migrating function as well. Life migration means VM can move to any working physic machine without suspending in-service programs. Life Migration is one of the advantages of OpenNebula. Therefore, we could migrate any VM what we want under any situation because we have a DRA mechanism to make the migration function more meaningfully.



::::Create VM::::

VM Name		Give a name for Virtual Machine
IP Address		Give a physical ip address
Memory Size	512MB	Memory Size
Root Password		eg. abc123

ID	USER	NAME	STAT	CPU	MEM	HOSTNAME	TIME	IP	Functions		
		vm001		1	1024			140.128.102.192	--Select-	BOOT	DELETE
		vm002		1	1024			140.128.102.193	--Select-	BOOT	DELETE
		vm003		1	1024			140.128.102.194	--Select-	BOOT	DELETE

Fig. 3. Virtual Machines Manager

4 Experimental Results

HPCC performance testing -We implement HPCC performance measurement in the same hardware with 4 different platforms.

- PM (Physical Machine)
- KVM
- VM Ware 8.0 in Linux
- VM Ware 8.0 in Win7 Professional

The host OS of (1), (2) and (3) is CentOS6.2 x64 (Desktop version). We can compare the differences of physical machine, KVM and VMware. And we choose the VM Ware because VMWare and KVM are the technique of full-virtualization. Moreover, we can compare the pop technique of full-virtualization in Linux and Windows.

4.1 Experimental Environment

The host list of the test environment as below.

Table 1. Environment Specification

	CPU	Memory	Disk	Network	OS	Software
Node01	i7 CPU 2.8 GHz	4GB	500GB	Gigabits	CentOS6 X64	OpenNebula Front End/Host
Node02	i7 CPU 2.8 GHz	4GB	500GB			OpenNebula-Host
Node03	i7 CPU 2.8 GHz	4GB	500GB			

4.2 Experimental Results

Experiment: the Comparison of Physical Machine- KVM and VMware

To avoid running service on testing environments which affects machine performance, we do not pick up the host from cloud platform. The HPCC testing is compared to physical machine, KVM and VMW which are on the same with independent hardware VMs. We compared the differenced of performance and computation by controlling the number of CPU. We list the hardware and software specification in the following:

Table 2. Environment of Experiment 3 Specification

	Host OS	Guest Os	H/W SPEC		
Physical Machine	CentOS6.2 x64 (Desktop)	N/A	Intel E3400 2.6GHz dual cores	Ram 8GB	HDD 500GB
KVM	CentOS6.2 x64 (Desktop)	CentOS6.2 x64 (Desktop)			
VM Ware 8.0	CentOS6.2 x64 (Desktop)	CentOS6.2 x64 (Desktop)			
VM Ware 8.0	Win7 Professional	CentOS6.2 x64 (Desktop)			

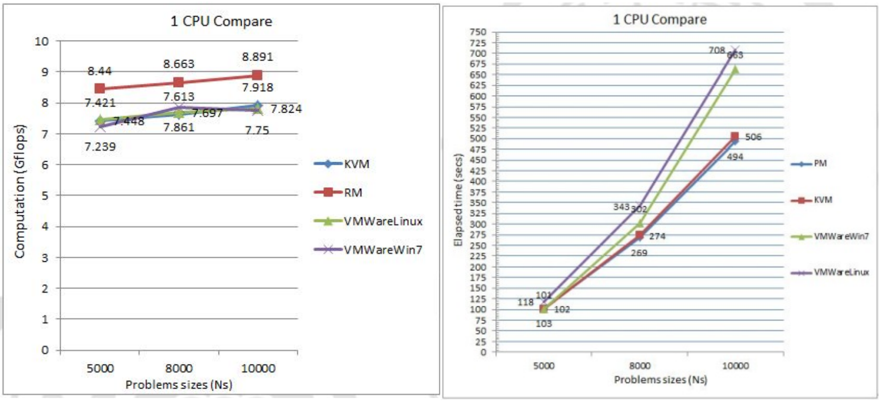


Fig. 4. The results of HPCC by one CPU in different platform

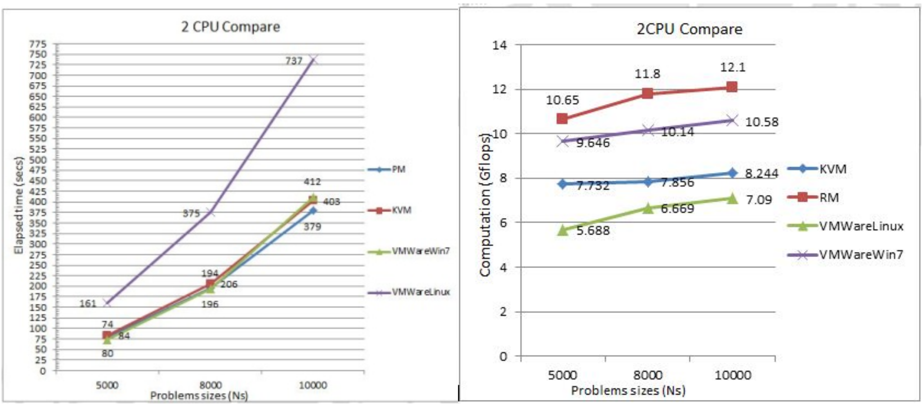


Fig. 5. The results of HPCC by two CPUs in different platform

Shown in Figure 4 for single core case, we find that the gap of these four platforms is small in low amount of computation (Ns =5000). By increasing the amount of computation, the tested value of KVM is closed to PM. However, the tested value of VMware lags far behind the KVM and PM. The result of the order is PM, KVM, VMware in WIN7, VMware in Linux. On the other hand, the gap of computation in different VMs is small other than the physical machine in Figure 4.

5 Conclusions

This paper implemented a cloud of KVM infrastructure and monitoring website, which offers users to apply for the use and monitoring of VM state, and the main page with easy to understand the type, the user in the application and monitoring, can obtained through the needs of the most simple steps in order to user friendly. Unlike the past, the usage of Xen as the virtualization technology, this paper tries to use KVM virtualization technology as a major. In addition, this paper also tested live migration and implementation of the efficiency of the test, although there is still a gap from the best performance, but the final results were very satisfactory.

Acknowledgements. This work is sponsored by Tunghai University “The U-Care ICT Integration Platform for the Elderly – No.101GREEnS004 – 2”, Aug. 2012. This work was supported in part by the National Science Council, Taiwan ROC, under grant numbers NSC 101-2218-E-029-004, NSC 101-2221-E-029-014, and NSC 101-2622-E-029-008-CC3.

References

1. Nagarajan, A.B., Mueller, F., Engelmann, C., Scott, S.L.: Proactive fault tolerance for HPC with Xen virtualization. In: Proceedings of the 21st Annual International Conference on Supercomputing, Seattle, Washington, June 17-21, pp. 23–32 (2007)

2. Zhang, B., Wang, X., Lai, R., Yang, L., Wang, Z., Luo, Y., Li, X.: Evaluating and Optimizing I/O Virtualization in Kernel-based Virtual Machine (KVM). In: Ding, C., Shao, Z., Zheng, R. (eds.) NPC 2010. LNCS, vol. 6289, pp. 220–231. Springer, Heidelberg (2010)
3. Matthews, C., Coady, Y.: Virtualized Recomposition: Cloudy or Clear? In: ICSE Workshop on Software Engineering Challenges of Cloud Computing, May 23, pp. 38–44 (2009)
4. Tseng, C.-H., Yang, C.-T., Chou, K.-Y., Tsaur, S.-C.: Design and Implementation of a Virtualized Cluster Computing Environment on Xen. Presented at the Second International Conference on High Performance Computing and Applications, HPCA (2009)
5. Waldspurger, C.A.: Memory Resource Management in VMware ESX Server. SIGOPS Oper. Rev. 36(SI), 181–194 (2002)
6. Bellard, F.: Qemu, a fast and portable dynamic translator. In: Proceedings of the USENIX 2005 Annual Technical Conference, FREENIX Track, p. 41 (2005)
7. Kecskemeti, G., Terstyanszky, G., Kacsuk, P., Nemetha, Z.: An approach for virtual appliance distribution for service deployment. Future Generation Computer Systems 27(3), 280–289 (2011)
8. Raj, H., Schwan, K.: High Performance and Scalable I/O Virtualization via Self-Virtualized Devices. In: The Proceedings of HPDC 2007, pp. 179–188 (2007)
9. Van, H.N., Tran, F.D., Menaud, J.-M.: Autonomic virtual resource management for service hosting platforms. In: ICSE Workshop on Software Engineering Challenges of Cloud Computing, May 23, pp. 1–8 (2009)
10. Oi, H., Nakajima, F.: Performance Analysis of Large Receive Offload in a Xen Virtualized System. In: Proceedings of 2009 International Conference on Computer Engineering and Technology (ICCET 2009), Singapore, vol. 1, pp. 475–480 (January 2009)
11. Smith, J.E., Nair, R.: The Architecture of Virtual Machines. Computer 38(5), 32–38 (2005)
12. Shafer, J., Willmann, P., Carr, D., Menon, A., Rixner, S., Cox, A.L., Zwaenepoel, W.: Concurrent Direct Network Access for Virtual Machine Monitors. In: The Second International Conference on High Performance Computing and Applications, HPCA, pp. 306–317 (2007)
13. Kertesz, A., Kacsuk, P.: Grid Interoperability Solutions in Grid Resource Management. IEEE Systems Journal 3(1), 131–141 (2009)
14. Adams, K., Agesen, O.: A Comparison of Software and Hardware Techniques for x86 Virtualization. In: ASPLOS-XII: Proceedings of the 12th International Conference on Architectural Support for Programming Languages and Operating Systems, pp. 2–13. ACM Press, New York (2006)
15. Rodero-Merino, L., Vaquero, L.M., Gil, V., Galán, F., Fontán, J., Montero, R.S., Llorente, I.M.: From infrastructure delivery to service management in clouds. Future Generation Computer Systems 26(8), 1226–1240 (2010)
16. Milojičić, D., Llorente, I.M., Montero, R.S.: OpenNebula: A Cloud Management Tool. IEEE Internet Computing 15(2), 11–14 (2011)
17. Luszczek, P., et al.: Introduction to the HPC Challenge Benchmark Suite. LBNL-57493 (2005)
18. Endo, P.T., Gonçalves, G.E., Kelner, J., Sadok, D.: A Survey on Open-source Cloud Computing Solutions. In: VIII Workshop em Clouds, Grids e Aplicações, pp. 3–16 (2011)
19. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the Art of Virtualization. In: SOSP 2003: Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles, pp. 164–177. ACM Press, New York (2003)

20. Qumranet, White Paper: KVM Kernel-based Virtualization Driver, Qumranet. Tech. Rep. (2006)
21. Montero, R.S., Sotomayor, B., Llorente, I.M., Foster, I.: Virtual Infrastructure Management in Private and Hybrid Clouds. IEEE Internet Computing 13, 16–23 (2009)
22. Soltesz, S., Potzl, H., Fiuczynski, M.E., Bavier, A., Peterson, L.: Container-based Operating System Virtualization: A Scalable, High-performance Alternative to Hypervisors. In: EuroSys 2007, pp. 275–287 (2007)
23. von Hagen, W.: Professional Xen Virtualization. Wrox Press Ltd., Birmingham (2008)
24. Emeneker, W., Stanzione, D.: HPC Cluster Readiness of Xen and User Mode Linux. In: 2006 IEEE International Conference on Cluster Computing, pp. 1–8 (2006)
25. Li, Y., Yang, Y., Ma, N., Zhou, L.: A hybrid load balancing strategy of sequential tasks for grid computing environments. Future Generation Computer Systems, 819–828 (2009)
26. Zhang, X., Dong, Y.: Optimizing Xen VMM Based on Intel Virtualization Technology. In: 2008 International Conference on Internet Computing in Science and Engineering (ICICSE 2008), pp. 367–374 (2008)
27. Dong, Y., Li, S., Mallick, A., Nakajima, J., Tian, K., Xu, X., Yang, F., Yu, W.: Extending Xen with Intel Virtualization Technology. Journal, ISSN, Core Software Division, Intel Corporation, 1–14 (August 10, 2006)
28. Hai, Z., et al.: An Approach to Optimized Resource Scheduling Algorithm for Open-Source Cloud Systems. In: 2010 Fifth Annual ChinaGrid Conference (ChinaGrid), pp. 124–129 (2010)
29. Chu, W.C.-C., Yang, C.-T., Lu, C.-W., Chang, C.-H., Chen, J.-N., Hsiung, P.-A., Lee, H.-M.: Cloud Computing in Taiwan. IEEE Computer 45(6), 48–56 (2012)
30. Yang, C.-T., Wang, S.-F., Huang, K.-L., Liu, J.-C.: On Construction of Cloud IaaS for VM Live Migration Using KVM and OpenNebula. In: Xiang, Y., Stojmenovic, I., Apduhan, B.O., Wang, G., Nakano, K., Zomaya, A. (eds.) ICA3PP 2012, Part II. LNCS, vol. 7440, pp. 225–234. Springer, Heidelberg (2012)
31. Yang, C.-T., Chen, B.-H., Chen, W.-S.: On Implementation of a KVM IaaS with Monitoring System on Cloud Environments. In: Kim, T.-h., Adeli, H., Fang, W.-c., Vasilakos, T., Stoica, A., Patrikakis, C.Z., Zhao, G., Villalba, J.G., Xiao, Y. (eds.) FGCN 2011, Part I. CCIS, vol. 265, pp. 300–309. Springer, Heidelberg (2011), doi:10.1007/978-3-642-27192-2_36
32. Yang, C.-T., Cheng, H.-Y., Huang, K.-L.: A Dynamic Resource Allocation Model for Virtual Machine Management on Cloud. In: Kim, T.-h., Adeli, H., Cho, H.-s., Gervasi, O., Yau, S.S., Kang, B.-H., Villalba, J.G. (eds.) GDC 2011. CCIS, vol. 261, pp. 581–590. Springer, Heidelberg (2011), doi:10.1007/978-3-642-27180-9_70.

Security Considerations in Cloud Computing Virtualization Environment

Sang-Soo Yeo[1] and Jong Hyuk Park[2,*]

[1] Division of Computer Engineering, Mokwon University, Daejeon 302-729, Korea
[2] Dept. of Computer Science & Engineering, SeoulTech, Seoul 139-743, Korea
sangsooyeo@gmail.com, parkjonghyuk1@hotmail.com

Abstract. Almost cloud service providers are having their own architectures for providing a variety of cloud services to their customers, cloud service clients. These various architectures and services increase the complexity of security management policies, frameworks, and systems, because they would require different aspects of security solutions for their own architectures and services. Consequently, such compatibility issues make us difficult to design a common security framework, security management system, or security evaluation system. Recognizing the need to solve such issues, we analyze common security elements to be required for cloud computing virtualization, and identify requirements for information protection in this paper. We also identify possible threats that may occur depending on different functions and roles over the cloud virtualization environments, and define security elements and requirements to deal with those issues. We show a set of common directions or approaches to prevent any possible treats to cloud computing, and provide more efficient and systematic method of managing and operating cloud computing system.

Keywords: Cloud Computing Services, Cloud Computing Architecture, Virtualization Security Layer, Security Requirement.

1 Introduction

Cloud computing system is a large scale dispersed computing paradigm based on economy of scale in which large IT resources such as storage, platform, and service are virtualized and dynamically expanded in which users can use through the interest with needed amount. The current clouding computing market is passing its early introduction phase and is already being used in web mails, blogs, web hard service, web hosting service, and etc. However, development of applications and services fitting user demand levels, expansion of linkage with original systems, relief of worries on security must be solved first to enter development phase. International market investigation organization IDC has investigated IT executives and answered that security must be solved for usage of cloud computing service. For this reason, research on cloud computing technology and security is being actively conducted nowadays, but

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 208–215, 2013.

problems on security element research and specific correspondence method research has occurred by different cloud architecture. Also, there exists flaw of mutual compatibility, linkage problems and deduction, application of security elements on virtualization which is the core function of cloud computing. For effective supplementation of these problems, architecture reflecting mutual functions of each vendor and organization architecture should be composed, virtualization security layer role and functions should be analyzed, and research on possible threats and correspondence plans is needed [1-3].

In this paper, we composes cloud computing architecture of common concept to present correspondence methods and information protection demands for possible threats by roles and functions related to virtualization and improvement of security of virtualization environments. In section 2, we introduce various researches on original cloud computing service architecture and its security, and we categorize them into several layers by its functionalities and roles. In section 3, we present well-known security issues for cloud computing mentioned in the prior research articles. In section 4, we analyze the existing research works and identify important security considerations and monitoring system requirements for cloud computing virtualization environment. Lastly in section 5, the conclusion and contribution of this paper is described.

2 Various Cloud Computing Architecture Models

In cloud computing standardization aspect, there is difficulty of application on mutual compatibility, portability, security between systems due to vendor dependence of various cloud computing platforms by provided platform dependent security solutions by vendors. For limited support where original system or mutual software is provided, stability of the system could be contained because of patches and enough security policies. However for cloud environment using integration of resources and virtualization technology, there was limit only with original patches or security policies. Therefore, stability of total system must be prior that analysis on mutual virtualization layer composition and functionality specifically reflecting core technology of cloud computing virtualization [1].

We compared and analyzed commonly used cloud computing building platforms such as cloud computing virtualization open source OS Xen, representative cloud services by IBM, Microsoft, RedHat and international organizations related to cloud computing or services, in order to find common characteristics by each layer. Composition of architecture presented by each vendor and organization was categorized from Layer 1 to Layer 6 shown in Table 1 and Table 2 by function and role by architecture layer. For common concept in Table 1, Layer 1 is physical equipment and facility. Layer 2 is virtualization of physical resources such as server, storage, and network. Layer 3 is providing and managing integrated and virtualized resources. Layer 4 is providing service by adding applications and middleware using allocated resources. Additionally, Layer 5 and 6 of architecture by the cloud service platform were categorized by additional middleware or application composition to provide PaaS and SaaS.

Through analysis of cloud computing architecture of Xen, IBM, MS, RedHat, CSA, IETF, and DMTF, we categorized all functions of cloud services into six (6) layers as shown in Table 1 and 2.

- Layer 1 is the infrastructure of cloud computing and includes data processing servers, inner/outer communication networks, data storage sets, and other physical resources.
- Layer 2 implements virtualization for providing capsuled resource views by integrating and abstracting physical resources such as server, storage, network into integrated.
- Layer 3 provides integrated resources such as virtual machines (VM), cluster, logical file system, and database system to upper layers.
- Layer 4 presents specialized tools and applications to users with integrated resources and allocation platforms.
- Layer 5 includes resource integration and middleware environment for providing application development frameworks, programming languages, tool functions (PaaS).
- Layer 6 is built on IaaS and PaaS stacks, and presents independent operating environments to users.

Table 1. Cloud computing categorization for Xen, IBM, MS, and RedHat [1]

	Xen	*IBM*	*MS*	*RedHat*	*common concepts*
Layer 1	Physical Host Hardware	System Resources	Servers Storages Networks	Physical Hardware (Servers, Storage, Networking)	Physical resources
Layer 2	Xen Hypervisor	Virtualized Infrastructure	Virtualization	RHEV (virtual servers, storage, networks, clients, applications, middleware)	Building virtualized resources
Layer 3	dom0 (Host Domain) domU (Guest Domain)	Virtualized Application	Virtualized Inframanagement, Cloud Service Platform, Infrastructure Service Platform	JBoss, Websphere Windows, RHEL	Providing and managing virtualized resources
Layer 4		Service Management	Cloud Service Presentation	Thousands of Certified Applications	Providing tools and applications on the virtualized resources

Table 2. Cloud computing categorization for CSA, IETF, and DMTF [1]

	CSA (Cloud Security Alliance)			***IETF (Cloud Reference Framework)***	***DMTF (Cloud Service Reference Architecture)***	***common concepts***
	IaaS	**PaaS**	**SaaS**			
Layer 1	Facilities	Facilities	Facilities	Physical Resource Layer	Firmware, Hardware	Physical resources
	Hardware	Hardware	Hardware			
Layer 2	Abstraction	Abstraction	Abstraction	Resource Abstract & Virtualization Layer	Software Kernel (OS, VM Manager)	Building virtualized resources
Layer 3	Core Connectivity & Delivery	Core Connectivity & Delivery	Core Connectivity & Delivery	Resource Control Layer	Virtualized Resources, Virtual image	Providing and managing virtualized resources
Layer 4	APIs	APIs	APIs	Application/ Service Layer	Cloud Applications	Providing and managing virtualized resources
Layer 5		Integration & Middleware	Integration & Middleware		SaaS PaaS IaaS	Resource integration for PaaS
Layer 6			Data, Metadata, Content Applications APIs Presentation Modality, Presentation Platform			Contents provisioning for SaaS

3 Well-Known Security Issues for Cloud Computing

We can find well-known security issues and countermeasures against them for data privacy and data protection in cloud computing environment in [4]. For data security in cloud computing, characteristics of servers of SaaS, PaaS, IaaS were analyzed and investigated on vulnerability or possible attacks. Security elements can be defined as follows; data security, network security, data integrity, data access, authentication and authorization, web application security, vulnerability in virtualization, availability, backup, identity management and sign-on process defined. Other than these elements, difference between expansion of resource usage and increase of authority range is considered in PaaS providing platform service resource and IaaS providing infra service different from SaaS condition in which environment of each service and additional security element was researched. However in this study, security elements on data process such as data security, data integrity, access, authorization, were considered as security elements for SaaS, PaaS, and IaaS data management, but was repeated or excluded in composed architecture by service type. Also, specific security element research on virtualization technology or operation such as monitoring metering, reporting is lacked due to computing storage, networking resources that are core technology of cloud computing system based on characteristics of SaaS, PaaS, and IaaS service.

We can see also a concept of CCOA (Cloud Computing Open Architecture) of cloud architecture, and architectural modules reflecting flexibility, expandability, and reusability of cloud computing in [5]. Also, functions and roles of architecture were categorized by considering expandable IT infrastructure and business values of management system based on integrated access providing and cloud computing base by users who are consumers of business services or companies. These prior studies defined categorized architecture layers, using SOA (Service Oriented Architecture) and business value concept of cloud computing.

4 Security Considerations and Monitoring System Requirements for Cloud Computing Virtualization Environment

The existing studies mentioned in the above section, only show simple definitions on operations and roles of categorized layers with specific functions. They do not provide security requirements or security elements to be considered in each architecture layer. Also, there are not enough security analyses on cloud computing virtualization, which is the core function of cloud computing.

In most of cases, a customer (service user) does not know where exactly his/her data is computed, processed, and stored in the cloud service provider's cloud farm. Over the Internet, the data can be transferred or computed over national wide range, and this situation can make more security threats and security audition issues. This situation can make also complicated billing issues. In other words, the service provider the customer's detailed usage of processors and storages over its Internet-based cloud computing environment.

For more secure and controllable cloud computing virtualization environment, we identified the following key security considerations from several research works [1-7] related to the cloud computing and its virtualization technology, especially from [7].

- Role-Based Access Control
  - Only authorized customers/users based on an RBAC mechanism can access to sensitive data on the platforms, in order to avoid data misuse or abuse.
- Data Isolation
  - An instance of customer data and information must be fully isolated from other customer data physically and/or logically
- Customer Privacy Protection
  - Any sensitive or privacy related information stored in cloud system must not be disclosed to other customers and even to the service provider itself.
- Exploit Code Blocking
  - The cloud service provider should prevent attackers to execute exploit codes on the cloud to access cloud customers' data or to take illegal privileges for further attacks.
- Backup and Recovery
  - The cloud provider has to provide an efficient replication scheme for safe back-up, and a rapid recovery mechanism to restore services, in order to mitigate the risks of uncontrollable natural, environmental, or societal disasters.
- Digital Forensic and Accountability
  - Even though cloud services are difficult to trace for accountability purposes, in some cases this should be considered as a mandatory application requirement for digital forensic and for other legal activities.

Especially, many people have been mentioning accountability can provide forensic aspects of the cloud system onto legal investigation parties, but reduce privacy aspects of the customers. In other words, a trade-off between privacy and accountability exists, since the latter produces action records or usage logs that can be examined by a third party when something goes wrong. Of course the customers can use obfuscation and privacy-preserving techniques to limit the information the VM exposes to the cloud. Still current cloud and its security system have open confidentiality issues with respect to the service provider or with respect to an attacker if he abuses the hosting platform.

Considering the above mentioned security issues, we can design and implement a common security monitoring system for cloud computing platform. For this design or implementation, we should take some technical issues into account. The followings are identified as the common set of requirements to be considered in security monitoring system for cloud computing virtualization environment. We have revised and elaborate the requirements described in [7].

- Effectiveness
  - The security monitoring system should be able to detect most kinds of attacks and integrity violations. The effectiveness means how many attacks it can

recognize. In other words, it should minimize false-negatives for maximizing its reputation and reliability.

- Precision
  - The security monitoring system should be able to avoid false-positives, so normal authorized activities could not be considered as malware activities. False-positives would make the customers very inconvenient.
- Transparency
  - The security monitoring system should minimize visibility from VMs and provide perfect transparency to service providers, customers, and even to potential intruders, so they should not be able to detect any monitoring activities and even to recognize the presence of the monitoring system.
- Self-Defense
  - The cloud system and its sub systems should not be possible to disable or alter the monitoring system itself.
- Interoperability
  - The security monitoring system should be deployable on the vast majority of available cloud frameworks with various configurations.
- Reaction Capability
  - After detecting an intrusion attempt over a cloud component, the security monitoring system should take appropriate actions against the compromised guest and his/her actions. Moreover, it should notify remote middleware security management components or security supervisors.
- Efficiency
  - The security monitoring system should not interfere with cloud and cloud application actions.

There is a trade-off between the above mentioned requirements. However, these can be included as a subset of regular guest maintenance capabilities, so they are virtually indistinguishable from regular load-balance based VM operations, from the point of view of the users and service providers.

5 Conclusions

After analyzing various cloud computing architectures and its security aspects, in this paper, we have identified a common set of security considerations which must be taken into account in cloud computing virtualization environment, and introduced important requirements for security monitoring system on the cloud. These considerations and requirements can be taken into account for managing virtualization middleware system of the existing cloud computing products and services, and also can be reflected in designing a novel and secure cloud computing virtualization framework.

Acknowledgement. This work was supported by NIA, KOREA under the KOREN program (1295100001-120010100).

References

1. Jeong, S., Chung, M., Cho, J., Shon, T., Moon, J.: A Research on Cloud Architecture and Function for Virtualization Security of Cloud Computing. Journal of Security Engineering 8(5), 627–643 (2011)
2. Williams, D.E., Garcia, J.: Virtualization with Xen: including XenEnterprise, XenServer, and XenExpress (2007)
3. Cloud Architecture Reference Models: A Survey, NIST CCRATWG 004 v2 (January 2011)
4. Subashini, S., Kavitha, V.: A Survey on Security Issues in Service Delivery Models of Cloud Computing. Journal of Network and Computer Application 34(1), 1–11 (2010)
5. Zhang, L.-J., Zhou, Q.: CCOA: Cloud Computing Open Architecture. In: 2009 IEEE International Conference on Web Services, pp. 607–616 (2009)
6. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud Computting and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility. Future Generation Computer Systems 25(6), 599–616 (2009)
7. Lombardi, F., Di Pietro, R.: Secure Virtualization for Cloud Computing. Journal of Network and Computer Applications (2010), doi:10.1016/j.jnca.2010.06.008

Medicine Rating Prediction and Recommendation in Mobile Social Networks

Shuai Li[1], Fei Hao[2], Mei Li[3], and Hee-Cheol Kim[1]

[1] Department of Computer Science, Ubiquitous Healthcare Research Center
Inje University, South Korea
ShuaiLi.sli@gmail.com, heeki@inje.ac.kr

[2] Department of Computing, School of Computing, Informatics and Media
University of Bradford, UK
feehao@gmail.com

[3] Diffverse (Beijing) Technology Co., Ltd, China
MeiLi.AngelLee@gmail.com

Abstract. During last few years we have witnessed a steady increase in medicine use for healthcare. The medicine experiences rated by other patients have huge potential to empower people to make more informed decisions. While the majority of previous research focused on rating prediction and recommendations on E-Commerce field, the area of healthcare or medical treatments has been rarely handled. Moreover, the geographical and temporal factors were not considered in their recommendation mechanisms. The rapid development of mobile devices, wireless networks, smart phones and ubiquitous wireless connections enable people to build and maintain mobile social interactions and relationships. In this paper, we identify and formalize the significant problem that exploits the over-the-counter medicine rating prediction and recommendation in mobile social networks. Then we devise the recommendation model and develop corresponding prototype of *iDrug*, reflecting a solution scheme of medicine rating prediction and recommendation in mobile social networks to increase the information accessibility for people's decision support.

Keywords: Machine learning, Medicine rating prediction, Mobile social network, Recommender system, Ubiquitous healthcare.

1 Introduction

With the rapid development of society and technology, people are becoming more healthy conscious in recent years, they usually take various medicines periodically in order to normalize serum cholesterol, glucose levels, or for the purpose of losing their weight [1] [2] [6]. As shown in Figure 1, the cost of medicines in the U.S. was 234.1 billion $ in 2008 which was more than double what was spent in 1999, indeed almost half of the populations take prescription medicines every month [4]. At the same time, these medicines often have debilitating and life-threatening side effects which are the

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 216–223, 2013.

factors should not be neglected, when a person takes multiple medicines and experiences a new symptom. It is not always clear which, if any, of the medicines or medicine combinations are responsible [6]. As new medicines are introduced continuously and still new patients for old medicines are found, more and more patients can improve their health and quality of life with the appropriate use of different medicines [2]. Obviously, the use patterns of current different kinds of medicines need to be better understood, especially the over-the-counter (OTC) medicines. Which can increase information accessibility for customer decision-support, e.g., to purchase healthcare or disease-treatment medicines etc.

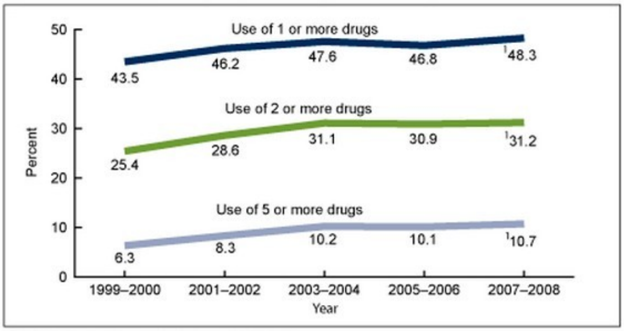


Fig. 1. Trends in the Percentage of People Take Medicines

In the past when people had a problem, they used to seek support and advice from family or friends. Nowadays they turn to smartphones or internet that can often make up by being less judgmental and more anonymous. A survey conducted by "Opinion Research Corporation"[1] reveals that 34% of people who search health information use mobile social resources, online forums and message boards etc. Meanwhile, according to "Pew Research Center"[2] 20% network users suffering from a chronic condition such as high blood pressure or diabetes, they try to find medicines of others with similar health concerns. Moreover, there are many medicines which can be purchased via online shopping where they can publish opinions and read many medicine reviews and comments [2].

There is no doubt that many researchers apply machine learning and data mining techniques to recommender systems. It has also gained some impact in tourism, restaurant, and entertainment [1] [3] [8]. However, recommendation techniques can be improved, by utilizing the geographical and temporal information to make medicine rating prediction and recommendation in mobile social network. These techniques have still been largely neglected [8]. Increasingly, the patients are turning to smart phones to seek medical suggestions. The wide usage of mobile social networks facilitates the development of social recommendations which are common

[1] `http://www.icrossing.com/articles`

[2] `http://www.pewinternet.org/reports/2011/p2phealthcare.aspx`

activities in our daily life, such as Facebook[3], Twitter[4]. In mobile social recommendation, rating prediction and item recommendation are two main research topics [8] [9]. In medical or healthcare domains, as for a new patient, medicine experiences reported by other former patients have great potential to empower medical consumers to make more informed decisions about medical medicines [6]. Issues like how to efficiently predict the rating for a certain medicine and to whom recommend some potential relevant medicines effectively with mobile social recommendation mechanism are a grand challenge [5] [7] [8]. Our major contributions are twofold: first, we identify and formulate the practical problem about medicine rating prediction and recommendation in our daily life; second, we devise an efficient recommendation model with considerations of geographical and temporal factors extracted from mobile social networks, and then implement a corresponding prototype of *iDrug*.

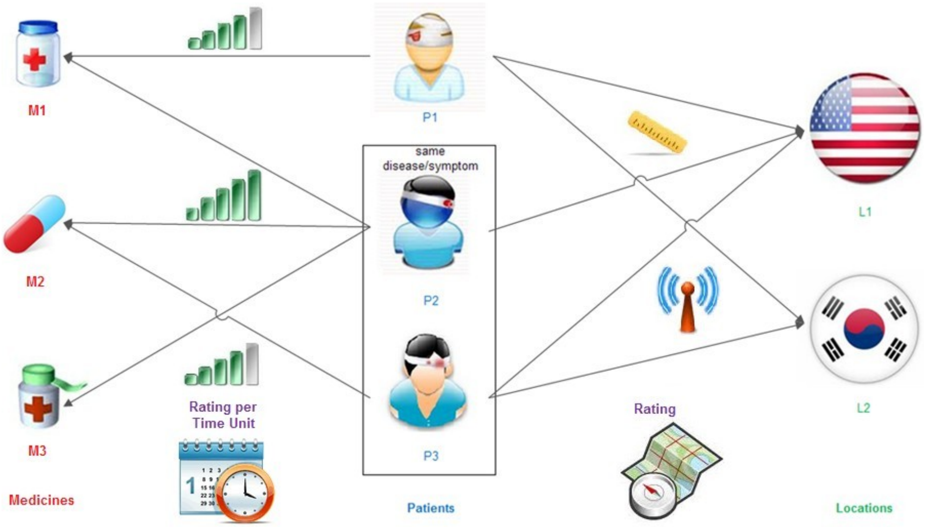


Fig. 2. Structure of Mobile Social Rating Network

2 Problem Statement

2.1 Mobile Social Rating Network

Definition 1: (Mobile Social Rating Network) MSRN is formalized as a six-tuple $\Omega =< \varphi, \delta, \lambda, \Phi, \Delta, \Lambda >$ with φ indicates the certain patient; δ indicates the certain medicine; λ indicates the certain location; Φ indicates the social relationships between different patients with the same certain disease or symptom, where $\varphi_{ij} \in \Phi$ denotes the relationship between patient i and j; Δ indicates a triple with patient,

[3] http://www.facebook.com

[4] https://twitter.com

medicine, and rating per time unit $\Delta=<\varphi,\delta,\frac{R^{\delta}}{\tau}>$ where R^{δ} is the rating (e.g., star grade, progress bar marking) on δ given by φ, and τ is the average elapsed time horizon from the other patients rated this certain medicine to current period on the occasion of the certain patient want to obtain some personalized suggestions of medicine recommendation; Λ indicates a triple with patient, location and corresponding rating (e.g., distance, longitude, latitude) $\Lambda=<\varphi,\lambda,R^{\lambda}>$.

In a broad sense, the "patient" mentioned means the customer who has already taken or will take the medicines respectively. More concretely, $\Delta=<\varphi,\delta,\frac{R^{\delta}}{\tau}>$ means the certain patient φ give the rating $\frac{R^{\delta}}{\tau}$ to a certain medicine δ; $\Lambda=<\varphi,\lambda,R^{\lambda}>$ means the certain patient φ give the rating R^{λ} to a certain location λ in terms of metric between patient and location.

Figure 2 depicts the structure of MSRN (Mobile Social Rating Network). In the left part of Figure 2, we notice that for a certain medicine M_2, there is a cluster of patients P_2 and P_3 who have given ratings or posted their reviews or comments on it. These patients come from the same community or clinic, which means they have similar symptom or disease. In the right part of Figure 2, for a given geographical location "U.S.", patients P_1, P_2 and P_3 who have given ratings according to a distance metric between the certain patient and the certain location of a local clinic or hospital. In other words, each kind of medicine in MSRN is associated with a community or single patient, as well as each kind of location in MSRN is associated with a community or a single patient [10]. Obviously, we refer to these communities as the potential medicine communities and local area communities respectively. Investigating the properties and customers' behavior in both medicine communities and local area communities is important to support the decision making of sales marketing and business analysts etc.

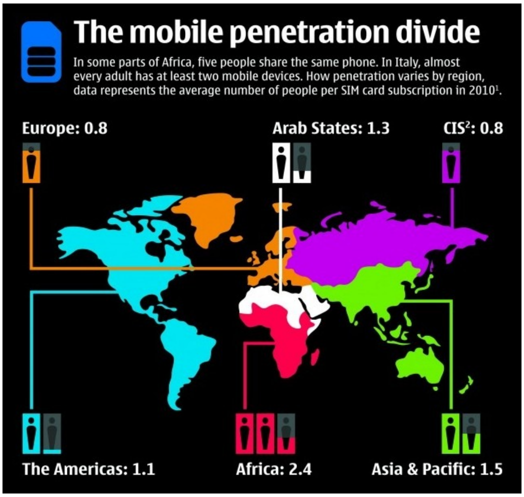

The mobile penetration divide
In some parts of Africa, five people share the same phone. In Italy, almost every adult has at least two mobile devices. How penetration varies by region, data represents the average number of people per SIM card subscription in 2010[1].
Europe: 0.8
Arab States: 1.3
CIS[2]: 0.8
The Americas: 1.1
Africa: 2.4
Asia & Pacific: 1.5

Fig. 3. The Distribution of Mobile Devices

2.2 Problem Formulation

Input: The Mobile Social Rating Network (MSRN) consists of different patients who have the same disease or symptom. A certain patient $\varphi \in \Phi$, Φ represents the mobile social relationships that they build. The mobile penetration rate in the world from "International Telecommunications Union (ITU)"[5] is shown in Figure 3. And the medicine $\delta \in \Delta$; the location $\lambda \in \Lambda$; the rating on the medicine δ given by patient φ divided by the time difference from rating moment to current time τ, called $\frac{r^{\delta}}{\tau}$; the rating on the location λ given by patient φ, called r^{λ}. Therefore, the training data set can be represented as

$$\Delta = \left\{\left(\varphi_1, \delta_1, \{\frac{r^{\delta}}{\tau}\}_1\right), \left(\varphi_2, \delta_2, \{\frac{r^{\delta}}{\tau}\}_2\right), \dots, \left(\varphi_M, \delta_M, \{\frac{r^{\delta}}{\tau}\}_M\right)\right\} \tag{1}$$

$$\Lambda = \{(\varphi_1, \lambda_1, \{r^{\lambda}\}_1), (\varphi_2, \lambda_2, \{r^{\lambda}\}_2), \dots, (\varphi_{\mathrm{N}}, \lambda_N, \{r^{\lambda}\}_N)\} \tag{2}$$

Learning: The goal of our medicine recommendation is to score a certain amount of relevant medicines in candidate set and return optimal affinity medicine for patients. However, the rating prediction is to derive rating predictions for a specific patient φ, we take into account ratings of the "top-K" similar patients to φ, where K is a patient-defined parameter [5] [7]. We can utilize these inter-dependencies to score the medicines using probability $\mathrm{P}(\delta|\varphi, \lambda)$.

Prediction: Suppose $\{s_{1,}s_2 \dots, s_K\}$ to be the corresponding final similarity values of the "top-K" similar patients $\{\varphi_{1,}\varphi_2 \dots, \varphi_K\}$ to φ; the predict ratings for the patient φ are defined as follows

$$\hat{r}_{\varphi,j} = avg + \frac{\sum_{i=1}^{K}\{s_i \times |r_{ij} - avg_i|\}}{\sum_{i=1}^{K} s_i}, \tag{3}$$

where j is any unrated medicine by the patient φ; r_{ij} refers the corresponding rating, and avg_i means the average ratings value of the patient φ_i, for the $\{\mathrm{i} = 1,2, \dots, \mathrm{K}\}$. avg denotes the average known ratings of the patient φ who wants recommendations. $s_i = f(*)$ is a mapping function from the similarities derived by $\{\Phi, \Delta, \Lambda\}$ to an overall similarity between φ_i and φ [7][9]. It aggregates three different similarities to obtain the optimal similarity between φ_i and φ.

Recommendation: Based on the score $\mathrm{P}(\delta|\varphi, \lambda)$ obtained in the learning step, the set of recommended medicines for a given user φ' and a given location λ' will be

$$\widehat{\Delta}(\varphi', \lambda') = argMax_{\delta \in \Delta}^{N} \mathrm{P}(\delta|\varphi', \lambda'), \tag{4}$$

where N is the number of recommended medicines in MSRN, $\widehat{\Delta}$ is the collection of recommended medicines. Finally, after sorting the predicted ratings $\hat{r}_{\varphi,j}$ of patient φ, it makes a suggestion list including the "top-K" medicines in the $\widehat{\Delta}$, where K is a desired cardinality value.

[5] http://www.itu.int

2.3 A Motivating Example

Figure 4(a) illustrates the motivating scenario: Prof. Charles is on a business trip from Korea to U.S. for attending a conference. Because different parts of the world have different climates, he caught a cold when he arrived in New York City. What can he do then? Actually there are 3 nearby medicine stores or hospitals where he can buy the corresponding medicines. In the past, other customers or patients may have also bought the similar medicines, and had the medicine use experience. Some of them may have already given rating scores to the taken medicines. Based on the aforementioned scenario, how can we solve Prof. Charles's practical healthcare or medical issue?

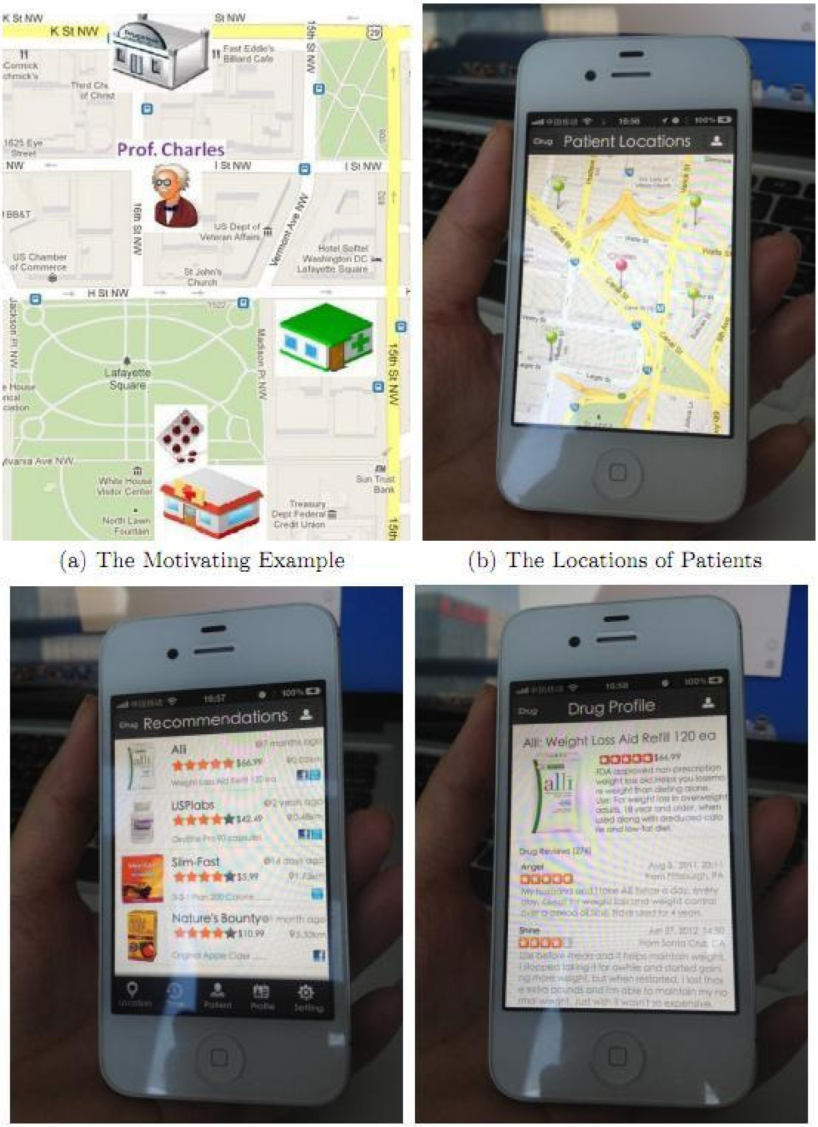
3 The Prototype of *iDrug*

To solve the problem mentioned earlier, based on previous recommendation model, this section presents the devised prototype (Data Set originates from “Drug Store”[6]) of *iDrug* for medicines rating prediction and recommendation in mobile social network. In Figure 4(b), this scenario is described as: in Canal Street of New York City, Prof. Charles wants to cure certain disease (or he just wants to invigorate his health), e.g. he wants to cure cold. With the help of his smart phone, he opens *iDrug* and inputs the keyword like “cure cold” in the text box. Then he will get a screen like Figure 4(b), where we can see there are another historical nearby customers or patients like Alice, Bob, and David, and corresponding pins represent a variety of locations (such as medicine stores or hospitals) when they can buy and rate medicines.

After Prof. Charles clicked the top left *iDrug* button, he enters screen like Figure 4(c), it depicts scenario that provides rating prediction and recommendation list for Prof. Charles to make useful decisions. In Figure 4(c), we can see medicine names and short text descriptions of functions, a specification, the rating which is represented by stars from one to five, the price, the average distance and the time period about all the previous customers or patients who rated this medicine, e.g., Nature's Bounty: Original Apple Cider Vinegar Diet 90 tablets. The mobile social network icon like “Facebook” or “Twitter” is facilitating Prof. Charles to share through any of the recommended medicines with his friends. Moreover, in the bottom there are some function buttons: 1) “location” button is to go back to the previous screen like Figure 4(b), 2) “time” button is to zoom in and displays the time difference from the patient's rating moment and current time among the different medicines, 3) “patient” button is to preview the patient’s account information, 4) “profile” button is to edit the patient’s profile, and 5) “setting” button is for the recommender system parameters tuning, e.g., to set the numbers of recommendation results that patients want to make a query for.

[6] `http://www.drugstore.com`

Finally, when Prof. Charles clicks his interesting medicines, e.g., he clicks the first item "Alli" then the screen jumps to Figure 4(d). This scenario is about specific concrete medicine information that he is interested in. We can see the high resolution medicine image, the average rating score, price and medicine description in detail, under which are a variety of medicine reviews and rating scores given by different patients such as Angel and Shine, then the date, and location information.



(a) The Motivating Example (b) The Locations of Patients

(c) Prediction and Recommendation (d) Medicine Profile and Reviews

Fig. 4. The Motivating Example and Prototype of *iDrug*

4 Conclusion and Future Work

In this paper, we identify and formalize a challenging and significant problem that exploits the OTC medicine rating prediction and recommendation in mobile social networks. We take into account the most critical geographical and temporal factors extracted from mobile social rating network. At the same time, we illustrate the corresponding user scenarios, devise and develop the prototype of *iDrug* to reflect the solution scheme for the given practical problem. In the future, we plan to adopt the "Nursing Home Compare and Patients' Hospital Experiences Data Set"[7], devise similarity measures, ameliorate a corresponding model and prototype, and deploy usability test for the ease-of-use of *iDrug*.

Acknowledgement. We would like to thank three anonymous reviews for providing us with constructive comments and suggestions. This work was supported by MKE (The Ministry of Knowledge Economy), Korea, under Global IT Talents Program supervised by NIPA (National IT Industry Promotion Agency, 2012-H0902-1013). It was also partially funded by the Korean Ministry of Knowledge Economy (#10033321).

References

1. Masashi, S.: Dimensionality Reduction of Multimodal Labeled Data by Local Fisher Discriminant Analysis. Journal of Machine Learning Research 8, 1027–1061 (2007)
2. Margaret, A.H., Francis, S.C.: The Path to Personalized Medicine. The New England Journal of Medicine 363, 301–304 (2010)
3. Fei, H., Min, C., Chunsheng, Z., Mohsen, G.: Discovering Influential Users in Micro-blog Marketing with Influence Maximization Mechanism. In: IEEE Proceedings of Global Communications Conference (GLOBECOM 2012), California, USA (2012)
4. Qiuping, G., Charles, F.D., Vicki, L.B.: Prescription drug use continues to increase: U.S. prescription drug data for 2007-2008 NCHS data brief. No. 42, Hyattsville, MD: National Center for Health Statistics (2010)
5. Sanjay, P., Yan, L., Jay Kuo, C.-C.: Collaborative Topic Regression with Social Matrix Factorization for Recommendation Systems. In: 29th International Conference on Machine Learning (ICML 2012), Scotland (2012)
6. Yueyang, A.L.: Medical Data Mining: Improving Information Accessibility using Online Patient Drug Reviews. Master's Thesis, Dept. of EECS, MIT (2011)
7. Jason, W., Chong, W., Ron, W., Adam, B.: Latent Collaborative Retrieval. In: 29th International Conference on Machine Learning (ICML 2012), Scotland (2012)
8. Francesco, R., Lior, R., Bracha, S.: Chap. 1: Introduction to Recommender Systems Handbook. In: Recommender Systems Handbook, pp. 1–35. Springer (2011)
9. Panagiotis, S., Eleftherios, T., Yannis, M.: Product Recommendation and Rating Prediction based on Multi-modal Social Networks. In: Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011), pp. 61–68. ACM (2011)
10. Yutaka, M., Hikaru, Y.: Community Gravity: Measuring Bidirectional Effects by Trust and Rating on Online Social Networks. In: Proceedings of the 18th International Conference on World Wide Web (WWW 2009), pp. 751–760. ACM (2009)

[7] https://data.medicare.gov

Cloud Browser: Enhancing the Web Browser with Cloud Sessions and Downloadable User Interface

Antero Taivalsaari[1], Tommi Mikkonen[2], and Kari Systä[2]

[1] Nokia, Visiokatu 5, Tampere, Finland
antero.taivalsaari@nokia.com
[2] Tampere University of Technology, Korkeakoulunkatu 1, Tampere, Finland
tommi.mikkonen@tut.fi, kari.systa@tut.fi

Abstract. The web browser has become one of the most important and frequently used computer programs that people use. The web browser has effectively assumed the role of an operating system. Yet there have been predictions that the web browser and the Web itself will effectively die. More specifically, it has been argued that the web browser will lose the battle against native, custom built web apps. In this paper we predict that the web browser may indeed disappear but for entirely different reasons. We present a concept and implementation of a cloud browser that moves the users' browser sessions and the user interface chrome of the web browser to the cloud. The benefits of a cloud browser are especially valuable to those people who use a plethora of web-connected devices, allowing the same web pages and applications to be used flexibly – and even simultaneously – from different devices.

Keywords: Web browser, web applications, session-based browsing, proxy browsing, cloud browser, HTML5.

1 Introduction

In a September 2010 Wired magazine article [1], Chris Anderson and Michael Wolff claimed provocatively that “the (World Wide) Web is dead.” They based this claim on two main arguments. The first was that the amount of (text-based) Internet traffic generated by web page downloads has decreased dramatically over the years in proportion to the traffic generated by video and music downloads. The second argument was that users will no longer surf web pages with a traditional web browser, because – for the vast majority of web services such as e-mail, news, Facebook and Twitter – they will prefer custom-built native applications (e.g., Flipboard for iPad) over open, unfettered web browser access. Anderson and Wolff argued that the trend toward such apps will be even more evident in the mobile device space, where – according to the authors – web browsers have already lost the battle against custom-built native apps.

Anderson and Wolff’s first argument has been widely refuted in the press and on the Web (see, e.g., www.smallfish-bigpond.com/2010/08/wired). The statement that the amount of text-based Web traffic is insignificant compared to other types of traffic, while literally true, is misleading because, at the same time, web page and web browser usage has increased dramatically – nearly exponentially, in fact. Consequently,

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 224–233, 2013.

the notion that video and music downloads generate the majority of network traffic is irrelevant and in no way confirms that the Web itself is dead.

Interestingly, Anderson and Wolff's second argument about the transition from open, browser-based Web access to custom-built native apps has generated much less debate. Given the current popularity of Apple's iPhone and iPad and Google's Android devices, many people seem to take it for granted that the success of custom-built native applications will predicate the demise of browser-based web applications and the use of the web browser more broadly. We believe that the trend toward such custom apps is only temporary, though. We predict that the use of Open Web applications – that is, those applications that run directly in a standard web browser without plugins or extensions [2-4] – will eventually surpass the use of custom native apps not only in the context of desktop computing, but also in mobile devices.

In this paper we postulate a different future for the web browser. We anticipate that the web browser itself will largely move to the cloud, so that the users' browser sessions can be used easily from a plethora of computers and devices simultaneously. This is critical, since in the future most people will use far more web-enabled devices in their daily use than just a laptop computer and a smartphone. Furthermore, we believe that the web browser's UI (browser chrome) may also become downloadable from the Web, so that the look-and-feel of the entire browsing environment can be adapted flexibly to each type of a device. In other words, the users will be able to carry their sessions from device to device, but the UI of those sessions can be tailored to each particular device and usage scenario. Such browser customization is already possible to a limited degree today, e.g., with downloadable browser color themes. However, ultimately the entire UI of the web browser could be fully dynamic, network-downloadable and customizable.

2 Making the Web Browser Disappear

Paradoxically, the best way to ensure the continued evolution and success of the web browser is to make the web browser "disappear". By this, we do not mean killing the web browser literally. We base our paper on two observations/trends.

(1) Towards a New Era with Multiple Device Ownership. First, we believe that the world will rapidly move from the current PC- and smartphone-centric era to an era in which the average user will have dozens of web-enabled devices. Today, the average user has perhaps two or three web-enabled devices, e.g., a laptop computer, a smartphone and possibly a tablet device. In the future, the number of web-enabled devices (computers, phones, tablets, TVs, car displays, game consoles, photo frames, wrist displays, etc.) that the average users will use in their daily lives will explode, and will likely be measured in double digits. This trend – when combined with the desire to access the same information and personal data from a multitude of different devices – will profoundly change the requirements for software platforms.

(2) Broad Variety of Devices with Different Screen Sizes, Input Mechanisms and Usage Situations. The second major trend that will have a significant impact on the characteristics of software platforms is the broad variety of that the future devices will have in terms of screen sizes and input mechanisms. With screens ranging from tiny phone displays to large TV screens, and input mechanisms ranging from T9 keypads and remote controls to touch displays and conventional QWERTY keyboards, one UI solution does not fit all. Moreover, the more the industry moves towards real web applications instead of just web pages, the more obvious the limitations of the standard web browser UI, with its tabs and arcane back, forward, and reload buttons will become.

Solution: Session-Based Browsing with "Downloadable Chrome". As a result of the trend towards multiple device ownership, we argue that the web browser should be cloud based, so that the users can create and access the same browser sessions from a plethora of different computers and devices. We refer to mechanisms that enable such behavior broadly as *session-based browsing*. In session-based browsing, the user's browser sessions persist on the Web independently of any specific device(s), so that the sessions can be used readily from a number of different devices. Without such capabilities, the users would have to explicitly open and manage their web sessions ("tabs" of pages and applications) on each device that they use, and typically do that again and again each and every time when they use a device.

Furthermore, to cope with a large variety of screen sizes, input mechanisms and usage situations, the top-level UI of the web browser, i.e., the browser "chrome" surrounding the actual web pages or applications, should become dynamic and network-downloadable. Although the classic UI of web browser is well suited (at least in terms of familiarity) to conventional computers and tablets, it is hardly ideal for devices that commonly require single-handed operation. With dynamically downloadable chrome and associated code, the UI can be customized to each type of a device and usage scenario. Our goal is to make the entire UI of the web browser fully dynamic, network-downloadable and customizable. When combined this session-based browsing discussed above, the users can carry their sessions from device to device, but with a top-level UI customized to each particular device.

We discuss session-based browsing and downloadable chrome (including the risks and issues associated with such mechanisms) in more detail in the following sections.

3 Cloud Browser Introduced

Figure 1 contains a screen snapshot of our cloud browser implementation running a number of HTML5 game applications. The cloud browser itself is a pure HTML5 web application (in other words, just a web page) running in a standard web browser, complemented with a server-side architecture that supports the creation of user-specific browser sessions and downloadable UIs. In this case the UI of the browser chrome is based on movable and resizable windows.

Cloud-Based Data System. A fundamental enabler for our cloud browser is the *Data API* explained in [5]. In this system the updates to the data are automatically synchronized to the server, and local copy of the data is automatically updated to repeat changes in the server. Individual applications can use this system, but we also use it to synchronize information about the browser session. The *Data API* includes also a *notification service* that allows applications to react to changes immediately.

It Is Just a Web Page. The cloud browser client itself is just a HTML5 web application, and therefore it does not require any installation. This means that the user can launch the system simply by entering its URL in an ordinary web browser. Upon starting the system, the system will download the currently selected UI, and then open the applications based on previously stored information.

Sessions are user-specific, meaning that each user of the system will have their own persistent sessions. Before using the system the users will have to register themselves and then login to our system based on their own user credentials.

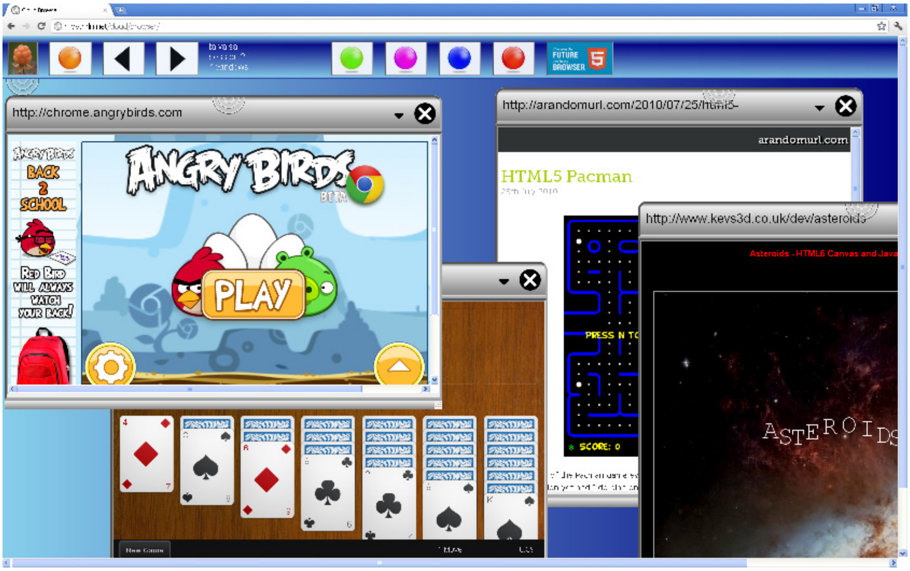

http://chrome.angrybirds.com
ANGRY BIRDS
BETA
PLAY
http://arandomurl.com/2010/07/25/html5-
arandomurl.com
HTML5 Pacman
PRESS N TO
SCORE: 0
http://www.kevs3d.co.uk/dev/asteroids
Asteroids - HTML6 Canvas and Java
ASTEROIDS
FUTURE BROWSER

Fig. 1. Cloud browser displaying a session of HTML games

It Can Support Any Number of User Sessions Containing Ordinary Web Content/Applications. The cloud browser is built around the notion of persistent sessions that can contain any standard web content. Each session consists of a number of web pages/applications opened by the user. Any number of sessions can be created for whatever purposes the users want. Common examples of session-based browsing include creating separate sessions, e.g., for social networking sites, e-mail accounts, games, financial news, photos, business (web) applications, and so on.

For example, the applications and pages displayed in Figure 1 are “off-the-shelf” HTML5 games available on the Web. The displayed set of games together with layout information of the windows constitutes one user session. In our current window-based cloud browser interface, the user can switch between different sessions by pressing the left and right arrow buttons on the top of the screen.

It Supports “Downloadable Chrome”. In Figure 1, we have intentionally left the native UI chrome of the web browser (in this case: the URL bar and the back and forward buttons of Google Chrome) visible to illustrate the point that the system runs in an ordinary web browser. However, typically we would use the web browser in full-screen mode, so that the user will deal only with the UI controls of our cloud browser, and not those of the underlying web browser. In Figure 1 we use a window-based UI style, with a number of built-in window controls/buttons that allow the windows to be arranged in a number of ways. All these buttons, as well as all the windows shown in Figure 1 are generated by the cloud browser, i.e., they are not native controls of the surrounding web browser.

It Supports a Number of Different Downloadable UI Styles. While a window-based UI is well suited to desktop computers, it is not ideal for mobile devices. In order to support mobile devices, we have implemented a number of different UI styles. One of our alternative UIs is built around tiles instead of windows. In a tile-based UI, applications are positioned automatically on the screen, i.e., they cannot be resized or moved. All our downloadable UIs leverage the same session information, i.e., the set of open applications remains the same regardless of the UI style.

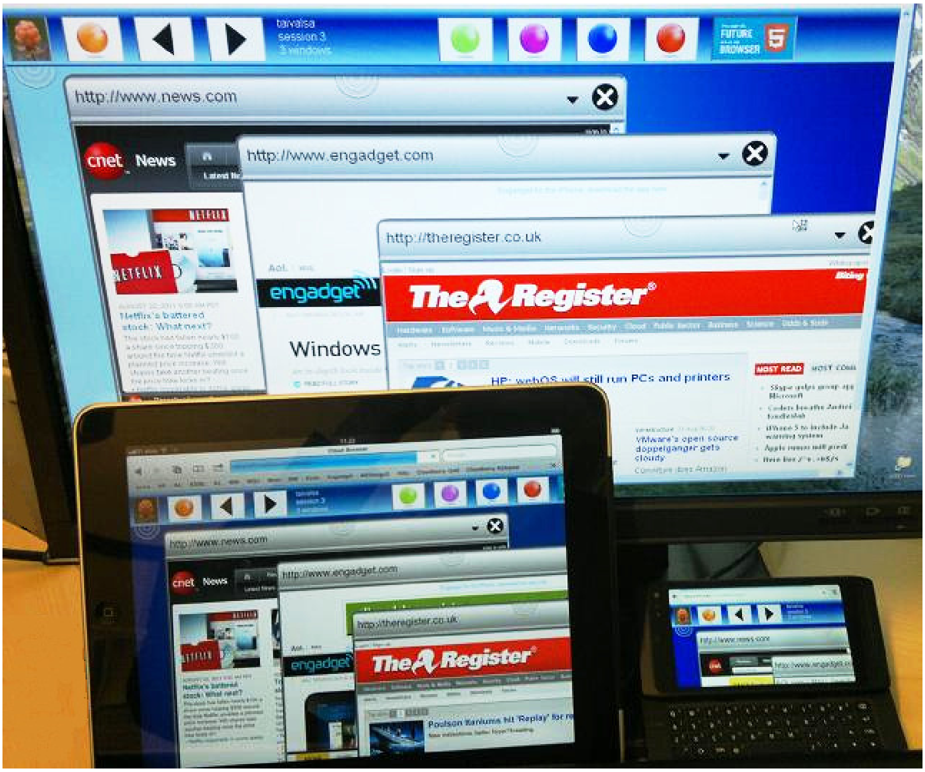


Fig. 2. Different kinds of terminal devices

It Supports Multiple Device Ownership. One of the central ideas in our cloud browser is that the users can "carry" their browsing sessions and applications from one device to another. Figure 2 shows a photo that illustrates our cloud browser running on three different devices simultaneously: a desktop PC, Apple iPad (bottom left), and Nokia N950 mobile phone (bottom right). Although the underlying web browser implementations on each device are different (Google Chrome, Safari, Nokia MeeGo browser, respectively), the system runs on each device without installation or any other native software than the web browser; the user can launch the system simply by entering its URL in the (native) web browser. In Figure 2, we have intentionally used the same top-level UI style in all the devices, even though a window-based UI is not well suited to devices with small screens. In a more practical scenario, mobile devices would use, e.g., a tile-based UI that we have also implemented.

Since our implementation is based on Cloudberry, the cloud browser can also work in off-line mode by using HTML5 caching. Similarly, if the individual applications have defined the HTML5 cache manifest, they can also work off-line. Another feature we inherit from Cloudberry is the security mechanism. The domain- and permission-based security model of Cloudberry provides protection to critical resources. More details about the off-line and security features have been described in [5].

It Runs Web Content Either on the Client Side or Server Side. By default, content execution in our current cloud browser takes place on the client devices, following the HTML5 Specification [6]. In this model, the HTML/CSS/JavaScript code of the web pages or applications is downloaded to the client device for execution. As an alternative execution model, pure server-side browser instances may be created as well. In this model, a server-side browser execution context is created whenever the user opens a new page or application on a client device. For example, a VNC-like "pixel streaming" approach is then used for displaying the generated browser content on client devices, and for passing events between the clients and the server. The two execution models have very different characteristics and tradeoffs especially when it comes to sharing and synchronizing content between multiple devices.

It Supports Synchronization of Content between Multiple Client Devices. In a world with multiple devices, people will have the implicit expectation to use their browser sessions from different devices even simultaneously, so that changes from one device are reflected automatically to the other devices viewing the same session and content. This way, they can immediately continue the work that they have been doing on one device or computer on another one, e.g., continue viewing or editing the same document or even continue playing the same game on a tablet or mobile phone after leaving from their computer at home or work. Such notification/synchronization features are familiar from systems intended for computer-supported collaborative work (CSCW), but unfamiliar to most web users today.

Depending on the content execution model discussed in the previous paragraphs, session/content synchronization can be either trivial or non-trivial. In the server-side execution model, the clients share exactly the same rendered and streamed content. As long as an active network connection exists, changes in the session are instantly visible in all the clients viewing the same session. In the client-side execution model, notification APIs provided by our system must be used to synchronize the state of the content running on different devices. We will discuss the implementation issues and tradeoffs related to the cloud browser concept in more detail in the following.

4 Implementation Issues and Remaining Challenges

Problems with the Server-Side Execution Model. There are a number of technical issues that make the implementation of a cloud browser more challenging than it should be. For instance, the use of a pure server-side execution model is impractical for mobile devices with intermittent, unreliable and potentially expensive network connections. If the network connection fails for any reason, the clients will stop seeing the streamed sessions pretty much immediately, and can no longer interact with any of the pages or applications. With intermittent connections, display streaming and event handling can become jagged and erroneous. Furthermore, the continuous streaming of pixels from the server to the client necessitated by this model can be expensive on those network connections that do not allow unlimited network use at a flat rate. In short, this model is suitable only for devices with reliable, fast, inexpensive network connections with low latency.

Problems with the Client-side Execution Model. The client-side execution model solves many of the problems associated with pure server-side execution. In particular, since the client-side model utilizes the host browser for executing web content, intermittent network connections do not pose any major problems. Offline use is supported according to the HTML5 Specification [6]. However, since application instances in different devices run independently of each other, synchronization of state can become a major challenge if the users wish to use the same sessions simultaneously from multiple devices, with the expectation to see changes made on one device reflected to the other devices immediately.

In our current system, we use a *Data API* with *notification service* – inherited from our earlier Cloudberry system [5] – that allows URL changes in one browser session to be automatically passed on to the other clients currently viewing the same session. This way, when the user clicks on a link in a certain web page or application on one device, the other device(s) will change the URL correspondingly. In our current window-based top-level UI, changes in window position, size and transformations (e.g., rotation, scaling) are also synchronized automatically between clients. As described earlier the same Data API with notification service can be used for synchronizing the data of individual applications, too. For instance, if the same web-based calendar application is running on a number of devices, changes in a calendar entry from one device can be automatically reflected to the other devices. The notification mechanisms must be used explicitly by the application developer. The state of arbitrary 3rd party applications will not be synchronized automatically, unless those applications are modified explicitly to support such behavior.

Limitations Arising from the Web Browser. The standard web browser places various restrictions on supporting “browser in browser” behavior as required by the client-side execution model. Currently, the only portable way to support "embedded sub-browser" instances that run within a web page (representing a top-level UI that manages the embedded sub-browsers) is to use *iframes* (the HTML <*iframe*> tag) [6, section 4.8.2]. Iframes suffer from well-known limitations. For instance, iframes do not provide any support for process isolation. This kind of a system is inherently less robust than a system in which web content arriving from different domains runs in separate native processes.

The same origin policy of the web browser [6, section 5.3] places additional limitations on the use of iframes. For instance, the main page (top-level UI) cannot access browser history information, or receive information about link clicks from those iframes (embedded sub-browsers) that display content arriving from different domains (servers) than the main page (cloud browser) itself. Although the forthcoming HTML5 Specification provides some mechanisms for bypassing these limitations, these features are not widely supported by commercial browsers yet. Many commercial web browsers also have iframe-related rendering bugs, i.e., content displayed within iframes is not rendered exactly the same way as it otherwise would.

Furthermore, many popular web sites (such as *www.facebook.com* and *www.gmail.com* explicitly prevent themselves to be embedded and used from iframes by using an "*X-Frame-Options deny*" response header.

Risks of Downloadable Chrome and Dynamic User Interface Customization. In a system with downloadable chrome, the user interface of the entire system can be changed even while the system is running. This kind of a feature can potentially be used in harmful ways, e.g., for different types of phishing attempts. For instance, browser chrome could be modified to look like a fake version of an application or site that the user is familiar with.

In practice, since our system allows top-level UI code to be downloaded only from the same domain (origin) as the cloud browser itself, these security issues arise primarily if hackers find their way into the server containing the cloud browser. This is not really any different from hacking into any major web server in order to modify the behavior of the site. In any case, security remains an interesting and relevant topic for further research in the cloud browser area.

Privacy of User Content. All systems that store users' data in the cloud need to secure the data against loss and misuse. On the other hand cloud browser approach ensures that the users do not lose their data in case of losing or breaking the device.

Our current system can use https protocol for protected access and users need to authenticate themselves. In addition the server has to be protected against hackers. Obviously there is still a lot to do.

5 Related Work

The term *cloud browser* has been used in a number of different contexts before. For instance, there exists a popular Cloud Browse application for the Apple iPhone (http://www.alwaysontechnologies.com/cloudbrowse/) that employs the use of a remotely streaming desktop browser. Unlike the native Safari browser on the iPhone, the remote browser is fully Flash and Java-enabled, allowing an iPhone user to access web sites that would not be usable on the iPhone otherwise. The iPhone Cloud Browse app displays only one web page at a time, i.e., it does not support persistent sessions, downloadable chrome, or other more advanced features that our system has.

On the academic side, the term cloud browser was introduced recently (and independently of our work) by Lu, Li and Shen from Microsoft Research Asia [7]. Their proposed approach focuses on server-side screen rendering and on streaming rendered content to multiple client devices. They do mention the possibility of a hybrid approach, in which screen rendering is performed partially in the cloud and partially in the clients. They do not pursue the client-side model further, though.

Ever since the World Wide Web became popular, there have been server-side *proxy browsers* that can be used for anonymizing the identity of the person accessing content on the Web. Examples of such servers are http://freeproxybrowsing.com/, http://www.proxsafe.net/, and http://www.proxybrowsing.com/. Proxy browsers have been especially fashionable in the mobile device space, in which they are commonly used for transliterating content from ordinary web sites to smaller form factors, so that bandwidth consumption can be reduced and content can be optimized (before downloading) to specific mobile devices and different screen sizes. For instance, Opera Mobile (http://www.opera.com/mobile/) uses such an approach.

Remote screen rendering and streaming are nothing new either. Desktop sharing/streaming was pioneered by the developers of the Virtual Network Computing (VNC) [8]. Web-based VNC clients such as noVNC (http://kanaka.github.com/-noVNC/) are available nowadays, so that VNC servers can be accessed from a standard web browser without installing any native client software. Many other streaming protocols, such as RDP (Remote Desktop Protocol) and RFB (Remote Framebuffer) exist for similar purposes.

Session sharing and notifications have been studied in the area of computer-supported collaborative work (CSCW), beginning from the mid-1980s after graphical UIs and online connectivity became widely available. In recent years rudimentary session synchronization mechanisms have started appearing in commercial web browsers, too. Nowadays, Mozilla Firefox and Google Chrome browsers offer mechanisms for synchronizing, e.g., bookmarks and the set of open windows/URLs across different computers.

6 Conclusions

The massive popularity of the World Wide Web is turning the web browser from a document viewing tool into a general-purpose host platform for various types of services, including desktop-style web applications. HTML5 web applications require no installation or manual upgrades, and they can be deployed instantly worldwide. These capabilities will allow application development and instant worldwide deployment without middlemen or distributors. Conventional binary applications are at a major disadvantage when compared to web-based software that can be deployed instantly across the planet.

In this paper, we have presented the concept and implementation of a *cloud browser* that will move the users' browser sessions and the UI chromes of the web browser to the Web. Our contributions include support for persistent web sessions, downloadable chrome, support for both server-side and client-side execution (with notification support), with a specific focus on multiple device ownership. The benefits of a cloud browser are especially valuable to those people who use a plethora of web-connected devices in their daily lives, allowing the same web pages and applications to be used flexibly – and even simultaneously – from different devices.

The cloud browser concept relegates the role of the native web browser primarily to that of a rendering engine. From the user's viewpoint, the actual web browser content is run inside the cloud browser, while the native browser is treated simply as a host and a "viewport" for the cloud browser itself. It will be very interesting to witness the reactions of major browser vendors and see whether this kind of architecture will start reaching widespread popularity.

References

1. Anderson, C., Wolff, M.: The Web is Dead: Long Live the Internet. Wired, 118–127, 164–166 (September 2010)
2. Mozilla, The Mozilla Manifesto (2011), `http://www.mozilla.org/about/manifesto.en.html`
3. Berners-Lee, T.: Long Live the Web: a Call for Continued Open Standards and Neutrality. Scientific American 303(4), 56–61 (2010)
4. Taivalsaari, A., Mikkonen, T., Ingalls, D., Palacz, K.: Web Browser as an Application Platform. In: Proc. 34th Euromicro Conference on Software Engineering and Advanced Applications (SEAA 2008), September 3-5, pp. 293–302. IEEE Computer Society (2008)
5. Taivalsaari, A., Systä, K.: Cloudberry: HTML5 Cloud Phone Platform for Mobile Devices. IEEE Software, 30–35 (July/August 2012)
6. World Wide Web Consortium, HTML5 Specification, candidate recommendation (2011), `http://www.w3.org/TR/html5/` (December 17, 2012)
7. Lu, Y., Li, S., Shen, H.: Virtualized Screen: A Third Element for Cloud-Mobile Convergence. IEEE Multimedia, 4–11 (April-June 2011)
8. Richardson, T., Stafford-Fraser, Q., Wood, K.R., Hopper, A.: Virtual Network Computing. IEEE Internet Computing 2(1), 33–38 (1998)

Visual Novels: An Methodology Guideline for Pervasive Educational Games that Favors Discernment

Francisco Lepe Salazar, Tatsuo Nakajima, and Todorka Alexandrova

Department of Computer Science and Engineering, Waseda University,
3-4-1 Okubo Shinjuku Tokyo 169-8555 Japan
{flepe,tatsuo}@dcl.cs.waseda.ac.jp,
toty.alexandrova@gmail.com

Abstract. Current educational games vary in how they present content, how they evaluate recently learned topics, and how student-teacher interaction is mediated. And while some treat educational games as an extra tool, others as virtual environments for practice, and some others as a replacement of the teacher, the areas of knowledge these are best suited for are usually abstract and technical. We present a method adapted from Visual Novels (VN), a sub-genre of Adventure Games born in Japan, that makes use of attractive characters, narrative engagement, puzzles and other interactive features to maintain user interest while submerging players in complex stories. With this our approach we are able to teach theoretical topics through discernment in multiple game scenarios, increasing knowledge and maintaining entertainment value. We show our results from experiments with a VN for Smart Pads we developed with Participatory Design, discuss our findings, limitations and talk about our future work.

Keywords: Educational Games, Visual Novels, Adventure Games, Pervasive Games, Game Scenarios, Participatory Design.

1 Introduction

Over the last decade, the research community has showed special interest in creating tools that support education [2]. Some of the products include pervasive games. Games are powerful mediators for learning, mostly thanks to their interactive, immersive, personalization and knowledge oriented characteristics [2]. One predictor of success in studies, is the amount of time spent absorbing the content, also referred to as time-on-task [5]. Video games stimulate cognitive processes including reading, deductive and inductive reasoning, problem solving, and inference making [2], and are able to motivate players to spend longer time-on-task [5]. Thus, they can be of direct educational or skill development value. Yiannoutsou [15] designed mobile pervasive applications, considering them more as a natural aid to learning, extra tools, that stimulate imagination and engagement. She classified educational games as: 1) suppliers of content, 2) enrichers of interaction or 3) task providers. To her the pre-established notion of 'static' content (i.e. written text), is a one-dimensional flow of

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 234–243, 2013.

information, advocated for an understanding of knowledge as interactive, favoring exchange and communication. With different tasks to be performed in a 3D world, a chat interface for 'socialization', and by taking the learning process to a simulated field, Belloti [2] assured assimilation and comprehension of topics by translating text-based studying to field practice in a virtual environment. As he suggests, making educational purposes part of the game, does not compromise the overall enjoyability. In order to avoid a chocolate-covered broccoli [5], neither fun nor educative, game and psychological techniques should be interrelated. Linehan presented clear guidelines from behavioral psychology [5], which enable developers to create viable applications to teach skills with the Applied Behavior Analysis (ABA). He believes that instruction should not be evaluated in blocks, but rather, the play itself is what should be examined, changing focus from what is taught to how well a student learns.

The three approaches teach while securing enjoyment. Nevertheless, the areas of knowledge to which they may be applied are somewhat limited, as they favor memorization/practicing over discernment. Yiannoutsou [15] used exhibits in a museum to make progress on-game, by storing, manipulating and exchanging specific information on them. SeaGame [2] gave students control over what and how to learn, with an event generator that activates tasks as the user gets near-by. And ABA formatted game design [5], consists of disciplined repetitive rehearsal, crucial to success in 'task/skill' oriented education. However, to our knowledge, a method to teach content, that requires the contextualization of subjects and discernment, has not been presented. For example, deciding if a special treatment would suit better a patient based on his or her antecedents, or making a judgement after evidence has been presented. We introduce Visual Novels (VN), a popular sub-genre of Adventure Games born in Japan, that facilitates information presentation and persuasion through a method we adapted using strong stories, dramatic tension, attractive characters, interactive puzzles, among other characteristics [7]. Persuasion has been suggested to be a valid mean to educate [5]. Therefore, with the use of educational game elements [5] together with narrative engagement [11], affect in messages [10], and coping techniques to elicit thought [14], we are able to increase knowledge and persuade. We developed a pervasive game for smart pads, in workshops conducted following recommendations of participatory design [12]. We present our scope, show results analyzed through qualitative grounded theory techniques [3], discuss our findings, and talk about limitations of our approach and future work.

2 Background

A game must have planning, enacting, feedback, rewarding, learning, practice and deduction features to be considered as entertaining [6]. Planning makes reference to the setting of short and long term goals, feedback to immediate and appropriate user-tool communication, rewarding to prices or stimulus obtained with desired behavior, learning to skill teaching before evaluating, practice to the gradual growth in complexity, and deduction to the obviation degree of the right course of action. One mistake made by educative pervasive game designers, according to [6], is to focus more on the educational content, translating standard textbooks into point-and-click applications. In order to obtain a balance between our educative, persuasive, game and pervasive elements, we made use of critique techniques from Participatory Design [12]. Like [5] suggests, persuasion is a viable mean to educate. For our work we make use of narrative engagement [11], affect in messages [10] and coping (fear) techniques [14], which also serve to maintain the entertainment value of the game. Through means of regulatory fit, a 'reader' gets involved with the plot, that is, the emotions of rightness and wrongness play an important factor to transport people to scenarios in the narrative on which persuasion messages are implicit. Affect consists in framing positive messages (desired behavior) as gain framed, and negative messages (undesired behavior) as loss framed [10]. Dissonance (coping) occurs when a threatening (fear) situation is presented, and the possible outcome is not favorable to the user, as when different routes are available, people prefer to reduce such dissonance by directly changing their attitudes/behaviors, rather than alleviating through self-affirmations [14]. Linehan [5] delimited requisites for pervasive educational games to successfully present, teach and evaluate information. He accentuated factors like fun, flow, engagement, feedback, goals, problem solving, balance and pacing, interesting choices and narrative, as essential to maintain player interest. With his group, they relied on positive/negative reinforcement and punishment to maximize results [5]. Reinforcement refers to the attainment or removal of stimulus, and punishment to the chastisement of undesired positive or negative behavior. Unlike Linehan [5], we preserve interest and enjoyment through the guideline of Visual Novels (VN), which make use of multiple branch storylines, attractive characters, puzzles and affordances to constantly challenge the player to explore and get involved with the plot(s) [7]. Instead of rewarding and punishing, we persuade with the story, characters and dramatic atmosphere of the game [11,10,14]. And unlike Linehan, we evaluate not based in overall performance, but in the deductive and discernment abilities shown by the player when solving puzzles and discovering secrets. For example, to prove the efficacy of our approach, we developed a game on which, knowledge on nutrition and health was to be taught and graded. On this game, knowing what fruits and vegetables are good was not sufficient, but providing good advice to a patient with anorexia on his or her personal diet was also required.

2.1 Related Work

The concept of teaching through narrative enhanced puzzles [6], or using comics (animated characters with a dialogue) to create a narrative [1] is not new. Marsh and his team [6], developed a flash-based set of interactive puzzles, with which, physical concepts of displacement and velocity were to be learned through continuous task repetition. They managed to show how interactive games with narration(s) perform higher in fun, attention and excitement. Andrews [1] on the other hand, aimed at portraying why branching narratives encourage interaction. On his application, a predefined set of sketches was provided to users, who would adapt text and create a narration at will, following fixed guidelines. He showed how, having too loose, or providing too much creative freedom to users, was actually a limitation. Thus, he suggests that some predefined elements should be at hand. Our method differs from [6] in that with our approach puzzles serve as means of evaluating, not teaching a concept, characters are in charge of teaching content, through contextualized learning, and we have a predefined multi-branch story. Our work differs from [1] in that we make users the drama managers, and our goal is education, not just fun.

3 Visual Novels (VN)

Premise of Visual Novels (VN) is to submerge players in a narrative, allowing them to decide outcomes, and the path the story takes, at will, depending on feedback [7]. VN make use of multi-branch storyline, as opposed to common single-line story. In VN, the user makes decisions for by responding, asking questions, choosing the next step or action to take, etc [7]. As they are story driven, characters are more than mere actuators, they are actors, who follow literary conventions of personal and emotional growth, instead of power or ability upgrade [7]. Players engage with characters by choosing one of pre-written options in a menu, and every action triggers a response [7]. Conversing reveals clues about how to solve a puzzle and disclose secrets [7]. Like any other game genre, high scores provide secondary goals, and serve as indicators for progression [7]. Each decision point provides the opportunity to alter the course of action, and lead to alternative outcomes [7]. Moving from one node to the next may expand from the original, or be a completely different proposition. While some aspects of the overall story are revealed to the user from the beginning, it is not until he or she uncovers different paths that the complete story may be finally understood [7]. Characteristic features of VNs include: narrative, attractive characters, puzzles and affordances [7]. Puzzles may be of inventory, dialogue, environment, or non-contextual type, depending on their goal and connection to the story [7]. Commercial examples of VN are available in different game hardware platforms, including the web [8]. As can bee seen in figure 1, non-playing character's attitude (NPC) will variate according to user feedback, and exploration is available [8]. Common affordances in VN include virtual currency, gifts or objects necessary on later stages of the game [7]. Unlike health/progress bars in action games, or popularity bars in simulations, 'love' or 'emotional' bars (Fig. 1) depend on the relations of the player with characters, not only on tasks performed.

3.1 VN Methodology for Education

Based on the precepts discussed so far, let us present the steps to create a Visual Novel (VN) for Education: **(1)** *target a specific audience,* **(2)** *determine a topic to teach,* **(3)** *elaborate on the purpose and objectives,* **(4)** *create characters (main-role, NPC, allies and enemies), and general plot,* **(5)** *test stories and characters with future users, to see how they engage with the visual, interactive and narrative elements,* **(6)** *carefully construct the issues the user will have to deal with (dramatic tension points and climax),* **(7)** *decide the implementation technicalities, based on the vision of the* *virtual world,* **(8)** *choose a conversation pattern (dialog bubbles, recorded sound, etc.), a graphic technology for display (2D, 3D, text-based, etc.), and other media (music, BGM, etc.),* **(9)** *elaborate all visual elements, based on the perspective of the game (1st or 3rd person),* **(10)** *select interaction mechanisms, and designate how items or objects will be used,* **(11)** *determine if maps or inventories will be necessary, resolve a way to track and record user-progress, and a point-counting method,* and **(12)** *design puzzles, rewards and secrets to unlock.*

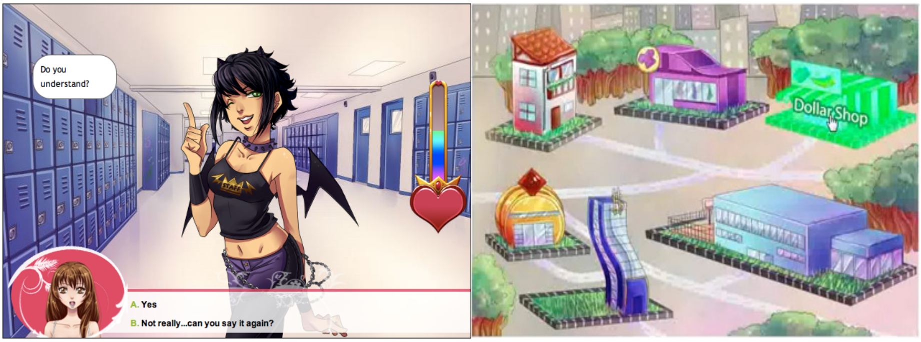


Fig. 1. Screen captures of 'My Candy Love' (art by Stephanie Sala, property of Beemov Games)

4 Amigo

To test our method, we created a game to teach nutrition to young japanese students with regular eating behaviors. Contextual teaching of nutrition and eating disorders may increase knowledge and persuade to have a healthier lifestyle [13]. While providing a predefined set of nutrition concepts to an audience may prepare them to use that knowledge in a specific occasion, presenting the same information when 'necessary' with game narrative, enabled us to help players reflect on content with multiple perspectives. To make our work culturally conscious, we created a demo using Japanese animation as inspiration for our characters, backgrounds, story and other media elements. Amigo is spanish for friend, and becoming friends with the main and non-playing characters is pivotal to our game. Our goal was to persuade participants, to make good decisions by helping the main character to recover from anorexia. We used regulatory fit for narrative engagement [11], presenting a fantasy-mystery story, on which all health/nutrition positive content was introduced as gains, and all bad behavior as losses [10], and the dangers of a poorly treated case of anorexia lead to unwanted outcomes, prompting players to use coping techniques [14]. As Fig. 2 shows, after our 'friend' had a specific problem (weakness due to self-imposed fasting), he or she would be taken by the player to the hospital, where the parent, a non-playing character (NPC), would introduce the user to the issue. Later other NPC like a professor or coach would present related nutrition information, and after a short time, this concept would be evaluated through a puzzle. This type of topic presentation is based on Wither's topic introduction method [13].

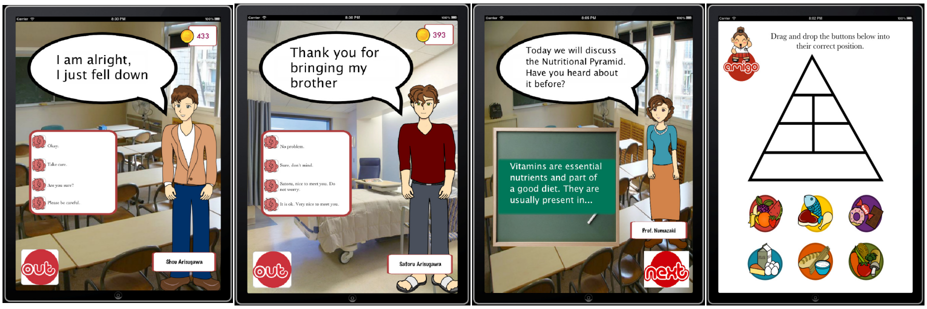

433
I am alright,
I just fell down
Okay.
Take care.
Are you sure?
Please be careful.
Shou Arisugawa
out
393
Thank you for bringing my brother
No problem.
Sure, don't mind.
Satoru, nice to meet you. Do not worry.
It is ok. Very nice to meet you.
Satoru Arisugawa
out
Today we will discuss the Nutritional Pyramid. Have you heard about it before?
Vitamins are essential nutrients and part of a good diet. They are usually present in...
Prof. Numazaki
next
amigo
Drag and drop the buttons below into their correct position.

Fig. 2. Teaching through contextual scenarios

4.1 Story

Amigo is the story of four young students who practice sports, suffer from anorexia nervosa, and try to keep an specific physical image and weight necessary to compete. Haruko, Shou, Henri and Momoka are their names. Haruko is an expert in gymnastics, and lives a very stressful life. Shou is the baseball team captain, and likes to go out every night. Henri is on the fencing team, and works at night. Momoka, on the other hand, practices Aikido yet is very into fashion, and aspires to become a model. To make our narrative interesting for boys and girls we decided on a fantasy-mystery story, in which, each main character is a protector of the magical alternate reality. This magical world is under attack by a mischievous clan, who seek to control both universes. Prof. Kudchenko is their first victim, a foreign researcher who, by accident, manages to create a portal to the fantasy world. Gameplay starts the first day of school, on which the famous foreign researcher is killed. Non-playing characters include parents, schoolmates, enemies, a detective, etc. Interaction with the actors allows the player to get involved, find clues, help their friend, defeat the enemy, and solve the mystery: who killed Prof. Kudchenko and why?. To maintain the story related with the educative goals, the way in which the main character remains strong, and defeat the bad guys, is by having a good nutrition. More scenes from the game can be appreciated on figure 3.

4.2 Teaching about Nutrition with Eating Disorders

Eating disorders are a persistent and severe disturbance, of regular eating habits, that impair physical health and functioning, and are not due to a medical or psychiatric conditions [4]. Well known disorders include bulimia, binge eating, anorexia nervosa, etc. For our experiment we concentrated on anorexia nervosa. Our intention was to elicit narrative engagement through a fantastic story with charming characters [11] who present health positive/negative behavior, and to persuade/educate through coping techniques [14], by helping students meditate and discern over nutrition content with a case in which such information is necessary. "Who would be able to use this type of information?", students and professors from areas in which, variables/conditions, may constantly change, or in which performance is subject to external factors. "Why would anyone take an approach like this?", although some may incline for a more traditional, less heterodox style of teaching, we believe that making the learning process fun and interesting may be beneficial. Two features must be present to make a diagnosis of anorexia: attitudes towards weight in which self-worth is valued based on physical appearance, and the active maintenance of low body weight [4]. People with anorexia eat limited amount of food, and constantly avoid those foods perceived as fattening. We sought to reflect with our main characters personality traits of anorexic people (perfectionism, steadfast determination and inflexibility) [4]. The most effective treatment is cognitive behavior therapy [4].

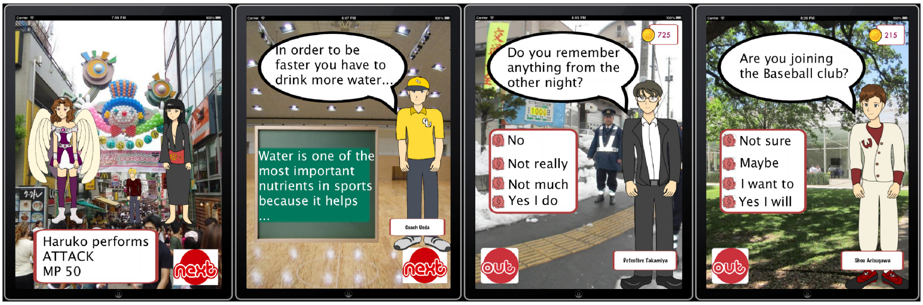

Haruko performs
ATTACK
MP 50
next
In order to be faster you have to drink more water...
Water is one of the most important nutrients in sports because it helps ...
Coach Ueda
next
725
Do you remember anything from the other night?
No
Not really
Not much
Yes I do
Detective Takamiya
out
215
Are you joining the Baseball club?
Not sure
Maybe
I want to
Yes I will
Shou Arisugawa
out

Fig. 3. Action and drama scenes in the game

4.3 Implementation

We developed an application for iPads using ObjectiveC, C++ and SQLite. Benefits of smart pads include: portability, accessibility, interactivity and availability on the market. By portability we refer to the possibility to take the pad or tablet to whatever location. Accessibility is related to internet connectivity, available in such gadgets. And interactivity is obtained thanks to its touch enabled screen. When looking for ways to present a VN, several ideas came to our mind. An online application was contemplated, yet discarded as it may be too space-restrictive, and it would be complicated to have a standardized use of sensors. Smart pads also gave as a possibility to update data, manage databases, tables and other structures. Like commercial games, Amigo contains a database, in which user progress, data and profile are registered. With user feedback we activated an event handler, in our case a touch-event reader, which in turn communicated to the narrative manager: a special class created to make sure that the correct node was accessed. Each node connected to the story container (on the database), and sent the action director (view) the correct content to display. Every character represented a different tree branch, and each node a new course of action.

5 Evaluation

To verify the effectivity of our method for educative Visual Novels (VN), we conducted a set of experiments and interviews with Amigo. We had questionnaires, surveys and follow-up interviews conforming to qualitative grounded theory [3]. We recruited 19 participants average age of 22.5 years, 15 males and 4 females, from the School of Engineering and School of Liberal Arts, with regular to normal eating habits and who were willing to learn about nutrition. To validate our approach, we randomly divided them into two subgroups: (T) Treatment, 10 students with the multiple scenarios version, and 9 students with the (C) Control version with linear scenarios. By multiple scenarios we mean the contextualized presentation of information through game scenes. Our control group was introduced to topics in a straightforward manner. We made an adaptation form Wither's 5 videotape method [13] by presenting 5 concepts with game scenarios, later evaluated with puzzles. To verify the success in learning we measured empathy, knowledge and intention to diet. All our questionnaires and surveys were 5-scale Likert instruments. Thanks to questions like “was [character] realistic?”, or “would you help someone in a situation similar to [character]?”, we were able to calculate empathy. With a questionnaire prior to start, and after play, corroborated with the results of the puzzles, we managed to measure knowledge. Intention to diet was obtained through questions on interviews like “do you plan to go on or continue to diet in the near future?”.

5.1 Results

The results we obtained can be appreciated with more detail on table 1. We calculated median and standard deviation for the first two variables. Average time of play per individual was of 58 minutes. For empathy, we made two samples, after a break for the game, and once finished. Knowledge was corroborated before playing (Sample 1), and after ending the game (Sample 2). Intention to diet was measured only once, and was asked at follow-up interviews. As can be appreciated in the table, there was little difference in empathy between treatment (T) and control (C) groups. This could be attributed to the aesthetic and visually attractive characteristics of Visual Novels, which engage players with their features. Differences are more visible on the knowledge part, as the treatment group outperforms control by 0.8, yet not too substantial. One could interpret that the multiple scenarios help, and are able to 'teach' correctly 4 out of every 5 contextualized topics, while Visual Novels with educative content presented in a straightforward manner in 3 out of 5. Intention to diet on the other side, was slightly higher with the control group, with 5 participants out of the 9 clearly expressing their desire to follow a diet, versus 5 out of 10 in the treatment group. The latter of course, may be ruled out as circumstantial, as control group only had 9 integrants. We intend to create a new version of Amigo with the suggestions and ideas given to us by participants. Follow-up interviews were scheduled days after the experiment. These were moderated using recommendations from qualitative grounded theory, to complete the categories we obtained through theoretical sampling [3].

Table 1. Results of the experiments with (T) treatment and (C) control groups

		Sample 1		Sample 2	
		M	SD	M	SD
Empathy	T=	3.3	1.25	4	1.41
	C=	3.3	1.5	4	0.71
Knowledge	T=	2.2	2.48	4	0.71
	C=	1.5	2.12	3.2	0.35
Intention to diet	T=	50%			
	C=	55%			

6 Discussion

Some volunteers related to the main characters in a very realistic way. One boy, from the Liberal Arts School, who does volunteer work, in the control group, said *"I wanted to help [character], I mean, I tried to, but I wanted to do more"*. When asked to elaborate on his reasons he said *"I do volunteer work, and I have worked with people who have a similar condition, so I felt complied"*. After checking his results, he ranked particularly high on empathy. A girl from the treatment group, said *"I had a roommate who was just like [character], she would also stay all day without eating... I mean, how can anyone do that to their body?"*. She was asked if she felt that characters were particularly realistic, to which she answered *"the images were static (not moving), but I mean, those situations I lived them before so, they made me remember my friend"*. Another boy, the youngest of the treatment group, said *"Wow! That was nasty! (referring to the murder scene), isn't it too realistic for a game?, maybe you could try making it more like other games"*. He was asked to be more specific, and responded *"well, you know... other games are less graphic"*. This effect may have occurred because of the transportation features of narratives, and their capacity to engage audiences [11]. People on both groups, shared their experiences with other participants, those who were nearer to them, and compared their results even though they weren't asked to do so. *"What did you get?"* asked a boy to a fellow classmate, *"I think it went well"*, responded his friend. Another guy asked *"did you get [secret]?"*, *"no it didn't appear to me"* answered the other. *"So! How was it?"* said a girl to other girl in her group, *"it was okay for me, and you?"* the other responded, *"To me it was long"* the first girl answered. This follows our social nature, and the fact that sharing and competing with scores, particularly after playing the same game or when being evaluated, seems to be part of expected player interaction [9, 15].

7 Conclusions and Future Work

We were able to teach contextualized concepts, with our Visual Novels (VN) for education using narrative engagement, affect in messages and coping techniques to persuade. VN are known for their attractive characters, interesting stories, puzzles and affordances. With our methodology, we developed a pervasive game for Smart Tablets, with which we taught young college students, with regular to normal eating habits, about proper nutrition. We conducted a set of experiments to validate our approach, and had follow-up interviews to build up on the categories we obtained through theoretical sampling. We presented results, discussed our findings, and showed strengths and limitations. Our process managed to teach 4 out of 5 topics on average, and motivated half of the students to put the knowledge into practice (i.e. intention to diet). For future work, we intend to develop a new version of the game with a design that is more in line with the tastes of our future participants.

References

1. Andrews, D., Barber, C., Efremov, S., Komarov, M.: Creating and using interactive narratives: reading and writing branching comics. In: Proc. of CHI 2012 (2012)
2. Belloti, F., Berta, R., De Gloria, A., Primavera, L.: Enhancing the educational value of video games. ACM Computers in Entertainment 7(2), Article 23 (2009)
3. Charmaz, K.: Constructing grounded theory – a practical guide through qualitative analysis. SAGE Editorial (2008)
4. Fairburn, C.G.: Eating disorders. Encyclopedia of Life Sciences, John Wiley & Sons Publishing (2001)
5. Linehan, C., Kirman, B., Lawson, S., Chan, G.G.: Practical, appropriate, empirically-validated guidelines for designing educational games. In: Proc. of CHI 2011 (2011)
6. Marsh, T., Xuejin, C., Li Zhiqiang, N., Osterweil, S., Klopfer, E., Haas, J.: Fun and learning: the power of narrative. In: Proc. of FDG 2011 (2011)
7. Rollings, A., Adams, E.: On game design. New Riders Editorial, ch. 15: Adventure Games (2008)
8. Sala, S., Beemov Games: "My Candy Love". Beemov Games (2012), Web game accessible from: `http://games.beemov.com` & `http://www.mycandylove.com`
9. Smith, I., Connsolvo, S., Lamarca, A.: The drop: problems in the design of a compelling pervasive game. ACM Computers in Entertainment 3(3), Article 4C (2005)
10. Van't Riet, J., Ruiter, R.A.C., Werrij, M.Q., Candel, M.J.J.M., De Vries, H.: The role of affect in message framing effects. European Journal of Social Psychology 40 (2010)
11. Vaughn, L.A., Hesse, S.J., Petkova, Z., Trudeau, L.: This story is right on: the impact of regulatory fit on narrative engagement and persuasion. European Journal of Social Psychology 38, 447–456 (2009)
12. Vines, J., Blythe, M., Lindsay, L., Dunphy, P., Monk, A., Olivier, P.: Questionable concepts: critique as a resource for designing with eighty somethings. In: Proceedings of CHI 2012 (2012)
13. Withers, G.F., Werthem, E.H.: Applying the elaboration likelihood model of persuasion to a videotape based eating disorders primacy prevention program for adolescent girls. Eating Disorders Journal (2004)
14. Wood, W.: Attitude change: persuasion and social influence. Annual Review of Psychology 51, 539–570 (2000)
15. Yiannoutsou, N., Papadimitriou, I., Komis, V., Avouris, N.: Playing with museum exhibits: designing educational games mediated by mobile technology. In: Proc. of IDC 2009 (2009)

An Optimal Radio Access Network Selection Method for Heterogeneous Wireless Networks

Glaucio H.S. Carvalho[1], Isaac Woungang[2], Md Mizanur Rahman[2],
and Alagan Anpalagan[3]

[1] Faculty of Computing, Federal University of Para,Belem, Para, Brazil
[2] Dept. of Computer Science, Ryerson University, Toronto, ON, M5B 2K3, Canada
[3] Dept. of Electrical Engineering, Ryerson University, Toronto, ON, M5B 2K3, Canada

Abstract. In Heterogeneous Wireless Networks (HetNets), different overlapped Radio Access Networks (RANs) can coexist with each other in the same geographical area with the goal to provide high rates pervasive access for mobile users via multi-mode terminals. This paper addresses the problem of initial RAN selection in HetNets with two co-located wireless networks which supports two different service classes. The framework of Semi-Markov Decision Process (SMDP) is used to formulate the problem as a Joint Call Admission Control (JCAC) optimization problem that involves the design of a total cost function that weights two criteria: the blocking cost and the energy consumption cost. Simulation results are provided, showing that our JCAC optimal policy often selects the less energy consuming RAN when more weight (50% or more) is given to the energy consumption cost in the total cost function.

1 Introduction

A HetNet integrates two or more different wireless networks into a single architecture, each having its own characteristics in terms of coverage, quality of service (QoS) assurance, implementation, operation costs, to name a few. The integrated networks provide overlap coverage in the same wireless service areas, allowing users to enjoy a large variety of innovative services based on their demands, in a cost efficient manner. Elementary to the operation of HetNet is the existence of modern multi-mode wireless terminals [1]. This type of terminals has more than one radio interfaces, each enabling them to access a different access technology.

The optimization of HetNets provides a substantially higher overall system capacity and can help saving energy when implemented into 4G wireless networks. Indeed, when a service request comes to a 4G system, it can direct the request to the network that best suits the user's requirements and complies with the status of the different networks. Moreover, the admission load can be balanced between the different networks in presence. This paper addresses the issue of a Joint Radio Resource Management (JRRM) algorithm for HetNets that is fully controlled by the network and performs the initial RAN selection task for two types of traffic classes in a HetNet with two co-located RANs. The goal of our model is to provide the optimal initial RAN selection based on a cost function (here composed of blocking cost and energy

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 244–252, 2013.

consumption cost). In order to minimize the blocking probability for each service class and the energy consumption, as well as maximize the system capacity of each RAN, the optimal RAN selection can decide which service class should be served by which RAN and when. In this work, we also consider the system as continuous and dynamic Markov decision process, which is governed by continuous arrival and departure of calls. Therefore, we use SMDP to model the system in order to take the decisions that are not fixed (i.e. discrete) but random. Finally, by using the Value Iteration Algorithm [2], the optimal RAN selection policy for system capacity and energy consumption are computed by our model.

The rest of the paper is organized as follows. Section 2 describes some related work. Section 3 presents the system and traffic assumptions. In Section 4, our SMDP model is described. In Section 5, simulation results are given. Section 6 concludes our work.

2 Related Work

Several methods for RAN selection in HetNets have been investigated in the literature. Representative ones are as follows. In [3], Giupponi et al. proposed a JRRM framework for achieving an efficient usage of a joint pool of resources belonging to different RANs. The framework includes a specific function called: RAN and cell selection - which aims at selecting the best RAN that maximizes the network resource utilization. In [4], Kandaraj et al. proposed a RRM framework made of resources monitoring, decision making, and decision enforcement functions – which can allocate resources to different classes of users in a satisfactory manner. In [5], Prez-Romero et al. proposed a policy-based RAN selection algorithm in which a function selects an initial RAN from a set of available RANs based on different inputs such as service class, load in each RAN, mobile speed. To avoid blocking possibilities when there is capacity available in other RANs, complex policies are proposed by combining basic policies in which the output is prioritized for a list of RANs. In [6], Olabisi and Anthony proposed a dynamic RAN selection algorithm for assigning a multimode terminal with a single call or group of calls to the most suitable RAN in an HetNet. To select the preferred RAN, the available RANs are rated using a multi-criteria group decision-making technique. In [7], Jin et al. proposed a RAN selection procedure for HetNets based on a fuzzy logic-based algorithm that achieves load balancing. However, a user has to wait until sufficient bandwidth is released by the network components before having access to the network. In [8], Marko and Borislav used a two-dimensional Markov chain to design an initial RAN selection approach using the service type, user mobility and network load as design criteria. Most of the above schemes do not consider the system as continuous i.e. the decision is taken at fixed time interval. Our work differs from the above in the sense that we here propose a cost function which accounts for two optimization criteria: blocking cost – that reflects the overloaded RAN and energy consumption cost – which is meant to save energy. Using SMDP, our model can handle situations where decision should be taken whenever a change (arrival or departure of calls) occurs in the network.

3 System and Traffic Assumptions

We consider a HetNet composed of 2 co-located RANs. The j^{th} RAN, $j=1,2$, has N_j radio resources. Each incoming service (call) is served by allocating the required amount of resources from one of the available RANs. In our design, the system capacity can be implemented by its effective (or equivalent) bandwidth, no matter which multiple access technology (FDMA, TDMA, CDMA, or OFDM) is used for implementing the radio interface. In this environment, when an incoming service connection (call) requests an access to the network, the optimal RAN selection method has to decide not only if it will be accepted, but also which RAN accepts it. The HetNets supports K classes of service connections, every class categorized by its bandwidth requirement, arrival distribution, and channel holding time. We consider two types of service connections $(i = 1,\ 2 \in K)$. We also assume that the i^{th} service connection arrives according to a Poisson process with parameter λ_i and requires b_i radio resources (bandwidth). The channel holding time (i.e. connection duration added to residence time) is assumed to follow an exponential distribution with mean value equal to μ_i. Finally, the traffic intensity is defined by λ_i / μ_i

4 Our SMDP Model

A SMDP model is completely defined by the following components: the state space, the decision epochs and the actions, the expected time until the next decision epoch, the state transition probabilities, and the cost function. The states of the SMDP are a five-tuple defined by:

$$S = (n_{11}, n_{21}, n_{12}, n_{22}, e)$$

The constraints associated to each RAN are as follows:

$$0 \le n_{11} \le \lceil N_1/b_1 \rceil$$

$$0 \le n_{21} \le \lceil N_1/b_2 \rceil$$

$$0 \le n_{12} \le \lceil N_2/b_1 \rceil$$

$$0 \le n_{22} \le \lceil N_2/b_2 \rceil$$

$$e = [0\ 1\ 2]^T$$

where M^T denotes the transpose of matrix M, n_{ij} is the number of calls of type i connection in RANj, N_j is the capacity of RANj, b_i is the bandwidth required by the type i connection (call) and e = 0 is the departure of connections and $e = 1$(resp. $e = 2$) is the arrival of connection of the type 1 (resp. type 2).

- ***Decision Epochs and Actions***: there are three possible actions for the JCAC policy, namely: block (B), accept in RAT-1 (AR1) or accept in RAT-2 (AR2). In each state s $\in$ S, the controller can choose one out of the possible actions:

$$A(x) = \begin{cases} B, & e = 0, 1, 2 \\ AR1, & e = 1, 2 \text{ and } b_{(i=(1,2))} + b_1 n_{11} + b_2 n_{21} \leq N_1 \\ AR2, & e = 1, 2 \text{ and } b_{(i=(1,2))} + b_1 n_{12} + b_2 n_{22} \leq N_2 \end{cases}$$

- ***Expected Time until the Next Decision Epoch:*** If the system is in the state $x \in S$ and the action a$\in A(x)$ is chosen, the expected time until the next decision epoch is determined as:

$$\tau(x, a) = \frac{1}{\lambda_1 + \lambda_2 + n_{11}\mu_1 + n_{21}\mu_2 + n_{12}\mu_1 + n_{22}\mu_2}$$

- ***Transition Probabilities:*** Let $p(x,y, a)$ be the probability that at the next decision epoch, the system will be in state y , y$\in S$ if action a$\in A(x)$ is chosen in state x. Let τ (x,a) be the expected time until the next decision epoch if action a$\in$ $A(x)$ is chosen in state x. The transition probabilities are then obtained as:

$$p(x, y, a) = \begin{cases} \lambda_1\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 1) \Rightarrow y = x, & \text{if } \ a = B \in A(i) \\ \lambda_1\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 1) \Rightarrow y = (n_{11} + 1, n_{21}, n_{12}, n_{22}, e), & \text{if } \ a = AR1 \in A(i) \\ \lambda_1\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 1) \Rightarrow y = (n_{11}, n_{21}, n_{12} + 1, n_{22}, e), & \text{if } \ a = AR2 \in A(i) \end{cases}$$

in case of arrival of type-1 call.

$$p(x, y, a) = \begin{cases} \lambda_2\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 2) \Rightarrow y = x, & \text{if } \ a = B \in A(i) \\ \lambda_2\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 2) \Rightarrow y = (n_{11}, n_{21} + 1, n_{12}, n_{22}, e), & \text{if } \ a = AR1 \in A(i) \\ \lambda_2\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 2) \Rightarrow y = (n_{11}, n_{21}, n_{12}, n_{22} + 1, e), & \text{if } \ a = AR2 \in A(i) \end{cases}$$

in case of arrival of type-2 call.

$$p(x, y, a) = \begin{cases} n_{11}\mu_1\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 0) \Rightarrow y = (n_{11} - 1, n_{21}, n_{12}, n_{22}, e), & \text{if } \ a = B \in A(i) \\ n_{21}\mu_2\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 0) \Rightarrow y = (n_{11}, n_{21} - 1, n_{12}, n_{22}, e), & \text{if } \ a = B \in A(i) \\ n_{12}\mu_1\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 0) \Rightarrow y = (n_{11}, n_{21}, n_{12} - 1, n_{22}, e), & \text{if } \ a = B \in A(i) \\ n_{22}\mu_2\tau(x, a),\ x = (n_{11}, n_{21}, n_{12}, n_{22}, 0) \Rightarrow y = (n_{11}, n_{21}, n_{12}, n_{22} - 1, e), & \text{if } \ a = B \in A(i) \end{cases}$$

in case of departures of calls.

Total Cost Function: If the system is in the state $x \in S$ and the action $a \in A(x)$ is chosen, the admission control incurs in the following cost function:

$$C(x,a) = \omega_1 g_{bc}(x,a) + \omega_2 g_{ec}(x,a),$$

where ω_1 and ω_2 are respectively the weight of the blocking cost function and the weight of the energy consumption cost function. These values may be viewed by wireless designers as a way to determine the relative importance of each system's objective. It is assumed that $\omega_1 + \omega_2 = 1$. The blocking cost function is defined as:

$$g_{bc}(x,a) = \begin{cases} BC_i, \; e = 1,2 \quad \text{and} \quad a = B \\ 0, \quad \text{otherwise} \end{cases}$$

where BC_i is the blocking cost of the i^{th} service class. The energy consumption cost function is defined as:

$$g_{ec}(x,a) = \begin{cases} \frac{E_j}{\max_j(E_j)}, \; e = 1,2 \quad \text{and} \quad a = AR_j \\ 0, \quad \text{otherwise} \end{cases}$$

where E_j is the energy consumed by the j^{th} RAN. It should be noticed that in the above equation, the energy is normalized so that its value less than or equal to 1.

5 Simulation Results

Performance measurements: The mean carried traffic is computed as

$$O_c^a = \sum_{x \in S; e=1,2; a=AR1,AR2 \in A(x)} \left(\sum_{j=1}^{2} \lambda_j + \sum_{j=1}^{2} \sum_{i=1}^{2} n_{ij} \mu_i \right) \pi_x$$

where π_x, for all $x \in S$, is the continuous time Markov chain steady state probability distribution under the optimal policy.

The probability that an arrival of a new type i service connection seeking admission into a RAN is blocked is called new connection blocking probability of service class i. Thus, the connection blocking probability of service class i^{th} is obtained as:

$$Pb_i = 1 - \frac{O_i^a}{\lambda_i}$$

The bandwidth utilization is defined as the ratio between the mean number of occupied channels and the total number of channels. The utilization of j^{th} RAN is computed as

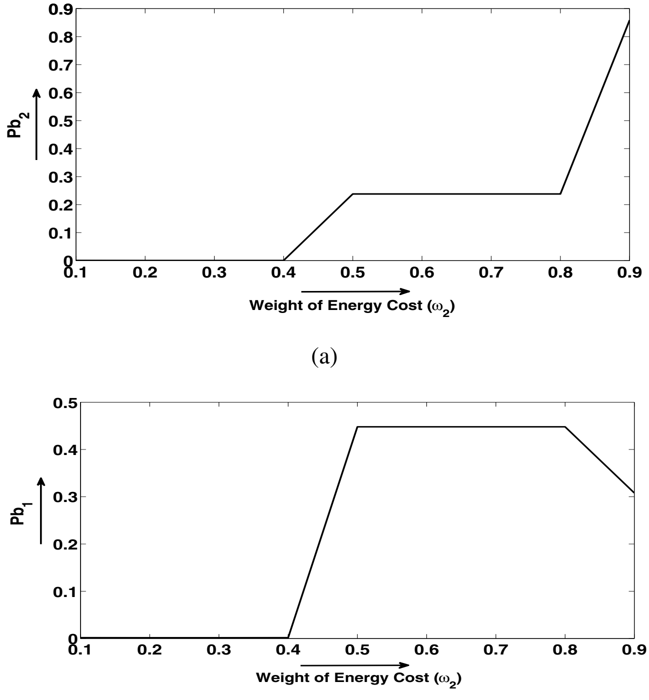
$$U_j = \frac{1}{N_j} \sum_{x \in S; a \in A(x): \forall i; \forall j; n_{ij} > 0} b_i n_{ij} \pi_x$$

System Configuration: We have considered two co-located networks: RAN-1 and RAN-2 and two service classes: class-1 and class-2. The blocking costs of class-1 and class-2 are 1 and 0.8 respectively, and the power consumption required for RAN-1 and RAN-2 are 3802 W (GSM) and 300 W (UTMS) respectively. Other parameters are shown in Table 1.

Table 1. System Configuration

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
N_1	20 channels	μ_1	1/120s (voice)	b_1	2 channels	ρ_1	5
N_2	10 channels	μ_2	1/120s (voice)	b_2	1 channel	ρ_2	3

The following scenario is considered: we vary the weight of the energy consumption cost (within the cost function given in Equation 7) and we measure the impact of this variation on the system's performance. The results are depicted in Fig. 1 to Fig. 3.



(a)

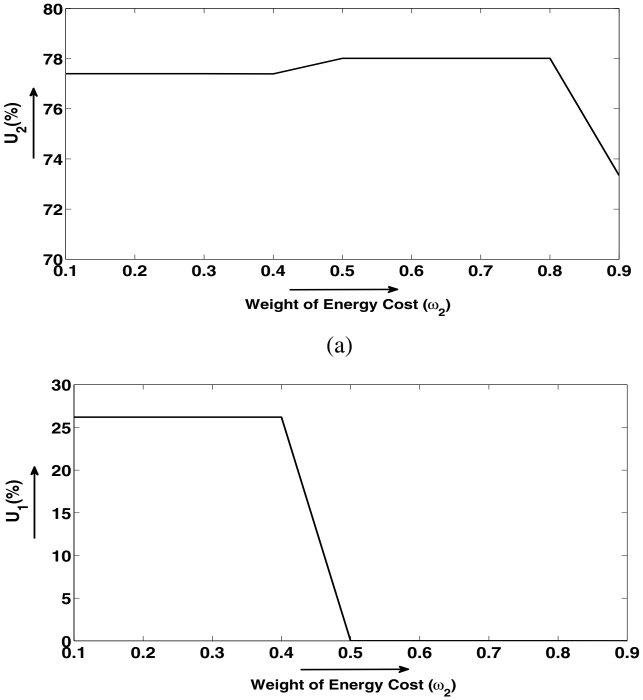
(b)

Fig. 1. Blocking probability versus weight for energy consumption cost (a) blocking probability for class-1 call, and (b) blocking probability for class-2 call

In Fig. 1, it can be observed that the optimal policy equally accepts both type of incoming calls (class-1 and class-2) when the weight of energy cost (i.e. ω$_{2)}$ is set from 0.1 to 0.4. However, when we give more emphasis on energy efficiency, i.e. ω$_{2 \geq}$0.5, the optimal policy starts to reject both type of incoming calls in order to save energy. It can also be seen from the figures that more class-1 calls are blocked by the optimal policy. The reason behind this trend is the amount of bandwidth required by it. However, for ω$_{2\,=}$ 0.9, figure 1 also reports that the class-1 call acceptance rate is higher than the class-2 call acceptance rate by the RANs.

In Fig. 1, it can be observed that the optimal policy equally accepts both class-1 and class-2 incoming calls when the weight of energy cost (ω_2) is set to a value in the range 0.1 to 0.4. However, when more emphasis is given on energy efficiency, i.e. $\omega_2 > 0.5$, the optimal policy starts rejecting class 1 and class 2 incoming calls in order to save energy.

In Fig. 2, it can be observed that the optimal policy utilizes more channels of RAN-2 (77%) than that of RAN-1 (26%). This is attributed to the fact that RAN-1 requires more power than RAN-2 does for operating, resulting to energy consumption savings. When ω$_{2\,\geq}$ 0.5, RAN-1 channel utilization reaches almost 0% and RAN-2 channel utilization increases slightly (above 77%) as more calls that were supposed to be



(b)

Fig. 2. RAN utilization versus weight of energy consumption cost in total cost: (a) RAN-1 utilization, and (b) RAN-2 utilization

served by RAN-1 are carried by RAN-2. It can also be observed that the channel utilization of RAN-2 decreases sharply when ω_2 is set to its maximum value.

In Fig. 3, it can be observed that the optimal cost increases when ω_2 is set to a value less than 0.5, and decreases thereafter. This is attributed to the fact that the system accepts more connections until when ω_2 is set to 0.5, after which the system starts blocking more calls in order to save some energy.

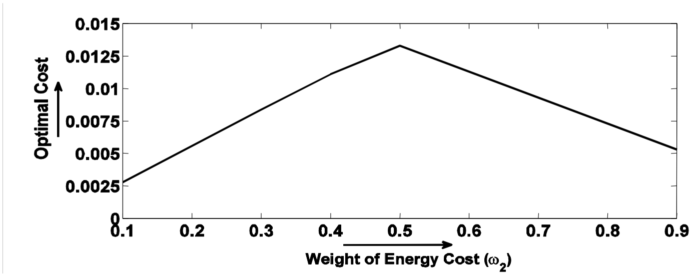


Fig. 3. Optimal Cost

6 Conclusion

We have proposed an optimal RAN selection for HetNets by using an SMDP framework. Our optimal RAN selection method considers a cost function that weights two parameters: the blocking cost and the energy consumption cost, leading to some flexibility in judging the level of energy consumption in the network. Simulation results have shown that variations in the weights of th energy consumption cost can greatly impact the system's capacity and the overall network utilization.

References

1. 3GPP Technical Specifications TR 21.910, Multi-mode UE issues; categories, principles and procedures, v3.0.0 (March 2000)
2. Tijms, H.C.: A first course in stochastic models, 2nd edn. John Wiley & Sons Ltd. (April 2003) ISBN-10: 0471498807
3. Giupponi, L., Agust, R., Prez-Romero, J., Roig, O.S.: A Novel Approach for Joint Radio Resource Management Based on Fuzzy Neural Methodology. IEEE Trans. on Vehicular Technology 57(3) (May 2008)
4. Kandaraj, P., Adlen, K., Jean-Marie, B., Cesar, V.: Radio resource management in emerging heterogeneous wireless networks. Computer Communications 34(9), 1066–1076 (2011)
5. Prez-Romero, J., Sallent, O., Agust, R.: Policy-based Initial RAT Selection algorithms in Heterogeneous Networks. In: Proc. of 7th IFIP Intl. Conference on Mobile and Wireless Communication Networks (MWCN 2005), Marrakech, Morocco, September 19-21 (2005)

6. Olabisi, E., Anthony, H.: Adaptive Bandwidth Management and Joint Call Admission Control to Enhance System Utilization and QoS in Heterogeneous Wireless Networks. EURASIP Journal on Wireless Communications and Networking (2007), doi:10.1155/2007/34378
7. Jin, S., Xuanli, W., Sha, V.: Load Balancing Algorithm with Multi-Service in Heterogeneous Wireless Networks. In: 6th Intl. ICST Conference on Communications and Networking (CHINACOM), Harbin, China, August 17-19, pp. 703–707 (2011)
8. Marko, P., Borislav, P.: Radio Access Technology Selection Algorithm for Heterogeneous Wireless Networks Based on Service Type, User Mobility, and Network Load. In: Proc. of TELSIKS 2011, Serbia, October 5-8, pp. 475–478 (2011)

Desktop Grid Computing at the Age of the Web*

Leila Abidi[1,2], Christophe Cerin[1], and Mohamed Jemni[2]

[1] Universitde Paris 13, LIPN UMR CNRS 7030, 99, avenue Jean-Baptiste Clement,
93430 Villetaneuse, France
[2] Universitde Tunis, LaTICE, ESSTT, 5 Avenue Taha Hussein,
BP, 56, Bab Manara, Tunis, Tunisie
{leila.abidi,christophe.cerin}@lipn.univ-paris13.fr,
mohamed.jemni@fst.rnu.tn

Abstract. In this paper we address the problem of deploying Desktop Grid (DG) middleware when we need it, where we need it, and on any device. One option is to put DG middleware into the cloud but at the condition that the code is suited for integration into the cloud. DG middleware were not generally developed with this option in mind. We propose an advanced prototype for a DG middleware able to run on small devices, i.e. smartphones and tablets as well as on more traditional computing devices (PCs). We explain, based on our experience, that the integration of existing DG middleware for small devices may be extraordinary challenging, resulting in rethinking the DG paradigm in terms of interactions between the components. We adopt a user-centric point of view in considering that the DG technology should be as simple as possible in its use. In another words, we are exploring the ways to offer DG as a service. Our prototype serves to illustrate our techniques and methodologies and to get a feedback and an analysis of our design.

Keywords: Service-based Grid Computing, Grid and Cloud middleware, Resource management, Cooperative systems, Redis, Publish-Subscribe paradigm.

1 Introduction

Originally, Desktop Grid [1] systems (DG) represented an alternative to supercomputers and parallel machines, and now, they serve, for instance, as data caches between grid systems. DGs offer computing power at low cost. They are built out of commodity PCs and use Internet as the communication layer. DGs aim at exploiting the resources of idle machines over Internet. Many DG systems [2] have been developed using a centralized model. These infrastructures run in a dynamic environment where the number of resources may change permanently.

In this paper, our contribution is to introduce, through a concrete example, good practices in developing DGs in the context of Web 2.0 and cloud technologies, we mean software applications that are built upon the Web as opposed to upon the

* Experiments presented in this paper were carried out using the Paris 13 experimental testbed.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 253–261, 2013.

desktop. The interaction schema between components is based on the Publish-Subscribe paradigm which is not a new concept but we merge it with other concepts originated from Web 2.0, as Redis, a no-SQL tool with capabilities for the publication and the subscription. Our scientific objectives are to rethink interactions between the components of any DG according to modern Web technologies and to discuss good practices to accomplish the task rather than to investigate advances of scheduling nor to investigate advances of scheduling mechanisms in large scale publish-subscribe systems. We assume that the implementation of the Publish-Subscribe paradigm scales well and we use it to fulfill our scientific objectives. However, the interactions are entirely specified in terms of publication-subscription notifications which is unconventional but relevant in the field of DG middleware.

This paper is organized as follows. After an introduction of the context of this work, we raise in Section 2 the principle issue of our work and we introduce resource coordination and the benefit of using Publish-Subscribe systems. Section 3 raises the issue of the integration of DG in the cloud. Section 4 is related to our contributions and introduces how we have organized our prototype. It presents details of the software prototype based on Redis and lessons learned. Section 5 concludes the paper.

2 Context

Traditionally, migrating desktop applications to Internet-centric technologies was hard to justify due to scientific, technological challenges and uncertain market value. Internet-centric applications require dedicated distributed control, in our case we need to monitor participants and simultaneously we have to check the results produced by participants on processor.

The increasing number of intelligent mobile devices with new business opportunities puts a significant pressure to make existing applications supporting these new platforms. New Web and Cloud technologies provide now feasible means to put almost any desktop functionality "in the Internet". However, we believe that a special attention is necessary at earlier stage of the design of the interaction between entities in order to get confidence in the Internet-centric application. In this work, we adopt a user-centric point of view in considering that the DG technology should be as simple as possible in its use. We expect that the user may not be a computer scientist nor a system administrator and DG application should be deployed with a single click In another words, we are exploring the ways to offer DG as a service.

2.1 Key Issues in Designing DG Middleware

Despite more than one decade of work [2], the design of DG middleware still faces many challenges. Some have been identified in the past but stay 'open', others are new (the role of Web 2.0 particularly the interfaces that allow developers with little technical knowledge to appropriate the new features of the Web) or have been introduced recently (formal approaches). The identified issues in the field and related to our work are:

- The communication paradigm (for coordination, not for exchanging data) adopted should provide a high level of asynchronism in order to promote scalability; The question is: what is the appropriate model for controlling and coordinating components of a DG middleware.
- Web 2.0 technologies are the future for designing distributed systems, hence it would be a benefit to take advantage of them. On the one hand, the Web 2.0 technologies assist to advertise DG goals and attract computational resources for DG communities. On the other hand, Web 2.0 systems need to handle heavy data traffic and complex relations, that require extraordinary large computational power. The question is: how Web 2.0 and grids technologies may merge.
- Grid technologies may serve as building blocks for Cloud technologies. In [3], we have explained how the DG paradigm is reused for the SlapOS system which is a provisioning and billing system for the cloud. The question is: which problems in clouds can be solved with grid technologies.

2.2 Resources Coordination

DGs are characterized by a dynamic environment due to the heterogeneity and volatility of resources. Usersdevices can join or leave the grid at any time, without any constraint. Each machine has its own properties such as its memory size, bandwidth, CPU/core numberwhich make the scheduling task difficult. The main problem with DGs is with coordination, especially when we have to execute applications that are modeled by a direct acyclic graph of tasks with precedence. To bypass these problems, we adopt a coordination mechanism based on the Publish-Subscribe paradigm which is an asynchronous mode for communicating between entities [4, 5]. This paradigm offers a total decoupling between the production and the consumption of services, and therefore, this increases the scalability by eliminating many sorts of explicit dependencies between participating entities.

3 Motivation for the Integration of DG in the Cloud

Beyong the integration algorithm introduced in the next section, we now briefly discuss the advantages of rethinking DG middleware in the context of cloud computing. We justify the work done in this paper according to the general objectives stated in the Introduction section: how to propose DG as a service.

DG as a service can be accomplished in integrating major DG middleware (BOINC, Condor, XtremWeb into a cloud. We have recently used the SlapOS [3] open source cloud, for this purpose. Different papers are under reviewing about that issue and, at this time, only one paper (in french) is publicly available [6]. The integration of DG into the cloud makes DG as services, and since we have used SlapOS we can possibly consider other SaaS (for instance for the purpose of billing) in a coherent way in order to propose a business application taken into account realistic features. At present time, BOINC, Condor, XtremWeb have dedicated and different means for accounting the resources used and for rewarding participants. Putting all these middleware in the SlapOS cloud offers an opportunity to tackle the problem since SlapOS is based on the open source ERP5 software. In some way, the goal is to extend DG features for the purpose of business.

The main features of SlapOS are as follows, in first approximation: a) It does not rely on virtualization b) It does not rely on data centers c) It reuses, in part, some concepts of DGs [2]: machines at home host services and data, a ?astercontains the services and publishes them in a directory d)The interoperability property is achieved in deploying the SLAPGRID daemon on nodes, for instance from Amazon, Azurethen installing on these nodes the good software versions. The integration of applications to SlapOS means that someone describes an automatic deployment procedure, we call it a recipe, allowing at the end that the services requested by the user are functional. The difficulties, related to SlapOS, are its inability to work in root mode; indeed BOINC and Condor recommend to create a dedicated user for instaling files and for the execution of daemons. Since it is not possible to work in root mode, then we must find a configuration with the user name provided with the SlapOS partition.

With the lack of virtualization for SlapOS partitions, isolation must be made sometimes with recipes, which may require to configure the components that will be used in particular ways. Furthermore the partition for the user has no root access (it is impossible to write in /etc, /opt and operates in a dedicated directory of the computer. This is often difficult for some components to proceed in this way. We also had problems with the use of IPv6 which is not yet properly supported by some components of BOINC and Condor. For example, it is impossible to use IPv6 addresses for the configuration of the BOINC client, although the `Curl` module used by BOINC was compiled with support for IPv6. On the other hand, the configuration of Condor with IPv6 also causes problems. The difficulty comes from the fact that we do not currently have the hostname for DNS resolution of our IPv6 address. This makes particularly difficult to configure Condor with SlapOS.

The compilation of BOINC and Condor component was also complicated, although using Linux, we can not use simple commands such as `apt-get install` or `zypper install`. The principle of isolation of components in SlapOS sometimes complicates their compilation. Despite the difficulties, we succeeded in the integration of BOINC and Condor but at the price of tight efforts and for a limited number of features. Thus, integrating existing middleware into a cloud in order to propose the concept of DG as a Service has proved to be a very difficult task. Hence, our choice to rethink/rebuild a new DG middleware that has more chances to be coupled with the SlapOS cloud. The reminder of this paper is devoted to the new DG middleware.

4 Contributions

4.1 The Interaction Algorithm

We introduce now a realistic interaction algorithm fully specified in terms of the Publish-Subscribe paradigm which far exceeds the one introduced in [7]. The obtained middleware has similar features compared to what we find in major DG middlewares. It manages scheduling strategies especially the detection of dependencies between tasks, the execution of tasks and the verification of results; since the results returned by workers can be manipulated or altered by malicious workers. The general objectives become:

- to use an asynchronous paradigm (Publish-Subscribe) that ensures as much as possible a total decoupling of the steps of the scheduling algorithm (for performance matters);
- to assert that the system is resilient in case of tasks duplication and actors duplication; We assume that since the system is asynchronous and tasks are duplicated, it should continue to progress. We also assume that actors are duplicated for resilient matters;

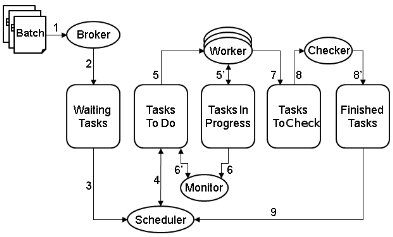


Fig. 1. Interactions between components

In Figure 1, we introduce the sequence of the different steps that a task will follow in our system. Tasks may have 5 status namely WaitingTasks, TasksToDo, TasksInProgress, TasksToCheck and FinishedTasks, and they are managed by 5 actors: a broker, a scheduler, a worker, a monitor and a checker. Taken separately, the behavior of each component may appear simple but we are interested by the interaction between components and this makes the problem hard to solve. One key idea is to allow the pluging of dedicated components (scheduler, checker into a general coordination mechanism in order to avoid to build a monolitic system. The behavior of the system as depicted in Figure 1 is as follows:

1. Submission of batches. Each batch represents a series-parallel graph of tickets. A ticket is simply a task to be executed.
2. The Broker extracts the tickets and publishes them on a channel named WaitingTasks
3. The scheduler listens to the channel WaitingTasks
4. The Scheduler starts by publishing independent tasks on a channel called TasksToDo
5. The Workers, already listening to TasksToDo channel, begin the execution of published tasks. Tasks are published on TasksInProgress channel 5).

6. During the execution, each task is under the supervision of the Monitor, which role is to ensure the smooth execution by checking if the node is alive. In the opposite case, it republishes the failed task into the TasksToDo channel 6).
7. Once the tasks executed, the worker publishes them on the channel TasksTo Check
8. The checker verifies the results returned, and publishes the corresponding tasks into the channel FinishedTasks 8).
9. The scheduler checks for dependencies between the tasks completed and those in waiting, if so repeat 4).

4.2 User Point of View, Implementation Details and Emulation

We present in this section the prototype we developed recently and which is more mature than the one introduced in [7]. The user submits an XML file describing his application which is represented by a dependency graph. Each node of the graph represents a task and each edge between two tasks represents a dependency between tasks. A node/task is described in terms of inputs, outputs, code to executeWe provide a template for the XML file, demonstrating how the application should be described. Thus, in our code, all the information related to the application to run are extracted directly from the XML file and automatically exposed to the internal graph structure. We use a python library named `parse from xml.dom.minidom` for that purpose. Currently, we construct automatically the XML file using `xml.dom.minidom` python library in the purpose to have in the long term a graphical interface in which the user can enter relevant data to the execution of its application. Behind, these data will be collected, and XML file is built to be parsed.

We have developed 5 classes representing our 5 actors: the broker, the scheduler, the worker, the monitor and the checker. These classes inherit from MachineClass which allows to set the properties of a machine (operating system type, amount of memory on the machine, processor type. The states that tasks may have are represented as channels in which we can publish events and subscribe to events. For the emulation, we use a MapReduce job that counts the number of occurrences of each word in a given input text. We use the emulation term and not simulation because we execute our code on a real machine but not through a simulator.

This application is represented by a graph of 8 nodes and 10 edges. It splits the input data-set into 4 independent chunks which are processed by the Map operation. The Reduce operation takes chunks by pairs to sum the different occurrences of words. Our program forks and starts a Broker, a Scheduler, a Monitor and a Checker as well as multiple instances of Workers.

The BrokerClass takes the graph as input, then it extracts the nodes. It associates to each node a list of predecessors. Then it publishes all the nodes and the predecessors in the channel WaitingTasks to which the Scheduler is listening.

The SchedulerClass is managed by two threads. The first one is used to listen to the channel WaitingTasksand to publish independent tasks in the channel TasksToDo The second one deals with finished tasks. In fact the Secheduler is listening to the channel FinishedTasks and at the reception of a new task from this channel it updates the dictionnary of waiting tasks on the purpose to detect new independant tasks which will be managed by the first thread. The two threads operate concurrently.

In parallel, Workers are launched. A worker is also managed by two threads. The first one is listening on TasksToDo channel in order to be notified by new tasks to execute. At the reception of this event, the worker publishes its identity to say that it is volunteer to start the execution of the task. The scheduler selects the first worker replying to the proposal. This functionality mimics the protocol for workers selection found in any DGs system. The second thread verifies if the worker is selected to execute the published task, if so, it launches the execution of the respective code which is downloaded from the Redis code repository. The Redis server is located somewhere in the Web but does not run on the same machine used for the emulation. Once the task execution has begun, the event is published by the worker in TasksInProgress channel. The task, the process number and the IP address of the worker are published in TasksInProgress channel in order to allow the monitoring. When the execution is finished, the worker publishes the corresponding task in TasksToCheck in order to run the result certification module for that task.

For the MonitorClass, we create one process for requests for monitoring and one process for stopping the monitoring. The first one is listening in TasksInProgress channel. When an event is published in that channel, the Monitor extracts the IP address of the worker and checks if the node is alive by pinging it every 2 seconds during all the time of the task execution. In the case we detect a worker which does not respond to a ping request, the Monitor publishes the corresponding task in the TasksToDo channel. The second process decides when stopping the monitoring. It is listening to the tasksToCheck channel in order to be notified by tasks that are already finished, and it kills the corresponding process .

The CheckerClass role is to certify the results. Our tool offers the possibility to duplicate the task k>1 times in order to execute the same task on different workers and compare returned results. In the emulation, the checker constructs a dictionary of similar tasks. We use `hashlib` library to compare MD5 of the output files returned by each worker. If the result of checking is not correct then the corresponding task is published again in the TasksToDo channel, else it is published on FinishedTasks channel. As said before, the interaction schema is not a trivial task to specify and to implement. The difficulty lies in the inter-relation between the components that notify their activities.

The code for the emulation is available online[1].

4.3 Analysis of Our Design, Feedback

The current graph engine is able to schedule series-parallel graphs. When running an application, the first step is duplicating nodes and edges in order to duplicate tasks. We recently received a demand for the ability to execute graph that may increase recursively. We do not know if such graphs have a precise definition. The application that requires such graphs is for instance the K-means clustering algorithm used in machine learning, which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. A parallel algorithm may consist in re-injecting an estimate about the clusters into the initial parallel steps of

[1] `http://www-lipn.univ-paris13.fr/~abidi/RedisDG.zip`

the algorithm to better estimate the cluster. We are currently working on the modeling of such underlying graphs in order to make the graph engine more general.

The Monitor class implementation consists in pinging, from a single site, all the machines/workers attached to an application. This is a bottleneck when considering the scalability criteria. We are currently working on the integration of Taktuk [8]. TakTuk is a tool for deploying parallel remote executions of commands to a potentially large set of remote nodes. The commands may be as simple as 'are you alive' which is what we want in the case of monitoring. Taktuk builds its own overlay network (a tree) and may broadcast and collect to/from nodes in an efficient way. Taktuk is used for instance to manage a cluster of 17000 machines. Our current implementation serves as a canvas for the future implementation. The management of the events for workers monitoring is in place, but we need to revisit the implementation of Taktuk to insert and remove nodes dynamically , i.e. when workers are selected. The current Taktuk system deals with static information: you need to know the list of IPs before starting Taktuk.

The current implementation is an emulation in the sense that all the components run on the same machine, except the Redis server. It is a first step to demonstrate that our prototype is functional. Thus we need to decompose the application into separate entities that share the description of the input graph. In the same vein, we are also working on the integration of our tool in the SlapOS cloud as explained above for BOINC and Condor and concurrently with the core code.

5 Conclusion

In this paper, we introduced the context of our work around the coordination of resources using the publish-subscribe paradigm. It is a step towards the development of DG middleware based on Web 2.0 technologies. We would like to clarify that our work focuses rather on managing the interactions fully based on eventsbetween components, and not on task scheduling algorithm. Thus, we can use in our tool any scheduling algorithm such as FCFS (First Come First Served) for instance. We introduce an interaction policy and we implemented it on top of Python and Redis modules. We have implemented the interaction algorithm based on tickets that we duplicate and manage through the Publish-Subscribe paradigm. Again, the controlling of the protocol is made exclusively through a Publish-Subscribe approach which is unconventional. We also analyzed our design and explained what are the limits, for instance in terms of scalability of the current implementation of the monitoring subsystem. To get confidence into our system, it remains to model our interaction framework in terms of colored Petri nets as done in [7, 9] for the BonjourGrid meta DG middleware[10]. In fact, we plan to take advantage of our formal modeling done at this occasion to verify formally our interaction framework.

The overall objective of the work is to offer DG services on demand, on any devices, indiscriminately, i.e. on smartphones, tablets and desktop PCs. Moreover the service should be deployed by a non expert ?n one clickand the management of the system should not be restricted to system administrators but widely open. The goal is to make this technology accessible to the greatest number of people in the e-Science community through automating the deployment.

References

1. Kondo, D.: Preface to the special issue on volunteer computing and desktop grids. J. Grid Comput. 7, 417–418 (2009)
2. Cerin, C., Fedak, G.: Desktop Grid Computing, 1st edn. Chapman and Hall-CRC (2012)
3. Smets-Solanes, J.P., Cerin, C., Courteaud, R.: Slapos: A multi-purpose distributed cloud operating system based on an erp billing model. In: [11], pp. 765–766
4. Eugster, P.T., Felber, P., Guerraoui, R., Kermarrec, A.M.: The many faces of publish/subscribe. ACM Comput. Surv. 35, 114–131 (2003)
5. Abbes, H., Dubacq, J.C.: Analysis of peer-to-peer protocols performance for establishing a decentralized desktop grid middleware. In: César, E., Alexander, M., Streit, A., Träff, J.L., Cérin, C., Knüpfer, A., Kranzlmüller, D., Jha, S. (eds.) Euro-Par 2008 Workshops. LNCS, vol. 5415, pp. 235–246. Springer, Heidelberg (2008)
6. Cerin, C., Takoudjou, A., Greneche, N.: Integration des intergiciels de grilles de pc dans le nuage slapos: le cas de boinc. CoRR abs/1211.6473 (2012)
7. Abidi, L., Cérin, C., Klai, K.: Design, verification and prototyping the next generation of desktop grid middleware. In: Li, R., Cao, J., Bourgeois, J. (eds.) GPC 2012. LNCS, vol. 7296, pp. 74–88. Springer, Heidelberg (2012)
8. Claudel, B., Huard, G., Richard, O.: Taktuk, adaptive deployment of remote executions. In: Kranzlmller, D., Bode, A., Hegering, H.G., Casanova, H., Gerndt, M. (eds.) HPDC, pp. 91–100. ACM (2009)
9. Abidi, L., Cerin, C., Evangelista, S.: A petri-net model for the publish-subscribe paradigm and its application for the verification of the bonjourgrid middleware. In: [11], pp. 496–503
10. Abbes, H., Cerin, C., Jemni, M.: Bonjourgrid as a decentralised job scheduler. In: APSCC, pp. 89–94. IEEE (2008)
11. Jacobsen, H.A., Wang, Y., Hung, P. (eds.): IEEE International Conference on Services Computing, SCC 2011, July 4-9. IEEE, Washington, DC (2011)

A Novel Model for Greenhouse Control Architecture

Miran Baek, Myeongbae Lee, Honggean Kim, Taehyung Kim, Namjin Bae,
Yongyun Cho, Jangwoo Park, and Changsun Shin*

Department of Information and Communication Engineering, Sunchon National University
{tm904,lmb,khg_david,taehyung,bakkepo,yycho,jwpark,
csshin}@sunchon.ac.kr

Abstract. This paper proposed the Greenhouse Control System (GCS) for high adaptability in greenhouse control devices and application services. The system is divided into the Greenhouse Control Engine (GCE) and the Crop Growth Engine (CGE). The GCE consists of Data Aggregator (DA), Greenhouse Information Storage (GIS), and Greenhouse Control Agent (GCA). The GCA includes Information Analyzer (IA), Control Device Selector (CDS), and Greenhouse Model (GM). The GCA selects control devices by referencing the aggregated greenhouse's information and the climate set-points. In this process, we apply the arbitrary greenhouse model to the GCA. And the CGE consists of Crop Status Information Storage (CSIS) and Crop Growth Agent (CGA). The CGA decides the climate set-points by applying the arbitrary crop growth model. The CGA has Crop Condition Predictor (CCP), Environment Set-points Decisioner (ESD), and Crop Growth Model (CGM). By interacting of each component, this system provides with the greenhouse control service and the crop growth prediction service. The greenhouse control service monitors the inside and outside climate of a greenhouse and controls the control devices of a greenhouse on the GCA. The crop growth prediction service predicts the crop growth status by considering the meteorological data and business data. Finally we showed the executing result by implementing the GCS.

Keywords: Greenhouse Control, Crop Growth, Greenhouse Monitoring, Greenhouse System Architecture, Greenhouse Service.

1 Introduction

Recently, the agricultural production environments is getting worse sharply result from agricultural products import opening (FTA: Free Trade Agreement), a decline in agricultural population, population ageing in farm village. However, ICT technology has applied in agricultural sector. As a result, it will increase value added and productivity in labour-intensive agriculture [1] [2].

Cultivation under structure (greenhouse) has capital with technology intensive and it is helped to foster our agriculture since south korea-united states FTA in the 21st century. These greenhouse is improved productivity to control the growing conditions (temperature, humidity, quantity of solar radiation, carbon dioxide concentration

* Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 262–269, 2013.

ration, and so on) in artificial. Therefore, the goal of the greenhouse is to harvest the crops whenever we want through the auto controlled production system [3-5].

We can examples of auto control system of the greenhouse environment. If the greenhouse environment is controlled to crop growth condition, it will increase the productivity. Also, it will decrease the cost of production. However, our greenhouse system is a simple automatic control system that control to set up information such as temperature, humidity, CO2 and so on through monitoring to greenhouse environment [6]. So, we want to the automatic control greenhouse system on crop growth rate at predict to crop growth environment.

We proposed greenhouse control system that supported greenhouse environment control service and crop growth state predict service using greenhouse control engine and crop growth predict engine.

This paper is organized as follows. We present our proposed technique in chapter 2 with structure and component of greenhouse control system, support available service. Chapter 3, the proposed system shows implementation and execution results. Finally Chapter 4 presents our conclusion and described for future work.

2 Greenhouse Control System(GCS)

2.1 Greenhouse Control System Architecture

In this paper, the proposed Greenhouse Control System (GCS) is consists of physical layer, middle layer and application layer. The physical layer exist sensors to measure the environmental and control devices and sensors to measure the status of the crop growth. The middle layer is divided into the Greenhouse Control Engine (GCE) and

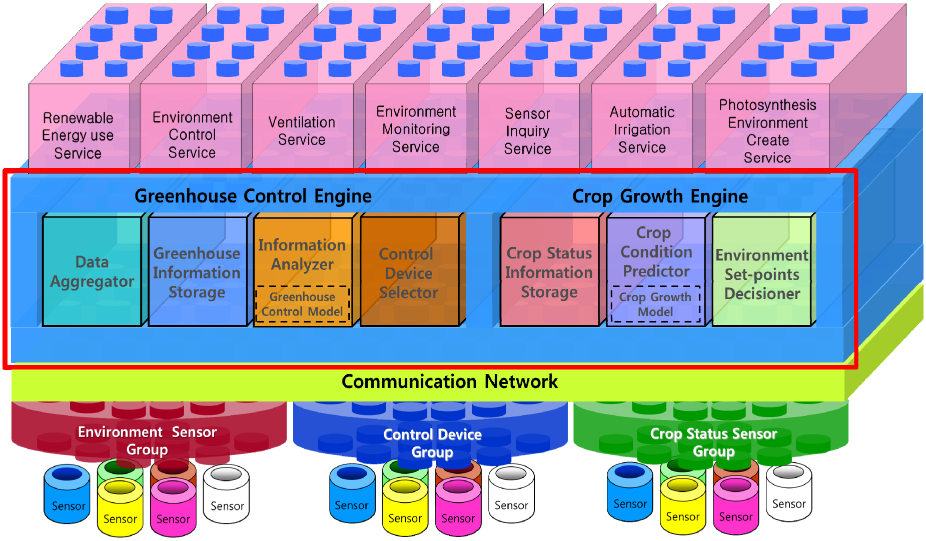


Fig. 1. Greenhouse Control Framework

the Crop Growth Engine (CGE). The GCE consists of Data Aggregator (DA), Greenhouse Information Storage (GIS), and Greenhouse Control Agent (GCA). The GCA includes Information Analyzer (IA), Control Device Selector (CDS), and Greenhouse Control Model (GCM). Also, the CGE consists of Crop Status Information Storage (CSIS) and Crop Growth Agent (CGA). The CGA has Crop Condition Predictor (CCP), Environment Set-points Decisioner (ESD), and Crop Growth Model (CGM). The application layer consists of greenhouse environment monitoring and greenhouse control services through sensors or control devices. In this paper, the middle layer design and implementation for GCS. Fig. 1 shows the architecture of the GCS.

2.2 Greenhouse Control System

The GCS is analyzed the collected greenhouse's information (internal environment and external environment and soil data) through environmental sensors based on the GCM. Selects control devices to control the greenhouse's environment by referencing the analyzed data and climate set-points. The value of the control device is delivered to the greenhouse will control the greenhouse's environment. And the controlled environmental factors affect the growth of crops. Also, it will collect status information of the crops through sensors that measure the growth status of the crop. The collected information predicts crop condition based on CGM. At this time, should be predicted by considering the meteorological data and business data. Based on predicted information are decided climate set-points of greenhouse and the system for optimal control of greenhouse set-points to fit. Table 1. and Table 2. display the factors of each process and Fig. 2 shows the diagram of the GCS.

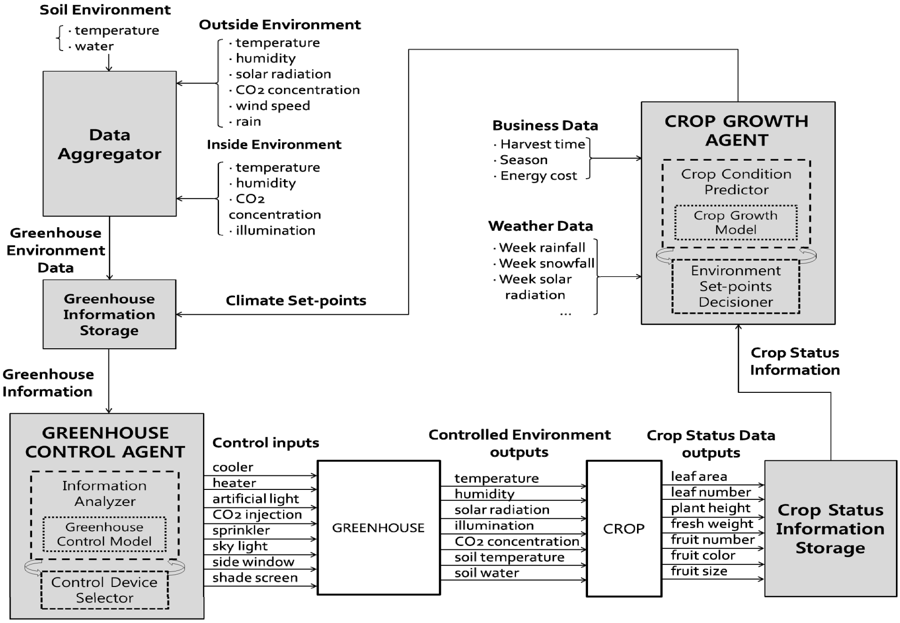


Fig. 2. Greenhouse Control System Diagram

Table 1. Factors of Greenhouse Control Engine

Category	Measurement Factors
Inside Environment	Temperature, Humidity, Solar radiation, Illumination
Outside Environment	Temperature, Humidity, Solar radiation, CO2 concentration, Wind speed, Rain
Soil Environment	Temperature, Water
Control Device	Cooler, Heater, Artificial light, CO2 injection, Sprinkler, Sky light, Shade screen

Table 2. Factors of Crop Growth Engine

Category	Measurement Factors
Growth Environment	Temperature, Humidity, Solar radiation, Illumination, CO2 concentration
Growth Status	Leaf area, Leaf number, Plant height, Fresh weight, Fruit number, Fruit color, Fruit size
Business Data	Harvest time, Season, Energy cost
Weather Data	Week rainfall, snowfall, solar radiation

2.3 Component of Greenhouse Control System

For function of each component of the GCE and the CGE is defined.

2.3.1 Component of Greenhouse Control Engine

The DA is to collect inside, outside and soil environment information through the inside, outside meteorological sensors and soil sensors. Are converted into data that can be used and the function is filtering of odd data. Also, the function is delivered to the collected data to the GIS. The GIS is stored processed data from the DA and filtered data, and stored set-up the climate set-points in the ESD. The GCA selects control devices by referencing the collected greenhouse's information and the climate set-points. The function of component included in the GCA is as follows. The IA is analyzed based on the GCM that call the odd data stored in the GIS. The IA is provides environment factors to be controlled in the CDS by referencing the analyzed data and climate set-points provided from the ESD. The CDS selects the appropriate control device for optimal control of greenhouse environment based on the data that is passed from the IA.

2.3.2 Component of Crop Growth Prediction Engine

The CSIS is stored growth status information of the collected crop through the sensors to measure the status of the crop growth. The CGA decides the climate set-points by applying the arbitrary crop growth model. The function of component included in the

CGA is as follows. The CCP is function to predict the future of crop condition based on CGM by considering the meteorological data and business data on the growth status of crop. The ESD plays a determinant role to regulate the greenhouse climate set-points based on extracted prediction data from the CCP.

2.4 Greenhouse Control System Service

The service of the GCS is divided into greenhouse control service and crop growth prediction service.

2.4.1 Greenhouse Control Service

The greenhouse control service is service that monitors the inside and outside climate of a greenhouse and to control the control devices of a greenhouse on the GCA. The driven process of greenhouse control service is as follows. The collected data through the sensor on the DA is stored in the GIS. Since the analyzed data based on the GCM at the IA by requested the necessary data on the GIS. The analyzed data is delivered to the CDS selects the device to be controlled of the greenhouse environment based on the analysis of data. The motion process of greenhouse control services is shown in Fig. 3.

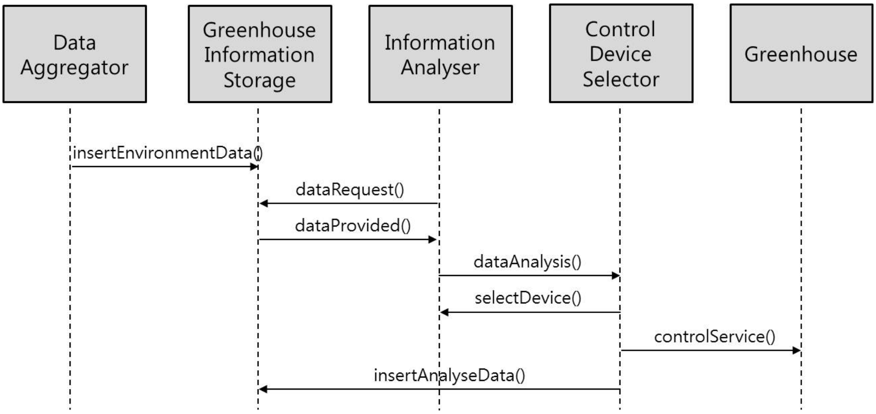


Fig. 3. Motion process of Greenhouse Control Service

2.4.2 Crop Growth Prediction Service

Crop growth prediction service is a service predicts the crop growth status by considering the meteorological data and business data. The driven process of crop growth prediction service is as follows. The current crop of status data is stored in the CSIS. After that, the predict future of the crop growth status based on the CGM at the CCP by requested the crop status information. Based on the predicted information is decided climate set-points at the ESD. Determined set-points are stored in the GIS. The motion process of crop growth prediction service is shown in Fig. 4.

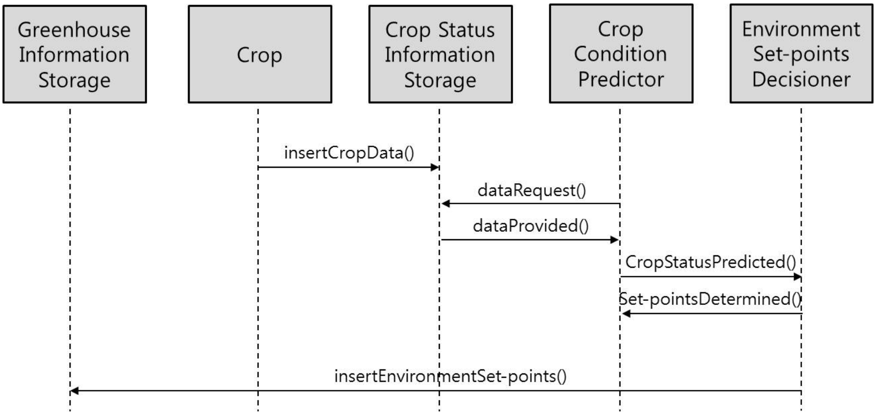

Greenhouse Information Storage
Crop
Crop Status Information Storage
Crop Condition Predictor
Environment Set-points Decisioner
insertCropData()
dataRequest()
dataProvided()
CropStatusPredicted()
Set-pointsDetermined()
insertEnvironmentSet-points()

Fig. 4. Motion process of Crop Growth Prediction Service

3 Results

3.1 Implementation Environment

This paper proposed to confirm performance of the GCS, we virtual greenhouse model was constructed as shown in Fig. 5. To collect data install temperature/humidity sensor, illuminance sensor, solar radiation sensor and to control the greenhouse environment control devices is make a model.

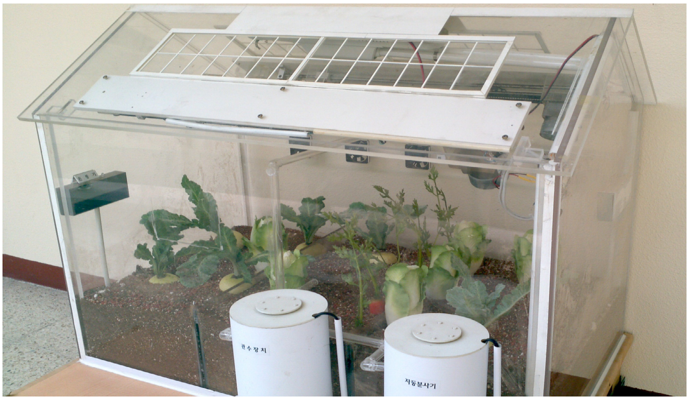


Fig. 5. Greenhouse model for tests

3.2 Execution Results

By implement a GUI for users, confirm execution results of system. Fig. 6 shows the smart terminal (iOS) GUI developed for this study. Through the installed sensors, the collected environment information and status of the control device are shown in real-time. The user to monitor the status of greenhouse and the user can see that control the greenhouse environment by working control devices.

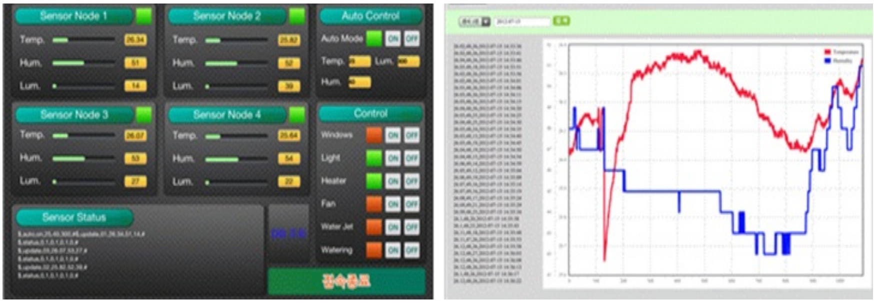


Fig. 6. iOS application GUI

4 Conclusion

In this paper, the GCS has been designed for optimal control of greenhouse environment by interacting greenhouse control and crop growth engine. The system is divided into the GCE and CGE. The GCE consists of DA, GIS and GCA. The GCA includes IA, CDS and GCM. The GCA selects control devices by referencing the aggregated greenhouse's information and the climate set-points. And the CGE consists of CSIS and CGA. The CGA has CCP, ESD and CGM. The CGA decides the climate set-points by applying the CGM. The GCS provides greenhouse control service for optimal control of the greenhouse environment using components of the GCE. Also, to provides crop growth status prediction service for predict the future of the crop growth status using components of the CGE. Greenhouse model constructed in order to verify performance of we proposed system. Through the developed of the GUI is accurately monitoring status of greenhouse and execution results showed that is controlled by a control device.

Future works we create a complete system is analysis of the requirement of the components and it is necessary a detailed to define the function. Also, it will need to apply of various algorithms.

Acknowledgements. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0014742).

This work was supported by the Industrial Strategic technology development program, 10040125, Development of the Integrated Environment Control S/W Platform for Constructing an Urbanized Vertical Farm Funded by the Ministry of Knowledge Economy (MKE, Korea).

References

1. Lee, M., Chin, C., Cho, Y., Yoe, H.: Greenhouse Environment Inte-grated Management System in Ubiquitous Agricultural. Journal of Information Science 27(6), 195–197 (2009)
2. Kim, D., Cho, H.: Control of environments in Greenhouse Using Programmable Logic Controller. Korean Society for Agricultural Machinery 3(1), 174–179 (1998)

3. Korea Society for Horticultural Science, http://horticulture.or.kr
4. Woo, Y.: Fusion of Mushroom and Protected Cultivation Environment Management Technology. Korean Society of Mushroom Science
5. Informatization village, http://greens.invil.org
6. Yang, J., Jeong, C., Hong, Y., An, B., Hwang, S., Choi, Y.: Implementation of Greenhouse Environmental Control Systems using Intelligence. Electronics Engineers of Korea 49(2), 29–37 (2012)
7. Rodriguez, F., Berenguel, M., Arahal, M.R.: A Hierarchical Control System for Maximizing Profit in Greenhouse Crop Production
8. Kwon, H., Kim, H., Kim, J.: Forecasting System Design for Tomato growth. Korea Information Processing Society 18(1) (2011)
9. Lee, Y., Kim, S., Son, K., Lee, I., Chin, C.: Implementation of Failure-Diagnostic Context-awareness Middleware for Support Highly Reliable USN Application Service. Korean Society for Internet Information 12(3) (2011)
10. Jeong, K.: Installation and Management of USN based Crop Growth Environment Management System. National IT Industry Promotion Agency, pp. 1–99 (2010)
11. Kim, J., Im, J.: A Design of Intelligent Plant Factory Control Structure based on Ontology for Growth Environment. Korean Society for Internet Information 11(2), 107–108 (2010)
12. Lee, Y., Seo, B., Kim, C., Kim, K., Park, Y., Chin, C.: Implementation of Facility Management System for Plant Factory. Korea Society of Computer Information 16(2), 141–151 (2011)
13. Standard for Greenhouse Control System-Part 3: Interface for Between Greenhouse Control Gateway and Greenhouse Operating System. Korea Association of RFID/USN Convergence, pp. 1–48 (2011)
14. Cunha, J.B., de Moura Oliveira, J.P.: Optimal Management of Greenhouse Environments. In: EFITA 2003 Conference, pp. 559–564 (2003)
15. Chin, C., Seo, J.: A Development of Proactive Application Service Engine Based on the Distributed Object Group Framework. Korean Society for Internet Information 11(1), 153–165 (2010)

Enhanced Search in Unstructured Peer-to-Peer Overlay Networks

Chittaranjan Hota[1], Vikram Nunia[1], Mario Di Francesco[2,3],
Jukka K. Nurminen[2], and Antti Ylä-Jääski[2]

[1] Dept. of Computer Science, BITS Pilani Hyderabad Campus, India
{vikram,hota}@bits-hyderabad.ac.in

[2] Dept. of Computer Science and Engineering, Aalto University, Finland
{mario.di.francesco,jukka.k.nurminen,antti.yla-jaaski}@aalto.fi

[3] Dept. of Computer Science and Engineering, University of Texas at Arlington, USA
mariodf@uta.edu

Abstract. Unstructured Peer-to-Peer (P2P) overlays are the most widely used topologies in P2P systems because of their simplicity and very limited control overhead. A P2P overlay specifies the logical connections among peers in a network. Such logical links define the order in which peers are queried in search for a specific resource. The most popular query routing algorithms are based on flooding, thus they do not scale well as each query generates a large amount of traffic. In this paper, we use heuristics to improve overlay search in an unstructured P2P file sharing system. The proposed heuristics effectively decide replica locations for popular resources based on the availability of computing and storage at a given peer, its neighborhood information, and the used routing strategy. Simulations performed over two different types of unstructured P2P network topologies (i.e., power law and random graphs) show significant improvements over plain flooding in terms of reduced network traffic and search time.

Keywords: Peer-to-peer, search, replication, unstructured, overlay networks.

1 Introduction

In a Peer-to-Peer (P2P) network participating nodes are both providers and users of services. The usage of P2P applications has grown steadily since their initial development, and recent empirical studies indicate that P2P and web together dominate today's Internet traffic. As reported in [1], P2P traffic accounted for almost 60% of Internet traffic worldwide in 2009. Motivated by the extent of their usage, researchers have focused on studying and improving the scalability and performance of P2P networks.

In order to facilitate direct data exchange and service execution between different peers, a logical overlay is usually imposed over the underlying physical network. There are two classes of P2P overlay networks: structured, and unstructured [2]. An unstructured P2P system consists of peers joining the network

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 270–279, 2013.

with some loose rules, without any prior knowledge of the topology. Unstructured P2P networks offer decentralization and simplicity, but may require $O(N)$ hops to search a file when the network is made of N nodes. In contrast, structured P2P overlay networks tightly control both the network topology and the placement of content. Specifically, the content is stored at specified locations based on distributed hash tables (DHTs), so as to improve the efficiency of the queries. Structured P2P overlays using DHTs are valuable for large scale distributed applications because of their search efficiency which is $O(\log N)$ for a network of N nodes. However, structured P2P overlays are not very suitable for file searching applications exploiting multiple attributes and involving a large number of peers with a high level of churn.

In this paper, we describe a novel replication heuristic which exploits a proportional replication policy to reduce search load in an unstructured P2P overlay. The replication heuristic considers resource popularity and also provides a query routing strategy. In our solution, replication is achieved by explicitly pushing resources to other peers. The replication heuristic is also enhanced with an intelligent neighbor selection heuristic. By using a power-law function, a peer selects n neighboring peers. Out of those neighbors, a peer further picks m most preferred peers by using chi-square similarity measure. The search algorithm used in this paper takes a hybrid approach that is a tradeoff between flooding, which alone is not inefficient and does not scale, and a random walk, which could take long time to find a resource. In our approach, we used k-random walks for resources with a low number of replicas, and a single random walk for resources with a higher number of replicas. Simulation results show that the search load is fairly distributed among the peers and that the cost to locate a resource in the network is very low.

The rest of this paper is organized as follows. Section 2 reviews the search and replication techniques commonly used in unstructured P2P networks. Section 3 details the overlay topology construction, while Sect. 4 presents our replication and neighbor selection heuristics, along with the k-walker search algorithm. Section 4 presents a performance evaluation of our proposed heuristics. Finally, Sect. 6 concludes the work.

2 Related Work

In order to reduce unnecessary flooding – which is, however, a widely used search technique in unstructured P2P – three major approaches have been proposed in the literature. In the first category, each peer uses heuristics to intelligently decide the peer which could likely provide the resource [3, 4]. In this case, the performance of the heuristics determines the search load. In our approach, we extend the search with a replication heuristic that effectively makes more copies of the most popular resources for reducing the search load. In the second category, a peer caches the resource IDs of other peers as a third-party query is routed through them, and uses these IDs to reduce the search load in subsequent requests [5–7]. The major drawback of indexing is represented by the additional storage requirement at the peers. In our approach, we try to reduce the storage demands at peers by judiciously deciding the number of replicas in a dynamic environment. The third category is based on overlay topology optimization, which has been attempted by many researchers through techniques that include end-system multicast [8] and clustering [9]. Even though clustering may scale, it does not guarantee the search scope. Our approach uses a similarity measure to decide upon similar peers in terms of their resource preferences. As a consequence, it is more likely that search queries will get a better response from well-connected (similar) peers.

Most P2P systems only construct peer connections according to the network constraints, and do not take user preferences into consideration. For instance, in [10, 11] social overlays were used to find out similar peers and build clusters accordingly in order to improve search. In these approaches, a peer selects another peer based on their similar interest in searching files. This social linking reduces search load on other peers as the file requests have a high probability of being fulfilled by the neighboring peer. In our approach, we used the chi-square statistics used in [10] to compute a similarity measure between two peers. In this work, we extend our earlier work in [12] by using a power-law distribution to compute the node degree and then select the peers which are similar.

3 Overlay Topology Construction

First, we create a P2P overlay topology, wherein each peer has a certain number of neighbors. We use two types of networks in our simulation:

1. **Power-Law Graph (PLG):** The node degrees follow a power-law distribution: when ranked from the most connected to the least connected, the i-th most connected node has C/i^{β} neighbors, where C is a constant, and β is a scaling factor such that $0 < \beta <1$. In the following, we will set $\beta = 0.7$. Once the node degree n is chosen, nodes are connected with the m “best” neighbors as described later.
2. **Random Graph (RG):** The node degrees are calculated randomly, and nodes are connected with m “best” nodes out of n nodes.

Many real-life P2P overlay networks are random graphs, i.e., neighbors are selected randomly. A plausible alternative to random overlay networks is to build a network based on a measure of similarity between the user’s resources [10]. Solutions available in the literature have already exploited social relations to find out the similarity of peers, such as in [10, 13]. We use an approach such as the one in [10] for computing the similarity measure between two peers. However, in contrast with that work, we use file types instead of style of files. The key observation is that, although the styles of files downloaded by two peers may be the same, the content within the file may be different, hence the file type provides a better similarity measure.

Each user is identified by a vector denoting the probability of sharing a file of each type. To this end, we first determine the background probability of a file being of a specific type, based on the type distributions in the entire network. The background probability is obtained as $\mathrm{P}_b(F_j) = \frac{F_j}{\tau_n}$, where F_j is the number of resources type j and τ_n total number of files shared in network. We calculate a similar probability for a user sharing the same type of file, namely, $\mathrm{P}_u(F_j) = \frac{F_j}{\tau_u}$, where τ_u is total number of files shared by user u. Finally, we also calculate the sharing probability for a user as $\mathrm{P}_u(S) = \frac{\tau_{du}}{\tau_{dn}}$, where τ_{dn} is the total data shared in network and τ_{du} is total data shared by user u. Given the number of downloads d_u and the downloaded amount of data D_u of a user, we can then calculate the number of expected files types downloaded by a user for the background probability, the similarity probability and the shared probability, as shown below:

$$\begin{aligned} E_b(F_j) &= \mathrm{P}_b(F_j) \cdot d_u \\ E_u(F_j) &= \mathrm{P}_u(F_j) \cdot d_u \\ E_b(S) &= \mathrm{P}_u(S) \cdot D_u \end{aligned}$$

By using the expected values computed above, we can then calculate two chi-square statistics to determine how the downloads of a users are are similar to the background type distribution and to their own shared distribution. In detail, it is

$$X_z^2 = \sum_{F_j} \frac{\left(d_{F_j} - E_z(F_j)\right)^2}{E_z(F_j)}, \quad z \in \{u, b\} \tag{1}$$

where d_{Fj} is total number of file downloads of type j. By using the difference between the two statistics, we can determine if a user is more like the network or more like the library of shared files [10].

If the user is more like the network, then the user will be connected with the m neighbors who have shared a large number of files and and a large amount of data by using the probabilities $E_b(F_j)$ and $E_u(S)$. Otherwise, user will be connected with m most similar nodes. We define the expected number of files that a sharer provides to a downloader as:

$$E(u, d) = \sum_{S_{ui}} \mathrm{P}_d(F_j) \cdot |F_u(f_j)| \tag{2}$$

where S_{uj} is the number of files of type j shared by user u, $P_d(F_i)$ is the probability of file type i being downloaded by a peer d, and $F_u(f_i)$ is the set of files shared by user u of a type i not already owned by d. For each downloader we can rank every other user based on the expected number of new files they might provide. Using this ranked list, we can select the m best neighbors for a user.

4 Algorithms for Replication and Search

In this section we will describe the heuristics behind the search and replication algorithms.

Algorithm 1. k-walker file search heuristics

```
Search file $f$ in shared folder;
if $f$ found then        // Call replication algorithm, save nodes and exit
    f_nodes ← File_Replicate(f) $FR \leftarrow FR \cup \{self\} \cup \{f\_nodes\}$ ;
    return $FR$ to source;
if source is self then        // Add file to request list and start walkers
    $RL \leftarrow RL \cup \{f\}$; calculate $k$ and create $k$-walkers;
    foreach walker $w$ in $k$ do        // Forward query to neighbor $N$
        randomly select neighbor $N$; N.F_Search($f$,source,1);
else        // Check whether the file was found or not
    if check==CHECK then
        if source.check_file(f)==True then exit;
        check ←0;
    check ←check + 1; randomly select neighbor $N$;
    N.F_Search($f$,source,check);        // Forward query to neighbor N
```

4.1 Search Heuristic

To avoid the message overhead of flooding, unstructured P2P networks use different types of random walks. In a random walk, a single query message is sent to a randomly selected neighbor. We call this message *walker*. A walker has a TTL value that is decremented at each hop. If the query finds the desired resource at some node, the search terminates successfully. If the query fails, as determined by timeout or a failure message from the node last receiving the query, the initiating peer chooses another random path. The standard random walk – which uses only one walker – can cut down the message overhead by one order of magnitude compared to flooding [14]. However, there is also an order of magnitude increase in the delay perceived by the user. To reduce the delay, we increase the number of walkers as in [14, 15]. That is, instead of just sending out one query message, a requesting node sends k query messages in parallel. More walkers find resources faster, but also generate more traffic when the number of replicas in the network is low. Furthermore, when the number of walkers is enough high, increasing it further slightly reduces the number of hops, but significantly increases the traffic. For every search request, the value of k depends on the replication probability calculated at the requesting node as follows. Let ρ_{F_u} be number of requests for a particular file type by user u, and ρ_u is total number of requests made by user u. Then, the replication probability is:

$$\mathrm{P}(R_{F_u}) = \frac{\rho_{F_u}}{\rho_u} \tag{3}$$

The key idea behind the choice of k is that its value should be lower when the replication probability is higher. In other words, $k = 1$ when a resource has the highest number of replicas, thus implying that only one walker is good enough to locate that resource. On the other hand, k will be maximum if no replica exists. Specifically, the value of k is expressed as:

$$k = \lceil K(1 - \mathrm{P}(R_{F_u})) \rceil \tag{4}$$

where K is a constant that defines the maximum number of walkers to be used in the search. As multiple random walks require some mechanism to terminate, each walker periodically checks with the original requester before walking to the next node. This method still uses a TTL, but the TTL is very large and is mainly used to prevent loops. Since there are a fixed number of walkers (1 to K), the walkers checking back with the requester will not lead to message implosion at the requester node. Of course, checking does have overhead; each check requires a message exchange between a node and the requester node. Indeed, simulation experiments in the next section show that checking once at every Max_hop_check step along the way achieves a good balance between the message overhead and the benefits of checking. The k-walker search heuristics is illustrated by Algorithm 1.

4.2 Replication Heuristics

File replication involves storing replicas of files in nodes other than the one sharing them. Replication improves the query success rate and reduces latency by making the shared files more likely to be available in the path of a search walk. In the following, we propose a proportional replication strategy coupled to the k-random walk search described in the previous section.

Since there is no well-known correlation between file popularity and capacity of nodes storing those files, 1-hop replication scheme is biased against files shared by peers with low capacity [16]. In a 1-hop scheme, replicas are stored on immediate neighbors. Our scheme overcomes this problem by replicating popular files at the nodes with high capacity, and by regulating the number of random walks dynamically. More random walkers are used when there are less replicas, while fewer walkers are exploited when there are more replicas in the overlay network.

The replication algorithm works as follows. Let be R the maximum number of replicas. In our implementation we use a *proportional* replication strategy, i.e., files are replicated proportional to the querying rate. If a resource is queried many times, more replicas should exist to reduce the associated search load. When a file f is found, the corresponding peer calculates the number of replicas r_f of f to be created as:

$$r_f = \begin{cases} \frac{R \cdot \mathrm{P}(R_{f_n})}{\mu} & \text{if } \mathrm{P}(R_{f_n}) < \mu, \\ R & \text{otherwise.} \end{cases} \tag{5}$$

In Eq. 5, $\mathrm{P}(R_{f_n})$ is the replication probability, R the maximum number of replicas and μ the average replication probability. The replication probability indicates the actual number of replicas to create for a given file f. Specifically, $\mathrm{P}(R_{f_n})$ is obtained as:

$$\mathrm{P}(R_{f_n}) = \frac{\rho_{f_n}}{\rho_n} \tag{6}$$

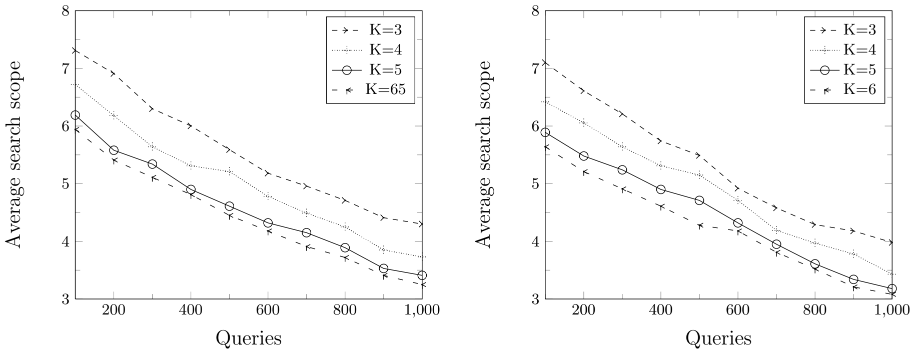
Algorithm 2. Replication heuristics

```
if $owner\_f$ != $self$ then
    return $f\_nodes$ ←owner_f.File_replicate(f); // send request to the owner
    $k$ ←0; $i$ ←0;
    foreach $node\ i\ which\ accessed\ f$ do
        calculate $\alpha$ for node i; $\alpha\_A[k]$ ←$\alpha$; $k$ ←k + 1;
    Sort $\alpha$_A in decreasing order; Calculate $r_f$;
    foreach $k$ in 0 to $r_f$ -1 do
        $f\_nodes$[k]←$\alpha$_A[i];          // Check if res available on f_nodes[k]
        if $Check\_resources(f\_nodes[k])==False$ then $k$ ←k-1;     // Ignore it
        $i$ ←i+1;
    // Replicate on nodes not having replica of file f
    foreach $node\ n$ in $\{f\_nodes\} - \{R\_nodes\}$ do replicate $f$;
    // Delete replica of $f$ from nodes not in f_nodes
    foreach $node\ n$ in $\{R\_nodes\} - \{f\_nodes\}$ do delete $f$;
```

where ρ_{f_n} is the number of requests for file f on node n, and ρ_n is total number of requests on node n. Now, node n calculates α for each node, and stores them in decreasing order in a sorted array. The value α is calculated as the probability that the file to be replicated will be accessed by the peer on which it will be replicated. High probability means that the file has been accessed more times by that node, which will probably access the file more in future too. The probability value α is then $\alpha = \frac{A_{f_j}}{A_f}$, where A_{f_j} is the number of accesses to the file f by node j, and A_f is total number of access of file f. The probability α is calculated for each node which accessed file f and stored in decreasing order in an array. The node n will select first r_f nodes from the sorted array which have enough resources to accommodate file f. Here the considered resources include secondary storage space, main memory and CPU load. After checking resources we will replicate file f only on the peers on which it has not been already replicated. Thus, a file f will be deleted from the nodes which have a smaller value of α. This makes our algorithm dynamic in nature. The replication heuristics is described in Algorithm 2.

5 Simulation Results

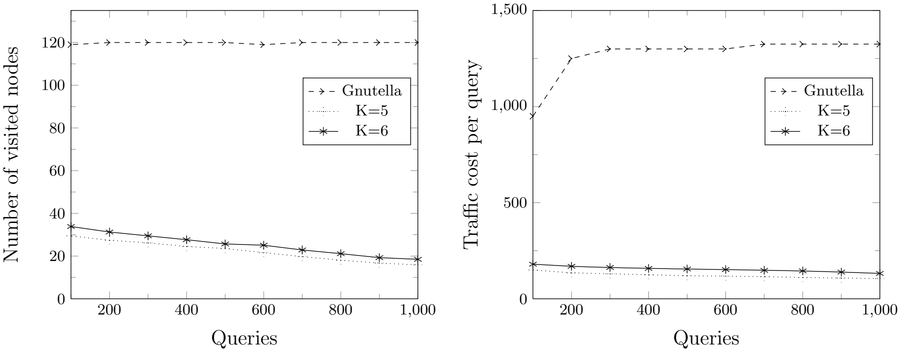
We performed experiments on a network of 120 peers with power law degree distribution and during the network lifetime degree being constant. There were 100 distinct items or files on each peer, and the same replica was not available at any other peer. To simulate our algorithms, we started with 120 peers and connected them randomly. During an initial transient phase, each peer performed 100 queries and no replication was performed. At the steady state, nodes were connected with the topology construction algorithm explained in Sect. 3. Unless otherwise stated, the number of walkers K was set to 3 and the maximum number of replicas R for each file was set to 3. The terminating condition was checked



(a) R=5

(b) R=6

Fig. 1. Average search scope for different values of maximum replicas



(a)

(b)

Fig. 2. Comparison of the k-walker algorithm and gnutella in terms of (a) visited nodes and (b) traffic cost

every two hops in the k-walker searching algorithm. Every 100 queries each peer was disconnected and reconnected to the best neighbors. In our simulation, out of 100 files on every peer, 40 files are music files, 30 are movie files and remaining 30 are miscellaneous files such as data files, pictures, and so on. We generated random requests according to such distribution. For comparison with Gnutella we implemented the flooding algorithm with TTL value large enough to search every resource in network.

To evaluate the search efficiency of the system, we considered the following metrics: the *search scope*, as the number of peers/hops a successful walker traverses during a search; the *replication ratio*, as the ratio of the total files replicated at a given peer to the total number of files on that peer; and the *traffic cost* as the total number of messages generated by the walkers for searching a file and the overhead traffic (e.g., asking for resource information or replicas). Simulations confirmed that k walkers after T steps reach roughly the same

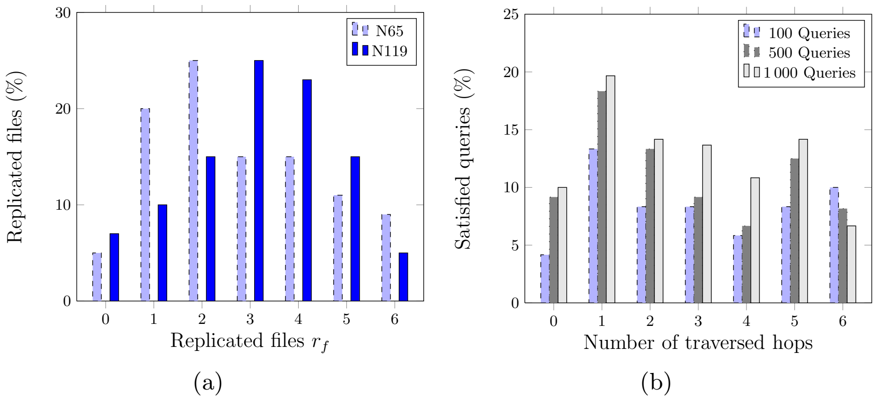


Fig. 3. (a) Replication ratio for nodes N65 and N119. (b) Satisfied queries for a sample node (N85) as a function of the traversed hops.

number of nodes as one walker after kT steps. Hence, by using k walkers, we can expect to improve the response time by a factor of k. We performed experiments with a different number of walkers and plotted the search scope as a function of the number of queries for different values of maximum replicas in Fig. 1. We can see that initially search scope is maximum, i.e., any successful walker traverses less hops on the average. As the number of queries increases, more replicas are created, thus decreasing the average search scope. Furthermore, from the figure it emerges that the search scope does not significantly changes when increasing K from 5 to 6, while the traffic clearly increases in the latter case. Finally, we can see that the average search scope does not actually depend on the considered values of R. Therefore, in the following, we will consider $K = 5$ and $R = 6$.

To measure the effectiveness of our heuristics, we also compared our solution with the standard flooding algorithm used in Gnutella as shown in Fig. 2. The average nodes visited and the traffic cost per query are significantly lower with our proposed approach. Figure 3a shows the replication ratio on two representative nodes – namely, N65 and N119 – after running 1,000 queries. From the plot we can observe that for N65 only 9% of files are replicated with the maximum factor. Similarly, at node N119 only 5% of files are replicated on 6 nodes, 15% files at 5 nodes and so on. Figure 3b shows the number of satisfied queries as a function of the hop distance for a sample node, namely, node N85. We can notice that the number of satisfied queries increases with the number of resources requested, and the increase is more significant when the number of hops is lower. As a consequence, most queries can be satisfied within two or three hops in all cases, thus, with low delay.

6 Conclusion

In this paper, we proposed replication and search heuristics to reduce the load for searching resources in unstructured peer-to-peer (P2P) systems. For the connection phase, we select the best nodes depending on the previous history, and dynamically adapt to the changing requests. After the connection phase, we proposed the k-walker algorithm, which dynamically determines the number of walkers and searches the files with a low network overhead. We used a proportional replication scheme built on top of the popularity of a file that is adaptive by nature. Experimental evaluation has shown that our techniques are effective at improving search efficiency.

References

[1] Schulze, H., Mochalski, K.: ipoque GmbH Internet Study (2008/2009), `http://www.ipoque.com/sites/default/files/mediafiles/documents/internet-study-2008-2009.pdf` (retrieved January 28, 2013)

[2] Tigelaar, A.S., Hiemstra, D., Trieschnigg, D.: Peer-to-peer information retrieval: An overview. ACM Trans. Inf. Syst. 30(2), 9:1–9:34 (2012)

[3] Zhuang, Z., Liu, Y., Xiao, L., Ni, L.: Hybrid periodical flooding in unstructured peer-to-peer networks. In: Proc. of ICPP 2003, pp. 171–178 (October 2003)

[4] Haribabu, K., Reddy, D., Hota, C., Ylä-Jääski, A., Tarkoma, S.: Adaptive lookup for unstructured peer-to-peer overlays. In: Proc. of COMSWARE 2008, pp. 776–782 (January 2008)

[5] Xiao, L., Liu, Y., Ni, L.: Improving unstructured peer-to-peer systems by adaptive connection establishment. IEEE Trans. on Computers 54(9), 1091–1103 (2005)

[6] Haribabu, K., Hota, C., Ylä-Jääski, A.: Indexing through querying in unstructured peer-to-peer overlay networks. In: Ma, Y., Choi, D., Ata, S. (eds.) APNOMS 2008. LNCS, vol. 5297, pp. 102–111. Springer, Heidelberg (2008)

[7] Patro, S., Hu, Y.: Transparent query caching in peer-to-peer overlay networks. In: Proc. of Parallel and Distributed Processing Symposium (April 2003)

[8] Chu, Y., Rao, S., Seshan, S., Zhang, H.: A case for end system multicast. IEEE Journal on Selected Areas in Communications 20(8), 1456–1471 (2002)

[9] Nakao, A., Peterson, L., Bavier, A.: A routing underlay for overlay networks. In: Proc. of SIGCOMM 2003, pp. 11–18 (2003)

[10] Fast, A., Jensen, D., Levine, B.N.: Creating social networks to improve peer-to-peer networking. In: Proc. of ACM SIGKDD 2005, pp. 568–573 (2005)

[11] Lin, C.J., Chang, Y.T., Tsai, S.C., Chou, C.F.: Distributed social-based overlay adaptation for unstructured P2P networks. In: IEEE Global Internet Symposium, pp. 1–6 (May 2007)

[12] Hota, C., Nunia, V., Ylä-Jääski, A.: Distributed algorithms for improving search efficiency in peer-to-peer overlays. International Journal of Computer Networks and Information Security 4(3), 1–7 (2012)

[13] Cholvi, V., Felber, P., Biersack, E.: Efficient search in unstructured peer-to-peer networks. In: Proc. of SPAA 2004, pp. 271–272 (2004)

[14] Lv, Q., Cao, P., Cohen, E., Li, K., Shenker, S.: Search and replication in unstructured peer-to-peer networks. In: Proc. of ICS 2002, pp. 84–95 (2002)

[15] Kitamura, H., Fujita, S.: A biased k-random walk to find useful files in unstructured peer-to-peer networks. In: 2009 International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 210–216 (December 2009)

[16] Chawathe, Y., Ratnasamy, S., Breslau, L., Lanham, N., Shenker, S.: Making gnutella-like P2P systems scalable. In: Proc. of SIGCOMM 2003, pp. 407–418 (2003)

CE-SeMMS: Cost-Effective and Secure Mobility Management Scheme Based on SIP in NEMO Environments*

Chulhee Cho, Jae Young Choi, Younghwa Cho, and Jong Pil Jeong

College of Information and Communication Engineering, Sungkyunkwan University
2066 Seobu-ro Jangan-gu Suwon Kyunggi-do, 440-746, Korea
chuli77@hanafos.com, {jychoi1001,choyh2285,jpjeong}@skku.edu

Abstract. The mobile Virtual Private Network (MVPN) of the Internet Engineering Task Force (IETF) is not designed to support NEtwork MObility (NEMO) and is not suitable for real-time applications. Therefore, architecture and protocols to support VPN in NEMO are needed. Therefore, in this paper we propose a cost-reduced secure mobility management scheme (CE-SeMMS) that is based on session initiation protocol (SIP) and designed for real-time applications with VPN. Our scheme to support MVPN in NEMO enables the session to be well maintained during movement of the entire network. Further, in order to reduce the authentication delay time in handoff operations, the signaling time which occurs to maintain the session is shortened through our proposed handoff scheme which adopts authentication using HMAC-based one-time password (HOTP). Our performance analysis results show our proposed scheme provides improvement in average handoff performance time relative to existing schemes.

Keywords: NEMO, mobile Virtual Private Network (VPN), SIP, PMIP.

1 Introduction

As the coverage area of wireless LAN (WLAN) expands, the demand from users is growing for access to the Internet anytime and anywhere. To satisfy this requirement, technologies that enable access to the Internet on trains, busses, ships, and other modes of transportations have come into the limelight. One such technology is NEMO (NEtwork MObility), an IP network mobility technology [1-3]. NEMO enables Internet connection service to be provided from the mobile router (MR) with all the nodes inside the network not recognizing the mobility, a standardization that is making progress in IETF based on IPv6. VPN service in NEMO has wide-ranging applications, providing stable access to the intranet for mobile networks. However, a method

* This research was supported by Next-Generation Information Computing Development Program (2012-0006426) and Basic Science Research Program (2011-0027030) through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology. Corresponding author: Jongpil Jeong.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 280–287, 2013.

for providing VPN service has yet to be identified in the NEMO working group of IETF. MVPN (mobile VPN) of IETF uses one IPSec [4] tunnels and two MIP tunnels. These three tunnels are major contributors to overhead during the real-time packet transfer. Thus, a new architecture and protocol are required to support the MVPN in safe NEMO. This paper proposes a cost-reduced secure mobility management scheme (CE-SeMMS) based on the SIP (Session Initiation Protocol) which is suitable for real-time application on MVPN and which shortens the signaling time. This design maintains the session continuously as the overall network moves. It integrates SIP-based MVPN and NEMO to provide efficient group mobility for high security and real-time services.

2 Cost-Effective and Secure Mobility Management Scheme (CE-SeMMS)

2.1 Architecture

The proposed CE-SeMMS comprises SIP, secure real-time transport protocol (SRTP) [5], multimedia internet keying (MIKEY) [6], and a Diameter server [7] to provide VPN services in NEMO.

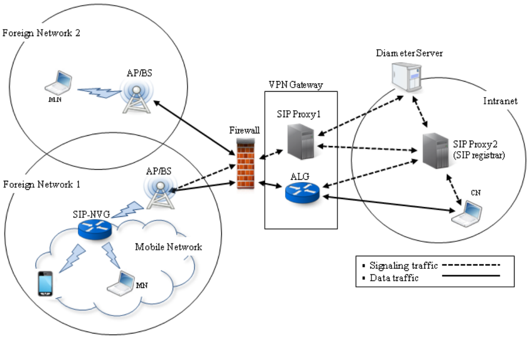


Fig. 1. System architecture

Fig. 1 depicts the architecture of the proposed CE-SeMMS. Fig. 1 shows a mobile network in a foreign network (Internet) connecting to the CN in the home network (intranet). The SIP NEMO VPN gateway (SIP-NVG) shown in the mobile network residing in Foreign Network 1 is the gateway of the mobile network to other networks. It follows the SIP standards and manages the traffic between the mobile network and the outside world. The VPN gateway (VPN GW) consists of SIP Proxy 1 and an application level gateway (ALG). SIP Proxy 1 is a SIP proxy server, which authenticates the incoming SIP messages through the Diameter server. It also routes messages to SIP Proxy 2 which is essentially a SIP registrar. In our proposed CE-SeMMS, we use SIP to authenticate and identify the mobile users. SIP also supports user mobility and terminal mobility [8]. Terminal mobility is achieved by sending new INVITE (re-INVITE) to the CN using the same call ID as that in the original session. The new INVITE contains the new contact address the MN has acquired in the new location. After receiving the re-INVITE, the CN will redirect future traffic to the MN's new location.

The authentication is done by the Diameter server rather than by delegating to a SIP server. HOTP-based authentication is adopted in the proposed CE-SeMMS to reduce authentication time, an element of delay time during handoff.

2.2 Operations

In the architecture shown in Fig. 1, the entire mobile network may move from one IP subnet to another. This is called network handoff. It is also possible that an MN moves into or moves out of the mobile network. This is called node handoff.

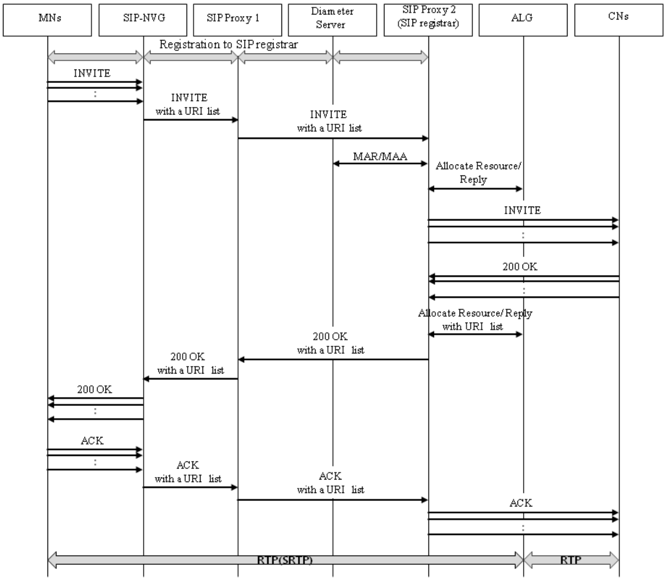


Fig. 2. Message flows when mobile network roams from home network to foreign network

Fig. 2 illustrates the flow when the mobile network moves from the intranet to the Internet. When a SIP-NVG moves to a foreign network, it must register with the SIP registrar using its new IP address. The SIP-NVG then checks whether there are MNs with active sessions inside the mobile network, according to the session table. The SIP-NVG must re-INVITE all CNs to recover all of the ongoing sessions. However, this process may cause substantial amounts of signaling messages in the wireless links. In order to reduce the signaling overhead, the SIP-NVG combines all contact addresses of CNs into a URI list. The URI list is then conveyed by the SDP embedded in one INVITE message.

3 Performance Analysis

In order to support secure communication in VPN, the proposed CE-SeMMS sends signaling messages carrying security information. It also sends signaling messages to maintain session continuity during handoff. In our proposed CE-SeMMS, the inter-realm roaming of a mobile network includes three types of handoff:

- From the intranet (home network) to a foreign network,
- From a foreign network to another foreign network, and
- From a foreign network back to the intranet.

They are represented as $\mathrm{H_{hf}}$, $\mathrm{H_{ff}}$ and $\mathrm{H_{fh}}$, respectively. We assume that the network topology is configured as shown in Fig. 3 such that the mobile network returns to the intranet after it moves across *N* - 1 foreign networks. To use in Analysis of the proposed CE-SeMMS, we define the following parameters:

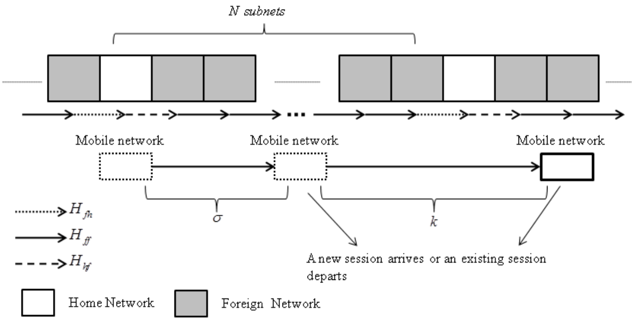


Fig. 3. Network topology for analysis

Table 1. List of Parameters Used in Analysis

Parameter	Description
N	Number of networks a mobile network visits before it goes back to the intranet.
λ	Session arrival rate for a mobile network
$1/\mu$	Average session service time
$1/\gamma$	Average network residence time
c	Maximum number of ongoing sessions in a mobile network

For a mobile network, let $\mathrm{f_m(t)}$ be a general density function for the network resi-dence time $\mathrm{t_M}$ in a subnet. Let $\mathrm{E[t_M]} = 1/\gamma$. Its Laplace transform is written:

$$f_m^*(s) = \int_{t=0}^{\infty} e^{-st} f_m(t) dt \tag{1}$$

For demonstration purpose, we assume that the network residence time follows a Gamma distribution. The Laplace transform of a Gamma random variable is expressed:

$$g_i = f_m^*(\pi_i) = \left(\frac{\gamma\beta}{\pi_i + \gamma\beta}\right)^{\beta} \tag{2}$$

In the proposed CE-SeMMS, when a mobile network moves across networks, it must perform registration with the SIP Registrar to update its location. It also must send re-INVITE messages to the CNs if there are ongoing sessions with the MNs in the mobile network. Hence, the cost comprises two parts: the registration cost for SIPNVG and the re-INVITE cost for maintaining session continuity. The registration cost is independent of the number of ongoing sessions in the mobile network, because the SIP-NVG can register with the SIP Registrar on behalf of the whole mobile network. On the other hand, the re-INVITE cost depends on the number of ongoing sessions in the mobile network. The cost increases when the number of ongoing sessions increases. However, because we design a URI list embedded in one re-INVITE message, the cost to really send a re-INVITE message to each individual CN is nearly constant, regardless of the number of ongoing sessions in the mobile network. We define the following parameters:

Table 2. Parameters for handoff signaling cost

Parameter	Description
S	Average handoff cost
R	Average registration cost of a mobile network
L	Average cost for the first part of re-INVITE
I	Average cost for the second part of re-INVITE of a session

Therefore, we can denote the signaling cost for handoff:

$$S_{hf}^{i} = R_f + L_{hf} + iI_{hf},\ \ S_{ff}^{i} = R_f + L_{ff} + iI_{ff},\ \ S_{fh}^{i} = R_h + L_{fh} + iI_{fh} \tag{3}$$

As discussed above, the arrival of sessions to a mobile network follows a Poisson process, and the session service time is exponentially distributed. In addition, there is a limit c for the maximum number of ongoing sessions allowed in the mobile network. Therefore, we can model the number of ongoing sessions in a mobile network as an M/M/c/c queuing system. The steady state probability that there are i ongoing sessions in the mobile network is then given by [9]:

$$P_i = \frac{\lambda^i}{i!\,\mu!}\left(\sum_{n=0}^{c}\frac{\lambda^x}{x!\,\mu^x}\right)^{-1} \tag{4}$$

As a result, the average handoff-signaling cost per unit time can be derived as:

$$\sum_{i=0}^{c} C_i P_i \pi_i \tag{5}$$

To evaluate the performance of the proposed CE-SeMMS, we define the following parameters:

Table 3. Parameters for CE-SeMMS Signaling Cost

Parameter	Description
a_x	The processing cost for SIP registration at Node x
b_x	The processing cost for SIP INVITE message at Node x
$A_{x,y}$	The transmission cost of SIP registration between Node x and Node y
$B_{x,y}$	The transmission cost of a SIP INVITE message between Node x and Node y
U	The total cost for SIP Proxy 1 to process and transmit UAR/UAA messages to the Diameter server
M	The total cost for SIP Proxy 2 to process and transmit MAR/MAA messages to the Diameter server

where x and y can be mn, nvg, pro, reg, alg, or cn which denote MN, SIP-NVG, SIP Proxy 1, SIP Proxy 2 (SIP Registrar), ALG, and CN, respectively. According to the signaling message flow described in Section 3, the above costs can be calculated:

$$R_h = a_{nvg} + a_{reg} + 2A_{nvg,reg} + M,$$

$$R_f = a_{nvg} + a_{pro} + a_{reg} + 2A_{nvg,pro} + 2A_{pro,reg} + U + M,$$

$$L_{hf} = 2b_{nvg} + 3b_{pro} + 4b_{reg} + 2b_{alg} + 3B_{nvg,pro} + 3B_{pro,reg} + 4B_{reg,alg} + B_{reg,cn} + M,$$

$$L_{ff} = 2b_{nvg} + 3b_{pro} + 2b_{reg} + b_{alg} + 3B_{nvg,pro} + 3B_{pro,reg} + 2B_{reg,alg} + M,$$

$$L_{fh} = 2b_{nvg} + 3b_{reg} + b_{alg} + 3B_{nvg,reg} + 2B_{reg,alg} + B_{reg,cn} + M,$$

$$I_{hf} = b_{mn} + b_{nvg} + b_{reg} + b_{cn} + 2B_{mn,nvg} + 2B_{reg,cn},$$

$$I_{ff} = b_{mn} + b_{nvg} + 2B_{mn,nvg},$$

$$I_{fh} = b_{mn} + b_{nvg} + 2B_{mn,nvg} + b_{reg} + b_{cn} + 2B_{reg,cn}; \tag{6}$$

In the architecture we propose, SIP-NVG manages the overall network mobility, registering the whole mobile network in the SIP Registrar when it moves to a new subnet. If there is no SIP-NVG, all MNs in the same mobile network must update their locations separately. This increases signaling cost. We can re-define the costs (3) when there is no SIP-NVG as follows. where m is the number of MNs connected to the mobile network.

$$S_{hf}^{i} = mR_f + iL_{hf} + iI_{hf},\ \ S_{ff}^{i} = mR_f + iL_{fh} + iI_{ff},\ \ S_{fh}^{i} = mR_h + iL_{hh} + iI_{fh}. \tag{7}$$

4 Numerical Results

This section provides the numerical results for the analysis presented in Section 3. The analysis was validated by extensive simulations using ns-2. As discussed in Section 3, the signaling cost function consists of the transmission cost and the processing cost. We assume that the transmission cost is proportional to the distance between the source and destination nodes, and the processing cost includes the processing and verifying SIP messages. Also, the transmission cost of a wireless link is higher than that of a wireline. Fig. 4 presents a comparison among the signaling costs with IETF MVPN, with SIP-NVG and HTTP, without SIP-NVG and with HOTP, and with CE-SeMMS (with SIP-NVG and HOTP). Fig. 4 shows that CE-SeMMS with SIP-NVG has lower signaling costs for handoff than in IETF MVPN. This is because IETF MVPN requires time to establish the three tunnels. Compared to the mobile network without SIP-NVG, as expected, the proposed CE-SeMMS reduces handoff signaling cost significantly, since SIP-NVG performs registration in the SIP Registrar on behalf of the entire mobile network when it moves to a new subnet, whereas, without SIP-NVG, all MNs must update their locations individually.

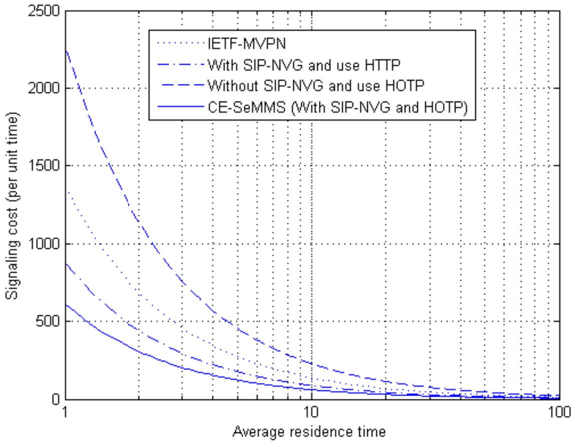


Fig. 4. Comparison of various signaling costs versus residence time

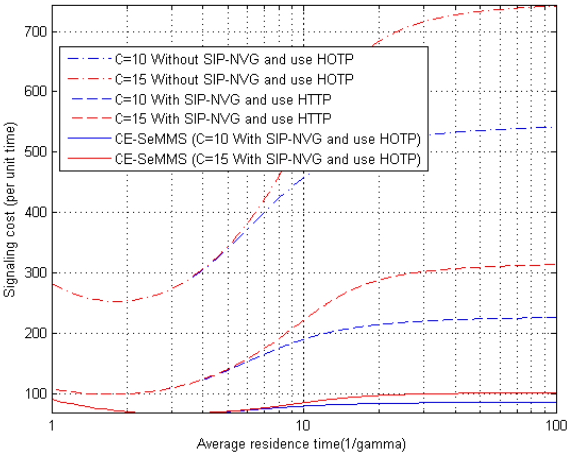


Fig. 5. Comparison of signaling cost with and without SIP-NVG and using HTTP or HOTP Method

Fig. 5 demonstrates the average signaling cost for handoff versus ρ, the number of sessions in the mobile network. We also see that when ρ increases, the average cost for SIP-based solutions increases too. The reason is that with more ongoing sessions, more re-INVITEs are needed to maintain session continuity. Besides, when ρ is larger than 20, the costs of all techniques presented in Fig. 5 remain almost constant. This is because, when ρ approaches 20, the number of ongoing sessions with each technique reaches the maximum number allowed in the mobile network.

5 Conclusions

Although the IETF standard has proposed a mobile VPN architecture, it is designed for the movement of a signal node only. In addition, IETF MVPN has large overhead for transmitting real-time packets, because it requires one IPsec tunnel and two MIP tunnels. We analyzed the design and performance of our proposed design, and results indicate that the proposed CE-SeMMS based on SIP is well suited to real-time service. Although SIP-based mobility management can easily support routing optimization, there may be an upswing in the handoff signaling costs, because many signaling messages are transmitted to maintain the session in progress with SIP in NEMO. In the proposed CE-SeMMS, a URI list is used to signify the SIP proxy server instead of transmitting signaling messages individually to each node. Therefore, the signaling cost is reduced.

References

1. Schena, V., Losquadro, G.: FIFTH Project Solutions Demonstrating New Satellite Broadband Communication System for High Speed Train. In: Proc. IEEE Vehicular Technology Conf., pp. 2831–2835 (2004)
2. WirelessCabin Project, `http://www.wirelesscabin.com`
3. Devarapalli, V., Wakikawa, R., Petrescu, A., Thubert, P.: Network Mobility (NEMO) Basic Support Protocol. IETF RFC 3963 (2005)
4. Kent, S., Atkinson, R.: Security Architecture for the Internet Protocol. IETF RFC 2401 (1998)
5. Baugher, M., McGrew, D., Naslund, M., Carrara, E., Norrman, K.: The Secure Real-Time Transport Protocol (SRTP). IETF RFC 3711 (2004)
6. Arkko, J., Carrara, E., Lindholm, F., Naslund, M., Norrman, K.: MIKEY: Multimedia Internet KEYing. IETF RFC 3830 (2004)
7. Calhoun, P., Loughney, J., Guttman, E., Zorn, G., Arkko, J.: Diameter Base Protocol. IETF RFC 3588 (2003)
8. Chen, J.-C., Zhang, T.: IP-Based Next-Generation Wireless Networks. John Wiley and Sons (2004)
9. Gross, D., Harris, C.M.: Fundmentals of Queueing Theory. John Wiley and Sons (1998)
10. Tuan-Che, C., Jyh-Cheng, C., Zong-Hua, L.: Secure Network Mobility (SeNEMO) for Real-Time Applications. IEEE Trans. Mobile Computing 10(8), 1113–1129 (2011)

A System-Level Approach for Designing Context-Aware Distributed Pervasive Applications

Kevin I-Kai Wang, HeeJong Park, Zoran Salcic, and Panith Ratnayaka

Department of Electrical and Computer Engineering,
University of Auckland, Auckland, New Zealand
kevin.wang@auckland.ac.nz

Abstract. Recent advances of embedded and wireless technologies make ubiquitous wireless sensor and actuator networks (WSANs) a reality. While individual sensor nodes are relatively easy to use, the overall system behaviors are difficult to model and program. This paper presents a novel system-level software design methodology using a concurrent and reactive programming language, SystemJ, in designing distributed pervasive applications based on WSANs with no requirement of any additional middleware. A context-aware pervasive service case study, where multiple fixed and mobile IPv6-enabled WSAN nodes are deployed in an office environment for RSSI-based localization and automated lighting control based on the user location, is designed using SystemJ.

Keywords: Distributed pervasive applications, Context-aware services, System-level design.

1 Introduction

Over the last decade, embedded technologies have achieved a tremendous breakthrough in terms of increasing computational power and minimizing physical size. More and more computing resources are embedded in the surrounding environment providing intelligent services, while at the same time exchanging context information to one another through wireless communication. The ubiquitous wireless sensor and actuator networks (WSANs) have extended the capability and complexity of the traditional Cyber-Physical Systems (CPS) [1] to another level, by merging pervasive and the Internet of Things (IoT) [2] technologies.

In this paper, we use as a motivating example a location-based automated lighting control system. The system is implemented with the Java-enabled WSAN nodes, SunSPOT [3], which allows Java Virtual Machine to operate on a bare metal without any operating system. The nodes communicate with each other using the IPv6-enabled 6LoWPAN protocol [4], which is provided as part of the SunSPOT Java API. Although SunSPOT is one of the most advanced wireless sensor nodes available and can be programmed in Java, the issues of concurrency and reactivity of distributed context-aware pervasive systems, require a system-level software design approach.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 288–298, 2013.

SystemJ [5] is a concurrent programming language based on the Globally Asynchronous Locally Synchronous (GALS) Model of Computation (MoC) [6] and is capable of designing highly reactive and concurrent systems. The synchronous parts of a SystemJ program are based on the Synchronous Reactive MoC [7], which makes formal verification of system functionalities possible. SystemJ programs are compiled into Java and can be executed on any machine that has a JVM. Unlike the other synchronous and reactive languages, such as Esterel [7], SystemJ provides an integrated solution to describe both control-dominated and data-dominated computations. The control-dominated operations are handled by SystemJ reactive statements, whereas the data-dominated operations are described in Java. Hence, the design approach of SystemJ [5] provides a more natural way to capture the model of distributed pervasive systems. Moreover, SystemJ provides higher level abstractions which allow a distributed system that consists of multiple wireless sensor and actuator nodes to be designed and implemented as a number of cooperating SystemJ programs, without employing any middleware. In this paper we show how SystemJ is ported to a Java-enabled SunSPOT-based WSAN and used in the design of a distributed pervasive system for location-based lighting control.

Section 2 of this paper introduces related works, while section 3 provides a brief overview of the SystemJ language. Section 4 describes the indoor localization and the location-based pervasive service implemented on the SunSPOT-based WSAN using SystemJ. Section 5 explains the platform specific runtime support of SystemJ operating on the SunSPOT nodes. Section 6 concludes the paper and indicates future directions based on the achieved results.

2 Related Works

The wide spread of Internet and increasing usage of WSANs result in merging Internet Protocol (IP) with low data rate sensor network protocols such as 6LoWPAN, which is one of the first IPv6-enabled protocols. Many emerging distributed pervasive applications are built on top of WSAN platforms. A comprehensive study on RSSI-based indoor localization using Wireless Sensor Networks (WSNs) is presented in [8]. The applications of WSANs in pervasive healthcare applications, such as patient and elderly care, are also exploited in previous works such as [9]. Various indoor and outdoor location-based services are developed and integrated with mobile devices to improve the quality of living [10][11]. However, despite of the progress in terms of the physical network infrastructure, the design of distributed pervasive applications still rely on different layers of abstraction, middleware, and device drivers [11][12]. The layered architecture poses many challenges in modeling, designing and implementing distributed pervasive systems. In this paper, a system-level design paradigm for pervasive applications, based on concurrent programming language, SystemJ, is presented. It allows modeling and implementing typical distributed pervasive systems with significantly reduced design efforts. The approach is demonstrated on an automated location-based lighting control system.

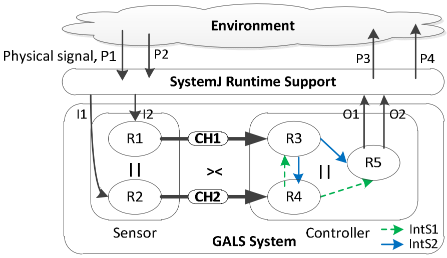


Fig. 1. Graphical illustration of a SystemJ program

3 SystemJ Overview

A GALS program designed in SystemJ consists of top-level design entities called clock domains (CDs), which represent concurrent asynchronous entities for constructing the overall system. Within each CD, there can be one or more synchronous program entities, called reactions, which execute concurrently by changing their states in lockstep according to the associated CD logical clock, and comply with Synchronous Reactive (SR) formal MoC [7]. Fig. 1 shows an illustration of a SystemJ program which models a small sensor and actuator system. In this program, there are two CDs, namely Sensor and Controller, running asynchronously to each other. Sensor CD gathers various data from the external environment, processes the information and transfers results to the Controller CD. The Controller CD generates appropriate control signals which are emitted back to the environment. Asynchronous parallel operator (><) forms boundaries between CDs within the program. Within each CD, there are multiple reactions composed using the synchronous parallel operator (||), which in turn can be applied within reactions in a hierarchical way (i.e. nested reactions) to any depth.

SystemJ reactions within a CD communicate using an abstract object called signal. A signal is represented by its binary status, which can be either true or false (i.e. present or absent). The reaction that emits a signal sets its status to true for only one logical tick. In Fig. 1, signals IntS1 and IntS2 are emitted by reaction R4 and R3 respectively. Any internal signal emitted by a reaction can be seen by all the other reactions in the same CD (i.e. it is broadcasted), as shown in Fig. 1. In addition to the binary status, a signal can also carry a value which can be of any Java object or primitive type. If a signal is a valued signal, the status along with its value is emitted and captured by all neighboring reactions within the same CD for one logical tick. Signals in SystemJ are also used as the communication mechanism between reactions and the external environment (shown as I1, I2 and O1, O2 in Fig. 1). These signals are called interface signals. Interface signals, which are processed by the SystemJ Runtime Support (RTS), are uni-directional and link a SystemJ program with its external environment. The SystemJ RTS, allows systems to be deployed across heterogeneous platforms without the need of an additional middleware. The high level signal abstraction allows system designers to focus on high level functionalities rather than low level communication details.

SystemJ CDs execute asynchronously at their own pace and, hence, signal broadcasting with duration of one tick would not be a reliable communication method between reactions in different CDs. Instead, CSP (Communicating Sequential Process) style message passing mechanism [5], based on rendezvous, is provided in form of another SystemJ object called channel, which handles communications between two synchronous reactions that reside in two different CDs (shown as CH1 and CH2 in Fig. 1). Channels provide simplex, point-to-point communication and can transfer any Java object or primitive data type.

The primary output of SystemJ program compilation is Java code, which is then compiled by standard Java compiler and runs on various JVM-enabled execution platforms. Java code generated by the SystemJ compiler is compliant with CLDC 1.1 specification. Platform dependent libraries, such as underlying communication methods for SystemJ signals are provided and maintained separately by the SystemJ Runtime Support (RTS), which is presented in Section 5.

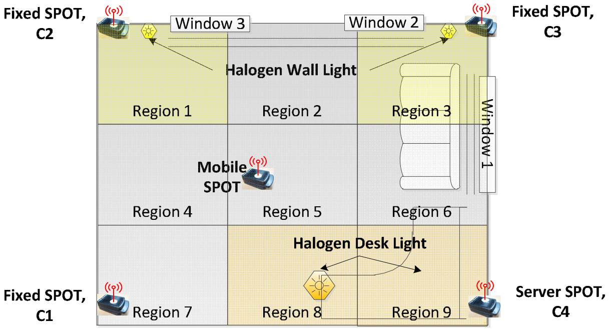


Fig. 2. Location-based lighting control system

4 Location-Based Pervasive Service

Referring to Fig. 2, a system that provides a location-based automated lighting control service is designed using SystemJ on SunSPOT nodes. Experimental testbed that creates the Ambient Intelligence (AmI) [13] consists of four fixed SunSPOT nodes, one of them being Server SPOT (C4) in the lower right hand corner. A mobile SunSPOT node attached to the user is sending broadcasting radiogram packets periodically. Upon receiving the broadcasted packets, the four fixed SunSPOT nodes retrieve and convert the RSSI information into estimated distances. The Server SPOT then polls the other three fixed SPOT nodes to collect the distance values, and carries out a trilateration-based algorithm to estimate the user location (one of the regions as shown in Fig. 2). Based on the user location, the corresponding lighting control command is issued by the Server SPOT, via a serial bus. Two sets of lighting fixtures, at the wall and above the desk, are adjusted according to the current user location.

Fig. 3 shows the graphical representation of the location-based lighting control system as designed and implemented in SystemJ. The dashed rectangles enclose the SystemJ program and the associated RTS running on each SunSPOT node. Within each SystemJ program, CDs are shown as solid rectangles and there may be multiple synchronous reactions (ellipses) within one CD. Referring to Fig. 3, the Mobile SunSPOT contains only one **RSSIsend** CD that emits *RSSIout* signal periodically, which is converted into a broadcasting radiogram packet by the RTS (Radiogram Sender). Thus, the *RSSIout* signal is received by each of the fixed SPOT nodes, including the Server SPOT, via the corresponding Radiogram Receiver RTS, which converts a physical RSSI event into a SystemJ signal. Upon receiving the input RSSI signal, the **RSSIcapture** CD converts RSSI into an estimated distance value, which is emitted through *Distout* signal when the Server SPOT polls for distance information.

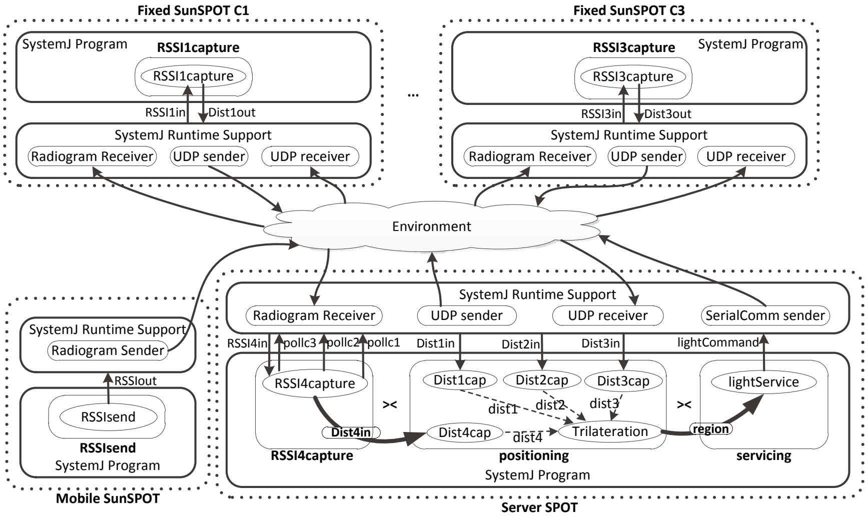


Fig. 3. System architecture of the location-based lighting service implemented in SystemJ

The Server SPOT provides additional functionality compared to the other fixed SPOT nodes. Referring to Listing 1, which shows the system declaration of Server SPOT SystemJ program, the Server SPOT contains three CDs, namely **RSSI4capture**, **positioning**, and **servicing**, running asynchronously to each other. The **RSSI4capture** CD receives RSSI and converts it into a distance value and also polls the other fixed SPOT nodes to collect their distance values (shown as *pollc1*, *pollc2*, *pollc3* output signals in Fig. 3 and Listing 1). All the distance values are sent to the **positioning** CD via interface signals and channel (*Dist1in*, *Dist2in*, *Dist3in*, and *Dist4in* as shown in Fig. 3 and Listing 2, line 1-4). The await statement (Listing 2, line 9) is one of the reactive statements provided by SystemJ, which demonstrates the ability of SystemJ to synchronize with external events. The distance values are passed to the location() method, which is a customized Java method performing trilateration computation and returning the user located region (Listing 2, line 11). The detail of the trilateration implementation is out of the scope of this paper and hence is not mentioned here. The located region is sent to the **servicing** CD via channel *region* as in Listing 2, line 12 and 20. The two asynchronous CDs are blocked during the synchronous channel communication and thus message delivery is guaranteed. Once the user located region is received via channel *region*, **servicing** CD emits *lightCommand* output signal, as shown in Listing 2 line 23 and 25, based on the received user located region.

Listing 1. System declaration of Server SPOT SystemJ program.

```
system{
  interface{
       input String signal Dist1in, Dist2in, Dist3in;
       output signal pollc1, pollc2, pollc3;
       input Double channel Dist4in;
       output Double channel Dist4in;
       …
  }
  {
       RSSI4capture(RSSI4,pollc1,pollc2,pollc3,Dist4in)
       ><
       positioning(Dist1in,Dist2in,Dist3in,Dist4in,region)
       ><
       servicing(region,lightCommand)
  }}
```

Listing 2. Clock domain example of Server SunSPOT.

```
1    reaction positioning (: input String signal Dist1in,
2                            input String signal Dist2in,
3                            input String signal Dist3in,
4                            input Double channel Dist4in,
5                            output Integer channel region){
6    …
7    {
8       while (true){
9         await(dist1||dist2||dist3||dist4);
10        if(dist1!=null&&dist2!=null&&dist3!=null&&dist4!=null){
11          regionNo = Calculate.location(values);
12          send region(regionNo);
13        }
14        pause;
15   }}}
16   reaction servicing (:  input Integer channel region,
17                       output String signal lightCommand){
18     …
19     while (true) {
20       receive region;
21       regionVal = (Integer)#region;
22       if(regionVal.intValue()==1||regionVal.intValue()==3)
23       { emit lightCommand(wallLightOn); }
24       else if(regionVal.intValue()==1||regionVal.intValue()==3)
25       { emit lightCommand(wallLightOn); }
26       …
27       pause;
28   }}
```

This case study demonstrates the strength of SystemJ in designing and implementing complex context-aware systems constructed using WSANs. For example, the SystemJ program for the entire Server SPOT is less than 200 lines of code. Each wireless node is modeled and controlled by a single SystemJ program and multiple programs communicate with each other using interface signals. Within an individual node, asynchronous behaviors, such as sensing and actuating, are implemented as separate CDs and communicate to each other using SystemJ channels, which guarantee synchronization of the communication [5]. Physical events and communications are converted into SystemJ signals by the RTS. It is important to note that the same SystemJ program can run on any Java-enabled platform, and only adaptation of the RTS is needed. The RTS removes the needs of additional middleware and allows system designers to focus on implementing system functionalities. SunSPOT specific RTS, necessary for implementing the aforementioned location-based pervasive service, is presented in the next section.

5 SystemJ Runtime Support for SunSPOT

SunSPOT is a wireless node based on ARM9 processor with Squawk VM [3] that does not require an operating system. It is capable of sensing acceleration, light intensity and temperature of its physical environment, and providing digital/analogue control outputs through I^{2}C, serial bus (RS-232) and PWM signals. Wireless communication is provided through the IPv6-enabled 6LowPAN protocol and includes TCP/IP, UDP and radiogram. In order to incorporate location awareness into a system, broadcasting radiogram communication is used to generate RSSI input signals (e.g. *RSSI3in* signal in Fig. 3) to each SystemJ program on the fixed SPOTs, which in turn enables to determine the current user location. Similarly, the UDP communication is used to create input and output signals (e.g. *Dist1in* and *pollc1* signals in Fig. 3), which are used for communication and exchange of messages between SystemJ programs running on different SunSPOTs. Based on the user location, a command to control the lighting circuit over a serial communication is issued through a SerialComm output signal (*lightCommand* signal in Fig. 3).

Every SystemJ interface signal emitted to or captured from the external environment is processed by the SystemJ RTS. The RTS is completely written in Java and abstracts the physical signals to the semantics of SystemJ signals. In order to incorporate new types of signals, the corresponding Java interface needs to be implemented. Each input signal has its corresponding GenericSignalReceiver class, whereas each output signal has the GenericSignalSender class. Each SystemJ program has an associated XML file, which defines all the interface signals and their corresponding attributes and RTS library classes. A typical structure of a XML file is shown in Listing 3. It is important to note that XML is parsed by a SystemJ program at runtime and hence the same logical signal can be mapped to a different physical signal when ported on a different execution platform.

In Listing 3, an input signal named '*RSSI*' and an output signal named '*lightCommand*', are bound to the Radiogram input and SerialComm output, respectively, which are handled by the SystemJ RTS (RadiogramReceiver and SerialCommSender classes). The input signal *RSSI* is defined with a set of attributes including Name, Port, and SignalClass. The Name attribute indicates the name of the signal. The Port attribute specifies the port for listening to the incoming radiogram packets. The value for SignalClass attribute determines the path of Java class that implements GenericSignalReceiver interface. For the *lightCommand* output signal, Port and Baud rate attributes specify the physical COM port and communication speed of the serial bus for delivering the physical message. The SignalClass shows this signal is handled by the SerialCommSender class, which implements the GenericSignalSender interface.

Listing 3. XML file contents

```
<SystemJProgram>
   <ClockDomain>
      <Inputs>
         <Signal
          Name="RSSI"
          Port="42"
          SignalClass="systemj.Signals.RadiogramReciever"/>
      </Inputs>
      <Outputs>
         <Signal
          Name="lightCommand"
          Port="COM1"
          Baud ="9600"
          SignalClass="systemj.Signals.SerialCommSender"/>
      </Outputs>
   </ClockDomain>
</SystemJProgram>
```

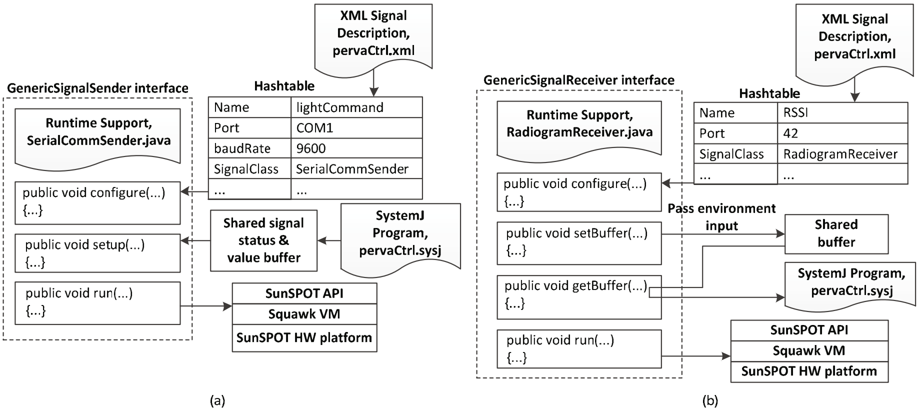

XML Signal Description, pervaCtrl.xml
GenericSignalSender interface
Hashtable
Name | lightCommand
Port | COM1
baudRate | 9600
SignalClass | SerialCommSender
... | ...
Runtime Support, SerialCommSender.java
public void configure(...) {...}
public void setup(...) {...}
public void run(...) {...}
Shared signal status & value buffer
SystemJ Program, pervaCtrl.sysj
SunSPOT API
Squawk VM
SunSPOT HW platform
(a)
XML Signal Description, pervaCtrl.xml
GenericSignalReceiver interface
Hashtable
Name | RSSI
Port | 42
SignalClass | RadiogramReceiver
... | ...
Runtime Support, RadiogramReceiver.java
public void configure(...) {...}
Pass environment input
public void setBuffer(...) {...}
Shared buffer
public void getBuffer(...) {...}
SystemJ Program, pervaCtrl.sysj
public void run(...) {...}
SunSPOT API
Squawk VM
SunSPOT HW platform
(b)

Fig. 4. Runtime support interface for (a) an output signal and (b) an input signal.

In order to port and execute a SystemJ program on the SunSPOT, an RTS, which maps physical I/O signals and wireless communication interface of the SunSPOT into SystemJ interface signals, needs to be implemented. SystemJ interface signals can perform either sending or receiving operation. Each type of signal needs to implement its corresponding Java interface, namely GenericSignalSender or GenericSignalReceiver, for sending and receiving operations, respectively. An output signal implements the GenericSignalSender class, which contains three methods, configure(), setup() and run(), as shown in Fig. 4(a). The configure() method initializes signal attributes from the stored Hashtable, which is generated by parsing the XML file. The setup() method defines a shared buffer object to hold the signal status and value emitted by the SystemJ program. The run() method implements the interface with the underlying physical platform by calling the corresponding SunSPOT API (or other necessary software routines depending on the execution platform). The run() method performs the operation to emit the signals' status and value, if needed.

Referring to Fig. 4(b), input signals are required to implement GenericSignalReceiver class, which consists of four methods, configure(), setBuffer(), getBuffer() and run(). Unlike to the sender routine, the external event (e.g. physical sensor value) is stored to the shared buffer object by calling setBuffer() within the run() method, which establishes the link between RTS and the external environment. The getBuffer() method provides the interface between RTS and SystemJ program to access the shared buffer object (i.e. the signal).

The RTS for signals necessary for the implementation of location-based services are explained in more details in the following subsections. Through the use of the RTS, different physical signals are converted into SystemJ signals and handled using standard SystemJ statements. The same SystemJ program can be executed on different platforms without any change, by providing the required XML file and RTS.

Listing 4. RTS for RSSI input signal

```
1     public void run() {
2       …
3       while (true) {
4         try {
5           …
6           if (packetType == RADIO_TEST_PACKET) {
7             Object buffer = new Double(rdg.getRssi());
8             Vector list = new Vector();
9             list.addElement(new Boolean(true));
10            list.addElement(buffer);
11            setBuffer(list);
12          }
13          …
14    }}}
```

5.1 Radiogram Receiver Signal (RSSI)

In the location-based lighting control system detailed in Section 4, the mobile SunSPOT node carried by a user broadcasts test radiogram packets periodically. The RSSI information can be retrieved through the received radiogram packets using the SunSPOT API. Referring to Fig. 4(b), the configure() method retrieves the signal attributes provided by the XML file, such as the incoming port number of radiogram packets. Referring to Listing 4, the run() method retrieves the RSSI information (line 7), encapsulates and stores the information to the shared buffer object through setBuffer() method (line 8-11). The *RSSI* signal is made accessible through the getBuffer() method.

Listing 5. RTS for SerialComm output signal

```
1     public void run() {
2       …
3       while (true) {
4         …
5         try {
6           commPort = selectedPortIdentifier.open(…);
7           serialPort = (SerialPort)commPort;
8           serialPort.setSerialPortParams(…);
9         }
10        catch (Exception e) {… }
11        output = serialPort.getOutputStream();
12        try {
13          output.write(lightCommand);
14          …
15        }
16        catch (Exception e) {… }
17    }}
```

5.2 SerialComm Sender Signal

In order to provide services based on the detected user location, output interface signal is necessary to perform actuation via analogue or digital interfaces such as PWM or serial buses. In this paper, RS-232 based serial communication is incorporated as an output signal to issue commands to a lighting circuit. The configure() method for an output signal retrieves signal attributes such as serial port number and baud rate for RS-232 communication. The setup() method initializes the shared buffer object. The run() method retrieves the lighting control command from the shared buffer and sends it over the serial port (as shown in Listing 5, line 5-17).

5.3 UDP Communication Signal

The UDP communication signal abstracts the 6LoWPAN-based UDP protocol, supported by the SunSPOT platform. This signal allows a SunSPOT node to establish communication links with a remote PC or other SunSPOT nodes. Unlike the other two signals, a UDP signal can be either an input or an output signal, and hence, both GenericSignalReceiver and GenericSignalSender need to be implemented. The main difference of UDP signal RTS is in the run() method, which tries to establish UDP connection for incoming and outgoing messages, rather than establishing radiogram or serial connections.

6 Conclusions and Future Works

In the paper we demonstrated how a new system-level design approach can be used to implement context-aware pervasive applications, based on the Java-enabled SunSPOT wireless sensor and actuator nodes. The user location can be detected via RSSI information broadcasted by a mobile SunSPOT to the other four fixed SunSPOT nodes. Trilateration-based algorithm is used to estimate the user location and automated lighting service can be provided via serial communication interface.

The entire system is designed using SystemJ, which is a concurrent system-level programming language based on GALS MoC and is capable of modeling and implementing complex reactive and concurrent systems. Physical sensing, controlling and communication interfaces, such RSSI, RS-232 and UDP are encapsulated in the SystemJ RTS to translate physical signals into SystemJ abstract signals. The functionalities of distributed pervasive systems can be easily modeled and implemented using SystemJ and without complex middleware. The designed system can also be ported to other Java-enabled platforms with the support of RTS. RTS for additional physical signals such as light intensity, temperature, motor actuation and TCP communication are being implemented as a part of standard RTS for SunSPOT. A more complex pervasive system will be targeted in the near future.

References

1. Cyber-Physical Systems Summit Report, CPS Summit: Holistic Approaches to Cyber-Physical Integration (April 2008), `http://varma.ece.cmu.edu/summit/CPS_Summit_Report.pdf` (acquired)
2. Vermesan, O., et al.: Internet of Things: Strategic Research Roadmap (2011), `http://www.internet-of-things-research.eu/pdf/IoT_Cluster_Strategic_Research_Agenda_2011.pdf` (acquired)
3. Smith, R.B.: SPOTWorld and the SunSPOT. In: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, Cambridge, Massachusetts, USA (2007)
4. Shelby, Z., Bormann, C.: 6LoWPAN: The Wireless Embedded Internet. Wiley (2009)
5. Malik, A., Salcic, Z., Chong, C., Javed, S.: System-level approach to design of a smart distributed surveillance system using System. ACM Transaction on Embedded Computing Systems (July 2011) (accepted for publication)
6. Chapiro, D.M.: Globally-Asynchronous Locally-synchronous systems. Ph.D Thesis-Computer Science. Stanford University (1984)
7. Berry, G., Gonthier, G.: The ESTEREL synchronous programming language: Design, semantics, implementation. Sci. Comput. Program 19(2), 87–152 (1992)
8. Wang, J., et al.: A study on wireless sensor network based indoor positioning systems for context-aware applications. Wireless Communications and Mobile Computing 12, 53–70 (2012)
9. Alemdar, H., Ersoy, C.: Wireless sensor networks for healthcare: A survey. Computer Networks 54, 2688–2710 (2010)
10. Marco, A., et al.: Location-based services for elderly and disabled people. Computer Communication 31, 1055–1066 (2008)
11. Flora, C., et al.: Indoor and outdoor location based services for portable devices. In: The Proceedings of the 25th International Conference on Distributed Computing Systems Workshops (2005)
12. Wood, A., et al.: Context-Aware Wireless Sensor Networks for Assisted-Living and Residential Monitoring. IEEE Network 22(4), 26–33 (2008)
13. Wang, K.I.-K., Abdulla, W., Salcic, Z.: AmI platform using multi-agent system and mobile ubiquitous hardware. PMC 5(5), 558–573 (2009)

Architecture of a Context Aware Framework for Automated Mobile Device Configuration

Md. Fazla Rabbi Opu, Emon Biswas, Mansura Habiba, and Cheonshik Kim

Samsung Bangladesh R &D Center (SBRC), Dhaka, Bangladesh
{fazla.rabbi,mansura.m}@samsung.com, emon.cse@samsung.com,
mipspan@mipsan@paran.com

Abstract. In this paper an architecture for context aware framework for mobile phones has been represented. The main goal of this proposed architecture is to enable phones to configure themselves based on self-initiated decisions which are directed by surrounding. This framework will reduce user's manual settings configuration. This will increase the flexibility level to great extent. Finally using the proposed framework phone will be self-configured. Most of the existing work regarding context aware mobile application is mainly based on location. However in this paper we have included a variety of context aware information along with inferred activities e.g. driving, running, meeting etc. in which users are engaged. In addition power management and memory management are two most important limitations of mobile devices. In this paper these two limitations are improved to a significant stage.

1 Introduction

Although smart phones are smart enough to provide several services to its owner, still user need to perform a lot of configuration and settings in order to get different services. Sometimes, this configuration has too much limited scope to provide user actual facility. For example, if user forgets to add reminder for any event appropriately, the device cannot remind him by itself about an important event. Therefore, mobile phones are still lacking of automated and self governing features. In this regard, the settings of any smart mobile phone have so many options to configure in order to provide user more services with flexibility. In spite of phones are becoming smarter day by day, users are being burdened with so many configuration options. However, a phone should be configured by itself. In this regard, user can manipulate phone's configuration if necessary but phone should be pre configured and help user to get rid of extra burden of configuring each and every settings option. In order to make smart phone self governed, phone should be able to recognize different context parameter such as noise, temperature, location, weather, speed/motion and brightness etc. based on the information extract from different context parameter phone will change its configuration. At the same time phone needs to save context parameter as history for adoption learning process as well as for future use. The main goal of future mobile phone is to be autonomic and to behave

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 299–308, 2013.

dynamically with the change of context. In order to reach this goal the main obstacles are: (1) there is no generic framework infrastructure to develop context aware application, (2) recognition of context parameter and extract information through calculation from context parameters is associated with energy cost, (3) calculation and information needs time, (4) lots of data need to be saved for further requirement and (5) context for smart phone is very dynamic and uncertain. This research has seven objectives to deal with, they are: (1) demonstrating unsupervised machine learning approaches in order to help mobile devices in identifying context (2) designing framework to extract context information in cost of minimum power and memory space (3) developing architecture framework to provide partially autonomic services which can be manipulate later by users (4) saving context information for future use (5) designing architecture framework to change phone configuration dynamically with the change in context (6) designing architecture framework to interpret higher level context information to lower level of information and (7) designing mechanism to share context information saved as history among mobile devices.

2 Literature Review

L. Baltrunus et al [3] has proposed a music recommendation in a car based on the traffic conditions, weather, driver's mood and way of driving. They have measured the impact of context information on the user's decision and recommended songs for driver. This calculation is static as well as they have used some defined parameters which are limited such as user can be driving in relax or in sporty. This kind of limited parameter is not always sufficient to take a decision. In the proposed paper dynamicity of context with the change of time has been considered as the main driving factor. Therefore the proposed framework can take decisions more dynamically than all existing framework. Y. Xiao et al [2] has proposed a cloud assisted based context aware power management system; this system can reduce power consumption by dynamically changing wi-fi access point with best signal. Moreover it takes help from cloud services to reduce power consumption by monitoring data download and dynamically off / on downloading. In the proposed system, instead of using continuous cloud services as well as keep running all services always, run-service-as-you- need method is applied. Therefore in the proposed framework no unnecessary service is running at any moment. In addition, all idle services for a certain period of time go off automatically. These existing context aware mobile frameworks focus on mainly location and provide different location based services. However our main goal is to deal with all possible context aware parameters and to change the phone settings dynamically.

3 Overview of Proposed Framework

The main design principles of proposed architecture are (1) Autonomy, (2) Dynamic (3) Re-configurability (4) Scalability (5) Extensibility (6) Personalization (7) Privacy and (8) on-site and off-site data storage.

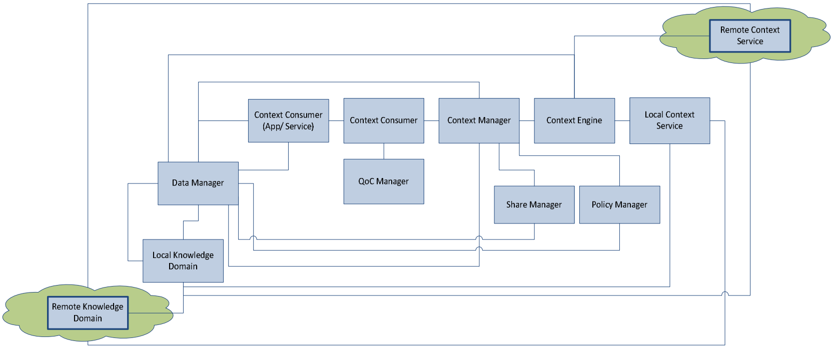


Fig. 1. Proposed Context aware framework Abstract level

Figure 1 depicts the proposed framework for agent based context aware mobile platform. Four agents have been used in this framework e.g. Context Consumer, Context Manager, QoC Manager and Data Manager. Context services typically reside in cloud and delivers context information to application or services of smart phone. In this framework, along with cloud based context services, some local context services also have been considered. All types of context information are extracted, simplified and categorized with the help of Context Engine. Later simplified context aware information is saved in either remote knowledge domain or local knowledge domain by the help of Data Manager. The components of proposed framework architecture are described as following

3.1 Context Consumer

In our proposed framework, any kind of mobile application or services are the Context Consumer (CC) who is in charge of gathering context information and put those in a form. CC needs different type of context information in order to adopt with the changing surrounding and react accordingly. There are 5 general WH questions in order to decide the requirement of CC regarding context information which is as (1) Who is the client, i.e application, service, web service or settings of the phone? (2) Where the client resides? (3)When client ask for context information and When that is extracted? (4)What is the activity of the client at the moment of asking for information i.e sensor activity? (5) Why this information is being invoked? (6) What is the device is being used to invoke information such as smart phone, PDA, Tablet, laptop or smart TV? (7) What are the auxiliary context parameters need to be looked for such as wind speed, temperature, rain measurement, traffic, route, places etc.? (8) What kind of security and privacy related policies are incorporated with? and (9) Is there any preference set prior to invoking the service by client? If yes then what are those preferences?

3.2 Context Manager

In this proposed framework, CC usually request for context information to Context Manager (CM) who runs a discovery service in order to find out the appropriate Context Owner (CO). As soon as the proper CO is found, CM binds the CC with the CO. Another major role of CM is to communicate Data Manager (DM) in order to retrieve and store context data. CM is also responsible for assigning CO to stored context information. CM sends both pull data and update data request to DM. On the response of request from CM, DM pulls data from remote or local knowledge domain or updates data in both knowledge domains.

3.3 Context Owner

Each manager has a particular number of CO defined by the framework based on different type of context information. Appropriate CO is selected by CM for CC based on the preference of CC. CO is responsible to store different remote context services of its type with corresponding Quality of Service (QoS) and Quality of Context (QoC). Furthermore, when CM binds CO to appropriate CC, CO is in charge of selecting correct context service that fits the requested QoS and QoC of CC.

3.4 QoC Manager

Each piece of context information needs to have a slandered QoC. In this proposed framework several QoC indicators are considered i.e.: precision, freshness, temporal resolution, spatial resolution, probability of correctness, probability to change, degree of dependency on other context information and frequency of use. QoC Manager (QM) is in charge of measuring the QoC of the context information simplified by context engine. After QoC measurement corresponding results are assigned with proper category by CM.

3.5 Context Engine

Context Parameter Recognition Mechanism: Figure 2 depicts the mechanism used in this proposed framework in order to detect context parameter. At first Read-context service runs using sensor periodically and collect context those are viable to change according to time. Then context parameter (CP) is extracted from context information. Next step is to classify all CP and categorize them into different classes. At first context aware information are classified into higher level and later they are classified into lower and unit CP. For example, at first stage acceleration is measured as High level Context Parameter (HCP) and later it is defined whether user is running or walking or driving as the Lower level Context Parameter (LCP). Once LCP is measured and stored, some uncertain CP are assumed and more details about LCP are gathered from environment. Finally all details about CP are saved.

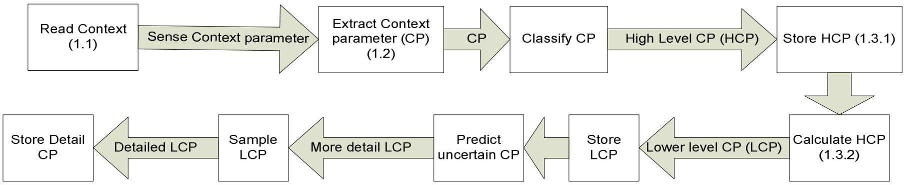


Fig. 2. Context parameter Recognition Mechanism

Context Parameter Inference Mechanism: Different kinds of CP are extracted from cloud services or sensors reside in phone. For example, location is the most changed and expensive parameter for mobile devices in terms of power consumption. Therefore we have defined the location inference mechanism to save extra power consumption in following two different ways:

GPS Tracing

In this technique, phone will enable GPS Location Identification Service (LIS) periodically to identify the current location. Another thing is as phone is learned about some regular location of its along with duration such as 9 hours at office in weekdays, 9 hours at home. Therefore phone can use LIS once in weekdays just to confirm whether it is in office during 9 hour working time. Similarly, after the time when usually user goes to sleep and phone remains idle, phone does not require LIS to run. In this way, the following rule limits energy consumption due to continuous using of LIS

- Periodical checking of LIS
- Phone should be learned about office time and sleeping time of user, so during office time and sleeping time only once LIS will run to confirm that the location is same.

Telecom Tracing

For LIS based services phone only need to know the current location. Telecom operators also informs user if he or she changed location and transferred to the inference location of another BSTI. The proposed framework can use this information along with GPS to detect current location. Using this feature, energy consumption will reduce to zero due to identify location. The proposed framework will also have manual re-configurability feature. So user can change his or her current location manually.

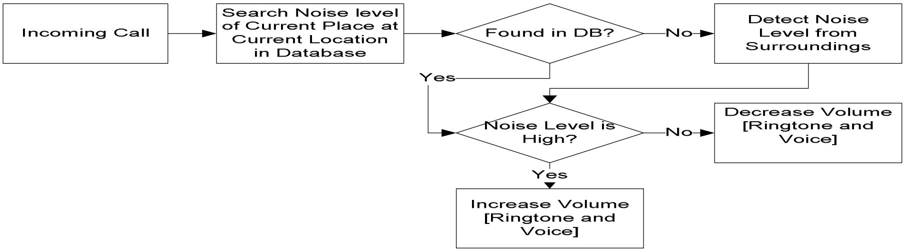

Incoming Call
Search Noise level of Current Place at Current Location in Database
Found in DB?
No
Detect Noise Level from Surroundings
Yes
Noise Level is High?
No
Decrease Volume [Ringtone and Voice]
Yes
Increase Volume [Ringtone and Voice]

Fig. 3. Noise Level Inference

Similarly there are some other CP such as noise, weather, temperature, driving mode and many other parameters can be detected using this proposed framework. Figure 3 describes the noise detection and inference procedure in this proposed framework.

3.6 Context Service

Both remote and local context services will be implemented for the proposed framework. They will be known as CAWS (context aware service) .Remote context services, typically residing in different clouds, deliver context information with various QoC and QoS. Sometimes existing remote context services are not enough and device need on site context information extraction. Therefore in this paper, local Context service is represented. Another reason of this novel idea is to overcome some outstanding limitations of mobile devices such as power limitation, continuous availability, dependency on infrastructure. Some context services are deployed locally; these services can be killed automatically if they are no longer required or are kept idle for certain amount of time and can be resumed when required.

3.7 Data Manager

Data storage, retrieval and updating as well as providing related context services corresponding updated data are the key role of DM. Data those are used occasionally are saved in remote data storage. In this regard, context information are of two type based on the frequency of use, Regular and Occasional. Occasional data are saved in remote knowledge domain while Regular data in local knowledge domain. In this way the phone memory has been saved from memory leakage problem.

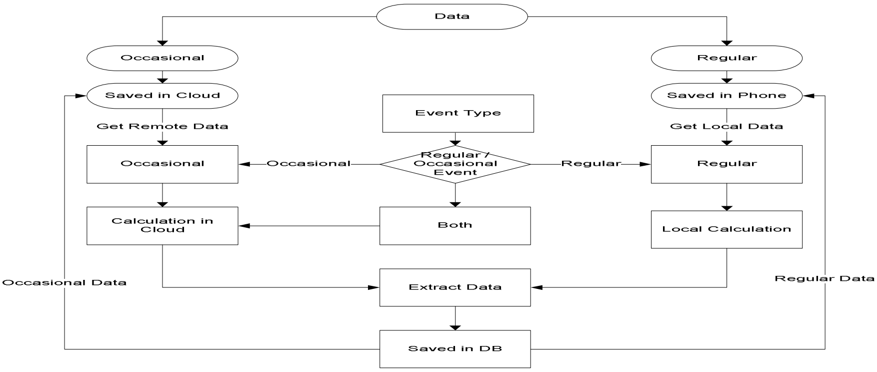


Fig. 4. Data Extraction and saving mechanism

Figure 4 describes how data are managed in phone memory and cloud as well. In this regard, data manipulation is less time consuming. As all searching, indexing and updating data services are resided in cloud for remote data. Only searching local data services resides in local phone. All other complex operation on data both local and remote is performed in cloud. Therefore all those services do not use phone memory or power for their actions. This way proposed framework can save memory and power.

3.8 Service Manager

Figure 5 describes how Service Manager (SM) can perform automated Configuration and manages changes

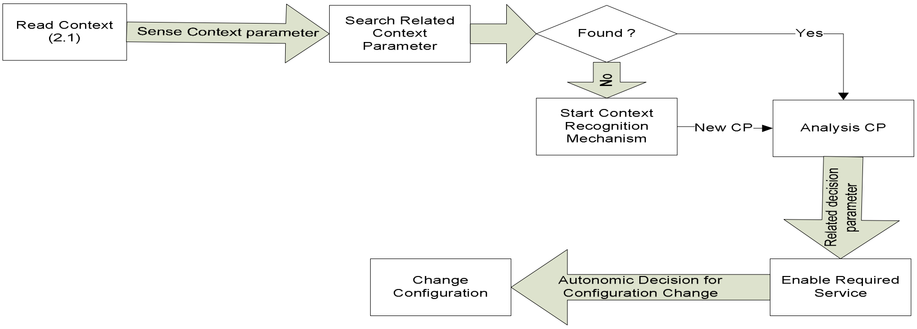

Read Context (2.1)
Sense Context parameter
Search Related Context Parameter
Found ?
Yes
No
Start Context Recognition Mechanism
New CP
Analysis CP
Related decision parameter
Enable Required Service
Autonomic Decision for Configuration Change
Change Configuration

Fig. 5. Autonomic Configuration Change Mechanism

3.9 Share Manager

Another important component of the proposed framework is Share Manager (ShM). This component is in charge to share context related information among devices through Bluetooth, Wi-Fi direct, Wi-Fi or even through internet.

3.10 Policy Manager

Security and privacy is one of the major concerns of smart phone users. Therefore, the proposed framework cannot avoid a Policy Manager (PM) who is responsible to define policies for security, privacy and collaboration among DM, ShM and CM. In case of binding CO and CC, CM asks PM to check whether they are compatible according to security as well as privacy. If there is any mismatch reported by PM, CM does not bind CO and CC in order to prevent privacy violation.

4 Scenarios of Using Proposed Framework

In this section some scenarios are described those will be benefitted once the proposed framework is implemented. Following are some scenario:

Scenario I: Bella gets a call in the middle of road where the surrounding is so noisy. Phone will automatically increase the volume level as soon as it detects the outer noise level.

Scenario II: Bella has prepared a shopping list and saved and she is passing a shopping mall, phone will reminder her if she wants to stop by shopping mall and complete her shopping. If she deletes that list, she can get that note according to the proposed framework as the same note is saved in cloud for years after years.

Scenario III: Bella has returned from a historical place, now one of her friends wants to visit the same place. Using the proposed framework, Bella can easily share the route she travelled, the hotel she stayed and the places she visited from her phone to her friend's phone. If she forgets the way to return back from where she started, the phone will guide her to trace the route.

Scenario IV: Bella unconsciously put her phone beside water or heater. If phone can detect the out temperature level is intolerable, it will ring to draw Bella's attention.

Scenario V: Bella wants to save all favorite phone numbers in her mobile phone. She does not need to do that manually. Based on the incoming and outgoing call frequency favorite numbers will be saved automatically.

5 Performance Analysis

5.1 Reduce Power Consumption

The proposed framework can reduce power consumption to a great extent. Periodically it will run a service for example after each two hour or less to detect whether there is any idle service as described in algorithm 1.

Algorithm 1. Detect idle service

```
1. For each two hours
    a. K = list of idle service
    b. For each service k in K
        i. If k is idle for T time T = true
        ii. If k has no dependent service P= true
        iii. If k will not be required for U time M = true
        iv. If (T ^ P ^ M )
            • Stop service k
```

Currently all service need to run for ever if user forget to stop that service after use and cost a lot amount of power. In this proposed framework idle service will no longer be running although user forgets to stop them or not according to algorithm 2.

Algorithm 2. Detect Location

```
1. For weekdays
    a. GPS will run automatically up to office start time and detect the current location
    b. If location is similar to other week days for last 15 days
    c. GPS service will be shut down automatically for q amount of time which is
       predefined and store the location, other services dependent on GPS system will
       use the saved location
    d. Else GPS service will run at every 1 hour
        i. Save current location Lcur and GPS service shutdown
        ii. If Lprev == Lcur
            • GPS service shutdown
        iii. Else update Lcur
        iv. All GPS dependent service will use Lcur as current location
```

Usually on weekdays we use to stay at office during day time and during night we use to stay at home. However the location is not being changed the running GPS location service in mobile phone still use to consume power. During 8 hours at office the location is office except lunch break, similarly at night 6-7 hours the location is home. In this system GPS system will not be running all day long.

Algorithm 3. Run GPS based Traffic System

```
1. Detect user is driving or running or walking or idle
2. If driving
      a. Run GPS Traffic Notification Service
      b. Change to auto answer mode through SMS or receive emergency
         call
3. Else Stop GPS Traffic Notification Service
```

Although algorithm 3 needs to run a service to detect the user activity, however it will be a local service and cost 67% less power than remote service GPS Traffic Notification. Figure 6 shows that the proposed framework can reduce power consumption around 69% than CasCap [2] and around 50% than the most popular smart phone S3. Moreover user need not to shutdown services from task manager of S3. The proposed framework will automatically detect unused services and shut them down.

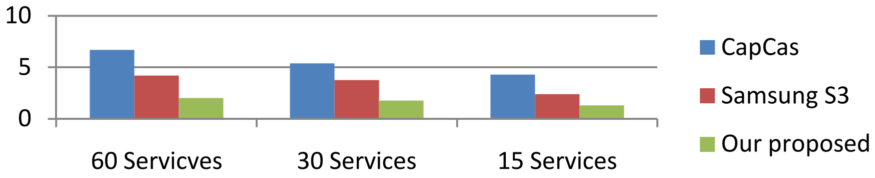


Fig. 6. Comparative power consumption analysis for proposed framework

The main objective of this proposed framework is to run services only when they are required and other time all unnecessary services will be stop state. Another main contribution of this proposed framework is user need not to configure phone about which service is required when, phone will take decision based on context aware knowledge domain.

5.2 Access Context Based Multimedia

W.Viana et al [1] describes their approach combines metadata extracted from the users' context along with annotations which is provided manually by the users and with annotations inferred by applying user-defined rules to context features. In this approach annotation is manual and multimedia needs to be saved in device. Moreover, user needs to pre-configure some rules and policies. However in this proposed framework all multimedia files will be categorized based on location, frequency of use, type, length and so many. User can automatically enjoy pictures of his current location and saved in his phone or in cloud. in this way managing media files will no longer need any manual configuration.

5.3 Better Reminder Service

This proposed framework can help user in many ways as an automatic reminder and user will need very limited configuration for that reminder service. For example, based on detecting how long user is running at Gym the proposed framework can remind him when to stop. Otherwise determining how long user is driving proposed framework can remind him what the remaining distance to destination is or what the fuel storage is and what the next fuel pump is. Conclusion and Future Discussion

This paper has presented a novel framework for the development and management of context aware services and applications for mobile devices to make those real smart and independent which is the ultimate requirement of future phone.. This framework will reduce user's interaction with their devices and take decision and configure itself based on context aware knowledge domain automatically, hence reduce user's initiatives. However as supervised learning is maintained in this proposed framework, still there will be options for manual configuration which will be reduced day by day as the learning grows mature and sufficient. We have designed a prototype system for this framework and demonstrated its effectiveness and several applications. We are now implementing the context inference service and learning mechanism for devices. We therefore planning to develop a complete set of services and applications based on this framework as our future work.

References

1. Viana, W., Miron, A., Moisuc, B., Gensel, J., Villanova-Oliver, M., Martin, H.: Towards the semantic and context-aware management of mobile multimedia. Multimedia Tools and Applications, ISSN: 1380-7501
2. Xiao, Y., Hui, P., Savolainen, P.: CasCap: Cloud-assisted Context-aware Power Management for Mobile Devices. In: Proceeding MCS 2011, Bethesda, Maryland, USA, June 28 (2011)
3. Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Aydin, A., Lüke, K.-H., Schwaiger, R.: InCarMusic: Context-Aware Music Recommendations in a Car. In: Huemer, C., Setzer, T. (eds.) EC-Web 2011. LNBIP, vol. 85, pp. 89–100. Springer, Heidelberg (2011)
4. Lu, C., Chang, M., Kinshuk, Huang, E., Chen, C.-W.: Usability of Context-Aware Mobile Education Game. Knowledge Managements and E- Learning: An International Journal 3(3), 448–477 (2011)
5. Sathan, D., Meetoo, A., Subramaniam, R.K.: Context Aware Lightweight Energy Efficient Framework 5(4), 249–255 (2010)
6. Buthpitiya, S., Luqman, F., Griss, M., Xing, B., Dey, A.K.: Hermes – A Context-Aware Application Development Framework and Toolkit for the Mobile Environment. In: 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA) (2012)
7. Chen, P., Sen, S., Pung, H.K., Xue, W., Wong, W.C.: A context management framework for context-aware applications in mobile spaces. International Journal of Pervasive Computing and Communications 8(2), 185–210 (2012)

AMM-PF: Additional Mobility Management Scheme Based on Pointer Forwarding in PMIPv6 Networks*

Seung Yoon Park, Jae Young Choi, and Jong Pil Jeong**

School of Information and Communication Engineering, Sungkyunkwan University, Korea
psy0624@skku.edu, jychoi@ece.skku.ac.kr, jpjeong@gmail.com

Abstract. In this paper, we propose additional mobility management schemes based on pointer forwarding for Proxy Mobile IPv6 (PMIPv6) networks with the aim of reducing the overall network traffic caused by mobility management and packet delivery. The proposed schemes are per-user-based, i.e., the optimal threshold of the forwarding chain length that minimizes the overall network traffic is dynamically determined for individual mobile user based on the mobility and service patterns. We show that there is an optimal threshold of the forwarding chain length given a set of parameters characterizing the specific mobility and service patterns of a mobile user. We also describe that our schemes yield significantly better performance than schemes that be applicable a static threshold to all mobile users. A comparative analysis shows that our pointer forwarding schemes outperform routing-based mobility management protocols for PMIPv6 networks.

Keywords: Mobility, PMIPv6, Pointer Forwarding, Dynamic Anchor.

1 Introduction

Mobile users want to access their personal files or the Web through their smartphones or tablet computers at any time and in any location, with the rapid growth of the internet industry as well as the increasing demand for mobile services. In order to communicate, all mobile devices must be configured with an IP address in accordance with the IP protocol and its addressing scheme. Problems occur when a user roams away from the device's home network and is no longer reachable using normal IP routing. This results in the active sessions of the device being terminated. A natural solution is to use IP layer mobility. Mobile IPv6 (MIPv6)[1] is the standard solution proposed by Internet Engineering Task Force (IETF) for handling terminal mobility among IP subnets. MIPv6 grants a Mobile Node (MN) to roam freely on the internet while still retaining the same IP address. However, MIPv6 requires additional stacks

[*] This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2012- 0006426)

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 309–316, 2013.

and signaling for the MN. This could add overhead such as a battery power and computational resource consumption.

A network mobility support mechanism is another way to solve mobility problems and to support IP mobility. This mechanism is called Proxy Mobile IPv6 (PMIPv6)[2] and is based on MIPv6. PMIPv6 enables IP mobility for a host without requiring its participation in any mobility-related signaling. There are two main components for the PMIPv6 networks: a Local Mobility Anchor (LMA) and a Mobile Access Gateway (MAG). The LMA carries out Home Agent (HA) roles, as defined in MIPv6, for the MN in the PMIPv6 networks domain. The LMA is the topological anchor point for the MN's Home Network Prefix (HNP), and it retains MN's binding state. The MAG carries out mobility-related signaling to the LMA for the MN and tracks the MN's movements. When an MN enters a PMIPv6 networks domain and attaches to an access link, the MAG retrieves the MN's profile using its current identifier. The MAG will then send a Proxy Binding Update (PBU) message to the LMA to register the current point of attachment of the MN. Accordingly, a Binding Cache Entry (BCE) and a tunnel for the MN's HNP will be created. The LMA then sends a Proxy Binding Acknowledgement (PBA) message with the MN's HNP. After receiving the Router Advertise (RA) message, the MN creates its IP address. For packet routing, the LMA will route all received packets over the established tunnel to the MAG. The MAG then forwards these packets to the MN. The MAG will then relay all the received packets over the tunnel to the LMA, and they will then be routed toward the Correspondent Node (CN). However, even with PMIPv6 networks the remote LMA signaling problem still remains unsolved. If an MN moves frequently within the PMIPv6 networks domain, the MAG incurs a high signaling cost in order to update the location of an MN to the LMA which is far from the MAG. This increases the network overhead on the LMA, wastes network resources, and lengthens the delay time. This problem becomes worse as the size of the PMIPv6 networks domain increasing. In this paper, therefore, we propose a pointer forwarding scheme[3] for minimizing signaling costs in PMIPv6 networks. Based on the analytic model, we formulate the location update cost and the packet delivery cost.

The remainder of this paper is organized as follows: Section 2 discusses previous PMIPv6 networks studies. Section 3 describes the proposed mobility management scheme using a mathematical model. Section 4 formulates signaling cost functions using the analytic model and the analysis of the results. Finally, conclusions are presented in Section 5.

2 Networks-Based Mobility Management Protocol

MIPv6 is designed to be a network-based mobility management protocol that does not require the handover[4] and signaling procedures associated with location registration. It thus has advantages that can reduce the load of MNs and mobility management latency. PMIPv6 networks adopts LMA and MAG as new components. The LMA, which acts an HA for an MN, manages all procedures in the PMIPv6 networks domain. The MAG, which is located between the LMA and an MN, carries out signaling procedures on behalf of the MN. In addition, MAG takes on routing function and the connection functions of a network for the MN. When an MN is connected to the networks, the MN attempts to access authentication and identification of the MN will be transferred to the MAG in this process. MAG gets its own profile using the identifier which can recognize MN after the authentication process with the Authentication, Authorization, Accounting (AAA) sever. The MAG then transfers the PBU message to the LMA then carries out the location registration process for the MN. If the LMA searches for information that is a relevant identifier of the MN in the BCE and if there is no information about the MN, LMA adds new information. The LMA then transfers the PBA message to the MAG and makes a bilateral tunnel. MAG that received the PBA message transfers the HNP allocated by the LMA and the RA massage containing the IP address information to the MN. Briefly, all messages are transferred through an LMA in PMIPv6 networks. Therefore, a MAG must carry out location registration at a distance, and this kind of location registration increases the load on the LMA. Furthermore, the total traffic that is transferred to the networks will be increased. In addition, lag time will increase based on location registration with increasing the LMA and the MAG distance.

3 AMM-PF in PMIPv6

3.1 Networks Architecture

A PMIPv6 networks consists of LMAs and MNs. LMAs are usually static and form the wireless mesh backbone of a PMIPv6 network. Some LMAs also serve as MAGs for MNs. One or more LMAs are connected to the internet and are responsible for relaying internet traffic to and from a PMIPv6 networks, and such LMAs are commonly referred to as MAGs. In this paper, we assume that a PMIPv6 networks has a single. In the proposed mobility management schemes, the central location database resides in the LMA. In a PMIPv6 network, there is an entry in location database storing the location information each roaming MN, i.e., the address of its Anchor MAG (AMAG). The MAG of an MN is the head of its forwarding chain. With the address of an MN's MAG, the MN can be reached by following the forwarding chain. Data packets sent to an MN will be routed to its current MAG first, which then forwards them to the MN by following the forwarding chain. Packet delivery in the proposed schemes simply relies on the routing protocol used. The concept of pointer forwarding[5] comes from mobility management schemes proposed for cellular networks. The idea behind pointer forwarding is minimizing the overall network signaling cost incurred by mobility management operations by reducing the number of expensive location update events. A location update event means sending a location update message to the LMA informing it to update the location database. With pointer forwarding, a location handoff simply involves setting up a forwarding pointer between two neighboring MAGs without having to trigger a location update event. The forwarding chain length of an MN significantly affects the network traffic cost incurred by mobility management and packet delivery, with respect to the MN. The longer the forwarding chain, the lower rate of the location update event, thus the smaller the signaling overhead. However, a long forwarding chain will increase the packet delivery cost because packets must travel a long distance to reach the destination. Therefore, there is a trade-off between the signaling costs incurred by mobility management versus the service cost incurred by packet delivery. Consequently, there exists an optimal threshold of the forwarding chain length for each MN. In the proposed schemes, this optimal threshold, denoted by K, is determined for each individual MN dynamically, based on the MN's specific mobility and service patterns. We use an MN parameter called the Service to Mobility Ratio (SMR) to MN's mobility and service patterns. For an MN with an average packet arrival rate denoted by λ_p and mobility rate denoted by σ, its SMR is formally defined as $SMR = \lambda_p/\sigma$.

As discussed in [5], internet traffic, i.e., the traffic between the MAGs and the LMA, dominates peer-to-peer traffic in PMIPv6 because PMIPv6 networks are expected to be a low cost solution for providing last-mile broadband internet access. Thus, we assume that for any MN, the internet session arrival rate is higher than the intranet session arrival rate and that the average duration of internet sessions is longer than that of intranet sessions. We use a parameter γ to signify the first assumption and another parameter δ to signify the second one. More specifically, γ denotes the ratio of the internet session arrival rate to the intranet session arrival rate, and δ denotes the ratio of the average duration of internet sessions to the average duration of intranet sessions. Also, we show that δ is also the ratio of the intranet session departure rate to the internet session departure rate, using the $M/M/\infty$ queue to model the process of session arrival at an MN.

3.2 Location Handoff

When an MN moves across the boundary of covering of two neighboring MAG areas, it de-associates from its old serving MAG and re-associates with the new MAG, thus incurring a location handoff. The MAG, it is newly associated with, becomes its current serving MAG. For each MN, if the length of its current forwarding chain is less than its specific threshold K, a new forwarding pointer will be set up between the old MAG and the new MAG during a location handoff. On the other hand, if the length of the MN's current forwarding chain has already reached its specific threshold K, a location handoff will trigger a location update. During a location update, the LMA is directed to update the location information of the MN in the location database by a location update message. The location update message is also sent to all active intranet MN correspondence nodes. After updating a location update, the forwarding chain is reset and the new MAG becomes the AMAG of the MN.

4 Performance Analysis

The Stochastic Petri Nets[6] model essentially captures the behaviors of an MN while it is moving within PMIPv6 networks. The places and transitions defined in the SPN model are given in Table 1. Here, we briefly describe how the SPN model is constructed.

Table 1. The Parameters and Notations used in Performance Modeling and Analysis

Parameter	Notation
σ	Mobility rate
λ_I/μ_I	Internet session arrival / departure rate
λ_L/μ_L	Intranet session arrival / departure rate
λ_{pu}	Average uplink (outgoing) packet arrival rate of internet sessions
λ_{pd}	Average downlink packet arrival rate of internet sessions
λ_{pL}	Average packet arrival rate of intranet sessions
N_{pI}	Average number of downlink (incoming) packets per internet session
N_{pL}	Average number of incoming packets per intranet session
N	Number of MAGs in PMIPv6
N_I	Instantaneous average number of active internet correspondence nodes per MN
N_L	Instantaneous average number of active intranet correspondence nodes per MN
α	Average distance (number of hops) between the LMA and an arbitrary MAG
β	Average distance (number of hops) between two arbitrary MAGs
γ	Ratio of the internet session arrival rate to the intranet session arrival rate
δ	Ratio of the average duration of internet sessions to the one of intranet sessions
ζ	Ratio of the downlink packet arrival rate to the uplink packet arrival rate of internet sessions
τ	One-hop communication latency between two neighboring MAGs
P_f	Probability that an MN moves forward
P_b	Probability that an MN moves backward
MinInt	Minimum cost of delay request after receiving message in PMIPv6
MaxInt	Maximum cost of delay request after receiving message in PMIPv6

4.1 Cost Modeling

The transitions forward and backward are associated with probabilities P_f and P_b, respectively. These probabilities depend on the network coverage model and the assumed mobility model. In this paper, we assume the hexagonal-grid mesh network model for PMIPv6 networks and the random walk model[7] for MNs. For the square-grid mesh network model, we assume that all MAGs have the same wireless range that covers the directly neighboring MAGs located in the four orthogonal directions.

Additionally, we consider a relatively large wireless mesh network simulated by a wraparound structure such that each MAG has four direct neighbors. Under these

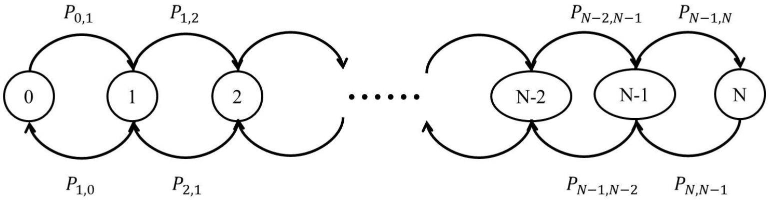


Fig. 1. State Diagram for the Random-walk Model

models, an MN can move randomly from the current MAG to one of the MAG's four neighbors with equal probability, i.e., 1/4. Thus, P_f and P_b are as shown in Figure 1 and calculated by

$$P_f = P_{r,r+1} = \begin{cases} 1-q & if\ r = 0 \\ (1-q)\left(\frac{1}{3}+\frac{1}{6}r\right) & if\ 1 \le r \le n \end{cases}$$

$$P_f = P_{r,r-1} = (1-q)\left(\frac{1}{3}-\frac{1}{6}r\right) \quad (1)$$

We use the total communication cost incurred per time unit as the metrics for performance evaluation and analysis. The total communication cost includes the signaling cost of the location handoff and update operations, the signaling cost of location search operations, and the packet delivery cost. For the static anchor scheme, the signaling cost of location search operations is incurred when a new intranet session is initiated with an MN. For the dynamic anchor scheme, the signaling cost of the location search operations represents the cost of tracking the current serving MAG of an MN and resetting the forwarding chain when new sessions are initiated with an MN. We use $C_{static}$and $C_{dynamic}$ to represent the total communication cost incurred per time unit by the static anchor scheme and dynamic anchor scheme, respectively. $C_{location}$, C_{search}, and $C_{transfer}$ represent the signaling cost of a location handoff operation, the signaling cost of a location search operation, and the cost to deliver a packet, respectively. Subscript *'I'* and *'L'* denote internet and intranet sessions, respectively. Subscript *'s'* and *'d'* denote the static anchor scheme and dynamic anchor scheme, respectively. For the static anchor scheme, the total communication cost incurred per time unit is calculated by

$$\begin{aligned} C_{static} = C_{location} \times \sigma + C_{search,L} \times \lambda_L \\ +C_{transfer,I} \times \lambda_{pd} + C_{transfer,L} \times \lambda_{pL} \end{aligned} \quad (2)$$

For the dynamic anchor scheme, the total communication cost incurred per time unit is calculated by

$$\begin{aligned} C_{dynamic} = C_{location} \times \sigma + C_{search,I} \times \lambda_I + C_{search,L} \times \lambda_L \\ +C_{transfer,I} \times \lambda_{pd} + C_{transfer,L} \times \lambda_{pL} \end{aligned} \quad (3)$$

The pointer forwarding scheme in PMIPv6 does not incur cost with $C_{search,I}$. However, there is a delay in movement detection. The MAG that supports mobility, which can transfer RA unrequested in movement detection delay[8], should define smaller MinInt(Min-Rtr-Adv) value and MaxInt(Max-Rtr-Adv) value. For simplicity, the average value[9] of the RA message unrequested in movement detection delay is used to define C_{find} and can be calculated by

$$C_{AMM-PF} = C_{location} \times \sigma + C_{transfer,I} \times \lambda_{pd} + C_{transfer,L} \times \lambda_{pL} + C_{find} \quad (4)$$

4.2 Numerical Result

In this section, we evaluate the performance of the proposed schemes, in terms of the total communication cost incurred per time unit. Additionally, we compare the proposed schemes with two baseline schemes. In the first baseline scheme, pointer forwarding is not used, meaning that every MN movement will trigger a location update event. Thus, it is essentially the same as having $K = 0$ in the proposed schemes. In the second baseline scheme, pointer forwarding is employed, but the same forwarding chain length threshold of K is preset for all MNs, e.g., $K = 4$ for all MNs. Table 2 lists the parameters and their default values used in the performance evaluation. The time is in seconds. All costs presented below are normalized with respect to $\tau = 1$.

Table 2. Parameter values for Performance evaluation

Parameter	Value	Parameter	Value	Parameter	Value
γ	10	δ	5	λ_I	1/600
μ_I	1/600	N_I	200	N_L	100
α	30	β	30	β	1000
τ	1	MinInt	7	MaxInt	30

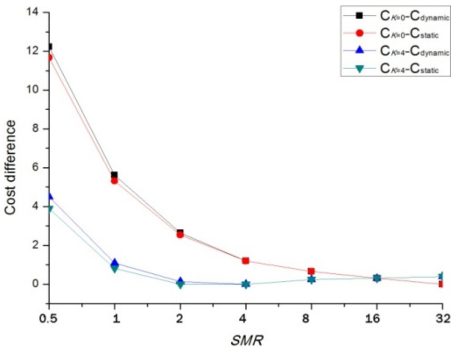


Fig. 2. Optimal ***K*** versus SMR

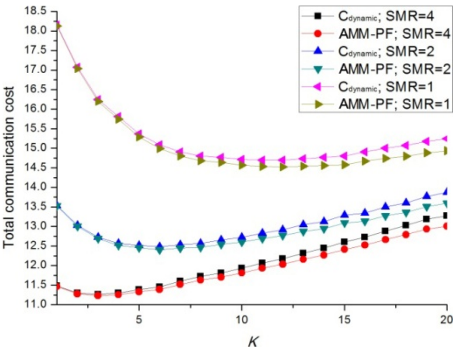


Fig. 3. Total communication cost versus ***K***

Figure 2 shows a plot of the optimal threshold K as a function of the SMR in both schemes. It can be observed that for both schemes, the optimal K decreases as the SMR increases. This is because the mobility rate decreases as the SMR increases, with fixed session arrival rates; thus, a short forwarding chain is favorable in order to reduce the service delivery cost. It is also interesting to note that the optimal K in the static anchor scheme is always smaller than or equal to that in the dynamic anchor scheme, due to the resetting of the forwarding chain of an MN upon a new session arrival in the dynamic anchor scheme.

Figure 3 shows the total communication cost as a function of K in both schemes, under different SMRs. As shown in the figure, there exists an optimal threshold K that results in a minimized total communication cost. For example, when SMR = 1, the optimal K is 11 for the static dynamic scheme, whereas it is 10 for the AMM-PF scheme. In addition, the total communication cost in both schemes decreases, as SMR increases. This is because given fixed session arrival rates, the mobility rate decreases as SMR increases, and thus the signaling cost incurred by location management as well as the total communication cost decreases. By comparing graphs in Figure 3, it can be seen that the dynamic anchor scheme and the proposed AMM-PF always shows excellent performance with the proposed method. A gentle curve accompanied by an increase in the value of K can be seen that indicates that it generates, and the more efficient entire communication costs.

5 Conclusion

PMIPv6 network is a network-based mobility management protocol to support mobility for IPv6 nodes without requirement for specialized software. In PMIPv6 networks, the MAG registers with the remote LMA when an MN moves frequently. This increases the network overhead of the LMA, wastes network resources, and lengthens the delay time. Therefore, we propose a new mobility management scheme for minimizing signaling costs using pointer forwarding. Our proposal can reduce signaling costs by registration with neighboring MAGs instead of the remote LMA though the use of pointer forwarding. Comparative Analysis of the static anchor scheme and the dynamic anchor scheme were performed with mathematical cost analysis, which showed that the performance of the proposed method is superior in terms of overall cost.

References

1. Johnson, D., Perkins, C.E., Akko, J.: Mobility Support in IPv6, IETF, RFC 3775 (2004)
2. Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., Patil, B.: Proxy Mobile IPv6, IETF, RFC 5213 (2008)
3. Jain, R., Lin, Y.-B., Lo, C., Mohan, S.: A forwarding strategy to reduce network impacts of PCS. In: Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1, pp. 481–489 (1995)
4. Park, C., Seong, H.: MIP-RA was used to transform the packet in HMIPv6 & MIH environment Efficient Handover. In: KAIS, vol. 1, pp. 251–254 (2011)
5. Nandiraju, N., Santhanam, L., He, B., Wang, J., Agrawal, D.: Wireless Mesh Networks: Current Challenges and Future Directions of Web-in-the-Sky. IEEE Wireless Comm. 14(4), 79–89 (2007)
6. Hirel, C., Tuffin, B., Trivedi, K.S.: SPNP: Stochastic Petri Nets. Version 6.0. In: Haverkort, B.R., Bohnenkamp, H.C., Smith, C.U. (eds.) TOOLS 2000. LNCS, vol. 1786, pp. 354–357. Springer, Heidelberg (2000)
7. Zang, L.J., Pierre, S.: Evaluating the Performance of Fast Handover for Hierarchical MIPv6 Cellual networks. Journal of Networks 3(6) (2008)
8. Perkins, C., Johnson, D.: Mobility(NEMO) Basic Suppot Protocol, RFC 3963 (2005)
9. Im, I., Cho, Y.-H., Choi, J.-Y., Jeong, J.: Security-Effective Fast Authentication Mechanism for Network Mobility in Proxy Mobile IPv6 Networks. In: Murgante, B., Gervasi, O., Misra, S., Nedjah, N., Rocha, A.M.A.C., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2012, Part IV. LNCS, vol. 7336, pp. 543–559. Springer, Heidelberg (2012)

The Evaluation and Optimization of 3-D Jacobi Iteration on a Stream Processor*

Ying Zhang, Gen Li, Yongjin Li, Caixia Sun, and Pingjing Lu

School of Computer, National University of Defense Technology, Changsha, 410073, China
{zhangying,genli,yjli,cxsun,pjl}@nudt.edu.cn

Abstract. Stream processors, with the stream programming model, have demonstrated significant performance advantages in the domains signal processing, multimedia and graphics applications, and are covering scientific applications. Jacobi iteration, which is widely used to solve partial differential equations, is an important class of scientific programs. As computers became more powerful, scientists have begun writing 3-D programs to solve PDEs. In this paper we examine the applicability of a stream processor to 3-D Jacobi iteration. In a stream processor system, the management of system resources is the programmers' responsibility. Compared with 2-D Jacobi iteration, some new issues must be considered, since reuse along the third dimension cannot fit in on-chip memory. We first map 3-D Jacobi iteration in FORTRAN version to the stream processor in a straightforward way. We then present several optimizations, which avail the stream program for 3-D Jacobi iteration, called StreamJacobi, of various aspects of the stream processor architecture. Finally, we analyze the performance of StreamJacobi, with different scales, and the presented optimizations. The final stream program StreamJacobi is from 2.43 to 11.48 times faster than the corresponding FORTRAN programs on a Xeon processor, with the optimizations playing an important role in realizing the performance improvement.

1 Introduction

Scientific computing plays an important role in the research and industry. Currently general purpose architecture processors cannot meet some of the demands of the scientific computing applications, including large amounts of bandwidth, large amounts of processing capability, low power and low price. Stream processors [1-2] have demonstrated significant performance advantages in media applications [3]. Many researchers are interested in the applicability of stream processors to scientific computing applications [4].

The stream processor architecture, which has many differences from the architecture of a conventional system, is designed to implement the stream programming model [5]. Although language implementations, such as streamC/kernelC [1], Brook, and Sequoia, exploit the model's features well, they do so at such a comparatively low-level; it is mainly the programmer's responsibility to manage system resources. Moreover,

* This work was supported by NSFC (61003075, 61103193, 61103011, 61103014).

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 317–325, 2013.

compared to other stream applications, such as media applications, scientific computing applications have more complex data traces and stronger data dependence. Therefore, writing a high-performance scientific stream program is rather hard and important to get right and high performance.

Jacobi iteration, using finite differencing techniques, is a popular and simple solver of partial differential equations (PDEs), an important class of scientific programs. As computers became more powerful, scientists have begun writing 3-D programs to solve PDEs. In order to examine the application of the stream processor to scientific programs and the utilization of stream processor features, this paper maps and optimizes the representative scientific application 3-D Jacobi iteration, to a stream processor in StreamC/KernelC.

As 3D stencil codes become widespread, numerical analysts discover that they have particularly poor memory behavior because accesses to the same data are usually too far apart, requiring array elements to be brought into on-chip memory multiple times per array sweep.

Fig. 1 presents the code for 2-D Jacobi iteration. Such a solver is also called *stencil* codes because they compute values using neighboring array elements in a fixed stencil pattern. This stencil pattern of data accesses is then repeated for each element of the array. The Jacobi iteration kernel consists of a simple 4-point stencil in two dimensions, shown in the first part of Fig. 2. On each loop iteration, four elements of the array are accessed in the 4-point diamond stencil pattern shown on the left. As the computation progresses, the stencil pattern is repeatedly applied to array elements in the column, sweeping through the array, as shown in the second part of Fig. 2.

```
A(N,N), B(N,N)
do J=2,N-1, I=2,N-1
  A(I,J)=
  C*(B(I-1,J)+
  B(I+1,J)+
  B(I,J-1)+
  B(I,J+1))
```

Fig. 1. Code for 2-D Jacobi iteration

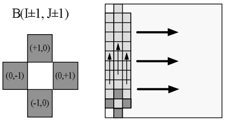


Fig. 2. Data traces for 2D Jacobi

```
A(N,N,N), B(N,N,N)
do K,J,I=2,N-1
  A(I,J,K) =
    C*(B(I-1,J,K)+
    B(I+1,J,K)+
    B(I,J-1,K)+
    B(I,J+1,K)+
    B(I,J,K-1)+
    B(I,J,K+1))
```

Fig. 3. Code for 3-D Jacobi iteration

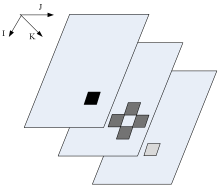


Fig. 4. Access pattern for 3D Jacobi

In comparison, the 3D Jacobi kernel, has a 6-point stencil which accesses six columns of B in three adjacent planes at the same time. With a distance of $2N^2$ between the leading B(I,J,K+1) and trailing B(I,J,K-1) array references, two entire $N \times N$ planes now need to remain in on-chip memory to exploit the reuse.

In this paper, we use the language streamC/kernelC [1] to map FORTRAN version of 3-D Jacobi iteration to the stream processor. A straightforward mapping method is first given to map 3-D Jacobi iteration to the stream processor; optimizations are then proposed to improve the overall performance of the mapped stream program. Finally, the performance of the stream program, called *StreamJacobi*, and the effectiveness of our optimizations are measured through a number of experiments. Compared with FORTRAN program on a Xeon, StreamJacobi finally achieves from 2.43 to 11.48 times speedup.

2 Background

The stream processor architecture is developed to speed up stream applications with intensive computations. The stream programming model divides a application into a stream-level program that specifies the high-level structure of the application and one or more kernels that define each processing step[6]. Each kernel is a function that operates on streams, sequences of records.

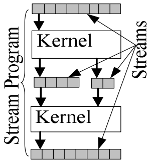


Fig. 5. Stream programming model

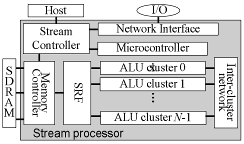


Fig. 6. A stream processor

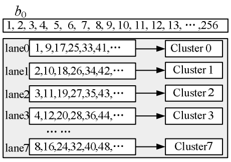


Fig. 7. Distribution of b0 among lanes

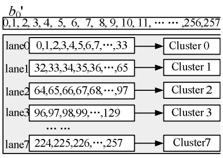


Fig. 8. Distribution of b0' with record reuse exploited

Popular languages implementing the stream programming model include StreamC/KernelC [1] for Imagine and Merrimac processor, Brook [7] for GPU and SF95 [4] for FT64 processor. These languages can also be used to develop stream programs for Cell Processors [8]. All these stream architectures have the characteristic of SIMD stream coprocessors with a large local memory for stream buffering. Fig. 4 shows a simplified diagram of such a stream processor. A stream-level program is run on the host while kernels are run on the stream processor. A single kernel that operates sequentially on records of streams is executed on clusters of ALUs, in a SIMD fashion. Only data in the local register files (LRFs), immediately adjacent to the arithmetic units, can be used by the clusters. Data passed to the LRFs is from the Stream Register File (SRF) that directly access memory. On-chip memory is used for application inputs, outputs and for intermediate streams that cannot fit in the SRF.

3 Straightforward Map and Optimizaiton

Straightforward Map

The implementation of mapping applications to the stream programming model can be thought of as a code transformation on programs that consist of a series of loops that process arrays of records. The data traces of different references in the innermost loop are extracted into different streams, and the computations performed by each loop are encapsulated inside a kernel. The remaining code composes the stream program.

When 3-D Jacobi iteration is mapped, the corresponding stream program declares five streams: four that correspond to the data accessed by four references to array B in I-loop and a fifth, a_0, that corresponds to the data accessed by the array *a* in I-loop.

After declaring the streams, the stream program then calls a kernel that processes the streams b_0, b_1, b_2 and b_3 to produce the stream a_0. The kernel is declared, taking four input streams ("istreams"), one output stream ("ostream") and one microcontroller variables as arguments. If first reads the values of the microcontroller variables, then loops over the records in the input streams computing records in the output stream.

Exploiting the Reuse of Record

The SRF is banked into lanes such that each lane supplies data only for its connected cluster. Records of a stream are interleaved among lanes. If a cluster needs data residing in another cluster, it gets the data by inter-cluster communication. Fig. 7 shows the distribution of data, accessed by the stream b_0 on j=0 iteration, among the lanes, with neighboring records residing in neighboring lanes.

In the FORTRAN code of 3-D Jacobi iteration, two neighboring elements, i.e. B(I+1, J, K) and B(I-1, J, K), of array B are involved in each iteration of I-loop. When mapping such loops, we have two choices as follow:

- Organize data as different streams to start and to end at different offsets into the original data arrays. The data covered by the two references, B(I-1,J, K) and B(I+1,J, K), in I-loop are organized into the streams b_0 and b_1 in referred order. Although the data in the records of every stream are almost the same, just displaced, all the streams must be loaded from off-chip memory.
- Organizing data covered by the two references in I-loop as a single stream. In this way, the streams b_0 and b_1 in *StreamJacobi* are merged into one stream $b((j+1)\times N, (j+2)\times N\text{-}1)$. However, during the kernel example execution, $cluster_i$ must communicate with $cluster_{i+2}$ to gather the needed record by inter-cluster communication, which causes clusters to stall while waiting for the data collection.

We reorganize the data distribution, with adjacent records distributed on the same lane. Thus, the optimized stream program has the same number of memory transfers with the second choice, but does not require any inter-cluster communication. The steps of exploiting the reuse of records for *StreamJacobi* are given below.

Step **1**. Organize all records covered by the two array references B(I+1, J, K) and B(I-1, J, K) as a new stream, with the same order as the array B.

Step **2**. Set the stride of the new stream be $Length_{stream}/N_{cluster}$ and the record length be $Length_{stream}/N_{cluster} + 2$, where $N_{cluster}$ is the number of clusters in the stream processor and $Length_{stream}$ is the length of the new stream. This means the original records from $i\times Length_{stream}/N_{cluster}$ to $(i + 1)\times Length_{stream}/N_{cluster} + 2$ in the new stream become the *ith* record of the new derived stream, b_0', residing in the *ith* cluster. The distribution of b_0' among lanes is shown in Fig. 8. Data distribution is transposed as shown, with neighboring records distributed on the same lane, such that $cluster_i$ gets neighboring records from the lane of itself without any inter-lane communication.

Step **3**. Divide all other stream references into $N_{cluster}$ parts by setting the stride be $Length_{stream}/N_{cluster}$ and the record length be $Length_{stream}/N_{cluster}$.

Step **4**. Update original kernel to process the records in the corresponding new order.

After the optimization, the final stream-level program, has little inter-cluster communication and less memory transfers.

In order to exploit stream reuse, the stream references in stream-level program are transformed with the same length, stride and record length, with redundant pad. Thus, the stream length, record length and stride of the streams b_2', b_3', b_4', and b_5' are changed as those of the stream b_0', with the changed streams named b_2'', b_3'', b_4'', and b_5''; correspondingly, the kernel is updated to process the correct records.

Exploiting the Reuse of Streams

The relationship among the locations, accessed by the streams b_0', b_2'', b_3'', b_4'', and b_5'' is described in Fig. 9. It is shown that the stream b_5'' on iteration j, the stream b_0' on iteration j+k×N, the stream b_2'' on iteration j+k×N-1, the stream b_3'' on iteration j+k×N+1 and the stream b_4'' on iteration j-k×N access the same locations. Since the values of the basic stream b are unchanged, the stream b_0' does not require accessing off-chip memory but accesses the SRF to get the values that are used by the stream b_5'' in N previous iterations. Similarly, b_2'' can access the SRF to get the values that are used by the stream b_5'' in N-1 previous iterations, b_3'' can access the SRF to get the values that are used by the stream b_5'' in N+1 previous iterations, and b_4'' can access the SRF to get the values that are used by the stream b_5'' in 2×N previous iterations.

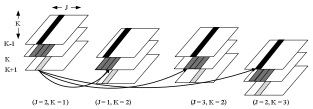


Fig. 9. Relationship among the locations accessed by b$_0$', b$_2$'', b$_3$'', b$_4$'', and b$_5$''

```
Stream<float> a(N*N*N),b(N*N*N);
Stream<float> a0', b5";
Stream<float> a00(T-2), b00(T), ..., b1T-1(T);
streamCopy(b(0, T),b00);
streamCopy(b(T, T),b01);
...
streamCopy(b(2*T*T,  2*T*T+1),b01);
for (jj=2; jj < N-1; jj = jj + T){
   for (ii=2; ii < N-1; ii = ii + T){
      for (k = 1; k < N-2; k++){
         for (j = jj; j < min(jj+T-1,N-1); j = j + 1){
         define recLen0, s0, recLen1, s1;
        //define stream references
            b5" = b((k+1)*T*T+(jj+j+1)*T+ii,
                          (k+1)*T*T+(jj+j+2)*T+ii,  recLen0, s0);
            a0' = a(k*T*T    +(jj+j+1)*T+1+ii,
                           k*T*T   +(jj+j+2)*T-1+ii,recLen1, s1);
            streamCopy(b5", b1T-1);
            //kernel call
            jacobi" (b11, b10,b12, b00, b1T-1, a0');

            //Rotate data between bij
            streamCopy(b01,b00);
            streamCopy(b02,b01);
            ... ...
            streamCopy(b1T-1,b1T-2);
} } } }
```

Fig. 10. Tiled 3D Jacobi iteration stream-level program

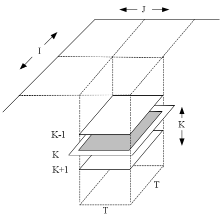


Fig. 11. Access pattern of tiled 3D Jacobi

However, stream compilers [1] cannot recognize and utilize the reuse supplied by the streams b_0', b_2'', b_3'', b_4'', and b_5''. This is because the start and end bound of these streams are variables, which means they are unknown when stream compilers allocate the SRF for streams. Therefore, the reuse supplied by these streams is all omitted by stream compilers

Stream Reuse Exploitation

We optimize it by introducing 2×N+1 basic streams, $b_{0,0}$, $b_{0,1}$, …, $b_{0,\text{N-1}}$, $b_{1,0}$, $b_{1,1}$, …, $b_{1,\text{N-1}}$ and a_{00}, initializing $b_{0,1}$, …, $b_{0,\text{N-1}}$, $b_{1,0}$, $b_{1,1}$, …, $b_{1,\text{N-2}}$ before the loop with the streams referenced by b_4" on iteration from 0 to 2×N-1, defining $b_{1,\mathrm{N}-1}$ to b_5" and a_0', loading the values referred to by b_4" to $b_{0,0}$, replacing the references b_0', b_2" b_3" and a_0' with basic streams $b_{1,1}$, $b_{1,0}$, $b_{1,2}$ and a_{00} respectively, saving the output a_{00} to the locations defined by a_0' and moving the values of $b_{0,j}$ to $b_{0,j-1}$, those of $b_{1,j}$ to $b_{1,j-1}$, and those of $b_{1,0}$ to $b_{0,N-1}$, at the end of the loop body. The function *streamCopy(s, t)* copies records of *s* to *t*. An SRF-to-memory copy generates a save of *s* to memory; a memory-to-SRF copy generates a load of *s* to the SRF; an SRF-to-SRF copy generates a save of *s* to memory and a load of *s* to the SRF buffer that holds *t*. We can effect the reuse by replacing streams that have unknown starts and ends with streams that have constant starts and ends, i.e. basic streams, and explicitly transferring original reuse to the reuse among streams with constant starts and ends. The stream compiler will recognize and utilize the reuse in the transformed code. However, it does so at the expense of introducing two expensive SRF-to-SRF data moves. Since these moves implement a permutation of values in the SRF, we can eliminate the need for moves by unrolling to the cycle length of the permutation, i.e. 2×N times, and permuting the stream references in each unrolled loop bodies. The stream compiler can capture reuse in the transformed stream program, thus efficiently reducing off-chip memory transfers.

Tiling the Stream Program

2xN+1 streams with the length of N, i.e. 2xNxN+N records totally, need to remain in the SRF for the stream processor simultaneously, in order to effect all the reuse. So only 3D Jacobi iteration of size less than 128×128×128 can fully exploit reuse for a 128KB SRF. Otherwise, data for stream B will need to be brought into the SRF two or more times each time the kernel is executed, reducing performance for larger problem sizes.

Tiling is a well-known transformation which improves locality by moving reuses to the same data closer in time. To exploit the reuse, we tile the J-loop and the stream length, i.e. I-loop in original FORTRAN code. First, J and the start record of each stream are strip-mined to form tile-controlling loops JJ and II. Next, JJ and II are permuted to the outermost level. Fig. 10 shows the tiled 3D Jacobi iteration stream-level program. Fig. 11 illustrates the stream access pattern of tiled 3D Jacobi iteration. After tiled, all the stream reuse is exploited, with only 2×T×T+T records remaining in the SRF.

Selecting Tile Sizes

Achieving reuse after applying our tiling transformation depends on the choice of tile dimensions T now. As discussed above, 2×T×T+T records need to remain in the SRF after tiled, in order to effect the reuse. Additionally, the stream of a_0', with the length of T-2, should also remain in the SRF in each iteration. Thus, 2×T×T+2×T-2 records need remain in the SRF if all the stream reuse is exploited. Let S_{rec} be the record size, and C_{SRF} be the SRF capacity. The max T that satisfies the following formula is the best tile size,

$$(2\times T\times T+2\times T-2)\times S_{rec}\leqslant C_{SRF}. \quad (1)$$

4 Experimental Setup and Performance Evaluation

In our experiments, we use Isim[2], a cycle-accurate stream processor simulator supplied by Stanford University, to get the performance of *StreamJacobi* with different scales and different versions. We perform experiments with four scales, 128×128, 256×256, 512×512, 1024×1024 and 4*K*×4*K*. The baseline configuration of the simulated stream processor and its memory system is detailed in table 1, and is used for all experiments unless noted otherwise. For comparison, 3-D Jacobi iteration kernel in FORTRAN is compiled by Intel's IA32 compiler (with max speed optimization option), and run on a Xeon processor, one class of the most popular machines used for scientific computing applications now. Table 2 shows the configuration of the Xeon processor.

Table 1. Baseline parameter of Isim

Parameter	Value
Number of clusters	8
Capacity of LRF	38.4KB
Capacity of SRF	512KB
Capacity of off-chip DRAM	4GB
Operating frequency	2GHz

Table 2. Xeon Configuration

Parameter	Value
Number of cores	8
Operating frequency	2.7GHz
L1 Cache	8×32K×2
L2 Cache	8×256K
L3	20M

Overall Performance

The performance of *StreamJacobi* with all optimizations is first presented to evaluate the stream processor's ability to process 3-D Jacobi. Fig. 12 shows the speedup yielded by *StreamJacobi*, with different scales, over FORTRAN 3-D Jacobi iteration. *StreamJacobi* yields from 2.43 to 11.48 times speedup, which indicates the stream processor can successfully process such class of scientific applications. This is because plenty of ALUs process computations in *StreamJacobi*; data reuse is all exploited; memory transfers are overlapped with kernel execution perfectly.

For our configuration of Isim, the 3-D Jacobi iteration with size of larger than 248 will be stripped for the exploitation of stream reuse; for the Xeon processor, the same data of the Jacobi iteration with size larger than 512 will be transferred to L2 cache multiple times, and that of the Jacobi iteration with size larger than 1.5*K* is transferred to L3 cache multiple times. Therefore, our mapping and optimization gets the highest speedup of 11.48 for the iteration with size of 4*K*, and a higher speedup for the iteration with size of 1024. For *StreamJacobi* with the size larger than 248 and smaller than 512, *StreamJacobi* loads some pad data in order to exploit the reuse. And therefore, they get relatively smaller speedup.

One key to achieving high performance on a stream processor is making kernel execution and memory transfers occur concurrently. Fig. 13 shows the time taken by *StreamJacobi* to execute kernels and to access memory, respectively (normalized to the total execution time). Due to the reuse of streams and the concurrent memory accesses, memory access delays become less in *StreamJacobi*. *StreamJacobi* is now bounded by both computing resources and memory access performance. In particular, *StreamJacobi* with size larger than 512 is bound by kernel execution now.

Effect of Exploiting the Reuse of Records

We now demonstrate the effectiveness of exploiting the reuse of records that reorganizes streams to reduce off-chip memory transfers. Fig. 14 demonstrates marginal speedup, due to the reduction of memory transfers. For *StreamJacobi* with 128×128 scale, this optimization reduces the number of loading streams, so reduces the preparing overheads, so gains largest speedup. But for the stream programs of the Jacobi application with other scales, their speedup is nearly the same.

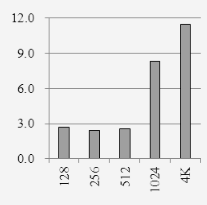


Fig. 12. Speedup of Isim over Xeon

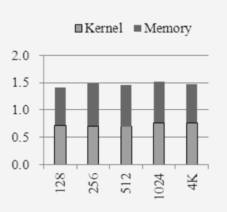


Fig. 13. Time distribution of kernel and memory access

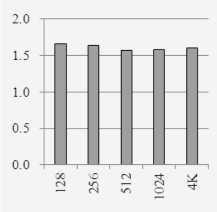


Fig. 14. Speedup of exploiting record reuse

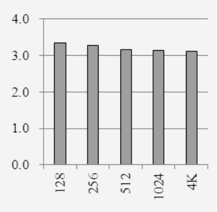


Fig. 15. Speedup of exploiting sreeam reuse

Effect of Exploiting the Reuse of Streams

As a stream processor reduces memory transfers only by capturing the reuse among streams in the SRF, this optimization is important. We evaluate the impact of exploiting the reuse of streams on program performance. *StreamJacobi* benefits greatly from this optimization. Fig. 15 demonstrates the speedup attained with this optimization.

Since five out of six input streams reuse the data generated on the previous iteration, a lot of memory transfers are reduced. Without the optimization, the stream compiler cannot identify the reuse, all input streams must be loaded from off-chip memory and each kernel must wait until its input streams are loaded from off-chip memory. Stream reuse removes the appearance of streams with unknown starts and ends, thus making the stream compiler able to identify reuse.

References

[1] Rixner, S.: Stream Processor Architecture. Kluwer Academic Publishers (2001)

[2] Kapasi, U., Dally, W., Rixner, S., Owens, J., Khailany, B.: The Imagine Stream Processor. In: Proceedings 2002 IEEE International Conference on Computer Design, pp. 282–288 (2002)

[3] Gordon, M., Maze, D., Amarasinghe, S., Thies, W., Karczmarek, M., Lin, J., Meli, A., Lamb, A., Leger, C., Wong, J., et al.: A stream compiler for communication-exposed architectures. ACM SIGARCH Computer Architecture News 30(5), 291–303 (2002)

[4] Fatica, M., Jameson, A., Alonso, J.: STREAMFLO: an Euler solver for streaming architectures. Submitted to AIAA Conference

[5] Kapasi, U., Rixner, S., Dally, W., Khailany, B., Ahn, J., Mattson, P., Owens, J.: Programmable Stream Processors. Computer 36(8), 54–62 (2003)

[6] Das, A., Dally, W.J., Mattson, P.: Compiling for stream processing. In: PACT 2006: Proceedings of the 15th International Conference on Parallel Architectures and Compilation Techniques, pp. 33–42. ACM Press, New York (2006)

[7] Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., Houston, M., Hanrahan, P.: Brook for gpus: stream computing on graphics hardware. ACM Trans. Graph. 23(3), 777–786 (2004)

[8] Kahle, J.A., Day, M.N., Hofstee, H.P., Johns, C.R., Maeurer, T.R., Shippy, D.: Introduction to the cellmultiprocessor. IBM J. Res. Dev. 49(4/5), 589–604 (2005)

DDASTM: Ensuring Conflict Serializability Efficiently in Distributed STM

Yu Zhang, Hai Jin, and Xiaofei Liao

Services Computing Technology and System Lab.
Cluster and Grid Computing Lab.
Huazhong University of Science and Technology, Wuhan, 430074, China
hjin@hust.edu.cn

Abstract. CS (Conflict Serializability) is a recently proposed relaxer correctness criterion that can increase transactional memory's parallelism. DDA (Distributed Dependency-Aware) model is currently proposed to implement CS in distributed STM (Software Transactional Memory) for the first time. However, its transactions detect conflicts individually via detecting cycles in PG (Precedence Graph) and cause extra runtime overhead, especially at the condition that the transactions access lots of objects or the PG is large. In this paper, we propose an approach to make each cycle in PG detected by those transactions, which construct this cycle, together in parallel way, instead of detecting cycle individually. Experimental results show that the average execution time and communication cost of all transactions, including aborted ones, in our approach, can be decreased to 76% and 78% of those in DDA respectively. Its speedup is up to 2.56× against baseDSTM, employing two-phase locking.

Keywords: Distributed software transactional memory, Conflict serializability, Parallelism.

1 Introduction

Transactional memory is a promising mechanism for simplifying shared memory parallel programming. Transactions improve performances of locks because they are easier to reason about, more composable, and in some cases they can provide progress guarantees without complex design. Distributed STM (*Software Transactional Memory*) is a natural fit for distributed memory system, such as partitioned global address space model. Yet, most current distributed STMs, such as Cluster-STM [1], DiSTM [2] and DecentSTM [3], still adopt 2PL (*two-phase locking*). It restarts or delays one transaction whenever two overlapping transactions access the same object and at least one access is a write. Furthermore, in distributed applications, long running transactions and the transactions needed to ensure the correctness of considerable data are also very common, such as transactions in operations of list, tree and graph. As a result, it has a very high abort rate, and bad performance of distributed applications.

Fortunately, DDA (*Distributed Dependency-Aware*) [4] model is proposed to leverage CS (*Conflict Serializability*) [5-8, 19] criterion in distributed STM for the first

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 326–335, 2013.

time, to relax the restriction of two-phase locking. It allows conflicting but conflict-serializable transactions to commit safely via maintaining a PG (*Precedence Graph*) of conflicting and uncommitted transactions and keeping it acyclic. However, whenever a transaction accesses a data object and inserts a dependency into PG, DDA needs to detect conflicts through detecting cycles in PG individually. Consequently, DDA induces much extra time to finish a transaction, especially at the condition that the transaction accesses lots of objects or the PG is complex for many active transactions and high contention.

In fact, to ensure CS, it is not necessary to make every transaction detect cycles in PG individually. It just needs every cycle to be detected respectively by one transaction in them and keeps PG of transactions acyclic. Hence, we can reduce the extra execution time, via making conflict-detect operations of transactions detect the cycle, constructed by those transactions, together in parallel way. The cycle will be detected out in self-loop by a transaction in it. To validate this approach, we implement it in a distributed STM, namely DDASTM. In summary, this paper proposes an approach that allows conflict-detect operations of transactions to detect the same cycle, which is constructed by these transactions, together in a parallel way, to spare extra execution time of transaction and also reduce communication cost. We present the details to efficiently employ our approach in distributed STM. Experimental results show that our approach can spare substantial execution time of transactions and communication cost against DDA, especially at the some conditions.

The rest of this paper is organized as follows. In section 2, the related work for transactional memory is surveyed. Section 3 presents our basic idea and challenges. Section 4 gives the experimental results of DDASTM and DDA. The conclusions are summarized in section 5.

2 Related Works

Currently, transactional memory has attracted much attention from researchers. They focus on STM [9], HTM [11, 13] and Hybrid TM [14, 15]. Some works [16, 17] try to optimize them to get good performance and scalability via relaxer correctness criterion and scheduler. Yet, most distributed STM systems still employ linearizability [18] to guarantee the correctness of concurrent transactions via emulating a technique known as 2PL. While easy to implement, this approach may lead to high abort rate, especially in situations with long-running transactions and contended shared objects.

Utku Aydonat [6] proposes to employ a relaxer correctness criterion, namely CS, to ensure serializability in transactional memory and guarantee the correctness of concurrent transactions. It does not impose any restrictions based the execution order of transactions, and performs more relaxed than linearizability and allows more concurrency. Some related works [5-8] implement CS in multicore environment and also demonstrate CS can reduce lots of spare aborts and performs better than 2PL, even though with more runtime overhead.

Bo Zhang [4] presents DDA and attempts to extend CS criterion to distributed STM. However, to accept any interleaving that is conflict-serializable, Hany E.

Ramadan [8] proves that it needs to maintain a PG of conflicting, uncommitted transactions, and keeps it acyclic. So, the key challenge to ensure conflict-serializable in distributed STM is cycle detection in a dynamic PG, which changes with the execution of transactions. Yet, DDA can not satisfy this requirement well, because it needs to detect cycles in PG whenever a transaction accesses a data object and inserts a dependency into PG. As a result, DDA has much redundant latency to finish a transaction, especially at the condition that the transaction accesses lots of objects or the PG is complex for many active transactions and high contention. Even though there are also many research papers about distributed cycle detection algorithm, all these algorithms [10] aim for deadlock and knots detection in distributed environment. None of them satisfy the requirements of cycle detection in the PG for distributed STM for bad performance or even can not correctly detecting out cycles. In this paper, we try to proposes an efficiently algorithm to detect out cycles on dynamic graph by several related operations together in parallel way.

3 Main Idea and Challenges of DDASTM

3.1 Main Idea

The main idea of our approach is inspired by the following inspection. To ensure CS, it just needs every cycle to be detected respectively by one transaction in them and keeps PG of transactions acyclic. Hence, when transactions are being committed, conflict-detect operations can gradually make nodes, representative of these transactions, out of PG correctly and compact the PG together in a parallel way, finally making each cycle detected in a self-loop by a transaction in it. Then extra execution time of transaction can be spared, and the communication cost to detect a cycle can be shared by transactions constructing this cycle. Note that to correctly compact a node T_i out of PG, we need to calculate an edge set $Dep(T_i)=\{<T_x, T_y>|T_x \in Pre(T_i) \wedge T_y \in Next(T_i)\}$ to substitute edge set $Pre(T_i)$ and $Next(T_i)$ in the edge set of PG, where $Pre(T_i)=\{T_x |<T_x, T_i> \in PG\}$ and $Next(T_i)=\{T_y |<T_i, T_y> \in PG\}$. Fig. 1 gives an example to describe this approach.

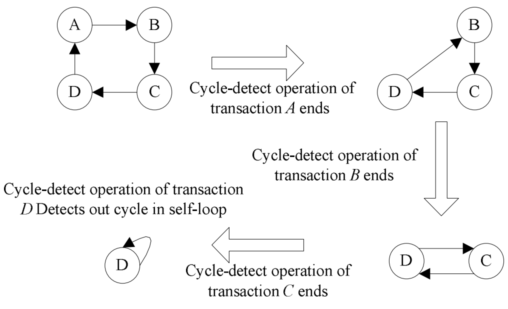
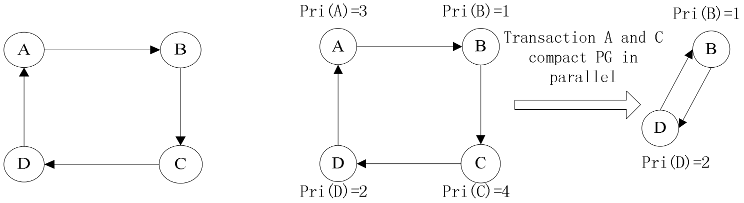

A
B
D
C
Cycle-detect operation of transaction *A* ends
B
D
C
Cycle-detect operation of transaction *B* ends
Cycle-detect operation of transaction *D* Detects out cycle in self-loop
D
Cycle-detect operation of transaction *C* ends
D
C

Fig. 1. Cycle-detection operation compacts PG



(a) Without priority (b) With priority

Fig. 2. Synchronization problem

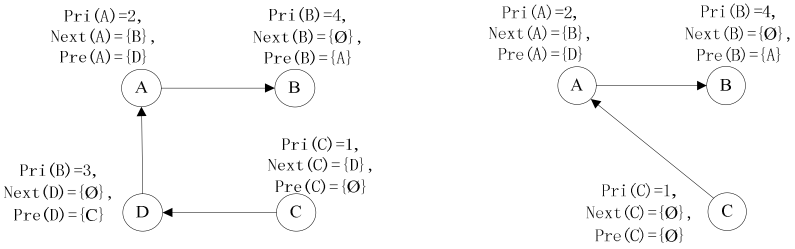
3.2 Challenges

In this part, we show the challenges to make operations compact the PG together in a parallel way. The details are shown as follows.

The first challenge is about synchronization. If compacting the PG in parallel way without correct synchronization, those operations will be unaware of other operations, which are also detecting the same cycle. These operations will detect the cycle individually instead of detecting the same cycle together, inducing much unnecessary latency and communication cost. For example, as the PG in Fig. 2(a), suppose the cycle-detect operation of transaction *A* tries to compact the PG, and needs to notify transaction *D* while the neighbor of *D* is not *A* anymore but *B*. Yet, cycle-detect operation of *D* likely proceeds in parallel with *A* and has left the node before the coming of notification. As a result, the operation of *D* is unaware of the change of its neighbor nodes. Then to correctly compact the PG, operation of *A* needs further communication to notify operation of *D*. It may be the same as operation of *D* as well. Then both operations may need further communication to correctly compact PG.

To eliminate such problem and correctly proceed in a parallel way, our approach gives every transaction a priority and makes their cycle-detect operations gradually compact the PG in an order, according to given priority of them and their neighbors. Cycle-detect operation of T_i needs to wait until operation of every transaction T_j in $Wait(T_i)$ has compacted indirect dependency among Pre_j and $Next_j$ into direct dependency Dep_j and notifies T_i. $Wait(T_i)=\{T_j|T_j\in Pre(T_i) \vee T_j \in Next(T_i)$, and $Pri(T_i)< Pri(T_j)\}$, where the priority of transaction T_x is $Pri(T_x)$, also the identity of T_x. Note that, we define the priority in this way just for simplicity and can define it in other ways as well. As in Fig. 2(b), because $Pri(A)>Pri(B)$ and $Pri(A)>Pri(D)$, in the first round, transaction *A* lets its cycle-detect operation compact indirect dependency between <*D*, *A*> and <*A*, *B*> into direct dependency <*D*, *B*>. Similarly, *C* proceeds in parallel with transaction *A*, and also compacts indirect dependency between <*B*, *C*> and <*C*, *D*> into direct dependency <*B*, *D*> in the first round. However, priorities of transaction *B* and *D* are lower than their neighbors respectively, thus they must wait and operation of *A* and *C* do not need further communication to notify the change of PG. As a result, the problem described above can be eliminated.

The second challenge is about consistency. The PG of transactions is dynamic and neighbors of each transaction are changing with the execution of transactions, inducing inconsistent neighbor information among transactions. Then some transactions can not correctly synchronize as discussed above, and even may wait the notification of neighbors forever. For example, as in Fig. 3(a), suppose transaction *A* accesses a data object and generates a dependency <*D*, *A*>, after transaction *D* has committed and collected all its information of neighbors. Transaction *A* makes *D* as its neighbor, but *D* is unaware of the existence of <*D*, *A*> and does not make *A* as its neighbor. If *pri*(*A*)<*Pri*(*D*), transaction *A* will wait the notification of *D*. As *D* is unaware of the existence of *A*, it will never notify and awaken *A*.



(a) *D* does not compacted itself

(b) *D* has compacted itself before the notification of *A*

Fig. 3. Information of neighbors inconsistent problem

To tackle this problem, we firstly need to know which transaction has inconsistent information of its neighbors, then make the information of this transaction consistent. Let us consider the above example. Because only the transaction *A* is aware of the existence of dependency <*D*, *A*> and knows that transaction *D* has collected information before the generation of such dependency. Hence, only transaction *A* knows the information of *D* is inconsistent, and can make the information of *D* consistent. So, after knowing the neighbor information of *D* is inconsistent, *A* should not wait for it, and just needs to find the current location of cycle-detect operation of *D* and makes its information consistent. Note that, as in Fig. 3(b), transaction *D* may has been compacted out of PG before the notification of *A*. Then the inconsistent neighbor information of *D* may be transferred to transaction *C*. Hence *A* needs further communication to make the information of *C* consistent. Moreover, *A* may need to locate the inconsistent information initially owned by *D* iteratively, because transaction *C* may have transferred this inconsistent information to other transactions.

4 Evaluation and Experimental Results

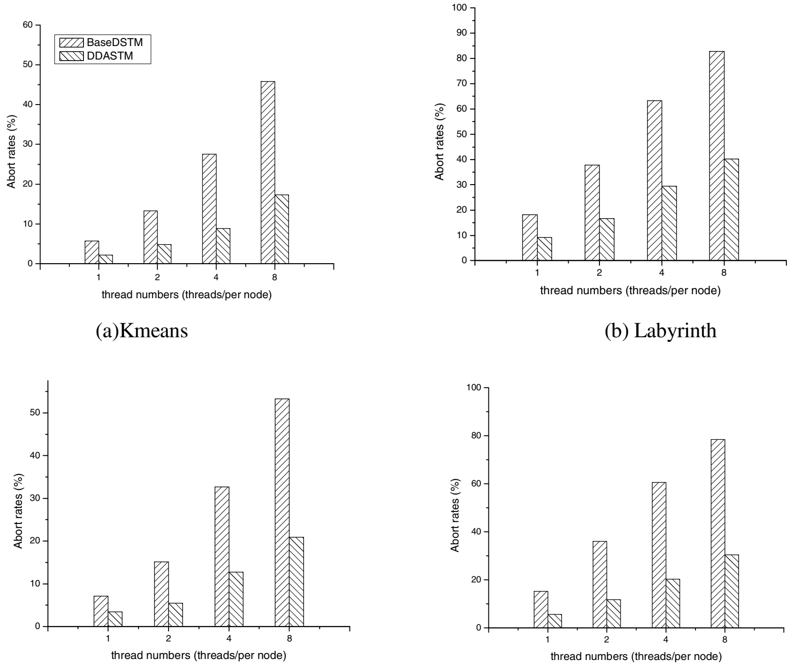
In this section, we firstly show the potential benefit of using our approach to ensure CS in distributed STM via comparing the abort rate of DDASTM with a distributed STM, named BaseDSTM, employing two-phase locking. Finally the runtime overhead and speedup for DDASTM and DDA are also shown.

4.1 Environments and Benchmarks

The hardware platform used in our experiments is a cluster with 40 cores residing on five nodes: the master node and four others while the network interconnection is a Gigabit ethernet. The master node and all the remaining worker nodes are 4 dual core Intel Xeon CPU at 1.60 GHz with 4GB of RAM each. Each node has 8 cores and thus a maximum of 8 threads are spawned to run transactions. All the nodes run Red Hat Enterprise Linux Server release 5.1. In experiments, we create from 1 to 8 threads per node in total from 5 (one thread per node) to 40 (8 threads per node) threads to execute following benchmarks.

Table 1. Transactional characteristics of benchmarks

Benchmarks	Length	Size of R/W Set	Contention
Kmeans	Short	Small	Low
Labyrinth	Long	Large	High
ssca2	Short	Small	Low
Vacation	Long	Large	High



(a)Kmeans

(b) Labyrinth

(c) ssca2

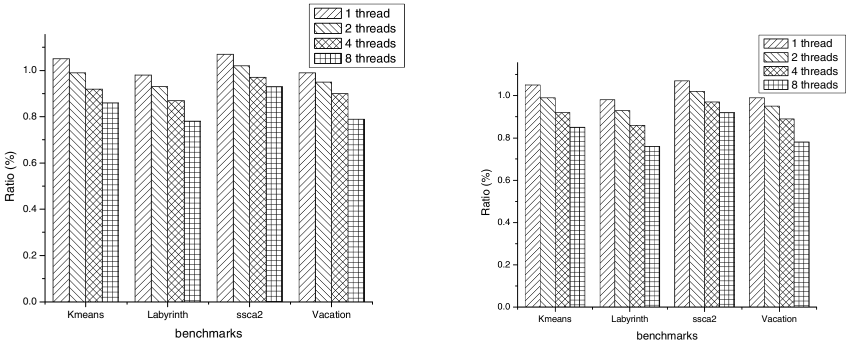
(d) Vacation

Fig. 4. Abort rates for BaseDSTM and DDASTM

In order to evaluate our system, some benchmarks from STAMP [12] suite have been ported to our system. In total four typical benchmarks have been used to evaluate our system: *kmeans*, *labyrinth*, *ssca2* and *vacation*. These benchmarks contain both short-running small R/W sets low-contention transactions as well as long running large R/W sets high-contention transactions. Table 1 summarizes the transactional characteristics of above benchmarks. These characteristics include the length of transactions (the average execution time spent for each successful transaction), size of the read and write sets for each successful transaction and amount of contention (abort rate).

4.2 Results

Fig. 4 shows the ratio of aborted transactions with respect to the total number of transactions for DDASTM and BaseDSTM using above benchmarks with 1, 2, 4 and 8 threads per node on five nodes respectively. These figures show that DDASTM has from significantly less abort rate compared to BaseDSTM for relaxer correctness criterion. Taking *kmeans* as an example, BaseDSTM almost aborts 45.8% of all transactions when the number of threads is 40, whereas DDASTM only aborts 17.3%. Furthermore, the abort rate of BaseDSTM is almost up to 82.8% when the benchmark is *labyrinth* executing on 40 cores. However, DDASTM almost only aborts 40.3%.

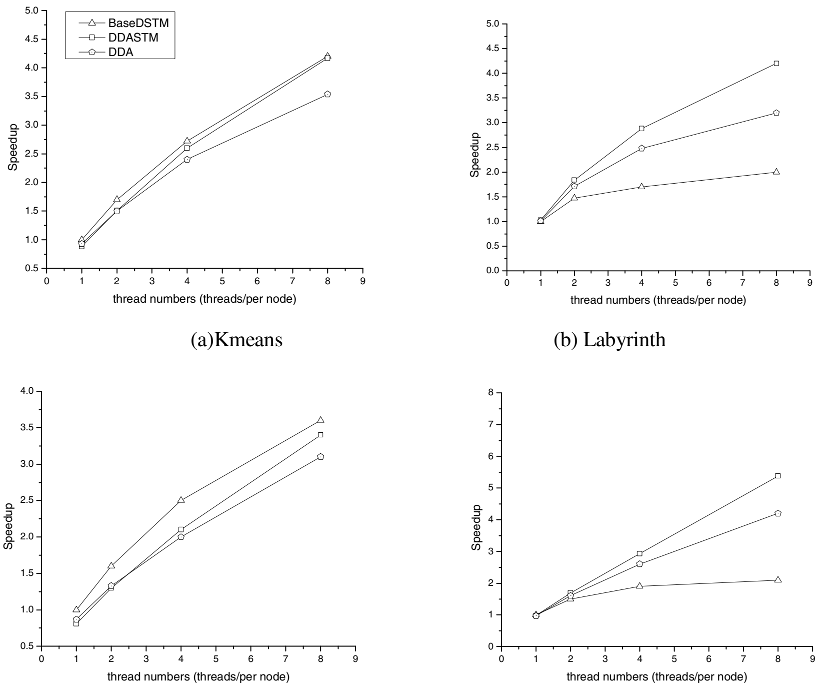


(a) Average execution time ratio for each transaction of DDASTM against DDA

(b) Average communication cost ratio for each transaction of DDASTM against DDA

Fig. 5. Runtime overhead of DDASTM against DDA

Fig. 5(a) and Fig. 5(b) show the average execution time and average communication cost of transactions, including aborted ones, for DDASTM normalized with respect to those of DDA respectively. A number of observations can be made from both figures. First, DDASTM causes more execution time and communication cost than DDA for benchmark *kmeans* and *ssca2* when the number of threads is small. This is because DDASTM needs to gather neighbor information at any condition and causes some fixed overhead. DDA only causes a little runtime overhead when the contention is low and the R/W set is small. The second observation is that the execution time and communication cost ratio of DDASTM against DDA both become smaller with the increase of threads. Finally, we can observe that DDASTM spares more execution time and communication cost for *labyrinth* and *vacation* than *kmeans* and *ssca2*. More specifically, the execution time ratio can be decreased to 76% and 78% for *labyrinth* and *vacation* respectively. The execution time ratio for *kmeans* and *ssca2* are 85% and 92% respectively. Hence we can conclude that DDASTM can spare more runtime overhead against DDA, at the condition that transaction has high contention and accesses lots of objects.



(a)Kmeans

(b) Labyrinth

(c) ssca2

(d) Vacation

Fig. 6. Speedup with respect to one thread of BaseDSTM per node

The speedup of BaseDSTM, DDASTM and DDA with respect to one thread of BaseDSTM per node for above benchmarks are shown from Fig.6(a) to Fig.6(d) respectively. Fig.6(a) and Fig.6(c) show that DDA performs worse than BaseDSTM for benchmark *kmeans* and *ssca2* respectively for much higher runtime overhead, even though with lower abort rates. DDASTM performs worse than DDA at the beginning for high runtime overhead, but performs better than DDA at last for more parallelism and lower average runtime overhead. However, its performance is still lower than BaseDSTM. Fig.6(b) and Fig.6(d) show that DDASTM performs better than DDA. Take *vacation* benchmark as example, when the speedup of DDA is 4.2×, the speedup of DDASTM has got up to 5.38× for lower runtime overhead. In *vacation* benchmark, we can also observe that DDASTM and DDA all perform better than BaseDSTM for lower abort ratio, except executing it with one thread per node.

5 Conclusions

Most current distributed STM systems still implement 2PL concurrency control algorithm, which is demonstrated limit concurrency of transactions. Recently, DDA model is proposed to guarantee CS in distributed STM to improve concurrency for the first time. However DDA induces much extra time to finish a transaction, especially at the condition that the transaction accesses lots of objects or the PG is complex for many active transactions and high contention.

In this work, we propose an efficient distributed STM, namely DDASTM, to ensure CS. We evaluate several benchmarks on DDASTM and DDA, and demonstrate that much redundant execution time and also much unnecessary communication cost can be spared via allowing several operations to correctly detect the same cycle together in parallel way. Experimental results also show that it has a clear performance advantage to the approach of DDA. Results prove that for transaction with high contention or accessing lots of objects, the speedup of distributed STM employing CS is up to 2.56× against distributed STM leveraging 2PL.

Acknowledgments. This paper is supported by National 863 High Technology Research and Development Program under grant No.2012AA010905, and National Natural Science Foundation of China under grants No.61272408 and No.61073024.

References

1. Bocchino, R.L., Adve, V.S., Chamberlain, B.L.: Software transactional memory for large scale clusters. In: Proceeding of 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, pp. 247–258. ACM Press (2008)
2. Kotselidis, C., Ansari, M., Jarvis, K., Luján, M., Kirkham, C., Watson, I.: DiSTM: A software transactional memory framework for clusters. In: Proceeding of 37th International Conference on Parallel Processing, pp. 51–58. IEEE Press (2008)
3. Bieniusa, A., Fuhrmann, T.: Consistency in hindsight: A fully decentralized STM algorithm. In: Proceeding of 24th IEEE International Parallel and Distributed Processing Symposium, pp. 1–12. IEEE Press (2010)
4. Zhang, B., Ravindran, B.: Brief announcement: on enhancing concurrency in distributed transactional memory. In: Proceeding of 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, pp. 73–74. ACM Press (2010)
5. Keidar, I., Perelman, D.: On avoiding spare aborts in transactional memory. In: Proceeding of 21th Annual Symposium on Parallelism in Algorithms and Architectures, pp. 59–68. ACM Press (2009)
6. Aydonat, U., Abdelrahman, T.: Serializability of transactions in software transactional memory. In: Proceeding of 2nd ACM SIGPLAN Workshop on Transactional Computing (2008)

7. Ramadan, H.E., Rossbach, C.J., Witchel, E.: Dependence-aware transactional memory for increased concurrency. In: Proceeding of 41st Annual IEEE/ACM International Symposium on Microarchitecture, pp. 246–257. IEEE Computer Society (2008)
8. Ramadan, H.E., Roy, I., Herlihy, M., Witchel, E.: Committing conicting transactions in an STM. In: Proceeding of 14th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, pp. 163–172. ACM Press (2009)
9. Perelman, D., Fan, R., Keidar, I.: On maintaining multiple versions in STM. In: Proceeding of 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, pp. 16–25. ACM Press (2010)
10. Krivokapić, N., Kemper, A., Gudes, E.: Deadlock detection in distributed database systems: a new algorithm and a comparative performance analysis. VLDB Journal 8(2), 79–100 (1999)
11. Quislant, R., Gutierrez, E., Plata, O., Zapata, E.L.: Multiset signatures for transactional memory. In: Proceeding of 25th International Conference on Supercomputing, pp. 43–52. ACM Press (2011)
12. Minh, C.C., Chung, J.W., Kozyrakis, C., Olukotun, K.: STAMP: Stanford transactional applications for multi-processing. In: Proceeding of IEEE International Symposium on Workload Characterization, pp. 35–46. IEEE Press (2008)
13. Tabba, F., Hay, A.W., Goodman, J.R.: Transactional conict decoupling and value prediction. In: Proceeding of 25th International Conference on Supercomputing, pp. 33–42. ACM Press (2011)
14. Riegel, T., Marlier, P., Nowack, M., Felber, P., Fetzer, C.: Optimizing hybrid transactional memory: The importance of nonspeculative operations. In: Proceeding of 21th Annual Symposium on Parallelism in Algorithms and Architectures (2011)
15. Titos-Gil, R., Negi, A., Acacio, M.E., Garcia, J.M., Stenstrom, P.: Zebra: A data-centric, hybrid-policy hardware transactional memory design. In: Proceeding of 25th International Conference on Supercomputing (2011)
16. Blake, G., Dreslinski, R.G., Mudge, T.: Proactive Transaction Scheduling for Contention Management. In: Proceeding of 42nd Annual IEEE/ACM International Symposium on Microarchitecture, pp. 156–167. ACM Press (2009)
17. Guerraoui, R., Kapalka, M.: On the Correctness of Transactional Memory. In: Proceeding of 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, pp. 175–184. ACM Press (2008)
18. Herlihy, M.P., Wing, J.M.: Linearizability: A correctness condition for concurrent objects. ACM Transactions on Programming Languages and Systems 12(3), 463–492 (1990)
19. Aydonat, U., Abdelrahman, T.S.: Hardware support for relaxed concurrency control in transactional memory. In: Proceeding of 43rd Annual IEEE/ACM International Symposium on Microarchitecture, pp. 15–26. IEEE Press (2010)

Research on Log Pre-processing for Exascale System Using Sparse Representation

Lei Zhu, Jianhua Gu, Tianhai Zhao, and Yunlan Wang

School of computer, NPU HPC Center, Xi'an, China
zeiier@126.com,
{gujh,wangyl,zhaoth}@nwpu.com

Abstract. With system size and complexity is growing rapidly, traditional passive fault tolerance can no longer guarantee the reliability of system because of the high overhead and poor scalability of these methods. Active fault tolerance is believed to be the most important fault tolerant approach for exascale systems. Aiming at system failure prediction, this paper proposes a system logs pre-processing method using classification via sparse representation (SRCP). Adopting the idea of vectorization, SRCP removes the details of each log and generates the corresponding Vectors. It uses TF-IDF (term frequency-inverse document frequency) method to Weight each keyword which can reveal more precise information about correlation between log records. In order to improve the accuracy and flexibility of pre-processing method, log vectors are processed by sparse representation classification. For generalization purpose, SRCP does not adopt any expert system or domain knowledge. Experimental results show that, SRCP can not only achieve both outstanding precision and F-measure, but also provide a satisfactory compression ratio.

Keywords: log pre-processing, active fault tolerance, exascale system, sparse representation.

1 Introduction

With system size and complexity are growing rapidly, the reliability issue of high performance computer has become extremely serious. The latest data shows that system mean time to failure (SMTTF) of the high-end system of TOP500 is less than 10 hours. If the scale of HPC continuous increasing and MTTF of single hardware remain stable, the scale of exascale system will much larger than the petascale systems and the SMTTF will as low as 1 hour [1].

Nowadays, passive fault-tolerant such as rollback-recovery is the dominant fault-tolerant method widely adopted in HPCs. However, this method has two drawbacks. First, its executing overhead is so large that contributes 15% to 50% of the overall processing cost [2]. Second, the scalability of these methods is poor.

To resolve these issues, researchers begin to distract their attention to active fault-tolerant method. These methods base on fault alert mechanism and predict possible failures in the near future. Comparing to passive fault tolerance, active fault

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 336–347, 2013.

tolerant methods can effectively reduce overhead and improve scalability of fault tolerance. By combining active and passive fault tolerant methods, the reliability of system will be significantly improved. Thus, such solution will be the most effective fault tolerant approach of next-generation extreme scale computer.

The effectiveness of active fault tolerant model is base on accurate failure prediction. Thus, many of predictive methods have been introduced, and researchers pay a lot of attention to system status analyzing method because of the effectiveness. System logs provide abundant source of information for system status analyzing. However, the raw logs can't be used directly, because it contains too many useless information and often unstructured for processing. Thus, the system logs pre-processing is indispensable to fault prediction. Fixed thresholding method is used by most of current pre-processing approaches. The precision and flexibility of these solutions can be improved.

Because of this, this paper focuses on the design of flexible and precise pre-processing method. The goal of this method is not only filtering out the useless information, but extracting the relevance between records. It can provide useful information for further executing such as failure pattern recognition.

This paper introduces a novel sparse representation classification based system logs pre-processing method (SRCP), which can not only filter out useless records, but has the ability to extract the relevance between logs precisely. SRCP removes the details of message in each log and generates the corresponding vectors. Then, SRCP process the log vectors by using sparse representation classification. For generalization purpose, SRCP method does not adopt any expert systems or domain knowledge. SRCP outperforms current log pre-treatment methods because it has high precision and flexibility.

2 Related Work

The most important step of active fault tolerance is failure prediction. According to different data sources, current system failure prediction is mainly executed in three ways, i.e. failure tracking, symptom monitoring and detected error reporting [3]. Regardless of which way system uses, accurate failure prediction based on system logs which are pre-processed. In order to eliminate useless records, Zhang and Yu [5] propose a pre-treatment approach which analyze RAS log and job log of Blue Gene/P in a cooperative manner, and identify a dozen important observations about failure characteristics and job interruption characteristics on the Blue Gene/P system. Liang et al [6] propose STF (temporal and spatial filtering) method which is widely used by current pre-processing approaches. This method is aimed at Blue Gene/L systems, and adopts expert system classify logs according to their sources.

All pre-processing methods introduced above use expert system or domain knowledge, which makes them relate to specific system closely. To overcome this problem, Liang et al. [7] introduce ASF (adaptive semantic filtering) method. ASF analyzes correlation of event logs based on semantics. It converts the log to two-value (0-1) vector according to their message and calculates Pearson correlation coefficient of two vectors. Then, ASF remove redundant records base on thresholds which relate to time interval between two vectors. Zhou and Jiang [3] propose IASF (improved ASF) approach base on original ASF method. Using the idea of VSM (Vector Space Model), IASF method removes the details in each log and generates its corresponding event vector which shows the present frequency of different keywords. IASF calculates the cosine of vectorial angle to evaluate the correlation between different event vector. Then it filters redundant records by using the same method of ASF.

All methods proposed above deal with useless information filtering, which is only one aspect of pre-processing. Pre-processing method needs to extract the relevance between logs too. Unlike these studies, this paper introduces a sparse representation classification based system logs pre-processing method. SRCP uses TF-IDF (term frequency-inverse document frequency) method to Weight each keyword. TF-IDF method can reveal more information about the similarity and dissimilarity between different log records because it sets the value of each vector element not only according to the present frequency of the keyword in corresponding log, but also according to the present frequency of the keyword in all logs. It can amplify the weighted values of rare keywords and diminish weighted values of common keywords. What's more, all studies introduced above use fixed thresholding method to filter out useless information which highly depend on appropriate threshold. Variable of system status can also affect the effect of pre-processing seriously, because the threshold can't be changed during processing. Thus, the flexibility of these methods is poor, which makes it unsuited for online failure prediction. SRCP proposed in this paper can overcome these problems.

3 Classification via Sparse Representation

Given training set $A = [A_1\, A_2\, A_3\, \cdots A_C\,]$. $A_i = [v_{i,1}\, v_{i,2}\, v_{i,3}\, \cdots v_{i,n_i}] \in R^{m\times n}$ is the ith object class. $y \in R^m$ is test sample. The sparse representation based classification method is based on the simple assumption that a new test sample y from class i lies in the same subspace with the training samples of the same class, Thus y can be represented by a linear combination of them [10]:

$$y = A_i x_i = \omega_{i,1} v_{i,1} + \omega_{i,2} v_{i,2} + \omega_{i,3} v_{i,3} + \cdots + \omega_{i,n_i} v_{i,n_i} \tag{1}$$

$x_\text{i} \in R^m$ is a coefficient vector whose elements are zero except those associated with the atoms belonging to the ground-truth class. But it is unknown most of the time, then (1) can be transfered to

$$y{=}Ax \tag{2}$$

The system is typically underdetermined ($m \ll n$) in practice, and the solution of (2) is not unique. The objective of sparse representation is to find the solution with smallest $\|x\|_0$. This optimization problem can be described as follows:

$$\hat{x} = argmin_x \|\mathrm{x}\|_0 \quad s.t\; Ax = y \tag{3}$$

where $\|\cdot\|_0$ represents l_0-norm. Usually, the test sample can only be approximately reconstructed because of noise. Thus (3) be reformulated as

$$\hat{x} = argmin_x \|\mathrm{x}\|_0 \quad s.t\ \|Ax = y\|_2^2 \leq \epsilon \tag{4}$$

where ϵ is the upper-bound of approximation error. However, (4) is an NP-hard problem. Typical methods for solving the problem are either approximating the original problem (2) with l_1-norm [11] or resorting to greedy schemes [12] to directly solve it. Ideally, $\hat{x}$ should be sparse, and only the atoms associated with the elements belonging to the ground-truth class are non-zero. However, in practice, $\hat{x}$ is generally dense, with large non-zero entries corresponding to training samples from many different classes [13]. Thus, the class label of y is decided by

$$\hat{I} = argmin_i \left\|y - \hat{y}^i\right\|_2 = argmin_i \left\|y - A\delta^i(\hat{x})\right\|_2 \tag{5}$$

where $\delta^i(x), i = 1,2,3 \cdots k$ is a new vector which sets all the elements of x to be zero except those corresponding to class i. This classification method can avoid overfitting.

Sparsity has long been exploited in many fields such as signal processing and pattern recognition. Recently, a classification method via sparse representation for Text Categorization is proposed by [14].

4 Sparse Representation Based System Log Pre-processing

The useless information of system logs should be filtered out. One part of them is the isolated events which is irrelevant to failures. Another part is redundancy. Redundancy can be divided into two types: temporal redundancy and spatial redundancy. Temporal redundancy is caused by log's burst feature in the time domain. Spatial redundancy is attributed to the parallelism of applications. When a job is running on multiple nodes, any record can be generated from multiple locations [4].

The relevance between logs is useful information for subsequential processing. Due to the number of logs collected by high-end system is massive (it's true even after pre-treatment), scanning operation is costly. However, pre-processing operation has to scan logs several times. If pre-processing method can record the relevance between logs, the overhead of subsequential processing will be reduced.

SRCP contains two steps: Event Vector Construction and vector processing.

4.1 Event Vector Construction

SRCP translates every record into corresponding vector because the original records can't be processed by machine.

Log Details Deletion. The message of the record is very important. But the details such as paths to the targets are not helpful to analyze correlation of log records. Thus, the following rules are adopted.

- All uppercase letters replaced are by corresponding lowercase letters.
- Remove punctuation, quotes and parentheses (include what it contains).
- All verbs are replaced by their simple present tense.

Vector Construction. Through scan all logs contained in training set, SRCP generates keyword table. Then, it constructs corresponding vector for each record. The elements of each vector are calculated as

$$TF \times IDF = TF_{i,j}\, log \frac{N}{DF_j} \tag{6}$$

where $TF_{i,j}$ is the present frequency of keyword j in record i, N is the number of logs contained in training set, and DF_j is the document frequency of keyword j. This method can amplify the weight of rare keywords. Thus it can reveal more precise information about the correlation between log records.

The location and job ID of the record is another part of corresponding vector. SRCP sets these two elements of all vectors to be zero first. They will be updated during classification process.

4.2 Vector Processing

Most of pre-processing methods use fixed thresholding method to filter out useless records. According to a given time interval, these methods filter out all records which contain identical message vector except the first one. It may lose useful information such as failure propagation. If all subsequent records are removed, some patterns will be filtered out. To solve this problem, instead of removing all subsequent records with same message according to T_{max}, SRCP method keeps all first records whose locations and job ID are different and filters out the rest of them. T_{max} is the fixed time interval which is decided by administrator. If the time interval of two records is larger than T_{max}, they are independent to each other no matter what they contain.

SRCP is a supervised learning method. But the training set extracting from original logs can't be used as dictionary directly because they do not contain the information about correlation of event logs. Thus, through analyzing the training set, correlation table $re = [re_1\, re_2\, \cdots re_l\,](l$ is the record number of $A)$ needs to be constructed. There are three sub-tables of re: redundancy table, relevance table and isolation table. Each sub-table contains the pointers which denote the IDs of corresponding subsequent records of each a_i. Base on that table, SRCP generates dictionary of every event vector $a_i \in A, i = 1 \cdots n_A(A$ is training set, n_A is number of different vector which contained in $A)$. The procedure of dictionary generating method is summarized in figure 1. Each a_i is processed identically.

Dictionary generating algorithm

```
Input: training set $A$, correlation table $re$, fixed time interval $T_{max}$, time slice $T$, event vector $a_i, i = 1 \cdots n_A$.
Output: dictionary $D_i$, $i = 1 \cdots n_A$
for all $a_i \in A$
    generate $R_i = \{r_{i,1}\, r_{i,2} \cdots r_{i,n_i}\} \leftarrow$ extract all vectors which are identical to $a_i$ from $A$
    generate $R'_i = [R'_{i,1}\, R'_{i,2} \cdots R'_{i,n_i}] \leftarrow$ extract all subsequent vectors of each element of $R_i$, and the time interval between them should not exceed $T_{max}$ (sub set $R'_{i,j}, j = 1 \cdots n_i$ is the set of all subsequent vectors of $j$th element of $R_i$ according to $T_{max}$ )
    if the locations or jobs ID of elements of $R_i$ and corresponding element of $R'_i$ are identical then
        set corresponding element of $R'_i$ to be 1
    else then
        set it to be zero
    end if
    according to $T$ and the time interval between each element of $R'_{i,j}$ and corresponding element of $R_i$, divide dictionary into several slices
    according to $re$, classify corresponding atoms of $R_i$ into three groups: ISOLATION, REDUNDANCY and RELEVANCE
    $D_i = R'_i$
end for
```

Fig. 1. Dictionary generating algorithm

Time slice T is used to divide dictionary into several slices, SRCP will perform similar operation during classifying test sample. The correlation between test sample and its correlated records can be processed using only corresponding dictionary slice. The overall procedure of classifying method is summarized in figure 2.

SRCP can achieve higher precision with a smaller T, but the size of B would be larger after removing duplicate elements. However, SRCP is a potential parallel method because each element of B is independent to each other during processing. Even though m is set larger, this feature can still reduce the time overhead of classifier because the size of dictionary slice is smaller. Clearly, the value of T can not be set too small, because each dictionary slice should be over-complete. Meanwhile, if T is set too large, the precision of SRCP would be affected.

SRCP assumes that an over-complete dictionary of each a_i can always be found. If the scale of the system is very large, the number of log will be massive after a period of collecting. In this condition, above assumptions are tenable because n_A is fixed.

Test sample classifying algorithm

```
1:  Input: training set $A$, fixed time interval $T_{max}$, time
    slice $T$, dictionary set $D = \{D_i\ D_2 \ldots D_{n_A}\}$
2:  Output: class label $I$, correlation vector $re\_f$
3:  for all $y \in Y$
4:      $re\_f = \emptyset$
5:      generate $B = [b_1\ b_2\ b_3\ \cdots b_n\ ] \leftarrow$ extract all antecedent
        vectors of $y$, and the time interval between them should
        not exceed $T_{max}$
6:      according to $T$, divide $B$ into several slices denoted
        by $B_1 \cdots B_m\ B_k = [b_{k,1}\ b_{k,2}\ \cdots\ b_{k,n_k}] \in B$, $k = 1 \cdots m$
7:      remove all identical elements from each $B_k$
8:      if  $B \neq \emptyset$  then
          extract $b_{n_b} \in B_k \in B, n_b = 1 \cdots n_k, k = 1 \cdots m$ ,delete $b_{n_b}$ from $B$
9:        if the locations or jobs ID of $y$ and $b_{n_b}$ are
          identical then
10:         set corresponding element of $y$ to be 1,
11:       else then
12:           set it to be zero.
13:       end if
14:       class label $\hat{I} \leftarrow$ classify $y$ using SRC(sparse
          representation based  classification) and
          corresponding dictionary slice
15:       if the class label $\hat{I} = red$(REDUNDANCY) then
16:           $I = red$, $re\_f = 0$, break
17:       else if $\hat{I} = rel$(RELEVANCE)  then
18:           $re\_t \leftarrow$ ID of $b_{n_b}$, go to 8;
19:       else if $\hat{I} = iso$(ISOLATION)  then
20:           go to 8;
21:        end if
22:     end if
23:     if $re\_t \neq \emptyset$ and $re\_t \neq 0$ then
24:       $I = rel$, $re_f = re\_t$
25:     else if $re\_f = \emptyset$ then
26:       $I = iso$
27:     end if
28: end for
```

Fig. 2. Classifying algorithm

SRCP only picks out the information whether two records are relevant, further process is beyond the scope of this paper.

5 Evaluation

In this section, SRCP method is evaluated using RAS log of Intrepid and the log is available at [8].

Intrepid is a 40-rack Blue Gene/P system. It consists of 40,960 computer nodes with a total of 163,840cores, which offer a peak performance of 556 TFlops. Table 1 contains the details of the log file.

Table 1. Details of system log

log	Start time	End time	Num of log	size
Intrepid	2009-1-15	2009-8-31	2084393	1.02GB

Four metrics are used to evaluate SRCP method: *compression ratio*, *precision*, *recall* and *F-measure*. The number of logs which are filtered out correctly is denoted as TP (true positive) and the number of logs which are filtered out incorrectly is denoted as FP (false positive). Meanwhile, TN (true negative) is the number of logs which are kept correctly and FN (false negative) is logs which are kept incorrectly. N is the number of records. The metrics can be formulated as follow.

Compression ratio is defined as ratio of the number of records which are filtered out, to the number of records of testing set.

$$Compression\ ratio = \frac{TP+FP}{N} \tag{7}$$

Precision is the ratio of the number of records which are filtered out correctly, to the total number of records which are filtered out

$$Precision = \frac{TP}{TP+FP} \tag{8}$$

Recall is defined as ratio of the number of records which are filtered out correctly, to the number of useless records.

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

Most likely, *precision* and *recall* can’t be improved simultaneously. To integrate the tradeoff between precision and recall, the *F-measure* was introduced [15].

$$F - measure \ = \frac{2\times precision\times recall}{precision+recall} \in [0,1] \tag{10}$$

For pre-processing methods, the *precision* is more important than *recall*. The overhead of subsequent process, such as failure pattern recognition, will be increased if the logs contain some redundant records. However, if pre-processing method filters out some useful records, corresponding pattern may not be recognized. That would affect the accuracy of prediction method. Thus, this paper also adopts $F_\beta - measure$.

$$F_\beta - measure \ = \frac{(1+\beta^2)\times precision\times recall}{\beta^2\times precision+recall} \tag{11}$$

If $\beta=0.5$, it is known as $F_{0.5}$-*measure* which puts more emphasis on *precision* than *recall*. The goal of this study is to achieve higher $F - measure$ and $F_{0.5} - measure$. The system parameters are set as follow:

$$T_{max} = 30\,\text{min} \tag{12}$$

$$T = 1\,\text{min} \tag{13}$$

SolveOMP algorithm of SparseLab is used to find sparse solutions to systems of linear equations [9].

SRCP is supervised learning method which is inferring a function from processed training data. An important issue is the amount of training data. If the classifier is learned from small amount of training data, the overall overhead of SRCP will be low. But the precision of SRCP will be affected. However, if the amount of training data is large, a learning algorithm with low bias and high overhead will be learned. This issue is exploited base on the log file of Intrepid. The results are shown in figure 3 and figure 4.

The result shows if the size of training set is larger than three months, the *precision* will not be improved significantly. The *compression ratio* is not monotone increasing. If the amount of training data is too small, many events are considered to be isolate event because SRCP can't find an over-complete dictionary for corresponding records.

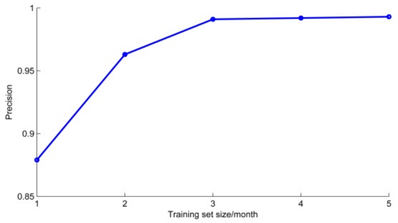


Fig. 3. Size of training set impact on *precision*

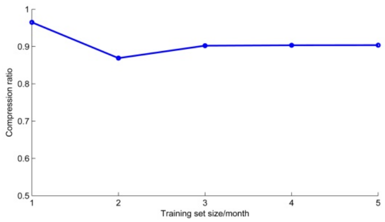


Fig. 4. Size of training set impact on *compression ratio*

Instead of choosing the first three month as training data, the data of February, March and July are chosen which makes the method more general. All further evaluations are based on this training data. To evaluate the effectiveness of our method, it is compared to ASF and IASF. The results are shown in figure 5.

The *compression ratio* of SRCP is the lowest, but it can sill filter out more than 90% of records. The main reason is that the *compression ratio* is the most important metric of ASF and IASF.

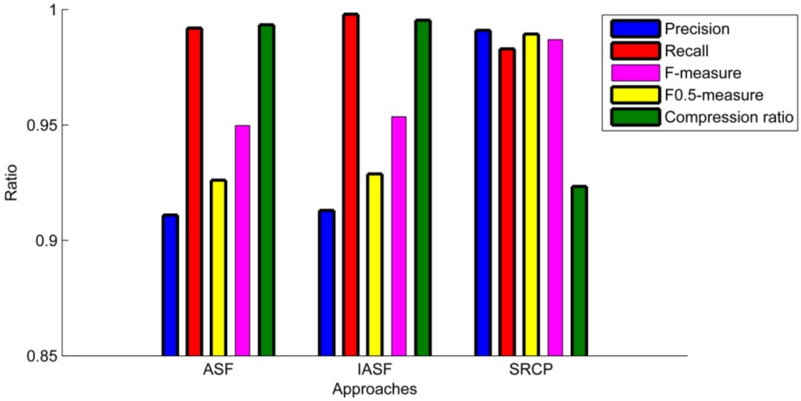


Fig. 5. The experiment results of pre-processing approaches

In order to compare the *precision*, *recall* and *F-measure* of these methods, the authors check the pre-treatment results of Intrepid. Figure 5 presents the results of evaluation.

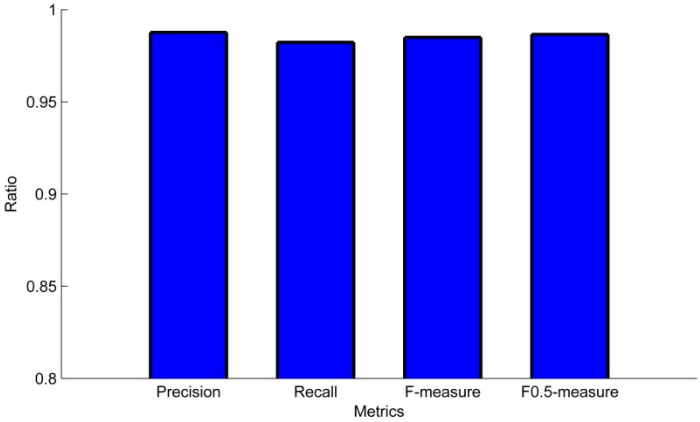


Fig. 6. Evaluation results of relevance extracting

The *recall* of ASF and IASF are outstanding because the *compression ratio* of these methods is all above 0.993. However, SRCP method greatly improves the *precision* of pre-treatment. For Intrepid, both *precision* and *recall* of SRCP method are above 0.95, and the *F-measure* of SRCP is the highest.

What's more, SRCP method can extract the relevance between logs. According to Figure 6, SRCP can extract the relevance between records precisely.

6 Conclusions and Future Work

This paper introduces sparse representation based system logs pre-processing method. SRCP removes the details of records and generates the corresponding vectors. It uses TF-IDF method to weighting each keyword which can reveal more precise information about the correlation between log records. Then, SRCP processes the vectors by using sparse representation classification. For generalizetion purpose, SRCP method does not adopt any expert systems or domain knowledge. Experimental results show that, SRCP can not only achieve both outstanding *precision* and *F-measure*, but also provide a satisfactory *compression ratio*.

However, SRCP method introduced in this paper could be further improved. First of all, the training set used in SRCP is processed manually. It is costly because the amount of training data is large. Besides, the effectiveness and correctness of sparse representation based classification heavily depend on the quality of dictionary. An automatic and accurate approach would be adopted to learn dictionary from the training data, which can give better performance. What's more, the time overhead of vector processing is high. Basing on the potential parallel feature of the algorithm, the time overhead would be reduced if the algorithm is processed on GPU. We will exploit these aspects in future.

Acknowledgments. The authors thank Ziming Zheng and Zhiling Lan from the Illinois Institute of Technology for making the log file of Intrepid available. The authors also would like thank the SparseLab team for making the greed algorithm available.

References

1. Cappello, F.: Fault tolerance in petascale/exascale systems: current knowledge, challenges and research opportunities. The International Journal of High Performance Computing Applications 23, 212–226 (2009)
2. Varela, M.R., Ferreira, K.B., Riesen, R.: Fault-Tolerance for Exascale Systems. In: 2010 IEEE International Conference on Cluster Computing Workshops and Posters (CLUSTER WORKSHOPS), Heraklion, Crete, pp. 1–4 (2010)
3. Zhou, H., Jiang, Y.: Research on Online Failure prediction Model and Status Pretreatment Method for Exascale System. In: International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp. 386–392 (2011)
4. Zheng, Z., Lan, Z.: System Log Pre-processing to Improve Failure Prediction. In: Proc. of ICDSN, pp. 572–577 (2009)

5. Zheng, Z., Li, Y.: Co-analysis of RAS Log and Job Log on Blue Gene/P. In: Proc. of IPDPS, pp. 840–851 (2011)
6. Liang, Y., Zhang, Y., Jette, M.: BlueGene/L Failure Analysis and Prediction Models. In: International Conference on Dependable Systems and Networks (2006)
7. Liang, Y., Zhang, Y., Xiong, H.: An Adaptive Semantic Filter for Blue Gene/L Failure Log Analysis. In: Parallel and Distributed Processing Symposium (2007)
8. The ANL Intrepid log, http://www.cs.huji.ac.il/labs/parallel/workload/l_anl_int/index.html
9. SparseLab software package, http://sparselab.stanford.edu/
10. Zhang, H., Nasrabadi, N.: Multi-View Automatic Target Recognition using Joint Sparse Representation. IEEE Transactions on Aerospace and Electronic Systems 48(3), 2481–2496 (2012)
11. Donoho, D.L.: For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. Communications on Pure and Applied Mathematics 59, 797–829 (2004)
12. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on Information Theory 53(12), 4655–4666 (2007)
13. Wright, J.: Robust Face Recognition via Sparse Representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(2), 210–227 (2009)
14. Sainath, T.N., Maskey, S., Kanevsky, D., Ramabhadran, B., Nahamoo, D., Hirschberg, J.: Sparse representations for text categorization. In: Proc. Interspeech, pp. 266–2269 (2010)
15. Salfner, F., Lenk, M., Malek, M.: A Survey of Online Failure Prediction Methods. ACM Computing Surveys 42(3) (2010)

Using Event-Based Style for Developing M2M Applications

Truong-Giang Le[1], Olivier Hermant[2], Matthieu Manceny[1],
Renaud Pawlak[3], and Renaud Rioboo[4]

[1] LISITE - ISEP, 28 rue Notre-Dame des Champs, 75006 Paris, France
[2] CRI - MINES ParisTech, 35 rue ST-Honoré, 77300 Fontainebleau, France
[3] IDCapture, 2 rue Duphot, 75001 Paris, France
[4] ENSIIE, 1 square de la Résistance, F-91025 Évry CEDEX, France
{le-truong.giang,matthieu.manceny}@isep.fr,
olivier.hermant@mines-paristech.fr,
renaud.pawlak@gmail.com,
renaud.rioboo@ensiie.fr

Abstract. In this paper, we introduce how to write M2M applications by using INI, a programming language specified and implemented by ourselves that supports event-based style. With event-based programming, all M2M communication can be handled and scheduled. Programmers may use existing built-in events or define their own events. We apply our approach in a real M2M gateway, which allows gathering and exchanging information between sensors and machines in the network. The results shows that our work proposes a concise and elegant alternative and complement to industrial state-of-the-art languages such as Java or C/C++.

Keywords: event-based programming, parallel programming, domain-specific languages, M2M applications, gateway.

1 Introduction

Machine-to-machine (M2M) refers to technologies that allow data communication and interaction between machine(s), device(s) or sensor(s) over a network without human intervention. The M2M connectivity market, a.k.a. the "Internet of Things", is growing worldwide. Analysts predict that there will be 25 billion connected IP devices by 2015, with M2M traffic expected to grow by 258% [2]. Another research estimates that M2M generates $35 billion in service revenues by 2016 [5]. It covers a wide array of applications including automotive, metering, remote management, IP multimedia subsystem, industrial data collection, health care, etc [16], [24]. For example, in agriculture, M2M applications are used to capture images to track crops' growth in the fields or to collect sounds to estimate insect quantity in the plants. Another example are medical centers, where the patient data such as blood pressure, heart rate, body temperature, and respiratory rate should be accumulated and sent periodically to the health care provider. In a factory, M2M sensors are used to track

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 348–357, 2013.

and monitor assets, equipment, materials, cargo and supplies. To understand more about M2M technologies, please refer to [19], [28]. Currently, M2M industry continues to looks for better and comprehensive M2M solutions.

Although M2M has attracted a large amount of attention over the years, developing M2M applications is still challenging. Besides, "existing M2M solutions are fragmented and usually are dedicated to a specific single application" [6]. In our work, we develop a novel programming language called INI to support developers to write M2M programs more easily. INI comes with event-based paradigm, which is an appropriate style to handle and schedule M2M communication [23]. Moreover, events handlers in INI run in parallel either asynchronously or synchronously to improve the performance and responsiveness of the system.

The rest of this paper is organized as follows. In Section 2, we give an overview of related work. In Section 3, we introduce INI and how it supports event-based style. Then, the case study of a M2M gateway written in INI along with the experiment tested on the real device are presented in Section 4. Section 5 concludes the paper.

2 Related Work

In recent years, the event-driven programming has been recognized as an efficient method for interacting and collaborating with the environment in ubiquitous computing [17]. Event-based programs are generally driven by a loop that waits for events and executes the appropriate callback [15]. Using event-driven style requires less effort and may lead to better performance, simpler, more manageable, portable code, and robust software [12]. This style is strong and convenient to write many kinds of applications: M2M applications, sensors applications, mobile applications, simulation systems, embedded systems, robotics, context-aware reactive applications, self-adaptive systems, etc.

Many M2M infrastructures frameworks, models, paradigms and services also have been proposed to ease the development of M2M systems. Herstad *et al.* [20] defined a service platform architecture for connected objects. The architecture exhibits a number of features to support scalability, rapid development, and technology and device independence. Cristaldi *et al.* [11] presented an interface platform, which is able to collect and process data from a wide variety of sensors and exchange information supporting different communication networks and protocols. Matson *et al.* [26] tried to create a model and architecture to support networking, communication, interaction, organization and collective intelligence features between machines, robots, software agents, and humans.

Currently, in order to build M2M applications, developers use classical programming languages (e.g. Java, .Net, C/C++, Perl, etc.) or their extensions. Besides, there has been several work on constructing event-based programming languages [10], [21], which can be applied to handle events happening in M2M systems. However, these languages are not fully comfortable for M2M applications since they lack a well-defined mechanism to support scheduled operations, which are essential in M2M communication. Another limitation is that events are not constructed and handled in an intuitive manner, i.e. they are mixed with other syntaxes and notations.

In our language called INI, events are defined and applied clearly. Furthermore, event actions can be scheduled easily with the help of two built-in events (e.g. `@every`, `@cron`) or by user-defined events. Another advantage is a flexible support of parallelism for events when running. Our next section will discuss in details.

3 Event-Based Programming with INI

3.1 Overview

Events are used to monitor changes happening in the environment or for time scheduling. In other words, any form of monitoring can be considered to be compatible with event-based style. In M2M communication, events are very frequent. Generally, three types of events are distinguished [27]:

- A timer event to express the passing of time.
- An arbitrary detectable state change in a system, e.g. the change of the value of a variable during execution.
- A physical event such as the appearance of a person detected by cameras.

For example, programmers may define an event to monitor the power level of their systems or to observe users' behaviors in order to react. They can also specify an event to schedule a desired action at preferable time. To understand more about event-based programming, please refer to [13], [14], 15].

INI (INI is Not ISEP) is a programming language developed by ourselves, which runs on Java Virtual Machine (JVM) but INI's syntax and semantics are not Java's ones. In INI, we support all these kinds of event as shown later. Event callback handlers (or events instances) are declared in the body of functions and are raised, by default asynchronously, every time the event occurs. By convention, an event instance in INI starts with `@` and takes input and output parameters. Input parameters are configuration parameters to tune the event execution. Output parameters are variable names that are filled in with values when then the event callback is called. They can be considered as the measured characteristic of the event instance. It has to be noticed that those variables, as well as any INI variable, enjoy a global scope in the function's body. Both two kinds of parameters are optional. Moreover, an event can also be optionally bound to an id, so that other parts of the program can refer to it. The syntax of event instances is shown below:

```
id:@eventKind[inputParam1=value1, inputParam2
  =value2,...](outputParam1, outputParam2,...)
   {<action>}
```

Programmer may use built-in events (listed in Table 1), or write user-defined events (in Java or in C/C++), and then integrate them to their INI programs.

Table 1. Some built-in events in INI

Built-in event kind	Meaning
@init()	used to initialize variables, when a function starts.
@end()	triggered when no event handler runs, and when the function is about to return.
@every[time:Integer]()	occurs periodically, as specified by its input parameter (in milliseconds).
@update[variable:T] (oldValue:T,newValue:T)	invoked when the given variable's value changes during execution.
@cron[pattern:String]()	used to trigger an action, based on the CRON patern indicated by its input parameter.

By developing custom events, we can process data which are captured by sensors. To illustrate user-defined events in INI, let us consider a simple health monitoring system that must monitor several information related to the patient such as body temperature, blood pressure, etc. We can design events to deal with each task as shown in Figure 1. The event `@temperatureMonitor` named `t` has one input parameter called `tempPeriod`, which is applied to set how long the event should sleep between two consecutive checks (time unit is in hours). Besides, it has one output parameter named `temperature` to indicate the current body temperature of the patient. Inside this event, based on the value of the current temperature, we can define several corresponding actions through the `n`-ary boolean `case` instruction (quite standard

```
function main() {
  //Monitor the temperature of a patient
  t:@temperatureMonitor[tempPeriod = 1](temperature) {
    case {
      temperature > ... {
        //Notify to a doctor or do other automatic
        //emergency actions ...
      }
      //A default action
      default { ... }
    }
  }
  //Monitor the blood pressure of a patient
  b:@bloodPressureMonitor[bpPeriod = 2](pressure) {
    //Do desired actions ...
  }
  ...
}
```

Fig. 1. A simple health monitoring system written in INI

and not described here, please refer to INI Language Reference Documentation [31]). The other event `@bloodPressureMonitor` named `b` has a similar structure. All events in our program run concurrently, which means that we can handle multiple tasks at one time. To learn how to write user-defined events in INI, interested readers may refer to [31].

3.2 Advanced Use of Events in INI

By default, except for the `@init` and `@end` events (see Table 1), all INI events are executed asynchronously. However, in some scenarios, a given event `e0` may want to synchronize on other events `e1,...,eN`. It means that the synchronizing event `e0` must wait for all running threads corresponding to the target events to be terminated before running. For instance, when `e0` may affect the actions defined inside other events, we need to apply the synchronization mechanism. Programmers may use the following code to ensure that the event `e0` synchronizes on `N` target events:

```
$(e1,e2,...,eN) e0:@eventKind[...](...) {<action>}
```

Besides, events in INI may be reconfigured at runtime in order to adjust their behaviors when necessary to adapt to changes happening in the environment. Programmers can call the built-in function `reconfigure_event(eventId, [inputParam1 = value1, inputParam2 = value2,...])` in order to modify the values of event's input parameters. For example, at some time during running, we want to collect patient's temperature data more frequently, we can adjust the input parameters of `t` by calling: `reconfigure_event(t, [tempPeriod = 0.5])`. Now our event will gather temperature data for every 30 minutes instead of one hour as before. Moreover, we also allow programmers to stop and restart events with the two built-in functions `stop_event([eventId1,eventId2,...])` and `restart_event([eventId1, eventId2,...])`. For instance, we may stop all data collection processes when the energy level of the system is too low and restart them later when the energy is charged again.

Last but not least, events in INI may be used in combination with a boolean expression to express the requirement that need to be satisfied before an event is executed. Programmers may use the syntax below:

```
<event_expression> <logical_expression> {<action>}
```

For example, if we want the event `@bloodPressureMonitor` to be executed only when the temperature is higher than some threshold:

```
@bloodPressureMonitor[bpPeriod=2](pressure) temperature
 >...{...}
```

To understand more about the above mechanisms and other aspects of INI (e.g. semantics, rules, type system, type checking, and built-in functions), programmers may have a look at [25], [31].

4 Case Study: A M2M Gateway Program

Many M2M applications are required to send data to a M2M server through network and a M2M gateway is a typical example [29]. In other words, a M2M gateway allows different types of networks to communicate with each other in order to provide data [30]. For example, users may use this device for industrial data collection or for surveillance purpose [13]. Since this kind of tasks does not create much added-value and maybe in dangerous or remote environment, taking advantages of M2M technologies is a good solution. In this section, we show how to apply INI to write a M2M gateway program, containing two basic steps as shown in Figure 2:

- Collecting data (e.g. images, sound, etc.), which are captured by sensors or peripherals.
- Transmitting data to the server through the network.

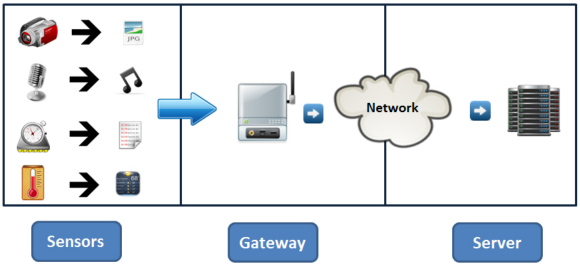


Fig. 2. The role of a gateway

All these operations above can be scheduled straightforwardly with the help of the two built-in events `@every` and `@cron` (see Table 1). The event `@every [time:Integer]()` can be applied to do an action periodically. The event `@cron [pattern :String]()` occurs on times indicated by the UNIX CRON pattern expression. CRON is a task scheduler that allows the concise definition of repetitive task within a single (and simple) CRON pattern [3]. A UNIX crontab-like pattern is a string split in five space separated parts, including minutes sub-pattern, hours sub-pattern, days of month sub-pattern, months sub-pattern, and days of week sub-pattern.

Some examples:

- "* * * * *": This pattern causes the event instance to occur every minute.
- "59 11 * * 1-5": This pattern causes the event to occur at 11:59 AM on every weekday (i.e. Monday, Tuesday, Wednesday, Thursday and Friday).

Our complete program is shown in Figure 3. The `main` function is composed of four events. The event `@init` (lines 3-13) is invoked to initialize necessary variables that will be used later. The variable `capturedDataFolder` indicates the folder where we put the collected data. If this folder does not exist, we create it (lines 7-9).

```
1  function main() {
2   //Initialization
3   @init() {
4    capturedDataFolder = file("data")
5    //Create a new data folder in case that it does not exist
6    case {
7     !file_exists(capturedDataFolder) {
8      mkdirs(capturedDataFolder)
9     }
10   }
11   zipFile = file("data.zip")
12   keepParentFolder = true
13  }
14  //Capture image from a camera using the library gphoto2
15  e1:@every[time = 60000]() {
16   exec("gphoto2 --capture-image-and-download --filename " +
17    "data/img" + time() + ".jpg")
18  }
19  //Capture sound from a microphone using the library alsa
20  e2:@every[time = 30000]() {
21   exec("arecord -d 30 -f cd " + "data/sound" + time() +
22    ".wav")
23  }
24  //Upload data to a FTP server by schedule
25  @cron[pattern = "0 09-18 * * 1-5"]() {
26   stop_event([e1,e2])
27   zip(capturedDataFolder,zipFile)
28   upload_ftp("server_address", "user_name",
29    "password", zipFile, to_string(time()) + "data.zip")
30   delete_file(zipFile)
31   delete_folder(capturedDataFolder, keepParentFolder)
32   restart_event([e1,e2])
33  }
34 }
```

Fig. 3. A M2M gateway program written in INI

The variable `zipFile` denotes a zipped data file. We compress the data before uploading to save network bandwidth.

The next two events are `@every` event kind. They are used to collect image and sound data (e.g. in a field or in a factory). The event `e1` (lines 15-18) is invoked every minute to get a picture captured by a camera (by using the library `gphoto2` [4]). The event `e2` (lines 20-23) is invoked every 30 seconds to use a microphone to record sound (by using the library `alsa` [1]). All files (i.e. pictures and sounds) are saved into the data folder. These collecting processes (i.e. two events) run in parallel to take advantage of multithreading for better performance.

The last event is a @cron (lines 25-33) event, which is employed to upload the data to a FTP server every hour during working hours (i.e. 9:00 AM - 6:00 PM) on every weekday. Inside this event, first, we stop the two events e1 and e2 (line 26). Next, we compress the data folder before uploading (line 27). Then we upload the compressed data to a FTP server (lines 28-29). Since the gateway is only a data-exchanging device with a limited storage capacity, after uploading, we delete the data to save storage (lines 30-31). Finally, we restart the two events e1 and e2 so that we can collect data again (line 32).

We tested our program on the real gateway in the scope of the MCUBE project [7]. The device is provided by Webdyn [9], a company dedicated to design, develop, and market this kind of product. We use Oracle Java SE Embedded for establishing the runtime environment of INI on the gateway. This platform is optimized for mid-range to high-end embedded systems and offers a high-performance virtual machine, full high-performance graphics support, deployment infrastructure, and a rich set of features and libraries [8]. During the experiment, our program worked well. All the data were captured and transmitted properly.

Now let us compare our INI program with a Java program that does the same tasks. In order to make all operations running in parallel in Java, we need to create some explicit threads for different tasks: capturing pictures, recording sounds, uploading data and time scheduling for all operations. Scheduling is a nontrivial task to implement with Java. In addition, we also need some thread pools in order to manage and synchronize those threads when needed. Although Java has a powerful support for concurrency, writing correct concurrent applications in this language is still challenging and error-prone, especially for novice programmers [18]. To decrease the difficulty and hide the unnecessary complexity, INI separates the thread issue (implemented in Java), and the event-handling issue (implemented in INI). This separation of concerns helps in making the INI approach clearer and less error-prone than a pure-Java program.

5 Conclusion and Future Work

In this paper, we presented how to write M2M applications using event-based style through INI, a programming language developed by ourselves. INI can be seen as an Architecture + DSL (Domain-Specific Language) for (multi-threaded) event handling and coordination. INI allows programmers to construct and define events in a convenient and straightforward way. Besides, events in INI run in parallel to perform multiple tasks concurrently. For testing, we built a gateway program and when running on the real device, this program completed required tasks adequately and appropriately.

For the future work, we will apply INI in more M2M applications and also in other domains like robotics or manufacturing systems. We also have a plan to evaluate quality and performance of INI programs. Currently, we are trying to better a tool called INICheck, which can convert a major subset of INI to Promela, the input modeling language of the well-known model checker SPIN [22]. Then SPIN can be used to verify some properties of INI programs like checking ranges of values for variables. This tool allows the programmer to have insurance on his code and its behavior.

Acknowledgments. The work presented in this article is supported by the European Union. Europe is committed in Ile-de-France with the European Regional Development Fund.

References

1. Alsa, http://www.alsa-project.org/
2. Cisco jumps into the M2M market, http://www.networkcomputing.com/wireless/231600077
3. Crontab, http://crontab.org/
4. Gphoto2, http://gphoto.org/
5. M2M to generate $35bn in service revenues by 2016, http://juniperresearch.com/viewpressrelease.php?pr=243
6. Machine-to-Machine (M2M): The rise of the machines, http://www.juniper.net/us/en/local/pdf/whitepapers/2000416-en.pdf
7. The MCUBE project, http://mcube.isep.fr:8080/
8. Oracle Java SE Embedded, http://www.oracle.com/technetwork/java/embedded/overview/getstarted/index.html
9. Webdyn, http://www.webdyn.com/en/
10. Cohen, N.H., Kalleberg, K.T.: EventScript: an event-processing language based on regular expressions with actions. In: Proceedings of the 2008 ACM SIGPLAN-SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems, LCTES 2008, pp. 111–120. ACM, New York (2008)
11. Cristaldi, L., Faifer, M., Grande, F., Ottoboni, R.: An improved M2M platform for multi-sensors agent application. In: Sensors for Industry Conference, pp. 79–83 (February 2005)
12. Dabek, F., Zeldovich, N., Kaashoek, F., Maziéres, D., Morris, R.: Event-driven programming for robust software. In: Proceedings of the 10th Workshop on ACM SIGOPS European Workshop, EW 10, pp. 186–189 (2002)
13. Denecke, K.: Event-Driven Surveillance: Possibilities and Challenges. Springer, Berlin (2012)
14. Etzion, O., Niblett, P.: Event Processing in Action. Manning Publications Co. (2010)
15. Faison, T.: Event-Based Programming: Taking Events to the Limit. Apress, Berkely (2006)
16. Fan, Z., Tan, S.: M2M communications for E-health: Standards, enabling technologies, and research challenges. In: 2012 6th International Symposium on Medical Information and Communication Technology (ISMICT), pp. 1–4 (March 2012)
17. Fischer, J., Majumdar, R., Millstein, T.: Tasks: language support for event-driven programming. In: Proceedings of the 2007 ACM SIGPLAN Symposium on Partial Evaluation and Semantics-Based Program Manipulation, PEPM 2007, pp. 134–143. ACM, New York (2007)
18. Goetz, B., Peierls, T.: Java concurrency in practice. Addison-Wesley (2006)

19. Hersent, O., Boswarthick, D., Elloumi, O.: The Internet of Things: Key Applications and Protocols. John Wiley & Sons (2011) (Incorporated)
20. Herstad, A., Nersveen, E., Samset, H., Storsveen, A., Svaet, S., Husa, K.: Connected objects: Building a service platform for M2M. In: 13th International Conference on Intelligence in Next Generation Networks, ICIN 2009, pp. 1–4 (October 2009)
21. Holzer, A., Ziarek, L., Jayaram, K., Eugster, P.: Putting events in context: aspects for event-based distributed programming. In: Proceedings of the Tenth International Conference on Aspect-Oriented Software Development, AOSD 2011, pp. 241–252. ACM, New York (2011)
22. Holzmann, G.J.: The SPIN model checker, Primer and reference manual, 1st edn. Addison-Wesley Professional (2003)
23. IoT-A: (Internet of Things – Architecture) Project Deliverable d3.1 - Initial M2M API Analysis (2012)
24. Kim, B.H., Ahn, H.J., Kim, J.O., Yoo, M., Cho, K., Choi, D.: Application of M2M technology to manufacturing systems. In: 2010 International Conference on Information and Communication Technology Convergence (ICTC), pp. 519–520 (November 2010)
25. Le, T.G., Hermant, O., Manceny, M., Pawlak, R., Rioboo, R.: Unifying event-based and rule-based styles to develop concurrent and context-aware reactive applications. In: Proceedings of the 7th International Conference on Software Paradigm Trends, Rome, Italy, July 24-27, pp. 347–350 (2012)
26. Matson, E., Min, B.C.: M2M infrastructure to integrate humans, agents and robots into collectives. In: 2011 IEEE Instrumentation and Measurement Technology Conference (I2MTC), pp. 1–6 (May 2011)
27. Mühl, G., Fiege, L., Pietzuch, P.: Distributed Event-Based Systems. Springer-Verlag New York, Inc., Secaucus (2006)
28. Roebuck, K.: Machine-to-machine (M2M) Communication Services: High-impact Technology - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors. Lightning Source Incorporated (2011)
29. Singh, S., Huang, K.L.: A robust M2M gateway for effective integration of capillary and 3GPP networks. In: 2011 IEEE 5th International Conference on Advanced Networks and Telecommunication Systems (ANTS), pp. 1–3 (December 2011)
30. Sosinsky, B.: Networking Bible, 1st edn. Wiley Publishing (2009)
31. Truong-Giang, L.: INI Online (2012), https://sites.google.com/site/inilanguage/

Retracted: Scheduling Optimization of the RFID Tagged Explosive Storage Based on Genetic Algorithm

Xiaoling Wu[1,*], Huawei Fu[1,2], Xiaomin He[2], Guangcong Liu[2], Jianjun Li[1],
Hainan Chen[1,2], Qianqiu Wang[1], and Qing He[1]

[1] Guangzhou Institute of Advanced Technology, Chinese Academy of Sciences
[2] Guangdong University of Technology, Guangzhou 510006
xl.wu@giat.ac.cn

Abstract. An on-line optimization method for explosives storage scheduling based on genetic algorithm is proposed to make management of explosives storage process more efficient, informative, secure, and intelligent. The information of explosives warehouse is acquired in real time by RFID technology, and an assignment strategy for the location of explosives in warehouse is proposed. The mathematical model of explosives storage optimization is constructed by analyzing operation characteristics of explosives storage and requirements, and the model is solved using the improved genetic algorithm. The simulation results show that the proposed method can improve the utilization rate of warehouse space, optimize the walking path in the process of the explosive delivery, as well as solve the operating problems under some constraints, such as the expire date of explosives.

Keywords: explosives storage, radio frequency identification technology, storage location assignment, genetic algorithm.

1 Introduction

Along with the rapid economic development of China, demands of industrial explosive materials are greatly increased, and so is the quick development of civil explosive industry. The literature [1] pointed out that this industry is under strict management of the regulatory authorities due to the characteristic of the civil explosive industry and other restrictions.

Explosives as a kind of special items need some special requirements in the storage process. There are few researches about the explosive storages. Yu Li [2] presented the application of RFID (Radio Frequency Identification) technology in safety management of the primer. Yuan Chen [3] studied the system model of dangerous goods logistics based on RFID and GPRS (General Packet Radio Service) technology. Explosive storage is a complicated process, in which combinatorial optimization problems need to be solved. The explosives of the same specification and the same production date need to be stored in the same area. To address the above problems, we use RFID technology to get explosive warehouse real-time information first, then

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 358–366, 2013.

partition the warehouse, finally, solve the optimization model by genetic algorithm and realize the scheduling optimization of explosive storage.

2 Problem Statement

Explosive products need to be temporarily stored in the warehouse in the production, circulation and use process. Many factors need to be considered when explosive products are passed in and out.

1) The expire date problem of the explosive is one of the important factors. In the warehouse explosives cannot be stored too long. If it's stored more than a certain time, it has to be delivered out of the warehouse first. The explosive products should follow the principle of first in first out;
2) If the production of explosives is more than the inventory capacity, the old explosive products have to be delivered out first, and then new products can be moved into the warehouse afterwards.
3) In order to get it delivered easily, the explosives with high probability to be delivered in and out should be placed close to the entrance.

3 The Calculation Model of Optimization Operation for Explosive Warehouse

Zhang Haijun, *et al* [4, 5] pointed out that normally there are many kinds of slotting distribution strategies in the explosive warehouse, one of which is the random classification. The characteristic of this distribution strategy is that each kind of goods is assigned to a fixed storage area, while in that specific storage area, goods distribution is random. The advantage is that it improves the slotting efficiency, and the defect is that the ins and outs of inventory and management are difficult. Fig.1 is a sketch of an explosive warehouse.

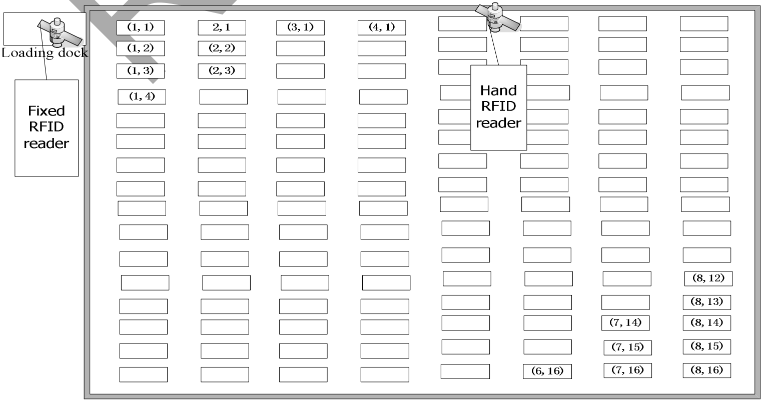


Fig. 1. The diagram of warehouse storage

The classification random storage method is adopted in this paper. RFID technology is applied to get the explosive warehouse real-time information. And we use genetic algorithm to process these large amount of data information, which can realize allocation optimization of the explosive location in the warehouse.

The specific strategy of location partition is as follows:

Step 1: According to the types of explosives, the partition number can be determined. The length and width of a warehouse slot are denoted as a and b, the distance that the operation person travels from the point of origin (warehouse gateway) to a slot is denoted as d_{pq} (slotting in line p column q), and the time spent is denoted as t_{pq}, where p, q are the code of slot. Set a time value t (t is the standard walking time in each area), v is the speed of delivery vehicle, which is the same in horizontal direction and vertical direction. If line p column q slot meets below (1) (2) (3) equations, then the slot is in k area.

$$d_{pq}=\text{（}\mathrm{p}\times\mathrm{a}+\mathrm{q}\times\mathrm{b}\text{）} \tag{1}$$

$$t_{pq}= d_{pq}/v \tag{2}$$

$$k\times t\leq t_{pq}-t\leq \text{（}k+1\text{）}\times t \tag{3}$$

Step 2: After the first step partitions each area may contain unequal number of slots. If the number of slots is too much different among different areas, then it must be regulated. If an area contains too few slots, the slots are taken from the adjacent area that contains more.

Step 3: Calculate the delivery frequency of the ins and outs of explosive of each specification. The type of explosives is equal to the number of partitions.

Step 4: Establish the weight matrix. The workload that the delivery vehicle takes to deliver some specific explosive in the unit time is related to the frequency of the goods ins and outs and the location of explosive storage. The frequency of goods ins and outs is multiplied by the time that operation person spends to arrive at the location, as a weight value, namely:

$$C_{kj} =f_k\, s_j$$

where f_k – the frequency of the explosive ins and outs at area k, s_j – the distance that the operation person moves from the point of origin to area j.

Consider that different explosive should be stored with different time in the warehouse and should be put in different position, the frequency of explosive ins and outs and the problem of goods expire duration, we denote the saving time of explosives in each area that is divided into K areas as P_k, so we can get a comprehensive factor W_{kj} in the process of explosive stored.

$$W_{kj}=P_kC_{kj}$$

After the above processing, the explosive storage becomes a 0-1 assignment problem. The mathematical model is as follows.

Objective function:

$$F{=}min \sum_{k}^{n} \sum_{j}^{n} W_{kj}X_{kj} \tag{4}$$

Constraint conditions:

$$\sum_{k=1}^{n} X_{kj} = 1 \tag{5}$$

$$\sum_{j=1}^{n} X_{kj} = 1 \tag{6}$$

where $X_{kj} = 1$ or 0, k = 1, 2... N; j = 1, 2... N, when $X_{\mathrm{kj}} = 1$ the j kind of explosives is put in area k, the constraint condition equation (5) means that an area can only put a kind of explosive, the equation (6) means that the same kind of explosive can be put only in the same area.

4 Operation Optimization Algorithm for Explosive Storages

4.1 Operation Optimization Algorithm Based on Genetic Algorithm

Step 1: Encoding
Literature [6, 7] claims that one hard problem of genetic algorithm is encoding. We use real number coding sequence representation method for encoding.
Step 2: Fitness function design
Because the value of the target function should be minimized, this paper applies "limit structure method" which uses a proper value C_{max} to minus the value of objective function. When the slot is divided into n areas, the corresponding weight matrix follows equation (7) to set up. Fitness function *Fit (f (x))* is as follows:

$$Fit(f(x)) = \begin{cases} c_{\max} - \sum_{k}^{n}\sum_{j}^{n} W_{kj} X_{kj} & f(x) < C_{\max} \\ 0 & otherwise \end{cases} \tag{7}$$

Step 3: The population initiation
According to the number of the partitions we can determine the length of the chromosome. The size of the initial population can be determined according to the number of partition size. If the division number is 15, namely the chromosome length is 15, we can generate the initial population number as 40. Wheel selection mechanism is

used in the selection process, and fitness function with an appropriate value C_{max} is used to minus the value of objective function.

Step 4: Genetic operator crossover

Based on the sequence encoding, if we use the conventional method of single point crossover, double point crossover or multipoint crossover, illegal chromosome will appear. Therefore, this paper constructs the above sequence encoding of crossover based on the crossover operator.

Step 5: Mutation

Mutation operators are mainly inversion, insert, shift, exchange, etc. Here we use simple exchange operation.

4.2 Operation Optimization Based on Improved Genetic Algorithm

To improve the convergence of the algorithm, and to avoid local optimization in the process of reaching optimization, an improved genetic algorithm is proposed based on the crossover rate P_c and mutation rate P_m variation within the scope of certain linearity. The equation (7) and (8) show that *GEN* represents the current evolution algebra and *MAXGEN* represents the largest evolution algebra.

$$P_c=（-0.6*GEN+180.6）/MAXGEN \tag{7}$$

$$P_m=(0.2*GEN+20)/MAXGEN \tag{8}$$

5 Experiment Results

Since the storage time of different kind of explosive is different, the storage time of 15 categories of explosives can be written as a vector denoted as P_k. Each digital unit is month. The ins and outs frequency of each category of explosive is f_k and the unit is ton. The average distance from each area to the entrance and exit is s_j, the unit is meter (each partition is at a distance from inward and outward). According to the above steps and the actual situation the coefficient of storage time P_k can be derived. The frequency of the ins and outs is f_k, the distance from each partition to the entrance and exit is t_j. The P_k, f_k, s_j value can be obtained through calculation and statistical analysis.

Table 1. The corresponding save time of different specifications of explosive p_k and the frequency of the ins and outs f_k

spec	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	K=11	K=12	K=13	K=14	K=15
P_k	2	4	3	3	2	6	4	5	3	4	5	5	6	3	5
f_k	3.0	3.5	3.3	4.3	5.0	6.2	5.8	8.0	9.0	3.3	11.0	6.0	12.2	10.0	8.2

We simulated the proposed algorithm for the storage allocation problem. The following four cases implemented with genetic algorithm are compared: crossover rate P_c = 0.8 mutation rate P_m = 0.2; crossover rate P_c = 0.8 mutation rate Pm = 0.1; crossover rate P_c = 0.9 mutation rate Pm = 0.2; crossover rate P_c = 0.9 mutation rate Pm = 0.1.

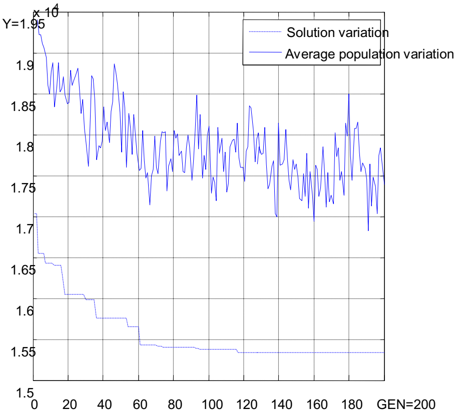


Fig. 2. $P_c = 0.8$; $P_m = 0.2$

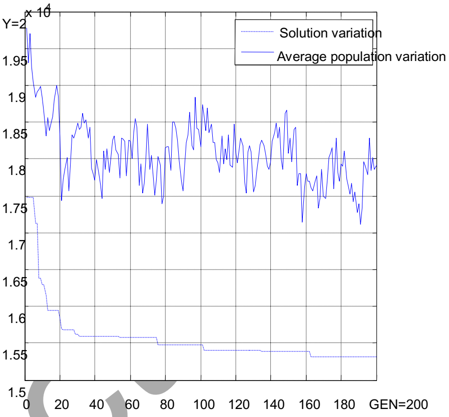


Fig. 3. $P_c = 0.8$; $P_m = 0.1$

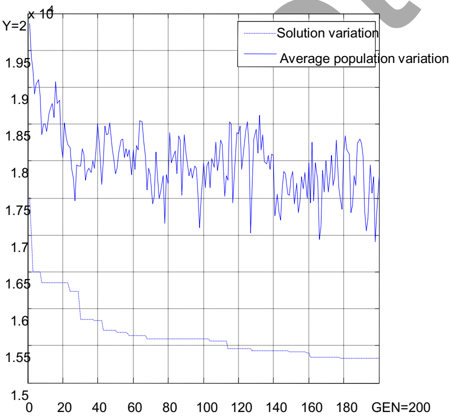


Fig. 4. $P_c = 0.9$; $P_m = 0.2$

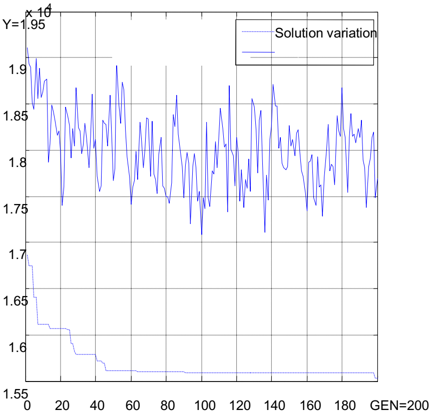


Fig. 5. $P_c = 0.9$; $P_m = 0.1$

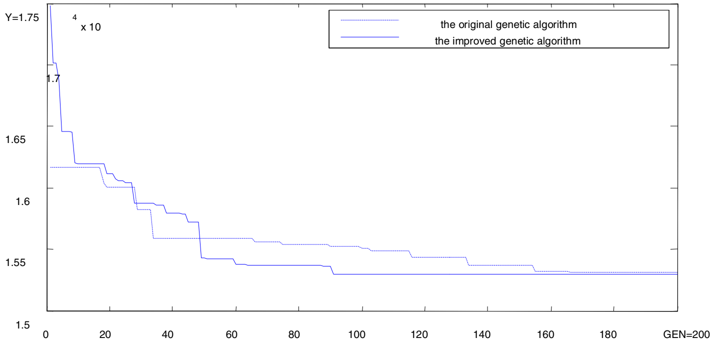


Fig. 6. The comparison between the improved genetic algorithm and the original genetic algorithm

From the above Fig. 2-5, we can obtain the optimal solution when evolving less than 200 generations every time, the biggest evolution generations as *MAXGEN* = 200, and at the same time, the generation gap *GGAP* = 0.9. Based on different value P_c, P_m, comparing the value *Y* objective function, the results of comparison as shown in Table 2. The results show that under the condition of the crossover rate *Pc* = 0.9 mutation rate *Pm* = 0.2 the genetic algorithm finds the optimal solution 15309 then the corresponding chromosome is Chrom= （15 11 12 13 14 5 9 3 7 10 2 8 1 6 4）;

Table 2. Different crossing rate and mutation rate corresponding to the objective function value

experiment	1	2	3	4
Crossing rate P_c	P_c=0.8	P_c=0.8	P_c=0.9	P_c=0.9
Mutation rate P_m	P_m=0.2	P_m=0.1	P_m=0.2	P_m=0.1
The function value Y	15333	15329	15309	15528

In Figure 6 the solid line represents the improved genetic algorithm solution variation, the dashed line represents the original genetic algorithm solution variation when crossover rate Pc is 0.9 and mutation rate Pm is 0.2. The improved genetic algorithm finds the optimal solution Y=15292 and the corresponding chromosome is Chrom = （15 10 12 13 14 5 9 4 8 11 2 6 1 7 3）; The order in which the 15 kinds of the explosives are put in the corresponding 15 areas is that the 1st kind of explosive is put in the 15th area, and the third kind of explosive is put into the 12th area, and so on, until the 15 kinds of explosive are all put into the corresponding 15 areas. The 15 kinds of explosives stand for 15 specifications, which can optimize the sequence of the 15 kinds of specification explosives, i.e., the explosives with the diameters 16mm，32mm，50mm，60mm，70mm，80mm，90mm，100mm，110mm，120mm，130mm，140mm，150mm，160mm，180mm respectively are put in the areas 15,10,12,13,14,5,9,4,8,11,2,6,1,7,3.

According to the above equations (1) (2) and (3), each partition contains the location of the specific rows and columns, such as the nearest distance from entrances is the 1st area as shown in figure 1, the farthest distance from entrances is the 15th area, each location of the 1st area and the 15th area in figure 1 use rows and columns, such as location (2, 1) which represents line 2 column 1. According to the analysis that the 16 mm specifications of explosive are put in the 15th area, 150 mm specifications of explosive are put in the 1st area.

From the experimental results, it can be easily concluded that the improved genetic algorithm can well check “premature” phenomenon, prevent falling into local optimum, and accelerate the convergence speed. Therefore, the improved genetic algorithm can gain the searching speed to find the optimal solution, especially has obvious advantages that prevent producing “premature” phenomenon.

6 Conclusions

This paper presents a new approach to establish the optimized mathematical model of explosive warehouse. The emulsion explosive warehouse is taken as an example and the genetic algorithm is applied to solve the problem. An improved genetic algorithm is also used in the experiment, which can gain the searching speed to find the optimal solution and prevent “premature” phenomenon. The simulation results verify the feasibility of this method, which can improve the warehouse space utilization rate, optimize walk path that the explosive access and solve the operation optimization problems on the condition of the different expire date of explosives.

Acknowledgement. This work was supported by Guangdong Innovative Research Team Program (No. 201001D0104648280), Shenzhen Sc. & Tech. Research Funds (JC201005280619A), and Guangzhou Nansha Research Funds, and Guangdong Foshan Shunde Research Funds (2012YSQ7).

References

1. Sun, Y., Zhong, F.: Current Situation and Development Prospects of Production Equipment of Emulsion Explosives. Blasting 27(3), 94–96 (2010)
2. Li, Y.: RFID technology in detonator product safety steward of the application of the principle. Beijing university of science and technology, Beijing (2008)
3. Chen, Y., Zhang, J., Huang, L.-F.: Research of dangerous goods logistics system model based on RFID and GPRS technology. Packaging Engineering 29(5), 78–80 (2008)
4. Hai-Jun, Z., Pu-Xiu, Y., Jie, Z.: A Realization of Optimal Order-Picking for Irregular Multi-Layered Warehouse. In: Pro. of International Seminar on Business and Information, pp. 28–32 (2008)

5. Zhang, H.-J., Liu, B.-W.: A New Genetic Algorithm for Order-Picking of Irregular Warehouse. In: Proc. of International Conference on Environmental Science and Information Application Technology, pp. 121–124 (2009)
6. Long, P., Lu, J.-G.: Order picking optimization of automated warehouses based on the colony genetic algorithm. Computer Engineering & Science 34(3), 148–151 (2012)
7. Poon, T.C., Choy, K.L., Chan, F.T.S., Ho, G.T.S.: A real time warehouse operations planning system for small batch replenishment problems in production environment. Expert Systems with Applications 38(1), 8524–8537 (2011)

Weighted Mining Association Rules Based Quantity Item with RFM Score for Personalized u-Commerce Recommendation System

Young Sung Cho[1,*], Si Choon Noh[2], and Song Chul Moon[2]

[1] Department of Computer Science, Chungbuk National University, Cheongju, Korea
[2] Department of Computer Science, Namseoul University, Cheonan-city, Korea, Korea
youngscho@empal.com, {nsc321,moon}@nsu.ac.kr

Abstract. This paper proposes a new weighted mining technique based quantity item with RFM(Recency, Frequency, Monetary) score for personalized u-commerce recommendation system under ubiquitous computing, or pervasive computing environment. Traditional association rule mining ignores the difference among the transactions. In this paper, it is necessary for us to consider the quantity of purchased data by each rank of RFM score in order to have different weights for different transactions, to generate weighted association rules through weighted mining association rules based quantity item with RFM score, and to recommend the items with high purchasability according to the threshold for creative weighted association rules with w-support, w-confidence and w-lift. To verify improved performance, we make experiments with dataset collected in a cosmetic internet shopping mall.

Keywords: Association Rules, RFM, Weighted Association Rules Mining.

1 Introduction

Along with the advent of ubiquitous computing, or pervasive computing environment and the spread of intelligent portable device such as smart phone, PDA and smart pad has been amplified, a variety of services and the amount of information has also increased. It is becoming a part of our common life style that the demands for enjoying the wireless internet are increasing anytime or anyplace without any restriction of time and place[1],[4]. The customers need a recommendation system that can recommend item which they really want on behalf of them. In the ubiquitous computing, or pervasive computing environment for u-commerce recommendation service, it is important to recommend the proper item among large item sets. Therefore, if the recommendation system can recommend the suitable item which they really want, the customers are satisfied with the system. The possession of intelligent recommendation system is becoming the company's business strategy. A personalized recommendation system using data mining technique based on RFM to meet the needs of customers has been actually processed the research[1-5]. We can improve the accuracy of recommendation through the weighted mining technique

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 367–375, 2013.

based quantity item with RFM score so as to be able to generate the associated items' rules. As a result, we propose a new weighted mining technique using weight based on the quantity of purchased data aggregated from the whole data with the item RFM score for recommendation in u-commerce under ubiquitous networking environment. The next Sect. briefly reviews the literature related to studies. The Sect. 3 is described a new method for personalized recommendation system in detail, such as system architecture with sub modules, the procedure of processing the recommendation, the algorithm for proposing method. The Sect. 4 describes the evaluation of this system in order to prove the criteria of logicality and efficiency through the implementation and the experiment. In Sect. 5, finally it is described the conclusion of paper and further research direction.

2 Related Works

2.1 RFM

RFM method is generally known in database marketing and direct marketing. It is easy for us to recommend the item with high purchasability using the customer's score and the item's score. The RFM score can be a basis factor how to determine purchasing behavior on the internet shopping mall, is helpful to buy the item which they really want by the personalized recommendation. One well-known commercial approach uses five bins per attributes, which yields 125 cells of segment. The following expression presents RFM score to be able to create an RFM analysis. The RFM score will be shown how to determine the customer as follows, will be used in this paper. The variables (A, B, C) are weights. The categories (R, F, M) have five bins.

$$\text{RFM score} = A \times R + B \times F + C \times M \tag{1}$$

The RFM score is correlated to the interest of e-commerce[2]. It is necessary for us to keep the analysis of RFM to be able to reflect the attributes of the item in order to find the items with high purchasability. In this paper, we can use weighted mining association rules with weight based on the quantity of purchased data aggregated from the whole data with the item RFM score to recommend the item they really want exactly.

2.2 Collaborative Filtering

Collaborative filtering comes from the method based on other users' preferences. There are two types of the method. One is the explicit method which is used user's profile for rating. The other is the implicit method which is not used user's profile for rating, The implicit method is not used user's profile for rating but is used user's web log patterns or purchased data to show user's buying patterns so as to reflect the user's preferences. There are some kinds of the method of recommendation, such as collaborative filtering, demographic filtering, rule-base filtering, contents based filtering, the hybrid filtering which put such a technique together and association rule and so on in data mining technique currently. The explicit method can not only reflect exact attributes of item, but also still has the problem of sparsity and scalability, though it has been practically used to improve these defects. In this paper, using a implicit method without onerous question and answer to the users, not used user's profile for rating, it is necessary for us to be able to reflect the attributes of the item in order to recommend the items with high purchasability.

2.3 Association Rules

Association rule mining aims to explore large transaction databases for association rules, which may reveal the implicit relationships among the data attributes. It was first introduced by Agrawal [6]. It has turned into a thriving research topic in data mining and has numerous practical applications such as market basket analysis including cross marketing, recommendation system in e-commerce. To select interesting rules from the set of all possible rules, the best-known constraints on various measures of significance and interest can be used. The constraints are minimum thresholds on support and confidence. Association rules which satisfy a minimum confidence threshold are then generated from the frequent itemsets. The association rules of the apriori algorithm can be divided into two steps. The first step finds a large itemsets larger than a minimum support. The second step finds larger rules than minimum confidence by creating all subsets of a large itemsets that are discovered at the first step. If the rule satisfies a specified confidence threshold requirement, then the candidate item is added to the recommendation set. Association rules which satisfy a minimum confidence threshold are then generated from the frequent itemsets. The traditional association rule mining employs the support measure, which treats every transaction equally. However, in our real world data sets, the weight / importance of a pattern may vary frequently due to some unavoidable situations. Usually in an association rule, it is expressed in the form of the rule *X*→*Y*. The rule of *X*→*Y* means that the transaction including the item of *X* tends to include the itemsets. And then in a weighted association rule, the w-support of a weighted association rule *X*→*Y* is defined as

$$\mathrm{WSUPP}(X \rightarrow Y) = \boldsymbol{WSUPP(X \cup Y)} \tag{2}$$

and the w-confidence is

$$\mathrm{WCONF}(X \rightarrow Y) = \frac{\boldsymbol{WSUPP(X \cup Y)}}{\boldsymbol{WSUPP(X)}} \tag{3}$$

Basically, w-support measures how significantly X and Y appear together; w-confidence measures how strong the rule is. An weighted association rule mining becomes an important research issue in data mining and knowledge discovery by considering different weights for different items. It is necessary to consider these dynamic changes in different application area such as retail market basket data analysis. Much effort has been dedicated to association rule mining with pre-assigned weights [8],[9]. It is crucial to have different weights for different transactions in order to reflect their different importance and adjust the mining results by emphasizing the important transactions.

3 Our Proposal for a Personalized u-Commerce Recommendation System

3.1 System Architecture

We can depict the system configuration concerning the personalized u-commerce recommendation system using weighted mining association rules based quantity item with RFM score under ubiquitous computing, or pervasive computing environment. This system had four agent modules which have the analytical agent, the recommendation agent, the learning agent, the data mining agent in the internet shopping mall environment. We observed the web standard in the web development, so developed the interface of internet to use full browsing in mobile device. As a matter of course, we can use web browser in wired internet to use our recommendation system. We can use the system under WAP in mobile web environment by using feature phone as well as using the internet browser such as safari browser of iPhone and Google chrome browser based on android so as to use our system by using smart phone.

3.2 Weighted Mining Association Rules Based Quantity Item with RFM Score for Personalized u-Commerce Recommendation System

In this part, we can depict weighted mining association rules based on the quantity item with RFM score for personalized u-commerce recommendation system. Our algorithm can consider situation where the weight / importance of a pattern may vary dynamically in e-commerce on the real world. It is necessary for us to consider the quantity of purchased data by each rank of RFM score in order to have different weights for different transactions and to generate weighted association rules through weighted mining association rules based quantity item with RFM score to recommend the items with high purchasability according to the threshold for creative weighted association rules with w-support, w-confidence and w-lift. At first, we can aggregate the quantity of purchased data by each rank of RFM score, divided by each 20%, which is aggregated counts of a rank from the whole data(sale) with the item RFM score. After that, we can make the rate of weight using aggregated counts of a section, that is, it is become the value of weight based on the quantity of purchased data.

It is crucial to have different weights for different transactions and adjust the mining results by emphasizing the important transactions. It is necessary for us to keep the scoring of RFM to be able to reflect the attributes of the item in order to use the dynamic weights in proposing method of mining. As a matter of course, we can use the weighted association rules mining (WARM) using the weight based quantity item with RFM score for generating association rules with w-support, w-confidence and w-lift through weighted mining association rules (WARM). The procedural algorithm for weighted mining based quantity item with RFM score for personalized u-commerce recommendation system is depicted as the following Table 1.

Table 1. The procedural algorithm for weighted mining based quantity item with RFM score for personalized u-commerce recommendation system

Step 1 : The RFM score of item is computed so as to to reflect the attributes of the item, consists of three attributes, each attribute has five bins divided by each 20%, exact quintile. As a result, we can make the RFM scores of items.

Step 2 : We can aggregate the quantity of purchased data by each rank of RFM score divided by each 20%, which is aggregated counts of a section from the whole data with the item RFM score, make the rate of weight.

Step 3 : We can make the rate of weight for items through aggregated counts of a section. This is the value of weight based on the quantity of purchased data.

Step 4 : Association rules are created by WARM

Step 5 : *Wsupport* = $\sum_i^N Weight_i/N$ X *Support count* /* *N is numbers of item in the rules */*

Step 6 : *We can create creative association rules with w- support, w-confidence and w-lift through weighted mining based on the quantity item of purchased data.*

Step 7 : We can scan whole database(sale) and generate association rules through weighted mining with weight based on the quantity with RFM score .

Step 8: We can recommend the item with high purchasability according to the threshold for creative weighted association rules with w-support, w- confidence and w-lift.

3.3 The Procedural Algorithm for Personalized u-Commerce Recommendation System

The login user can read users' information and recognize the code of classification such as demographic variables : age, gender, occupation, region.and user’s score. The system can search the information in the cluster of the code of classification equal to the login user. It can scan the preference as the average of brand item in the cluster, suggest the brand item in item category selected by the highest probability for preference as the average of brand item. This system can create the list of recommendation with TOP-N of the highest preference of item to recommend the item with purchasability efficiently. This system can recommend the items with efficiency, are used to generate the recommendable item according to the basic threshold for weighted association rules, with w-support, w-confidence and w-lift. It can recommend the associated item to TOP-N of recommending list if users want to have the cross-selling or up-selling. This system takes the cross comparison with purchased data in order to avoid the duplicated recommendation which it has ever taken.

3.4 The Analysis of Application for Weighted Mining Based Quantity Item with RFM Score for Personalized u-commerce Recommendation System

In this part, we can have the experimental analysis of validity to do the process of mining for experimental evaluation. We have made two experimental tasks of mining process in order in the same condition to compare the result of mining association rules. One is the test of Ordinary mining association rules in original data (sale). The other is the test of proposing weighted mining association rules with weight based on the quantity of purchased data aggregated from the whole data with the item RFM score so as to make the solution of mining improved by the performance as the metrics with support count, rule count, support, confidence and lift. The result of experimental test is the metrics of association rules with support, confidence and lift as the following Table 2. The result of proposing mining is higher average rates of association rules with support count, rule count, support, and lift than the ordinary mining even if the proposal is lower 6.892 in average of confidence than the original.

Table 2. The result on the performance for mining

	sale count	support count	Rule count	average sup_rate	average conf_rate	average lift_rate
Proposal	1,600	16,908	1,606	0.249	34.487	6.024
Ordinary	1,600	15,570	406	0.227	41.379	5.233

We use the same original data(sale), so both of method is similar processing time for mining at the first phase. And also, the result of proposing mining for processing time is faster than existing mining(Ordinary) because proposing method(Proposal) create association rules in Table 2 using frequent patterns mining rapidly when new data are added persistently at the second phase. It is important for us to process the large databases in e-commerce on the real world. As a result of that, we can obtain the result of the performance of mining as the metrics with support count, rule count, support and lift though confident rate is less than the mining of original method. We can describe the evaluation of this system in order to prove the criteria of logicality and efficiency through the implementation and the experiment in next Sect. 4.

4 The Environment of Implementation and Experiment and Evaluation

4.1 Experimental Data for Evaluation

We used 319 users who have had the experience to buy items in e-shopping mall, 580 cosmetic items used in current industry, 1600 results of purchased data recommended in order to evaluate the proposing method. In order to do that, we make the implementation for prototyping of the internet shopping mall which handles the cosmetics professionally and do the experiment. We have finished the system implementation about prototyping recommendation system. The experimental datasets could be made by the learning data set for 12 months and testing data set for 3 months in a cosmetic internet shopping mall[4]. To verify improved performance, we made experiments with dataset collected in real datasets environment under a cosmetic internet shopping mall.

4.2 Experiment and Evaluation

In this part, we can describe the experimental methodology and metrics, we can use to compare different mining algorithms; and present the results of our experiments.

We can make the task of clustering of item category based on purchased data for preprocessing under ubiquitous computing environment. We carry out the experiments in the same condition with dataset collected in a cosmetic internet shopping mall. The 1st system of weighted mining association rules based quantity item with RFM score, is proposing method(W_ARM) called by "proposal", the 2nd system is the original method(O_ARM) using the ordinary association rules mining, the third system is existing system. The proposing method's overall performance evaluation for recommendation is precision, recall and F-measure as comparing proposing method using (W_ARM) and the original method using (O_ARM). The performance was performed to prove the validity of recommendation and the system's overall performance evaluation. The metrics of evaluation for recommendation system in our system was used in the field of information retrieval commonly[10].

Table 3. The result for table of precision, recall, F-measure for recommendation ratio by each cluster

Cluster	Proposal(W_ ARM)			Original((O_ARM)			Existing		
	Precision1	Recall1	F-measure1	Precision2	Recall2	F-measure2	Precision3	Recall3	F-measure3
C1	35.24	70.00	46.88	46.20	55.70	45.90	48.79	31.32	35.64
C2	33.18	62.42	43.33	43.20	52.53	45.76	49.36	29.54	35.06
C3	34.00	65.66	44.80	40.99	23.93	30.05	44.26	21.81	27.65
C4	36.39	68.84	47.61	33.56	37.23	34.68	50.93	36.60	39.64
C5	45.78	66.60	54.26	53.94	27.27	34.90	47.41	26.81	32.26
C6	39.78	71.90	51.22	45.07	37.23	38.29	43.60	36.60	37.82
C7	42.52	75.49	54.40	64.08	28.45	37.19	46.68	25.19	30.28
C8	31.89	74.75	44.70	60.00	20.69	30.77	46.53	18.32	25.10
C9	44.90	80.23	57.57	73.85	62.50	64.42	67.23	55.34	57.10

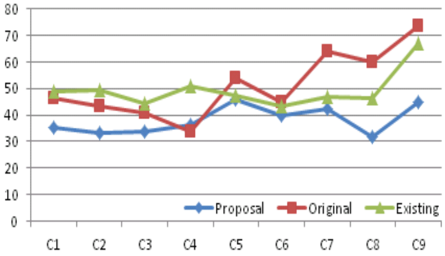


Fig. 1. The result of recommending ratio by precision

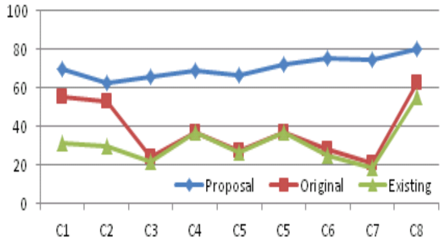


Fig. 2. The result of recommending ratio by recall

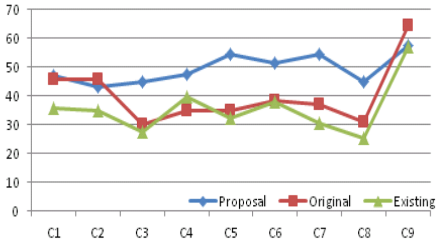


Fig. 3. The result of recommending ratio by F-measure

Fig. 4. The result of recommending items of cosmetics

Above Table 3 presents the result of evaluation metrics (precision, recall and F-measure) for recommendation system. The weighted mining association rules based quantity item with RFM score is improved better performance of proposing method using (W_ARM) than the original method using (O_ARM). The proposed higher 32.26% in recall even if it is lower 13.02% in precision than the original method using (O_ARM), higher 9.2% in F-measure than the original method. After that, it shows that our algorithm is very efficient and scalable for the recommendation system. Above figure 4 is shown in the result of screen on a smart phone. The performance of proposing mining method was improved more counts of support and rule than the original method, it was especially worthy of notice, in the rule counts, had an effect about 4 times what the original mining method did before. As a result, it was efficient for us to recommend the items of association because it is strong cohesion of the attribute of item based weighted association rules using the weight based quantity item with RFM score. So, we could have the recommendation system to be able to recommend the items with high purchasability. Above figure 4 is shown in the result of screen on a smart phone. The performance of proposing method is improved although it is less in average of confidence (average confi_rate), however it is efficient for us to recommend the items of association because it is strong cohesion of the attribute of item because of using a new weighted mining technique based on the quantity item with RFM score.

5 Conclusion

Recently u-commerce as a application field under ubiquitous computing, or pervasive computing environment, is in the limelight. Existing algorithms for weighted association rule mining are based on fixed weight, do not reflect the weight / importance of a pattern, and do not consider these dynamic changes in different application area such as retail market basket data analysis. It was necessary for us to keep the scoring of RFM to be able to reflect the attributes of the item in order to use the dynamic weights in proposing method of mining. As a result, we proposed a new weighted mining technique using the weight based quantity item with RFM score for personalized u-commerce recommendation system in real datasets environment in order to to improve the accuracy of recommendation with high purchasability. As a matter of course, we have described that the performance of the proposing method with mining using the weight based quantity item with RFM score is improved better than the original method and existing system. To verify improved better performance of recommendation, we carried out the experiments in the same dataset collected in a cosmetic internet shopping mall. It is meaningful to present a new mining technique using the weight based quantity item with RFM score for personalized u-commerce recommendation system under ubiquitous computing, or pervasive computing environment. The following research will be looking for a personalized recommendation in semantic web environment by neutral network clustering approach to increase the efficiency and scalability under ubiquitous computing, or pervasive computing environment**.**

Acknowledgements. This paper was supported by funding of Namseoul University

References

1. Cho, Y.S., Ryu, K.H.: Personalized Recommendation System using FP-tree Mining based on RFM. In: 17th KSCI, vol. 2 (February 2012)
2. Jin, B.W., Cho, Y.S., Ryu, K.H.: Personalized e-Commerce Recommendation System using RFM method and Association Rules. In: 15th KSCI, vol. 12, pp. 227–235 (December 2010)
3. Cho, Y.S., Moon, S.C., Noh, S.C., Ryu, K.H.: Implementation of Personalized recommendation System using k-means Clustering of Item Category based on RFM. In: 2012 IEEE International Conference on Management of Innovation & Technology Publication (June 2012)
4. Cho, Y.S., Moon, S.C., Jeong, S.P., Oh, I.B., Ryu, K.H.: Clustering Method using Item Preference based on RFM for Recommendation System in u-Commerce. In: Cho, Y.S., Moon, S.C., Jeong, S.-P., Oh, I.-B., Ryu, K.H. (eds.) Ubiquitous Information Technologies and Applications. LNEE, vol. 214, pp. 353–362. Springer, Heidelberg (2012)
5. Cho, Y.S., Moon, S.C., Ryu, K.H.: Mining Association Rules Using RFM Scoring Method for Personalized u-Commerce Recommendation System in Emerging Data. In: Kim, T.-h., Ramos, C., Abawajy, J., Kang, B.-H., Ślęzak, D., Adeli, H. (eds.) MAS/ASNT 2012. CCIS, vol. 341, pp. 190–198. Springer, Heidelberg (2012)
6. Agrawal, R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Datasets. In: Proc. ACM SIGMOD 1993, pp. 207–216 (1993)
7. Agrawal, A., Faloutsos, C., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington, D.C., pp. 207–216 (May 1993)
8. Ramkumar, G.D., Ranka, S., Tsur, S.: Weighted Association Rules: Model and Algorithm. In: Proc. ACM SIGKDD (1998)
9. Tao, F., Murtagh, F., Farid, M.: Weighted Association Rule Mining Using Weighted Support and Significance Framework. In: Proc. ACM SIGKDD 2003, pp. 661–666 (2003)
10. Herlocker, J.L., Kosran, J.A., Borchers, A., Riedl, J.: An Algorithm Framework for Performing Collaborative Filtering. In: Proceedings of the 1999 Conference on Research and Development in Information Research and Development in Information Retrival (1999)

Priority-Based Live Migration of Virtual Machine

Bangjie Jiang[1], Junmin Wu[1,2], Xiaodong Zhu[1], and Die Hu[1]

[1] Department of Computer Science and Technology
University of Science and Technology of China, Hefei, China
bjjiang@mail.ustc.edu.cn
[2] Suzhou Institute for Advanced Study
University of Science and Technology of China, Suzhou, China
jmwu@ustc.edu.cn

Abstract. Live migration of Virtual Machines (VMs) has been a powerful tool to facilitate system maintenance, load balancing, fault tolerance, and power saving. In this paper, we describe the design and implementation of a novel, priority-based approach for the live migration of VM that can greatly reduce VM service downtime. Our approach is mainly used in the desktop virtual environment where there are more than one application running in the VM. Our scheme offers applications that demand short service downtime the high migration priority while applications that can tolerate long VM service downtime are assigned with the low migration priority. During the iterative copy and the stop-and-copy phases, our scheme only transfers all dirty pages that belong to the high priority applications, so the service downtime will be less than that of pre-copy. Compared with pre-copy based live migration, the proposed approach can significantly reduce 57% of the service downtime of high priority applications.

Keywords: Priority-based, Migration Time, Live Migration, Virtual Machine.

1 Introduction

Live migration of VMs allows an administrator to move a running VM between different physical machines without disconnecting the client. The most important is the migration of VM memory. To achieve this, we copy all the memory pages from source to the destination while the VM is still running on the source. In case that some memory pages are modified (i.e. dirty pages) during the memory copy process, they will be retransmitted until the rate of retransmitting pages is not less than the rate of generating dirty pages. And then stop the VM and copy the remaining dirty pages. This is the principle of VM live migration using pre-copy method. Many hypervisor such as VMware, XEN and KVM adopt the pre-copy based approach for live migration of VMs [2].

When migrating a VM in the desktop virtualization environment where there are more than one application running in the VM, the system may have high memory write rate and produces large amounts of memory dirty pages, leading to a long service downtime and total migration time in low-speed network. Among the multiple

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 376–385, 2013.

applications, some are expected to be migrated quickly while the other can better tolerate VM service interrupt. For instance, when a VM that runs music playing application is migrated, the users generally expect small service downtime to ensure satisfying user experience. The use of pre-copy based on live virtual migration is difficult to meet this requirement in the low-speed network.

In this paper, we propose a novel, priority-based VM migration approach to accelerate the migration of some of the applications in the VM. Our approach is based on the pre-copy mechanism [1] [3]. In the pre-copy the VM is migrated as a while, while our approach gives priority to migrate applications in the VM that are sensitive to downtime. In order to distinguish different applications in VM, we proposed a scheme which manages different applications under VMM without modifying the guest OS. Through this scheme, we are able to separate the dirty pages generated by different applications in VM. When a VM is migrated to the target node, our technique transfers those dirty pages generated by the high-priority applications first. The dirty pages generated by the low-priority applications are temporarily recorded on the local host and then are transmitted to the destination host at the appropriate time. When the number of dirty pages for high-priority applications reduces to a certain threshold, we suspend the source VM, and then transfer all remaining dirty pages generated by high-priority applications and all devices states. At last, in the target node we start the target VM. For those low-priority applications, as part of the dirty pages are not transmitted to the destination node, we suspend the execution of these applications through the signal mechanism. Then we transfer the remaining dirty pages generated by low-priority applications while running VM. When the transmission of the rest of the dirty pages is completed, we continue to run the rest applications through signal mechanism.

The service downtime of the live VM migration mainly depends on the dirty pages transferred at the stop-and-copy phase. By our approach, we only transfer all dirty pages generated by high priority applications at the stop-and-copy phase, so the service downtime of these high priority applications will be also extremely short.

2 The Design of Priority-Based Live VM Migration

In this section, we present the architectural overview of priority-based live VM migration. In our approach, a network-accessible storage system (such as SAN or NAS) is employed. Only memory and CPU states need to be transferred from the source node to the target one.

2.1 An Overview

Our priority-based VM live migration consists of two parts. As is shown in Figure 1, the first part, called MAVM, is mainly responsible for managing different applications in the VM. The second part is responsible for managing the priority-based live VM migration. The migration daemon in the source node is primarily responsible for setting the priorities of applications and migrating the VM to the target node while the migration daemon in the target node is responsible for receiving the dirty pages transferred from the source host and run the VM. However, the most important role for migration daemon in the target node is controlling the execution of the low-priority applications via the signaling mechanism.

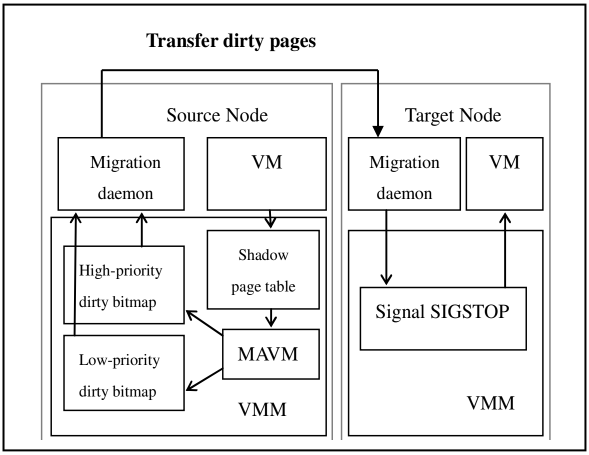


Fig. 1. The overall architecture of priority-based live migration of VM

2.2 The Design

The logical flow of the proposed approach is summarized in Figure 2. We take an innovative approach to the management of migration. To achieve this goal, we view the migration process as a transactional interaction between the two hosts involved.

Stage 0: Pre-Migration. An active VM is running on physical host A. When a request is issued to migrate the VM from host A to host B, our approach confirms that the necessary resources are available on B and reserves a VM container of that size.

The next step is to set the priorities of applications in the VM. The migration daemon sets the priorities of all applications as needed based on the messages of applications stored by MAVM.

Stage 1: Iterative Pre-Copy. The copying of all dirty pages to target host occurs. Initially, our approach sets all memory pages in the VM to dirty pages. This suggests that our method copies all memory pages to target host during the first iteration. During the following iterations, our technique copies only those pages modified by the high-priority applications during the previous transfer phase. For those dirty pages generated by low-priority applications, our scheme records them in the low-priority dirty bitmap.

Stage 2: Stop and Copy. When the dirty pages generated by high-priority applications fall below a threshold, our scheme suspends the running VM. As described earlier, CPU state and any remaining inconsistent memory pages generated by high-priority applications are then transferred. At the end of this stage, all memory pages modified by low-priority applications are also stored in the source host.

Stage 3: Run the VM. Note that our scheme suspends the low-priority applications while running the VM in the target host since all memory pages modified by the low-priority applications have not yet been transferred to the target host. Our scheme sends the SIGSTOP signal to suspend low-priority applications. At this time there are only high-priority applications running normally in the VM while low-priority applications are suspended.

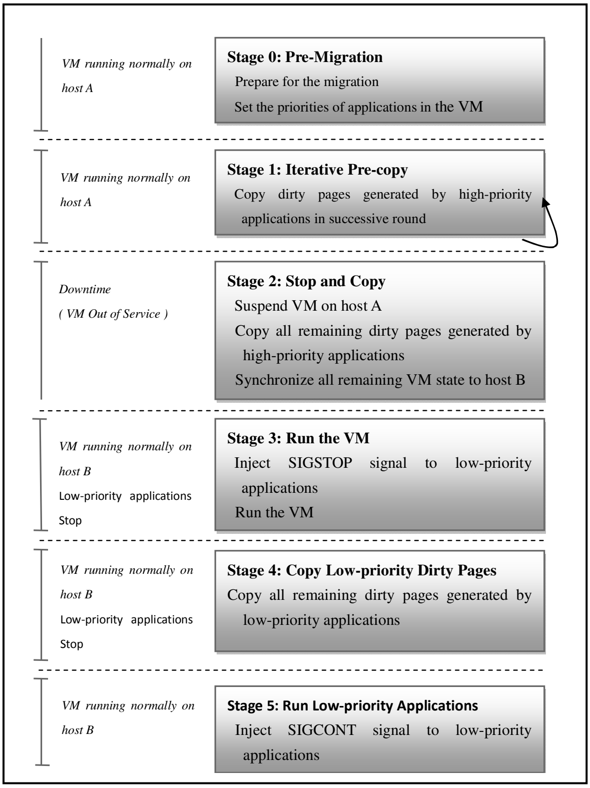

VM running normally on host A
Stage 0: Pre-Migration
Prepare for the migration
Set the priorities of applications in the VM
VM running normally on host A
Stage 1: Iterative Pre-copy
Copy dirty pages generated by high-priority applications in successive round
Downtime
(VM Out of Service)
Stage 2: Stop and Copy
Suspend VM on host A
Copy all remaining dirty pages generated by high-priority applications
Synchronize all remaining VM state to host B
VM running normally on host B
Low-priority applications Stop
Stage 3: Run the VM
Inject SIGSTOP signal to low-priority applications
Run the VM
VM running normally on host B
Low-priority applications Stop
Stage 4: Copy Low-priority Dirty Pages
Copy all remaining dirty pages generated by low-priority applications
VM running normally on host B
Stage 5: Run Low-priority Applications
Inject SIGCONT signal to low-priority applications

Fig. 2. Migration timeline

Stage 4: Copy Low-Priority Dirty Pages. For the source host, our scheme initializes the transmission of all remaining low-priority dirty pages to target host and the target host receives dirty pages while running the VM.

Stage 5: Run Low-Priority Applications. When all dirty pages generated by low-priority applications are transferred to the target host, we can run the low-priority applications. In this paper we continue to run these applications by sending the SIGCONT signal to them in the VM.

2.3 Control of Low-Priority Applications by Signaling Mechanism

Our approach of priority-based live VM migration sets different priorities for the applications. When we run the VM in the target node, some dirty pages generated by low-priority applications have not been transferred to target node. Therefore, we must suspend these low-priority applications; otherwise they will perform incorrectly when running the VM. When all low-priority dirty pages are transferred to target node, we will also wake up these low-priority applications. In this paper, we achieve these functions through signaling mechanism.

We use the signal of SIGSTOP and SIGCONT to control the low-priority applications. The primary role of the former is to stop the execution of the process while the later signal is used to resume the execution of process, which has been suspended.

Before running the VM in the target node, the signal domain in the corresponding process descriptor of low-priority applications is set to SIGSTOP. When guest OS schedules these low-priority applications, it will detect the corresponding signal in the process descriptor and then suspend the execution of these applications. So in the guest OS there are only high-priority applications running normally while those low-priority applications are suspended. When all low-priority dirty pages are transferred to target node, we wake up these applications. Unfortunately, we can’t wake up them under VMM. Because these applications have been suspended, they will not take the initiative to deal with the corresponding signal. So we resume the execution of applications through sending a SIGCONT signal to these applications in the VM, so that the guest OS will take the initiative to activate these applications.

2.4 Dynamically Setting the Priority

In the above description, before the VM is migrated, we must manually set the priorities of applications. This is very troublesome and sometimes the migration of VMs is done automatically without the participation of the managers. So we propose a dynamical method to set the applications priorities.

Processes in the Linux system are divided into the ordinary processes and real-time processes. Real-time processes have real-time requirements which need to be responded quickly. And real-time process has a high scheduling priority. So when the VM is migrated, these real-time processes, such as video and audio applications, are set to high priorities automatically while other applications are set to low priorities according to the scheduling priority of the process in kernel automatically without the participation of the migration manager.

3 Managing Applications in the VM (MAVM)

Currently, in the fully virtualized environment, The VMM manages the Guest OS as a whole. Live migration of VM does not discriminate between different applications in the VM. It transfers all dirty pages generated by applications running in the VM to target node. While in this paper we implement a novel approach priority-based VM migration which needs to set different priorities to the applications. Therefore, we have to manage applications in the VM. In this section, we propose a scheme for managing the different applications in VMM without modifying the Guest OS. The module of MAVM mainly consists of two parts. One part is obtaining the process descriptor that belongs to the process running in the guest OS while the other part is primarily responsible for recording the dirty pages generated by the different priority-based applications.

3.1 Getting the Process Descriptor that Belongs to the Process in Guest OS

Our scheme is implemented in Kernel-based Virtual Machine (KVM) [7] [8] [9] and takes Ubuntu 11.04 as guest and host OS. By obtaining the process descriptor which belongs to the process in guest OS under VMM, we can achieve the process of management of the Guest OS under the VMM. Note that our approach is implemented in the VMM which does not modify guest OS.

As illustrated in Figure 3, Process descriptor is represented by a *Task_struct* structure in Linux kernel. All process-related information is stored in this structure. *Thread_info* structure is introduced in the 2.6 Linux kernel which is used to store process information frequently accessed.

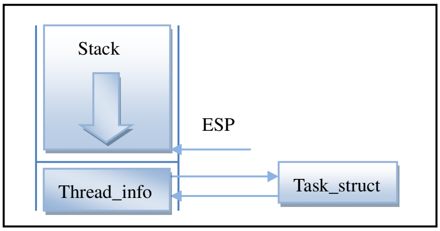


Fig. 3. The relationship of Task_struct, *Thread_info* and kernel stack

When guest OS switches the process in the VM, VMM will intercept the operation of setting page directory base address. At this point, the value of ESP register can be obtained. And then it is easy to obtain the corresponding process descriptor (*Task_struct*) through executing the following three commands according to the Figure 3. Through this method, we can get all of the information for processes running in the VM [10].

```
movl $0xffffe000,%ecx /*0xfffffe000 for 4KB kernelstack*/
andl %esp,%ecx        /* esp stores kernel stack */
movl (%ecx),p         /*p pointes to current process de-
                      scriptor */
```

3.2 Recording the Dirty Pages Generated by the Different Priority-Based Applications

In the pre-copy approach based on live VM migration, we iteratively transfer dirty pages generated by the VM. These dirty pages are recorded using a dirty pages bitmap. When the data in the memory is modified, our method set the corresponding bit to 1 in the dirty page bitmap. In this paper, we use high priority bitmap and low priority bitmap to keep track of dirty pages generated by applications of different priorities.

When the process in the VM writes the data to memory, VMM will intercept this operation. Since our method records the process currently running in the VM and set the priorities of all process, it is easy to know the current priority of the process. Then it can set the dirty page to the corresponding priority bitmap.

4 Evaluation

In this section, we present a detailed evaluation of our priority-based VM migration implementation and compare it to KVM's pre-copy migration. Firstly we describe our experimental setup. And then we describe the detailed experimental results based on the desktop virtualization environment.

4.1 Experimental Setup

We conduct our experiments on several identical server, each with 2-way quad-core Xeon E5620 2.4 GHz CPUs and 8GB DDR RAM. The machines have Intel Corporation 80003ES2LAN gigabit network interface card (NIC) and are connected via switched gigabit Ethernet. We used ubuntu11.04 as the guest and host OS with kernel 2.6.38 in all cases. All the VMs are configured to use 512MB of RAM. We primarily consider three performance metrics: migration downtime, total migration time, and the whole amount of data transferred during live migration of VM.

4.2 Application Scenarios

As our approach is mainly applied to the desktop virtualization environment, we select some of the daily personal applications running in the VM. The experiments use the following VM workloads:

1) *VLC* Media Player the media player is playing the music.
2) *Soffice* the office software is used to edit documents under Linux system.
3) *Firefox* we use firefox browser to browse news.
4) *Neverputt* a famous game is running in Ubuntu guest OS

Here, we emphasize on the four applications running in the VM together because we will set different priorities for the four applications when the VM is migrated. However, there are also other processes running in the VM, such as the kernel daemon. For simplicity, in our priority-based VM migration, we always set the rest processes to high priority.

Before describing the experiment, we first explain the meaning of the legend in the following figure. *Pre-copy* represents that the VM is migrated by pre-copy while the others denote that we migrate the VM to target node by priority-based live VM migration. For example, *VLC High Priority* means that we set the application of *VLC* to high priority while the other applications, such as *Firefox*, *Soffice* and *Neverputt*, are set to low priority. *Priority-based* refers to the priorities that are set dynamically.

Total Data Transferred. Figure 4 shows that our priority-based scheme reduces total data transferred during the whole migration process, compared with pre-copy. Although our scheme and pre-copy have the same dirty pages in the first round, smaller amount of data have been sent to the target by priority-based live migration of VM from the second round due to not transmitting low priority dirty pages, resulting that dirty pages generated by low priority are only transferred once. Experimental results show that our scheme reduces the total transferred data of VM migration by about 8%.

Total Migration Time and Downtime. Total migration time and downtime are two key performance metrics that clients of VM service care about the most, because they are concerned about service degradation and the duration that service is completely unavailable.

To compare the migration downtime of our scheme with pre-copy scheme, the same four workloads are running in the VM, which is migrated with the different speed LAN to transfer dirty pages. The test result in Figure 5 shows that our approach yields much less downtime than pre-copy.

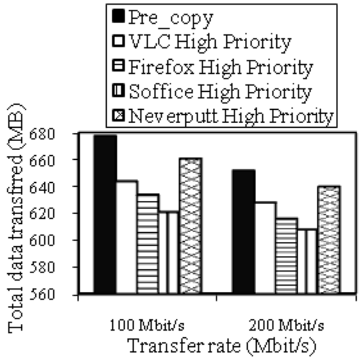


Fig. 4. Total transferred data of pre-copy and priority-based during live migration for different transferring rate

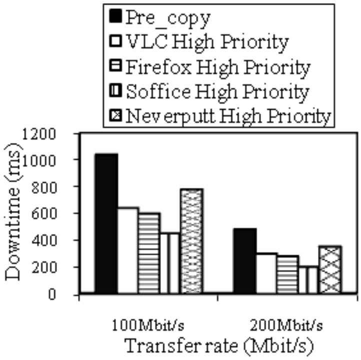


Fig. 5. The downtime of pre-copy and priority-based during live migration for different transferring rate

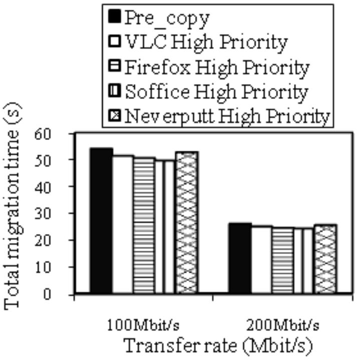


Fig. 6. Total migration time of pre-copy and priority-based during live migration for different transferring rate

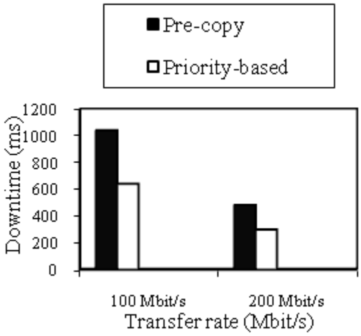


Fig. 7. The downtime of pre-copy and priority-based during live migration for different transferring rate by setting priorities dynamically

We see that it reduces the migration downtime by more than 57% when we set the application of *Soffice* to the high priority and the other applications to low priority. When we set the application of *Neverputt* to high priority and the other applications to low priority the downtime will only be reduced by 12%.

The test results in Figure 6 show that, compared with pre-copy, our scheme has shorter total migration time. This should be attributed to smaller rounds and less data transferred in each round. Our system reduces total migration time by an average of 5.5%.

4.3 Dynamically Setting the Priority

In order to verify the function of dynamically setting the priorities of applications, we remain of the four applications described above running in the VM. As is shown in Figure 7, the downtime is 298ms with the transferring rate of 200Mbit/s by priority-based live migration of VM. This is approximately equal to downtime when we set the application of *VLC* to high priority by priority-based live migration of VM as shown in Figure 5. This indicates that when VM is migrated without setting priorities of applications manually, the system will set the application of *VLC* to high priority automatically.

5 Related Work

Clark et al. [1] discussed live migration of VMs using pre-copy mechanism, which maintains small downtime by minimizing the amount of dirty pages generated by VM that needs to be transferred. Although pre-copy minimizes downtime, it reduces effectiveness and increases total migration time since pages that are repeatedly modified may have to be transmitted multiple times.

Zawet al. [2] achieved efficient working set prediction by pre-copy. A working set prediction algorithm is proposed as a preprocessing step, which postpones the transmission of those dirty pages modified frequently in order to reduce the total migration time.

In post-copy [4], all memory pages are transferred only once during the whole migration process and the baseline total migration time is achieved.

MECOM [5] first introduced memory compression technique into live VM migration. Based on memory page characteristics, MECOM design a specific memory compression algorithm for live migration of VMs.

Liu et al. [6] described the design and implementation of a novel approach CR/TR-Motion that adopts checkpointing/recovery and trace/replay technology to provide fast, transparent VM migration.

6 Conclusions and Future Work

In this paper, we present the design and implementation of a priority-based technique for live migration of VMs, which speeds up the applications for migration in VM by prioritizing the applications. Our approach is mainly used for the desktop virtual environment with low-bandwidth network. In order to distinguish different applications in VM, we propose a scheme which manages the different applications in VM without modifying the guest OS. Experimental results show that our approach can get better average performance than KVM by pre-copy: up to 57% on VM downtime.

We implement live migration of dynamically setting the priority. In the future, we will extend the technique to real-time systems which are used to meet the real-time requirements of the service.

This work is supported by the Fundamental Research Funds for the Central Universities (WK0110000020).

References

1. Clark, C., Fraser, K., Hand, S., Hansen, J.G., July, E., Limpach, C., Pratt, I., Warfield, A.: Live Migration of Virtual Machines. In: Proceedings of the 2nd USENIX Symposium on Networked Systems Design and Implementation (2005)
2. Zaw, E.P., Thein, N.L.: Live virtual machine migration with efficient working set prediction. In: 2011 International Conference on Network and Electronics Engineering (2011)
3. Khaled, Z.I., Hofmeyr, S., Iancu, C., Roman, E.: Optimized Pre-Copy Live Migration for Memory Intensive Applications. In: SC 2011 (2011)
4. Hines, M.R., Gopalan, K.: Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning. In: Proceedings of the ACM/Usenix International Conference on Virtual Execution Environments (VEE 2009), pp. 51–60 (2009)
5. Jin, H., Deng, L., Wu, S., Shi, X., and Pan, X.: Live virtual machine migration with adaptive memory compression. In: Proceedings of the 2009 IEEE International Conference on Cluster Computing (2009)
6. Siripoonya, V., Chanchio, K.: Thread-Based Live Checkpointing of Virtual Machines. In: 2011 IEEE International Symposium on Network Computing and Applications (2011)
7. Qumranet Inc. KVM: Kernel-based virtualization driver, `http://www.qumranet.com/wp/kvmwp.pdf`
8. Quramnet Inc. KVM: Migrating a VM, `http://kvm.qumranet.com/kvmwiki/Migration`
9. Kivity, A., Kamay, Y., Laor, D.: kvm: the linux virtual machine monitor. In: Proc. of Ottawa Linux Symposium (2007)
10. Kang, H.: `http://blog.csdn.net/kanghua/article/details/1820785`

Improvement of the MCMA Blind Equalization Performance Using the Coordinate Change Method in 16-APSK

Youngguk Kim and Heung-Gyoon Ryu

Department of Electronic Engineering
Chungbuk National University
Cheongju, Korea 361-763
coolfeelyg@naver.com, ecomm@cbu.ac.kr

Abstract. In this paper, we propose the coordinate change schemes for improving the performance of blind equalizer such as MCMA (modified constant modulus algorithm) which compensate for ISI channel effect. ISI (inter symbol interference) is generated due to user's movement in the mobile satellite communication environment. Satellite communication systems do not use pilot signals for channel estimation. Blind equalization techniques such as MCMA are well known that it is possible to estimate and to compensate for channel without pilot signals. It is necessary to improve the blind equalizer performance. Therefore, we propose the coordinate change schemes for improve the equalization performance in 16-APSK. We confirm that this proposed method has better equalization performance than conventional MCMA.

Keywords: blind equalization, coordinate change, BER performance, 16-APSK, MSE.

1 Introduction

In digital communication system, it's important to transmit more information data. According to the given power, the amount of information is limited based on information theory. Channel noise and inter symbol interference (ISI) are main factors to limit amount of information. Conventional adaptive equalizations are using the training sequence to estimate the channel characteristic. Through the channel characteristic, we estimate the characteristic coefficient of reverse channel. After then, transmit signals are passed, have a characteristic coefficient of reverse channel, the filter. Using this method, we reduce the ISI and random phase rotation influence. Therefore, the equalization can improve overall performance. Training sequence is promised signal between transmitter and receiver. In other words, training sequence is additional information. So, Bandwidth efficiency is decreased.

In blind equalization method, bandwidth efficiency problem is partially solved because of transmitted signal does not use the training sequence. Many researches

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 386–392, 2013.

about blind equalization have been studied to compensate channel effect using only received signal without the training sequences. Blind equalization of 16-QAM signal through coordinate change has already been studied [1]. We have to use the blind equalizer if receiver is in mobile status when there is no training sequence. In this blind equalization, it's very important to improve MSE performance to apply the blind equalization in mobile satellite communication system. Coordinate change is improved the equalization performance by reducing the modulus.

In the blind equalization, using the cumulative rate of received signal and using the constant modulus algorithm (CMA) is represented in a way. Inter symbol interference (ISI) and the phase rotation can be restored at the cumulative rate method. However, it requires high-level operation. So, high speed transmission may have a problem as equalization. In the CMA, ISI and phase rotation compensate is impossible at a time. However, this method has the advantage of reduces the amount of computation. CMA equalization method for updating the equalizer coefficients, using the LMS adaptive filtering algorithm the actual implementation is very simple. LMS method the Eigen value distribution of the correlation matrix of the input signal is large; the rate of convergence is slow. CMA blind equalization algorithm is one of the most used techniques. CMA can’t compensate phase rotation problem. but The MCMA accomplishes the correction of phase error and frequency offset with the modified cost functions.

This paper is organized as follows.

Section 2 MCMA introduced, and Section 3 describes the propose method. Section 4 through simulations evaluating the performance of the propose system, and finally concludes.

2 Modified CMA Algorithm

CMA algorithm is one of the most used algorithms [1]. The MCMA was proposed for correcting phase error based on CMA. Figure 1 shows a block diagram of MCMA.

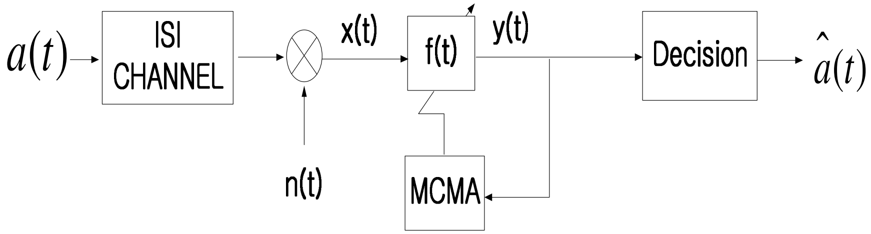


Fig. 1. Block diagram of MCMA blind equalization system

$a(t)$ is the transmitted signal, n(t) stands for the channel noise. $\hat{a}(t)$ is the signal after passing equalizer determined.

Input vector is

$$x(t) = [x(t), x(t-1), \cdots, x(t-N+1)]^T \tag{1}$$

Equalizer output signal $y(t)$ is as follows.

$$y(t) = f^T(t)x(t) \tag{2}$$

N-tap equalizer coefficients are defined as follows.

$$f(t) = [f_0(t), f_1(t), f_2(t), \cdots, f_{N-1}(t)]^T \tag{3}$$

The cost function of MCMA can be expressed as

$$J(n) = E[(|\operatorname{Re}(y(n))|^2 - R_{2,R})^2] + E[(|\operatorname{Im}(y(n))|^2 - R_{2,I})^2] \tag{4}$$

The following is the error function of the MCMA.

$$\begin{aligned} \mathrm{e}_R(t) &= y_R(t)(|y_R(t)|^2 - R_{R,2}) \\ \mathrm{e}_I(t) &= y_I(t)(|y_I(t)|^2 - R_{I,2}) \\ e(t) &= e_R(t) + je_I(t) \end{aligned} \tag{5}$$

s(t) is the transmitted symbol, the constant modulus of $R_{2,R}$ and $R_{2,I}$ are given by

$$R_R^{\;2} = \frac{E[|a_R(t)|^4]}{E[|a_R(t)|^2]}, \quad R_I^{\;2} = \frac{E[|a_I(t)|^4]}{E[|a_I(t)|^2]} \tag{6}$$

The tap coefficients are updated through the following equation (7).

$$f(t+1) = f(t) - \mu e(t)x(t) \tag{7}$$

μ is the step size value.

3 Coordinate Change Method

3.1 Coordinate Change of 16-APSK

Coordinate change is proposed to be used for 16-APSK signal. 16-APSK is composed of inner circle and outer circle. Inner circle have four symbols and outer circle have twelve symbols. The ratio of the inner circle and outer circle is expressed as follows.

$$\gamma = \frac{R_2}{R_1} \tag{8}$$

γ of 16-APSK signal has a value of 2.85, each symbol has a value of $\{\pm 1 \pm i$, $\pm 2.0153 \pm 2.0153i$, $\pm 2.7529 \pm 0.7376i$, $\pm 0.7376 \pm 2.7529i\}$. Change method can be seen in Table 1.

We need to obtain angular information before change the coordinate of the signal in the data values. Obtaining the angle is as follows.

$$\theta(t) = \tan^{-1}(y(t)) \tag{9}$$

Table 1. Coordinate Change for 16-APSK

Original Coordinates	New Coordinates	Original Coordinates	New Coordinates
1+i	1+i	1-i	1-i
2.0153+2.0153i	1+i	2.0153-2.0153i	1-i
2.7529+0.7376i	1-i	2.7529-0.7376i	1+i
0.7376+2.7529i	-1+i	0.7376-2.7529i	-1-i
-1+i	-1+i	-1-i	-1-i
-2.0153+2.0153i	-1+i	-2.0153-2.0153i	-1-i
-2.7529+0.7376i	-1-i	-2.7529-0.7376i	-1+i
-0.7376+2.7529i	1+i	-0.7376-2.7529i	1-i

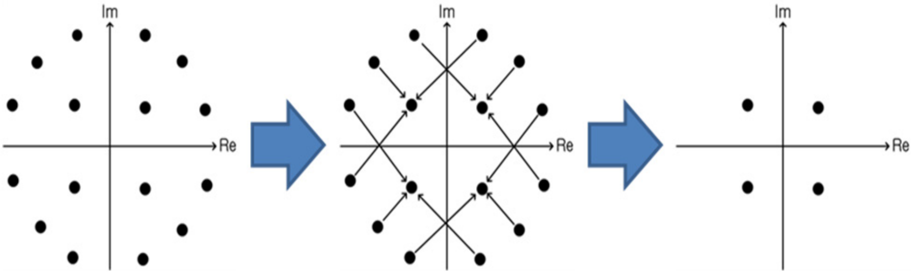


Fig. 2. Coordinate change constellation of 16-APSK

Coordinate change data can be obtained using angle information. Conversion equations are as follows.

$$\begin{aligned}
&\mathit{if}\,((0 <|\,\theta(t)\,|<= 30 \,||\, (150 <|\,\theta(t)\,|<= 180))\,\&\,\&\,|\,X\,|>= 2.13\\
&Y = [X_r - 1.7529 sign(X_r)] + i[X_i - 1.7376 sign(X_i)]\\
&\mathit{elseif}\,((0 <|\,\theta(t)\,|<= 30 \,||\, (150 <|\,\theta(t)\,|<= 180))\,\&\,\&\,|\,X\,|< 2.13\\
&Y = X\\
&\mathit{elseif}\,((30 <|\,\theta(t)\,|<= 60 \,||\, (120 <|\,\theta(t)\,|<= 150))\,\&\,\&\,|\,X\,|>= 2.13\\
&Y = [X_r - 1.0153 sign(X_r)] + i[X_i - 1.0153 sign(X_i)]\\
&\mathit{elseif}\,((30 <|\,\theta(t)\,|<= 60 \,||\, (120 <|\,\theta(t)\,|<= 150))\,\&\,\&\,|\,X\,|< 2.13\\
&Y = X\\
&\mathit{elseif}\,(60 <|\,\theta(t)\,|<= 120)\&\,|\,X\,|>= 2.13\\
&Y = [X_r - 1.7376 sign(X_r)] + i[X_i - 1.7529 sign(X_i)]\\
&\mathit{elseif}\,(60 <|\,\theta(t)\,|<= 120)\&\,|\,X\,|< 2.13\\
&Y = X
\end{aligned} \tag{10}$$

X means received signal. Y means coordinate changed signal.

3.2 Proposed MCMA Algorithm

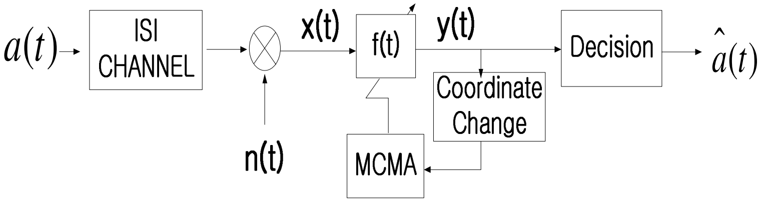


Fig. 3. Block diagram of the proposed scheme in MCMA

Coordinate change of $R_2^{'}$ is defined as follows.

$$R_{2,R}{}' = \frac{E[|[a_R{}'(t)|^4]}{E[|[a_R{}'(t)|^2]} , \quad R_{2,I}{}' = \frac{E[|[a_I{}'(t)|^4]}{E[|[a_I{}'(t)|^2]} \tag{11}$$

The output signal of the equalizer is

$$y(t) = f^T(t)x'(t) \tag{12}$$

The proposed error function is

$$\begin{aligned} e'_R(t) &= y'_R(t)(|y'_R(t)|^2 - R'_{2,R}) \\ e'_I(t) &= y'_I(t)(|y'_R(t)|^2 - R'_{2,I}) \\ e'(t) &= e_R(t) + je_I(t) \end{aligned} \tag{13}$$

Cost function of the proposed MCMA is as follows.

$$J'_{CMA}(f) = E[\{e'(t)\}^2)] \tag{14}$$

The tap coefficients are updated through the following equation.

$$f'(t+1) = f'(t) - \mu e'(t)x(t) \tag{15}$$

μ means the step size.

4 Simulation Results

In this paper, through a coordinate change of 16-APSK, examines the change in MSE performance and BER performance by reducing the modulus.

Figure 4, 5 compare equalization performance of the propose method and the conventional method such as BER and MSE.

Table 2. Simulation Parameter

Modulation	16-APSK
Channel	ISI Channel [0.8, 0.3, 0, 0.2+j0.2, 0, 0]

In the simulation, the ISI channel was used. SNR is 30dB. Equalizer has 21 tabs. Step size 0.00005 was used.

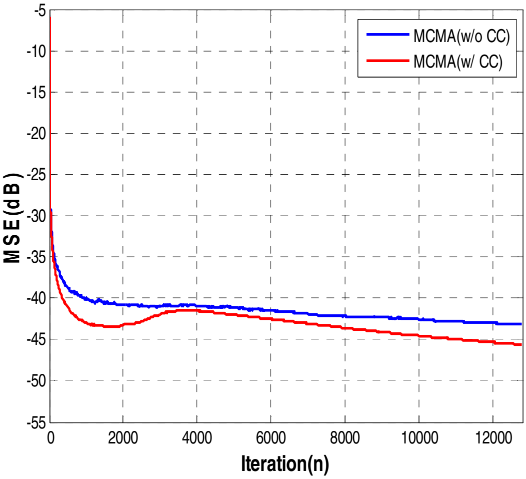


Fig. 4. Comparison of the MSE performances

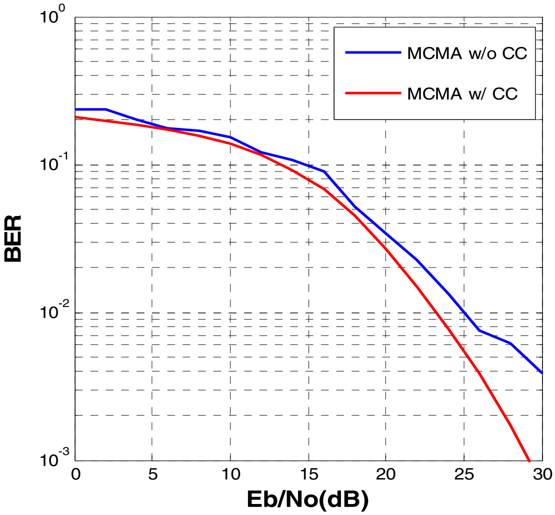


Fig. 5. Comparison of the BER performances

5 Conclusions

In this paper, we propose the MCMA algorithm and the coordinate change method to improve MSE and BER performance in 16-APSK system. We confirm that the proposed scheme achieves the MSE performance enhancement, compared with that of conventional MCMA blind equalization system at SNR=30dB. Proposed MCMA has better BER performance than conventional MCMA. The proposed scheme has a little error function because modulus value is decreased by using coordinate change. So, SAG-MCMA and MCMA with coordinate change has better receive performance than that of MCMA without coordinate change.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2012017339).

References

1. Rao, W., Yuan, K.-M., Guo, Y.-C., Yang, C.: A Simple Constant Modulus Algorithm for Blind Equalization Suitable for 16-QAM signal. In: The 9th International Conference on Signal Processing, vol. 2, pp. 1963–1966 (2008)
2. Godard, D.: Self-recovering equalization equalization and carrier tracking in two dimensional data communication systems. IEEE Trans. Communications COM-28, 1867–1875 (1980)
3. Johnson Jr., C.R., Schniter, P., Endres, J.T., et al.: Blind equalization Using the Constant Modulus Criterion: A Review. Proceedings of the IEEE 86(10), 1927–1949 (1998)
4. Rao, W., Guo, Y.-C.: New constant modulus blind equalization algorithm based on variable segment error function. Journal of System Simulation 19(12), 2686–2689 (2006)
5. Suzuki, Y., Hashimoto, A., Kojima, M., Sujikai, H., Tanaka, S., Kimura, T., Shogen, K.: A Study of Adaptive Equalizer for APSK in the Advanced Satellite Broadcasting System. In: Global Telecommunications Conference. GLOBECOM 2009, pp. 1–6. IEEE (2009)

Postural Transition Detection Using a Wireless Sensor Activity Monitoring System*

Richelle LeMay, Sangil Choi, Jong-Hoon Youn, and Jay Newstorm

College of Information Science and Technology, University of Nebraska at Omaha
Omaha, NE USA 68182
{rlemay,sangilchoi,jyoun,jnewstorm}@unomaha.edu

Abstract. Mobility health is an important aspect of the overall health status of a person. Many tests exist that determine the mobility health of a subject, but there are several issues associated with these tests, such as human error. Much work is being done to develop a mobility classification system which consolidates these tests, and circumvents the associated issues. Even so, many of these systems in development are complicated and lack the calculation of important postural transition measurements. The goal of this project was to remove the errors associated with current mobility tests, and to make the system as simple and energy-efficient as possible. In addition, we wanted this system to be able to detect with accuracy of over 90% six mobility states in addition to postural transition information. These goals were accomplished by using a waist-mounted triaxial accelerometer that processed data on-board using a well-developed classification algorithm.

Keywords: activity classification, mobility monitoring, sensor networks.

1 Introduction

Various tests exist to discern between health levels in the elderly. These tests typically rely on human observation, self-recording, or bio-mechanical observation of a patient. Both human observation and self-recording are prone to human error, while mechanical observation is typically very expensive as it requires both large and highly technical equipment [1]. As the capability of technology has increased, so has our ability to create a condensed, wearable, inexpensive methodology that predicts the mobility health level of a patient without human error or expensive equipment.

It has been shown that a triaxial accelerometer can categorize many types of human movement [8, 9, 10]. Creating a log of daily activities using this sensor would not only allow for the consolidation of human observation, self-recording, and bio-mechanical observation tests, but would also eliminate the associated human error.

Typical mobility states that are detectable using a wireless sensor include: sitting, standing, lying, walking and running. While most of these states cover the typical movement of a person throughout their day, a key factor used in determining mobility

* This work was partially supported by NSF grant CNS-1062995.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 393–402, 2013.

health is lacking: postural transitions. Postural transition are movement that result in the trunk being a different angle in relation to the bottom half of the body, such as moving from sitting to standing.

A common observational test currently in use is the timed up and down test. In this test, the subject is asked to stand from a sitting position, walk forward, turn around, walk back to their chair, and finally sit back down. An observer tracks the total time it takes for the subject to perform the entire task, including the postural transitions from sitting to standing and standing to sitting [1]. The longer the period of time it takes to perform, the less healthy the subject is likely to be. In order to get a complete picture of a subject's mobility health, it must be possible to measure postural transition frequency and postural transition time in addition to their daily activities.

In this study, a mobility classification system was developed using a single waist-mounted wireless triaxial accelerometer. This single sensor collects and processes real-time acceleration data, classifies the mobility activity from this data, then transmits this information to a base station.

The remaining sections of this paper are organized as follows: Section 2 discusses related work, Section 3 provides details on the physical and software system set-up, Section 4 discusses the classification algorithm, Section 5 discusses the experiments, and Section 6 provides a conclusion and discussion of future work.

2 Related Works

Several mobility monitoring systems are currently in development. These systems typically vary by the number of sensors used, the position of the sensors on the body, the detectable mobility states, and the features generated.

Lyons et al., Veltink et al. and Culhane et. al. used two wireless triaxial accelerometer sensors at varying positions on the body. These groups were able to distinguish between both static and dynamic activities using these sensors [2, 3, 6]. However, other groups have produced results with similar and even higher accuracy using a single sensor [1, 4, 5]. As a goal was to keep the system simple and wearable, only a single sensor was used as it often produces similar results to two sensor systems.

Haché was able to create a system using only a single waist-mounted wireless triaxial accelerometer that detects wide range of mobility states, including transitions and stair ascent and descent. Several features were used, including standard deviation, inclination angle, and Signal Magnitude Area. The accuracy of this system was 96.4% [1]. Karantonis et al. also created a system using a single waist mounted triaxial accelerometer. The overall accuracy of their system was 90.8% [4]. While the accuracy of these systems was within our goal, signal magnitude area and some of the other features used in Karantonis et al. are overly complex and, as this work shows, unnecessary to compute.

3 Mobility Monitoring Physical and Software Set-Up

The mobility monitoring system developed to accomplish our goal uses the triaxial accelerometer inside of a Shimmer sensor. First, calibration is done on the acceleration data, and then features are calculated on the sensor. These features are then fed into a classification algorithm to determine the current mobility state of the subject. This state can then be transmitted from the sensor to a base station or stored on a MicroSD card.

3.1 Shimmer

The Shimmer sensor was used in this system as it is both wireless and easy-to-wear. Shimmer was created for uses much like the one in this project, and as such, it has worked well in the developed system. The Shimmer contains a 3-axis MMA7361 accelerometer in addition to a 8MHz MSP430 CPU and a storage device [11].

3.2 TinyOs

The operating system used on the Shimmer device is TinyOS. This operating system is commonly used in wireless sensors, and as such, was easily incorporated into our system. Its small size and ease in use for programming allowed it to be used without conflicting with the original goals of making a small, energy-efficient system.

4 Activity Classification Algorithm

The system can currently detect with a high degree of accuracy sit, stand, lie, walk, run and fall states. Distinguishing these six states is done by calculating only standard deviation and inclination angle from the calibrated acceleration values. These features are common to most mobility detection systems [7, 8, 9, 10]. The position the sensor is worn will affect the calculations of these features. With the position we chose, the Shimmer's x-axis represents the vertical axis while the Shimmer's y-axis represents the horizontal axis.

4.1 Features

Standard Deviation

Standard deviation of the vertical axis can be used to determine the degree of dynamic activity of the wearer. It is a common feature used in mobility classification. Changes in the vertical axis are associated with movement, such as walking and running. The higher the change over the x-axis is, the higher the degree of activity currently being performed. As such, standard deviation over the x-axis is an appropriate measurement for the degree of activity being performed.

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{1}$$

After much observation, it was concluded that low degrees of dynamic movement rarely had a standard deviation greater than 0.3 or lower than 0.075. These values were then decided to be the dynamic threshold 2 and dynamic threshold 1 respectively. Figure 1 shows the relation of these thresholds to varying degrees of activity: standing, walking and running. Standing is static, walking is moderately dynamic, and running is highly dynamic. As such, the standard deviation associated with standing is below dynamic threshold 1, the standard deviation associated with walking is between dynamic threshold 1 and 2, and the standard deviation associated with running is above the dynamic threshold 2.

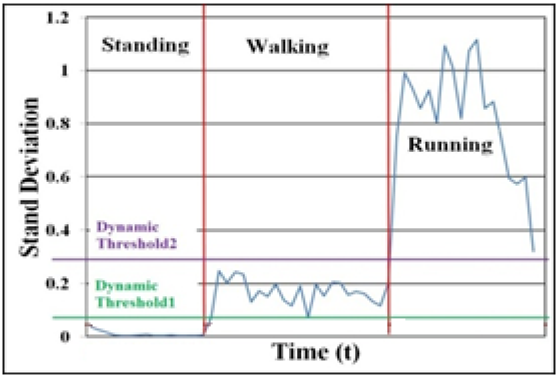
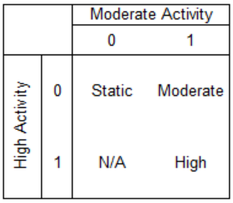


Fig. 1. Standard deviation of various activities

From these thresholds, two binary values are calculated: moderate activity and high activity. If the standard deviation has surpassed dynamic threshold 1, the movement is at least moderately active. If the standard deviation surpasses dynamic threshold 2, the movement is highly active. However, if the standard deviation does not surpass dynamic threshold 1, then the activity being performed is neither moderately or highly dynamic, and is then assumed to be static. These binary values are combined as shown in table 1 to determine the degree of activity of the subject. As seen in table 1, if moderate activity is 0 and high activity is 0, the result is that the subject is static.

Table 1. Resulting activity degree from moderate and high activity binary values



		Moderate Activity	
		0	1
High Activity	0	Static	Moderate
	1	N/A	High

Inclination Angle

Inclination angle is used to determine the posture of the wearer. This is done by simply calculating the arc-tangent of the horizontal axis of the device over the vertical axis of the device. Below is the formula used in the code where Ay represents the acceleration over the horizontal axis, and Ax represents the acceleration over the vertical axis. Please see figure 2 for more details.

$$\Phi = \arctan\frac{A_y}{A_x} \qquad (2)$$

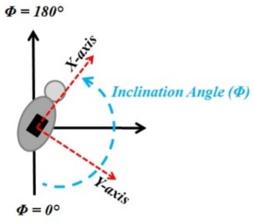


Fig. 2. Inclination angle

After ten samples are received, the average angle is then calculated. The resulting average is then compared to threshold values for standing and lying. It is assumed that if the resulting angle does not surpass the threshold for standing or lying, the activity state would be sitting.

4.2 Mobility Detection Algorithm

The results from the calculated features are combined to determine the current activity state. For example, if the standard deviation calculated indicates a moderate level of activity, and the inclination angle indicates a standing posture, we then assume the current activity is walking.

As discussed prior, each of the features generates two binary values. Standard deviation gives us the binary values of moderate degree of activity and high degree of activity. Inclination angle gives us the binary values of standing and lying. These four values are combined into a decimal value, which has a mapping to an activity state.

As an example, assume the features generated produced a standard deviation of 0.10 and an inclination angle of 125 degrees. The binary values for moderate activity, high activity, standing and lying would all be zero. These are then combined to form the binary value 0000. The sensor wearer is not moderately active, not highly active, not standing and not lying. The system then calculates the correlating decimal value of zero, which is mapped to sitting.

Falls can also be easily detected using only these four binary values as seen in figure 3. When inclination angle indicates lying and the standard deviation indicates high activity, the associated state would be a fall.

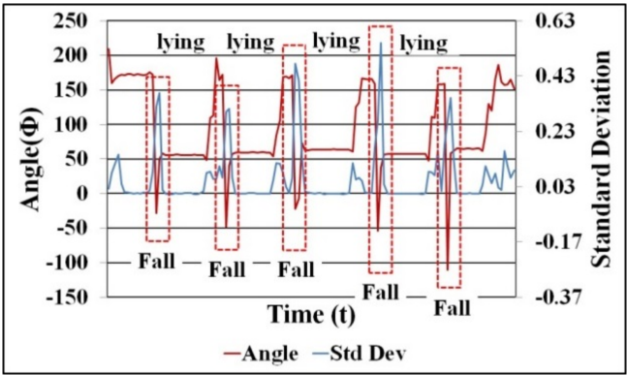


Fig. 3. Fall detection

4.3 Postural Transition Detection Algorithm

Explanation

The feature used in transition detection is the angle of inclination, which is also used to determine the posture of the wearer. As transitions are simply changes in posture, we then look at changes in inclination angle to determine if the wearer is in a transition state. This is simply done by taking the abstract value of the current angle less the previous angle. In the formula below, C represents the current angle and P represents the previous angle. If the resulting angle difference is greater than zero, this would indicate a postural change, but may not indicate a postural transition.

$$Angle\ Different = |C - P| \tag{3}$$

Algorithm

As the goal is to detect not only healthy transitions, but slow gradual transitions, it is important to differentiate small changes in inclination angle due to postural changes from small movements that are not transition related. An example would be rocking back and forth in a chair. The inclination angle will change, but there is no actual postural transition. If an algorithm were used that indicate any postural change as a transition change, this movement would be incorrectly classified as a postural transition.

Our solution involves adding a potential transition flag to smaller differences in transitions, but still labeling the larger differences in inclination angle as transitions. A potential transition angle difference is between 1 and 20 degrees, while a large transition angle difference is over 20 degrees. If any potential transition states precede or follow a state without a transition flag, the potential transition is relabeled as a regular transition. This is seen in second 7 in Table 2. If a potential transition flag is preceded or followed by several other flags, then at least one transition state, the states with the potential transition flag are relabeled as transitions.

Table 2. Postural transition algorithm results

Time (sec)	Original State	State after Postural Transition Algorithm
1	Sit	Sit
2	Sit - PT	Transition
3	Transition	Transition
4	Transition	Transition
5	Stand - PT	Transition
6	Stand	Stand
7	Stand - PT	Stand
8	Stand	Stand

4.4 Combined Classification Algorithm

The algorithms used for mobility classification and for postural transition detection are then combined to form the final classification algorithm. Inclination angle and standard deviation are first calculated. From this point, the posture and overall activity level of the subject is then determined. In figure 4, inclination angle is seen to feed into three tests: standing, lying, and postural transition. Standard deviation feeds into a high dynamic and low dynamic. These all produce binary values, from which a decimal value is calculated and mapped to a correlating mobility state as discussed in sections 4.2 and 4.3.

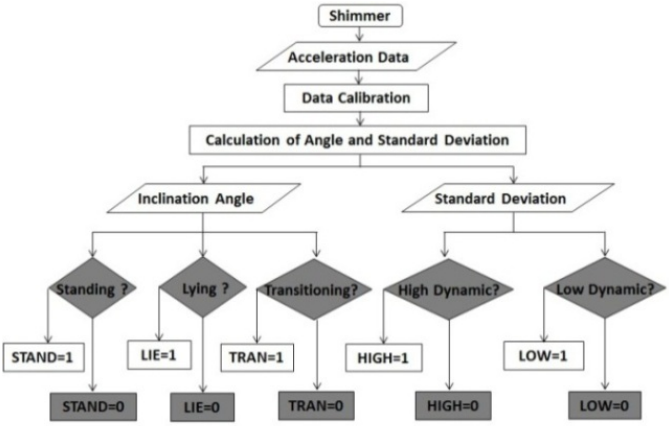


Fig. 4. Final classification algorithm

5 Experiments

Two experiments were conducted using this system. The first experiment tested the accuracy of the mobility detection algorithm without concerning postural transition time or frequency. The second experiment tested the accuracy of the system's ability to detect postural transitions.

5.1 Mobility Classification Experiment

An experiment on the system confirmed that the accuracy of detecting sitting, standing, lying, walking, and running using the above methodology is above 95%.

Both a male and a female wore the sensor while performing a list of activities in tandem over approximately 10 minutes. These activities were performed both indoors and outdoors in order to confirm that surface level did not affect accuracy. Table 3 shows the duration of each activity performed by each subject, the order in which the activities were performed, as well as the location.

Table 3. Activities performed

Activity	Place	Time (seconds)	
		Subject I	Subject II
Walking	Outdoor	***360***	***300***
Standing	Outdoor	***20***	***20***
Running	Outdoor	***30***	***15***
Standing	Outdoor	***30***	***120***
Walking	Outdoor	***300***	***300***
Sitting	Indoor	***360***	***340***
Walking	Indoor	***30***	***45***
Lying	Indoor	***72***	***60***
Total time		***1200***	***1200***

The accuracy of detecting the performed activities was high in both subjects. Table 4 summarizes the percentage of time the subject performed each activity in second column and the percentage of time the system detected each activity in the first column. The accuracy of mobility detection in both subjects is above 95%.

Table 4. Accuracy of activities performed

	Subject I		Subject II	
	Detected	Performed	Performed	Detected
Walking	*54%*	*57*	*52*	*54*
Sitting	*32*	*30*	*29*	*28*
Lying	*7*	*6*	*6*	*5*
Standing	*5*	*4*	*12*	*12*
Running	*2*	*3*	*1*	*1*

5.2 Postural Transition Experiment

The postural transition classification experiment involved a single subject performing three sit-to-stand transitions and three stand-to-sit transitions. These transitions were performed in increasing duration. The first transition of its type was performed in one second, the second in three seconds, and the final in five seconds. This was done to verify the system's ability to detect not only postural transitions, but also their duration. The results of this experiment can be seen in figure 5. The test graph represents the states detected by the device during the experiment, while the goal represents the states performed by the subject.

The overall accuracy of the transition detection is only at 87%, which is below the original goal of greater than 90%. While the majority of transitions resulted in an accuracy of 100%, the longer duration sit-to-stand transition is driving the accuracy down. This can be seen in table 5.

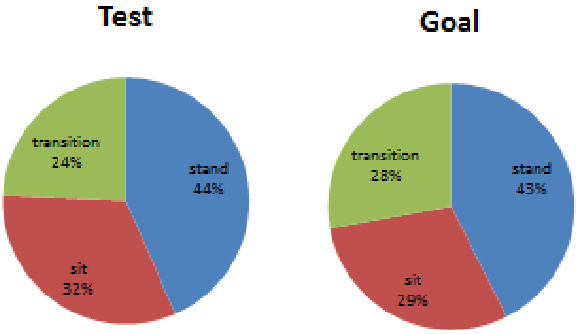


Fig. 5. Postural transition experiment results

Table 5. Postural transition experiment accuracy breakdown

Duration (sec)	Sit-to-Stand Accuracy	Stand-to-Sit Accuracy
1	100%	100
3	100	100
5	67	100

6 Conclusion and Future Work

Current methods used in determining mobility health are out-of-date, time consuming, and expensive. These methods can be easily improved with the mobility detection system discussed in this paper. In addition to the states that are typically classified in similar mobility monitoring systems, we have added both postural transition time and frequency to data gathered from the system.

The system discussed in this paper can detect sitting, standing, lying, falling, walking, running and postural transitions with an overall accuracy of over 90% using a single waist-mounted triaxial accelerometer. This is done by calculating only two features: standard deviation and inclination angle. Many related systems use more than one sensor and features that are heavier in calculations with a lower accuracy.

The accuracy of transition detection is lower than the overall accuracy of the system, and more work will initially be done to improve transition detection. This will be done by revising the current algorithm used in handling potential transition state flags.

We wish to include both stair ascent and descent to the states that are currently detected. This has been attempted in several other systems with a relatively lower accuracy. With the addition of stair ascent and descent, new features will also need to be calculated. Typically, skewness and eccentricity calculations have been used in stair detection, but these are much more complex than the features we currently use.

The most promising possible step would be using the mobility data to generate an overall health profile of the subject. This would most likely require collaboration with a medical professional with experience in mobility health. Once complete, we will introduce a training algorithm on the generated health profiles and mobility classification in order to increase accuracy for larger populations.

References

1. Hache, G.: Development of a wearable mobility monitoring system. Master Thesis, University of Ottawa (2010)
2. Veltink, P.H., Bussmann, H.B.J., De Vries, W., Martens, W.L.J., Van Lummel, R.C.: Detection of static and dynamic activities using uniaxial accelerometers. IEEE Trans. Rehabil. Eng. 4(4), 375–385 (1996)
3. Lyons, G.M., Culhane, K.M., Hilton, D., Grace, P.A., Lyons, D.: A description of an accelerometer-based mobility monitoring technique. Med. Eng. Phys. 27(6), 497–504 (2005)
4. Karantonis, D.M., Narayanan, M.R., Mathie, M., Lovell, N.H., Celler, B.G.: Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. IEEE Trans. Inf. Technol. Biomed. 10(1), 156–167 (2006), `http://www.shimmer-research.com`
5. Mathie, M.J., Coster, A.C.F., Lovell, N.H., Celler, B.G.: Detection of daily physical activities using a triaxial accelerometer. Med. Biol. Eng. Comput. 41(3), 296–301 (2003)
6. Culhane, K.M., Lyons, G.M., Hilton, D., Grace, P.A., Lyons, D.: Longterm mobility monitoring of older adults using accelerometers in a clinical environment. Clin. Rehabil. 18(3), 335–343 (2004)
7. Najafi, B., Aminian, K., Paraschiv-Ionescu, A., Loew, F., Bula, C.J., Robert, P.: Ambulatory system for human motion analysis using a kinematic sensor: Monitoring of daily physical activity in the elderly. IEEE Trans. Biomed. Eng. 50(6), 711–723 (2003)
8. Bouten, C.V.C., Koekkoek, K.T.M., Verduin, M., Kodde, R., Janssen, J.D.: A triaxial accelerometer and portable data processing unit for the assessment of daily physical activity. IEEE Trans. Biomed. Eng. 44(3), 136–147 (1997)
9. Baek, J., Lee, G., Park, W., Yun, B.-J.: Accelerometer signal processing for user activity detection. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 610–617. Springer, Berlin (2004)
10. Vaidya, S., Youn, J., Ali, H.: Real-Time Fall Detection and Activity Recognition Using Wireless Sensors. In: International Conference on Computer and Software Modeling, ICCSM 2010 (December 2010)
11. `http://www.shimmer-research.com`

A Dedicated Serialization Scheme in Homogeneous Cluster RPC Communication

Yong Wan, Dan Feng, Fang Wang, and Tingwei Zhu

Wuhan National Laboratory for Optoelectronics
School of Computer Science and Technology
Huazhong University of Science and Technology
Wuhan, China
{wanabc,tingwzh}@foxmail.com, {dfeng,wangfang}@mail.hust.edu.cn

Abstract. RPC is a communication technology which has been widely used in distributed systems. It has been employed as an essential component of distributed systems. However, the performance of traditional RPC technology will be seriously decreased in high-speed network based cluster system. The main reason is that the network of cluster has many obvious features such as high bandwidth, low latency, and high reliability, etc, which are different from normal network environment. Therefore, how to improve the performance of RPC technology in cluster system has caught increasing attention.

In this paper, we have carefully studied the traditional RPC technology, which suggest that the decreasing of cluster network performance is mainly caused by the ***serialization/deserialization*** process of RPC technology. Thus we proposed a dedicated ***serialization/deserialization*** scheme which can run on homogeneous cluster system. This scheme can well improve the performance of cluster network by reducing the number of date copy operations of RPC protocol. We have evaluated our improved scheme in our real-world cluster system. And our evaluation results show that our scheme can significantly promote performance of bandwidth by up to 43% in our cluster system when the size of transmitted data block is large.

Keywords: Serialization, Homogeneous cluster, RPC.

1 Introduction

A computer cluster [2] is a type of parallel and distributed processing system, which consists of a collection of interconnected stand-alone computers working together as a single, integrated computing resource. Network component is an important part in distributed system. RPC (remote procedure call) [1, 14] technology is the most commonly used network middle layer.

However, at the beginning of the RPC technology was proposed, it was designed for using in common network environment, not specifically designed for high-speed cluster network environment. The cluster network has the obvious features such as short physical path, high bandwidth, low latency, and high reliability. These features

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 403–412, 2013.

lead to the huge difference between cluster network and ordinary network environment. So, when the traditional RPC technology is used in cluster network, it often works with a low efficiency.

Therefore, there are lots of works have been carried on to simplify or modify the RPC technology on cluster system [3, 9, 10]. However, the RPC systems that most of these works are based on have much difference with the traditional RPC, it brings some other problems such as hard to understand, use, and transplant. Being different from them, we proposed a dedicated serialization scheme which based on the traditional RPC system, it can run on homogeneous cluster system and can get great performance promotion on bandwidth.

The main contributions of our paper are as follows:

(1) We make a detailed analysis on the process of traditional RPC technology, and we find that the ***serialization/deserialization*** operation—one step of the RPC process, is one of the main overheads.

(2) We proposed a new simplified ***serialization/deserialization*** method, which can run on homogeneous cluster system, and get a much higher performance.

(3) We make a detailed performance evaluation and analysis on the new simplified ***serialization/deserialization*** method in our real-world cluster system, and make our conclusion.

The rest of the paper is organized as follows: in Section 2, we describe the background and explain the existing problems. In Section 3, we give a detailed analysis on the overhead of the traditional ***serialization/deserialization*** process. In Section 4, we propose a new simplified ***serialization/deserialization*** scheme that can achieve much high performance in cluster system which has high-speed network environment. The experimental results and analyses are in Section 5. The conclusions are presented at the end of this paper.

2 Background and Motivation

As we have mentioned above, even since the RPC technology was used in cluster system, there are lots of works were carried on to simplify or modify the RPC technology on cluster system. In Panasas system [9, 16], it uses a special lightweight RPC to provide fast communication between the Metadata Server and Clients, and get a good performance. In Lustre system [10], it implemented a layered software module which named LNET (Lustre networking). And LNET integrated a dedicated RPC in it, to provide a very good performance to user.

Although lots of achievements have been made on RPC improvement in cluster system, there are still some problems in this aspect. For example, about Lustre, it provides a MPI interface to user, enclosed main components in a large middle layer. RPC is integrated in this layer, it is not only difficult to understand and transplant, but also difficult to do secondary development on it. So it is urgent to design and implement a simple and dedicated RPC which is based on traditional RPC, easy to understand and can be easily applied to the cluster system.

SUN microsystems had defined the RPC Protocol Specification [14], it has been the de facto RPC standard, so we choose a typical open source version in SUN RPC products, TI-RPC (transport-independent remote procedure call) [7, 8], as the research example. The TI-RPC makes RPC applications transport-independent by enabling a single binary version of a distributed program to run on multiple transports. And, it can be regarded as a typical representative of traditional RPC [8].

After carefully studied the source code of TI-RPC, we depict the general steps of data processing in TI-RPC as in Figure 1.

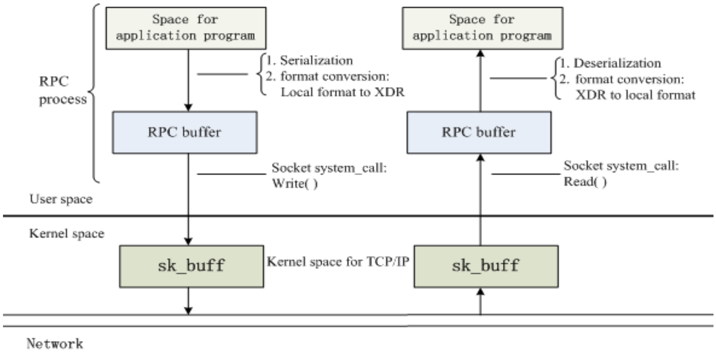

Space for application program
Space for application program
RPC process
1. Serialization
2. format conversion: Local format to XDR
1. Deserialization
2. format conversion: XDR to local format
RPC buffer
RPC buffer
Socket system_call: Write( )
Socket system_call: Read( )
User space
Kernel space
sk_buff
Kernel space for TCP/IP
sk_buff
Network

Fig. 1. The steps of data processing in TI-RPC

During the RPC communication, the data ***serialization/deserialization*** operation is needed. The ***serialization*** is used to convert the structure of arguments data in RPC, from all kinds of structure (such as ***char, pointer, struct***, etc.) format to bytes stream which are convenient to transmit over network. Without *serialization*, it is difficult to transmit the data which store as all kinds of structure, especially like the pointer type data. The ***deserialization*** is used to restore the structure of arguments data, from bytes stream to its original data structure.

As shown in Figure 1, during the argument *serialization* operation, the XDR [6] format conversion for these arguments data is also been taken. The module which is been used to complete the *serialization* operation and the XDR operation, is named ***RPC encode*** [5, 11]. In receiver, the module which is been used to complete the ***deserialization*** and the data format conversion(from XDR format to local format) operation, is named ***RPC decode*** [5, 11, 13]. And, during the process of *RPC encode and decode*, both of the sender and receiver will allocate a memory space used to store the arguments data. We can call this memory space as "***RPC buffer***". During the process of data encoding, the sender convert all types of arguments data into XDR format according to the *serialization* rule, then copy them to RPC buffer in turn. And in receiver, the data will be received from network to RPC buffer, then according to the *deserialization* rule, these data will be converted into local format, and then copy to the space of application program in turn.

From the above analysis, we can learn that in sender and receiver, they both have the copy operation to all arguments data. And it is well-known that the memory copy operation is one of the main overhead in network protocol process [4, 15]. So the *serialization/deserialization* process, which includes memory copy operation, is obvious one of the main overhead in RPC protocol process.

Therefore, if the copy operation can be removed, it will significantly improve the performance of RPC in cluster system. However, in the entire course of RPC operation, the data is passed through different layers, from application program to RPC protocol, then to socket, and in each layer the data has different structure and format. So if we want to reduce the copy operation in traditional RPC process, it is need to change the data structure and the operation steps of the RPC protocol greatly, and it is difficult to implement and hardly get good effect.

However, it is very common that the computers are homogeneous in cluster system. In this special environment, the situation is simple, the problem above depicts maybe has particular solution. The XDR format converting is used in RPC because it is need to consider the arguments data maybe passed in different architectures of computer. So the XDR format converting is no longer need in homogeneous cluster system. Meanwhile, though the *serialization* process is still needed in homogeneous cluster system, in this special environment, we can change it to reduce the copy operation, so as to let the network performance get great improvement.

3 The Detailed Analysis of Traditional Serialization Process

In this section, we analyzed the detailed ***serialization/deserialization*** process in traditional RPC. Because we can remove the XDR process in our dedicated RPC scheme design, we omitted the related analysis about XDR process in this section.

In fact, all types of arguments data in RPC can be capsulated as data structure, so we can take a typical data structure as an example in the following analysis. The data structure defined as below:

```
struct data_arg {  int     data_int;
                   char   *data1;
                   long    data_long;
                   char   *data2;    };
```

In the process of ***serialization***, it must *serialize* all arguments data, include the data that are pointed by pointer ***data1, date2***, convert all data to a bytes stream, then sent them to network [11]. Figure 2 below shows this process: (dashed line indicates that the data is changed, and the solid line indicates that the data is no change, directly copy).

The detailed steps are as below:

1. Copy ***data_int*** to RPC buffer;
2. The *serialization* of ****data1***;

Because **data1* is a pointer type, the length of data that it point to is also need to transmit to the receiver, RPC system place the length(*strlen(data1)*) into RPC buffer firstly, then copy the actual data that it point to into RPC buffer;

3. Copy ***data_long*** to RPC buffer;
4. The *serialization* of ****data2***, same to *data1.

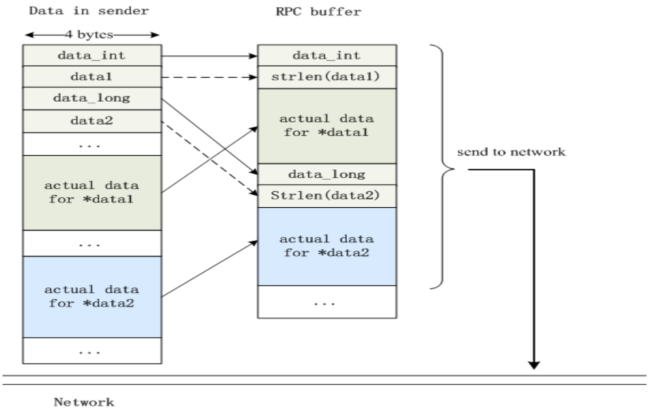


Fig. 2. The encode process of arguments data in TI-RPC

The final results in RPC buffer after conversion is shown in Figure 2. After the *serialization* operation is completed, the data in RPC buffer will be sent into network.

To the receiver, the process of *deserialization* is just recovering the serialized data to their original structure.

Therefore, in traditional RPC, the length of data block(***strlen(data1)***, ***strlen(data2)***) must be send to receiver, together with the *struct data_arg*. Only after getting the value of ***strlen(data1)***, the receiver can extract the content of the **data1* from the bytes stream received from network. In essence, this method add new members (***strlen(data1), strlen(data2)***), these new members together with the content of *struct data_arg* constitute the total arguments data. So, if there is no copy operation, it is hardly to arrange and store these data orderly in memory.

At the beginning of RPC ***encoding***(*serialization*, and the XDR format converting) at sender, a function is been called to allocate RPC buffer. Then the encoded data is stored in this buffer. When the buffer is full, or the encoding is completed, the socket function (such as ***write()***) is called to sent the data in buffer to network. We can call the buffer as **RPC *send buffer***, and the default value of its size is 64KB.

In a RPC call, all the data that is transmitted into network constitute a ***record***, it also name as RPC ***message***. In fact, in addition to the arguments data in a RPC ***record***, it also include a RPC message ***header***, and the data in *header* is used by RPC protocol itself, not by application program. When there is a large amount of arguments data in a RPC call, the size of send buffer may be far less than the size of arguments data. In this case, the process of encoding will be taken many times, and at one time, RPC system only encodes a part of arguments data, then copy it into the buffer. When the buffer is full, the data in buffer will be sent into network, then the buffer can be reused to continue encode the rest arguments data. RPC system repeated these steps, until all the arguments data has been encoded and sent into network.

The figure 3 below depicts the encode process for overall arguments data.

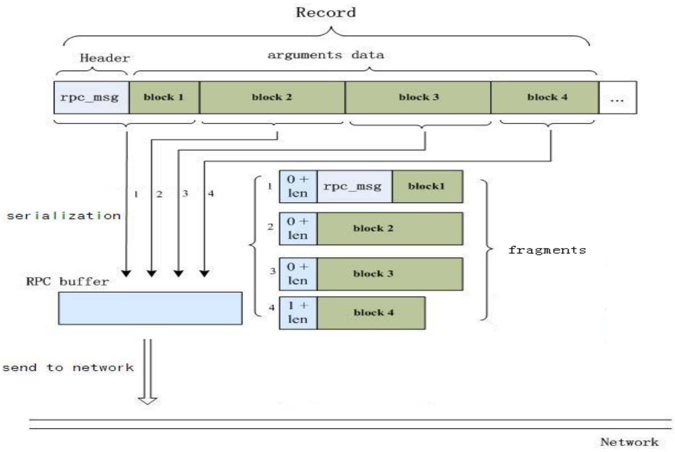


Fig. 3. The encode and send process in TI-RPC

In figure 3, the arguments data is been divided into 4 parts: block 1-4, the encoding and sent operation is been done four times, one time for a block. When each time the data has been filled in send buffer, it is referred as a ***fragment***. In the front of every fragment, there is a 32 bit field, "***flag + length***", the length of "***flag***" is one bit, indicates that if current fragment is the last fragment in the record, and the low 31 bit indicate the ***"length"***- size of current fragment.

4 The Design and Implementation of Dedicated Serialization Scheme

In this section, we proposed a new simplified ***serialization/deserialization*** scheme, it can achieve much high performance in cluster system which has high-speed network environment. We still take the data structure ***data_arg*** as example to explain the process of our simplified serialization scheme. There are 3 blocks of data in structure *data_arg* in memory, the first block is the ***struct data_arg***, second and third block are the data that the pointer ***data1, data2*** point to, respectively.

4.1 Improved Encode Method

Firstly, we need define a constant ***threshold_of_copy*** to distinguish the size of a block data is small or huge. Then, the detailed encoding process is shown in Figure 4 below:

1. Because the length (***sizeof(data_arg)***) of ***struct data_arg*** is small, we copy the data of block 1 to RPC buffer directly. Meanwhile, we use the length (*strlen(data1), strlen(data2)*) of the data block 2, 3, to replace the original value of pointer variable(***data1, data2***), respectively.

2. Send the data in RPC buffer into network.

3. About the following block2, block3 data:

a. If the lengths of them are small, we still copy them to RPC buffer and send them to network directly, because in this case, the overhead of copy operation is small.

b. If the lengths of the two blocks are huge, we send the data of the two blocks into network directly, to avoid the copy overhead.

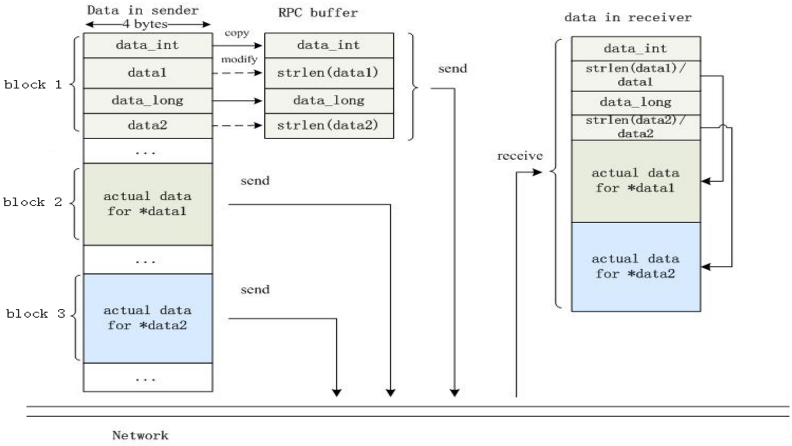


Fig. 4. Dedicated serialization scheme

And, in this scheme, the serialization needs to take the data structure as the operation unit. That is, to a given data structure, need a corresponding serialization method to handle it. In fact, when the serialization scheme is defined, we can easily write all corresponding serialization function to all kinds of arguments data structure.

At receiver, according to the ***strlen(data1), strlen(data2)***, it can identify the end position of the ***data1, data2*** block. So it can restore the value of ***struct data_arg*** by only restore the value of point variable, no longer need copy operation again. We will illustrate it in below.

4.2 Improved Decoding Method

After the receiver has received arguments data from network, the decoding process begin. In this process, because all the arguments data has been received, and the data is arranged in a continuous stream, the data of ***block2, block3***, is next and behind the ***struct data_arg***. Therefore, we can calculate the address of the ***block2, block3*** according to its length and the beginning address of ***struct data_arg***.

Figure 4 depicts the process. The arguments data stored in memory, and is divided into three blocks. The length of the first, second, and the third data block is *len1, len2, len3*, respectively, the beginning address of these data in receive buffer is ***addr0***. So the address of ***block2, block3*** is ***addr0 + len1, addr0 + len1 + len2***, respectively.

Therefore, in the decoding process, we only need to change the value of pointer ***data1, data2*** in ***struct data_arg*** as ***addr0 + len1, addr0 + len1 + len2***, respectively, then the decoding process is completed. We can return its beginning address ***addr0*** to upper layer application program directly.

5 Performance Evaluation and Analysis

In this section, we tested the performance of the traditional TI-RPC and our simplified RPC which use dedicated ***serialization/deserialization*** scheme, and analyzed the results of tests. In this section, we call the simplified RPC as S-RPC.

We construct test platform on a computer cluster in our laboratory. We select two computers in the cluster, and test the RPC performance between them. The specific configuration of each node is as table 1 below:

Table 1. The spceific configuration of each node in test platform

CPU	2 CPUs (X5560, 2.8GHz × 4 cores)
Bus	PCI-E 2.0 ×16
Operating System	Linux RedHat Enterprise, kernel 2.6.27
Network card	ConnectX InfiniBand adapter Cards, 40G bits/s
Network switch	InfiniScale IV IS5030 QDR 36-Port

5.1 The Tests of RPC Transmission Bandwidth

In the tests, the RPC server registered the procedure ***rpc_write()***. This procedure allocates a buffer in server, and receives the data that sent from client. Specifically, at the beginning of test, the client also allocates a buffer, and set the content of this buffer is characters zero, and take the contents as a data block, then client remotely call the procedure *rpc_write()*, take the data block as the arguments data, write them to server side. The size of data block is set as 4K to 2M Bytes in turn in every test. In once test, the total transferred data is 1G Bytes, so the repeated execution times of procedure *rpc_wirte()* = 1G / size of data block.

During the tests, we also tested the performance of Socket application, so as to take it as a reference to RPC. The figure 5 shows the bandwidth of SOCKET, TI-RPC, and S-RPC.

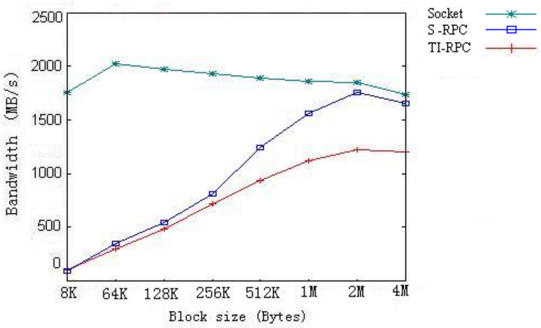


Fig. 5. The bandwidth of Socket, S-RPC, TI-RPC

We can see that, in every time of RPC call, when the size of data block for transmission is small, the performance of S-RPC has no advantages compared to TI-RPC. When the size of data block gradually increase, S-RPC shows its advantages, when the size of data block is larger than 256KB, the advantage is obvious. When the size of data block equals 2MB, the bandwidth of S-RPC reach 1.75GB/s, this bandwidth is already close to the bandwidth of Socket, which is 1.85GB/s. And the bandwidth of TI-RPC is 1.22GB/s, so the improvement is about 43%.

In our tests, the size of *fragment* is 64KB, the size of threshold (setup by ***threshold_of_copy,*** section4.1) is 16KB.

In S-RPC, when procedure ***rpc_write()*** is been called, a little judge operation has been added in program, so it brings some additional slight overhead. These overhead is unrelated with the size of data block, and in general, its value is often stable.

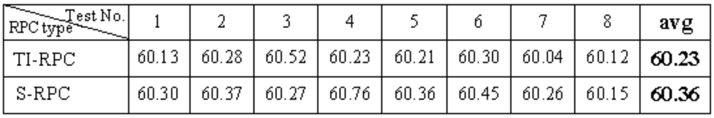
When the size of data block for every time of transmission is less than threshold, the sending process is similar with TI-RPC. But because there is additional control overhead in S-RPC, its performance is slightly lower than TI-RPC.

When the size of data block is large than threshold, these data blocks will be send to network directly. ⅰ) When the size of data block is between the threshold and fragment, the performance is upgrade, but not obvious. ⅱ) When the size of data block is larger than fragment, in this case, the size of data is very large in once ***rpc_write***() operation, so the overhead of copy and underlying transmission occupy a big proportion in overall overhead. Compared with it, the additional overhead brings by S-RPC is trivial, so the reduced overhead made by S-RPC is far more than the increased overhead, and the performance improvement is obvious.

5.2 The Tests of RPC Transmission Latency

In this part, we test the delay of RPC API. In tests, the client program call a procedure named *NULLPROC*, this process has no parameters. And the server only gives a send reply.

Table 2. The latency of TI-RPC, S-RPC



Test No. / RPC type	1	2	3	4	5	6	7	8	avg
TI-RPC	60.13	60.28	60.52	60.23	60.21	60.30	60.04	60.12	**60.23**
S-RPC	60.30	60.37	60.27	60.76	60.36	60.45	60.26	60.15	**60.36**

The table 2 is the test results. The test be taken 8 times, and we calculate the average latency value, there are small differences among every test values. The average value of S-TI is 60.36 μs, very close to the average value of TI-RPC, which is 60.23μs. The increasing rate of S-RPC to TI-RPC is about 0.22%.

6 Conclusions

In traditional RPC, during the process of ***serialization/deserialization***, there is corresponding data copy operation, so the performance is not very good. Our simplified

RPC removed these copy operations, when the size of data block for transmission is huge, it has obvious performance upgrade, compared to traditional RPC.

Acknowledgments. We would like to thank the anonymous reviewers for their suggestions and comments. This work was supported by the National Basic Research 973 Program of China under Grant No. 2011CB302301, the National High Technology Research and Development Program ("863" Program) of China under Grant No. 2013AA013203, NSFC No. 61025008, No. 60933002, No. 61232004.

References

1. Birrell, A.D., Nelson, B.J.: Implementing Remote Procedure Call. ACM Transactions on Computer Systems 2(1) (February 1984)
2. Buyya, R.: High performance cluster computing. Posts & Telecom Press (2002)
3. Chen, H., Shi, L., Sun, J.: VMRPC: A High Efficiency and Light Weight RPC System for Virtual Machines. In: 8th International Workshop on Quality of Service
4. David, D., Clark, V., Jacobson, J.: An Analysis of TCP processing overhead. IEEE Communications (June 1989)
5. Douglas, E., Comer, D.L.: Internetworking with TCP/IP. Client-Server Programming and Applications, vol. 3. Publishing House of Electronics Industry (2008)
6. Eisler, M.: XDR: External Data Representation Standard. RFC 4506 (May 2006)
7. http://docs.oracle.com/cd/E19683-01/816-1435/oncintro-3/index.html
8. http://nfsv4.bullopensource.org/doc/tirpc_rpcbind.php
9. Nagle, D., Serenyi, D., Matthews, A.: The Panasas ActiveScale Storage Cluster: Delivering Scalable High Bandwidth Storage. In: Proceedings of the 2004 ACM/IEEE Conference on Supercomputing (2004)
10. Braam, P.: The Lustre Storage Architecture (2004), http://www.lustre.org/docs/lustre.pdf
11. Richard Stevens, W.: UNIX Network Programming, 2nd edn. Interprocess Communications, vol. 2. Posts & Telecom Press (2009)
12. Sun Microsystems, Inc. LUSTRE^{TM} NETWORKING High-Performance Features and Flexible Support for a Wide Array of Networks, White Paper (November 2008)
13. Andrew, S., Van Steen, M.: Distributed systems: principles and paradigms. Tsinghua University Press (2008)
14. Thurlow, R.: RPC: Remote Procedure Call Protocol Specification Version 2, RFC5531 (May 2009)
15. Wan, Y., Feng, D., et al.: An In-depth Analysis of TCP and RDMA Performance on Modern Server Platform. In: The 7th IEEE International Conference on Networking, Architecture, and Storage
16. Welch, B., Unangst, M., et al.: Scalable Performance of the Panasas Parallel File System. In: 6th USENIX Conference on File and Storage Technologies (FAST 2008) (2008)

Friends Based Keyword Search over Online Social Networks

Jinzhou Huang and Hai Jin

Services Computing Technology and System Lab.
Cluster and Grid Computing Lab.
School of Computer Science and Technology
Huazhong University of Science and Technology, Wuhan, 430074, China
hjin@hust.edu.cn

Abstract. Online social networks are rapidly becoming popular for users to share, organize and locate interesting content. Users pay much attention to their close friends, those direct or two-hop friends. Users of Facebook commonly browse relevant profiles and the homepages, which are inefficient in obtaining desired information for a user due to the large amount of relevant data. In this paper, we propose a summary index with a ranking model by extending existing Bloom filter techniques, and achieve efficient full-text search over large scale OSNs to reduce inter-server communication cost and provide much shorter query latency. Furthermore, we conduct comprehensive simulations using traces from real world systems to evaluate our design. Results show that our scheme reduces the network traffic by 94.1% and reduces the query latency by 82.4% with high search accuracy.

Keywords: Online social network, Keyword search, Stream dynamic Bloom filters, Summary index, Friends-based selection.

1 Introduction

Popular *online social networks* (OSNs) such as Facebook and Twitter are changing the way users communicate and interact with the Internet. Hundreds of millions of users have started to use OSNs to harness desired information through social links. Users of online social networks pay much attention to their friends. The main purpose is to “keep in touch with old friends” and “finding out what old friends are doing now” [11]. Furthermore, due to the privacy policy of online social networks, some systems (e.g., Facebook, MySpace, and LinkedIn) merely allow a user to visit his/her close friends, e.g., those direct or two hop friends [6]. A recent study by Benevenuto et al. show that users browsing the profiles and the homepages of their friends dominates the behaviors on Facebook with the share of 92% in total operations [2]. Therefore, in the systems, users mainly visit their close friends and gain interested information via browsing.

Based on the analysis of a large topology trace we collected from Facebook (shown in Fig.1), the average number of two-hop friends of a user is 3.1×10^4, where over 40% users have more than 1.0×10^4 two-hop friends. Due to such large amount of

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 413–422, 2013.

possible relevant data, the simple operation of browsing is inefficient in obtaining desired information for a user. Therefore, there should be an efficient key-word search for users to avoid useless browsing.

Previous wisdom for large scale search such as Google is to collect the relevant information into a centralized repository and build indices for searching. However, due to the privacy problem of OSNs, it is impossible to build an OSN search engine using existing indexing schemes. Popular OSN systems, such as Facebook and Twitter, commonly utilize consistent hashing based scheme to partition users' data across world-wide data centers. For example, Facebook uses Cassandra [13] as default to randomly partition users' data among tens of thousands of servers across multiple data centers. Specifically, Cassandra uses the key-value model to organize the users' data in world-wide data centers, where a key is the unique identifier of a user, and the value corresponding to the key is the user's data with flexible schemas. Based on the consistent hashing mechanism, it is easy to implement user-name based data location using DHTs. However, it is difficult to provide keyword-based content search. Due to the de-facto random partition strategy, a simple query processing within a user's close friends may need to exhaustively contact a large number of servers across the data centers, raising heavy inter-server communication cost.

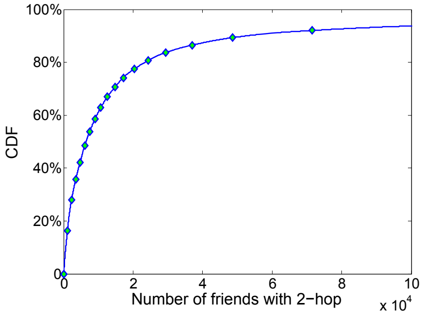


Fig. 1. Scope of 2-hop friends distribution

To address this problem, we propose an efficient keyword search scheme over OSN systems. Instead of exhaustively transmitting a query message to all relevant servers for each query, we summarize the content of each relevant friend of a user using a Bloom filter [4] and build a light-weight index of all the relevant friends for the user. Based on the summary index, we propose a ranking model which identifies the servers most likely to return the desired results. By only transmitting the query messages to the top-ranked servers, our scheme avoids significant unnecessary message caused by exhaustive search, greatly reducing the inter-server communications cost during query processing.

We conduct comprehensive simulations to evaluate the performance of this design based on the Facebook traces collected. Results show that our scheme can significantly reduce the inter-server communication cost for keyword search over OSNs, while achieving satisfactory search quality.

All in all, the contributions of this paper are twofold: 1) we propose a lightweight summary index over two-hop friends scale with a ranking model by extending existing Bloom filter techniques, and achieve efficient full-text search avoiding unnecessary queries over a mass of servers; 2) we conduct comprehensive simulations using traces from real world systems to evaluate our design. Results show the efficiency of this design.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 introduces system design. Section 4 presents the performance evaluation. Section 5 concludes this work.

2 Related Work

Existing social networks already have some user-name based search mechanisms. For examples, Facebook provides *inbox search*. *inbox search* is a feature that enables users to search through their Facebook Inbox by name of friends [13]. In Orkut, users can add several restrictions to their queries to filter the search results, such as hair color, age or geographical locations [18].

Recently, there are some literatures focusing on integrating search and social media. Mislove et al. [14] study how to integrate social network search with web search in order to complement search results. Also, how content publishing and locating influences the overall searching experience in the web perspective and in the social network context is discussed. Horowitz and Kamvar [10] present Aardvark, a social search engine, to find the right person, rather than the right document, to answer questions. Gehrke et al. [3] make a first step towards the problem of keyword search in social networks with access control. However, there is lack of key-word based content search over online social networks.

3 System Design

We first give a brief overview of the design. In the system, every user maintains a compact summary index of his/her accessible friends. More specifically, each user caches a succinct table of his/her accessible friends. Such a light weighted table contains the pairs of the name of a friend and a *Stream Dynamic Bloom filter* (SDBF) for the friend. The SDBF of a friend summarizes the set of stemmed non-stop terms from his/her documents with the information of the recency of the documents. The SDBF extends the dynamic Bloom filter [8] by integrating time information in a space efficient way. When a query comes, we first look up through the summary index table of the user locally and filter the friends less likely to return relevant content based on their SDBF. By contacting the servers hosting the top relevant friends, the scheme retrieves the matched documents and achieves more accurate ranking of the final results set. As it can be seen, the SDBF-based summary index is the core of our design.

3.1 Summary Index

Due to the privacy problem as aforementioned, it is difficult, if not impossible, for OSNs to maintain a centralized index like Google. Instead, our scheme maintains a summary index for each user. The basic idea is to summarize the content (a set of terms) of each friend and pre-compute an index of all the friends for a user. By using such an index, when a query comes, it simply forwards the queries to those friends most likely to return relevant documents. Such a solution can avoid a significant amount of bandwidth during query processing.

Considering the features of OSN systems, the goal of the summary index design should meet the following requirements: 1) The summary index should be space efficient owing to the large population in OSNs. 2) The summary of a user should support the representation of dynamic sets because a user can append content containing new terms at any time. 3) The summary should contain the recency information of the elements based on the observation that OSN users care the recency of information much [12]. 4) The update and maintenance of the summary index should be handled in a cost economical way to save bandwidth. Therefore, we design the SDBF that extends the traditional Bloom filter techniques to summarize the content of a friend.

It is well known that Bloom filter [4] is a space-efficient randomized data structure for concisely representing a set. It supports the membership verification within a constant delay. The *standard Bloom filter* (SBF) just focuses on representing a static set. Thus, we proposed using a scalable set of homogenous SBFs to represent *Dynamic Bloom Filter* (DBF) [8]. Although the DBF can support the representation of dynamic sets, it cannot reveal the recency of the elements, which is an important factor of OSN data.

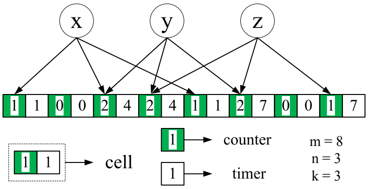


Fig. 2. Basic component of SDBF

To solve this problem, we propose to use SDBF, which extends the DBF by embedding the recency information of elements. Different from DBF, we use a variant of counting Bloom filters [7], named as *time-based counting Bloom filters* (TCBF), as the basic component of the SDBF, where each *cell* of the hashing space consists of a *counter* as well as a *timer*. Figure 2 shows an example, where *m* is the number of cells, *n* is the bits of *counter*, *k* is the number of hash functions. The *counter* records the times the hash function hits the cell, while the *timer* indicates the recency of the update of the counter. With a *counter*, the Bloom filter can support the element deletion operation [7]. In practice, very few bits, such as four bits, will be sufficient for the counter of each cell for the probability that the value of counter *c* of a given cell is larger than *j* decreases exponentially with *j*,

$$\mathrm{p}(c<j) \le \mathrm{m}(\frac{e\ln 2}{j})^{j} \qquad (1)$$

By using SDBF, each user maintains the SDBF of its two-hop neighbors and thus establishes an index to the summaries of all accessible contents.

3.2 Ranking Model

In the ranking model, the content of a user is represented as a stream of an un-bounded number of terms: $u = (t_1, t_2, ..., t_n)$, and a query q is a set of keywords, $q = (k_1, k_2, ..., k_m)$. Due to the random partition strategy of users in existing large scale OSN key-value stores, a straightforward exhaustive search scheme needs to contact a large amount of servers to find matched content, raising significant unnecessary inter-server communication cost. Our search scheme processes a query in two steps. First, a group of friends with potential answers to the query is detected using the summary index. Second, the query is submitted to the servers hosting the most relevant friends likely to return relevant documents. By using the two-stage search process, our scheme aims to achieve the performance of exhaustive search while limiting the size of the summary index and minimizing the inter-server communication cost for searching.

The ranking model of our scheme considers two factors: the content relevance and the recency of the content. When a user u_j issues a query q, the system browses the summary index of u_j and selects friends using the following model,

$$R(u_i, u_j, q) = \sum_{t \in q \cap t \in SDBF_i} ts(u_i, t) \cdot qm(u_i, t) \tag{2}$$

where $R(u_i, u_j, q)$ denotes the relevance between a friend u_i of u_j and the query q. The factor $qm(u_i, t)$ computes the relevance between the content of u_i and term t. The factor $ts(u_i, t)$ quantifies the recency of the matched content of u_i. Then, we introduce the two factors in the proposed model in detail.

1) Content relevance. It implements a content ranking algorithm of $qm(u_i, t)$ using the vector space model [17]. In the vector space model, a query is modeled as a vector of distinct terms $q = (t_1, t_2, ..., t_m)$, while a document is modeled as another term vector $d = (t_1, t_2, ..., t_n)$. The vector space model measures the similarity between the query q and the document d using the cosine of the angle between the two vectors, which can be computed using the following equation,

$$sim(q,d) = \frac{\sum_{t \in q} w_{t,q} \times w_{t,d}}{\sqrt{|q| \times |d|}} \tag{3}$$

where $w_{t,q}$ represents the weight of term t for query q and $w_{t,d}$ represents the weight of term t for document d. The notation $|q|$ denotes the number of terms in the query q, while $|d|$ is the length of the document d.

The most popular method for assigning term weights for a document in the corpus is the *term frequency-inverse document frequency scheme* (*TF*×*IDF*) [16]. Specifically, the factor *TF* denotes the term frequency property that is local and content-oriented to a document,

$$w_{t,d} = TF_t = 1 + \log(f_{d,t}) \tag{4}$$

where $f_{d,t}$ is the number of times that the term t appears in the document d.

The factor *IDF* quantifies the fact that terms appearing in many documents in a collection are less important for a query,

$$w_{t,q} = IDF_t = \log(1+\frac{N}{f_t}) \tag{5}$$

where N is the total number of documents in the corpus and f_t is the number of documents that contain term t.

In the design, due to the limit of Bloom filter structure, the summary index does not contain the term frequency information $f_{d,t}$ and the term to document mapping, $t \rightarrow d$, necessary for the *TF*×*IDF* scheme. We approximate the *TF*×*IDF* scheme by processing the query in two steps: the first step ranks the likelihood of a friend to contain relevant content and the second retrieves the documents from the servers hosting the top-ranked friends.

We introduce a measure called *inverse friend frequency* (IFF) for ranking the most relevant friends in the first step of the search process. IFF_t is computed as follow:

$$qm(u_i,t) = IFF_t = \log(1+\frac{F}{F_t}) \tag{6}$$

where F is the total number of friends of a user in his summary index, F_t is the number of friends who has content containing t. Similarly with *IDF*, a term that appears in every friend is useless for differentiating the terms for a certain query. Different from *IDF*, *IFF* can be computed using the summary index of a user conveniently. The parameters F and F_t can be easily computed by using the summary index, where F is the number of entries in the summary index and F_t is the number of friends contains an element matching the term t.

2) Recency of content. We mentioned that the SDBF supports not only the succinct representation of the terms extracted from a friend's content, but also the reflecting of the recency of the terms.

When executing membership queries against the SDBF for a term, if matched, we can also obtain the timer of the term. The timer reflects the time an item is inserted or updated. For example Facebook, users pay more attention to the relevant information in the past month [1]. Based on the observation, in the design of SDBF we consider a much succinct representation of a timer, which uses a few bits to represent a time window. Initially the timer of the cell will be set to *MAX_T*, which represents the size of the window. As time goes by, when a cell is set by an inserted/updated element at time *T*, the timer will be decreased to the value of *MAX_T-T*. Thus, the timer can reflect the recency of the element compressed inside the Bloom filter. When verifying the membership of an element, the recency of a matched element is determined by the timer with the largest value among the number of *k* timers. Therefore, achieving the timer of the term, we can present the time as follow:

$$ts(u_i,t) = ts_t = 1+\frac{T_t}{MAX\_T} \tag{7}$$

where ts_t is the time of the term t, T_t is the timer of term t, *MAX_T* is the maximum timer. In this design, we use 5 bits to represent the time window of about one month. Thus, the granularity of the timer is one day. So the maximum timer is 31, and the ts_t is a value between 1 and 2, and the bigger the timer, the larger the value.

Above all, by using IFF, we then come to rank the content relevance between a friend u_i of user u_j by:

$$R(u_i, u_j, q) = \sum_{t \in q \cap t \in SDBF_i} (1 + \frac{T_t}{MAX\_T}) \cdot \log(1 + \frac{F}{F_t}) \tag{8}$$

4 Performance Evaluation

4.1 Simulator Setup

A custom simulator in Java is developed to compare our retrieval algorithm with exhaustive retrieval as baseline. In order to better represent the real world OSN system, we consider the real data trace with the underlying data center characteristics. Previous studies have shown that the existing large-scale data centers commonly use a fat tree network architecture [15]. The fat tree as a whole is split into *k* individual pods, with each pod supporting non-blocking operation among $1/4k^2$ hosts. In the evaluation, we set *k*=50, which can maintain 30,000 servers. The number of servers is set at 1,000 at the beginning and it is changed to evaluate the performance with the network size increasing.

After developing the underlying data center network, we can simulate the data center of Facebook with thousands of servers. Then, we randomly partition the Facebook trace collection among these servers with the Cassandra scheme. Next, we assign each user 100 documents randomly selected from the WT10G data collection [9] and give each document a time factor to present the recency of the document. We set the time of documents with uniform distribution in the evaluation. Then, each user maintains a summary index of his/her two-hop friends with SDBF. By using the MD5 hash algorithm, we hash the terms of documents of one's friend into the counter of the SDBF, while hash the time of documents into the timer. The counter is set at 4 bits, while the timer is set at 5 bits of a cell.

We use the query logs of a commercial search engine to evaluate the performance [5]. When a query comes, we first look up through the summary index table of the user locally and filter the friends less likely to return relevant content based on ranking model described in section 3. By contacting the servers hosting the top relevant friends, the scheme retrieves the matched documents and achieves a more accurate ranking of the final results set.

4.2 Results

There are two kinds of metrics to be considered in the evaluation: one is the communication cost, the other is the accuracy of the results.

We use two metrics, *traffic* and *latency* to measure the communication cost. The traffic of OSN has a significant impact on the underlying data center network.

$$Traffic = M \sum_i \frac{L_i}{B_i} \tag{9}$$

where M is the size of the message, and L_i and B_i represent the length and the bandwidth of the *ith* physical link that the message travels on the underlying physical network during one hop in the overlay, respectively.

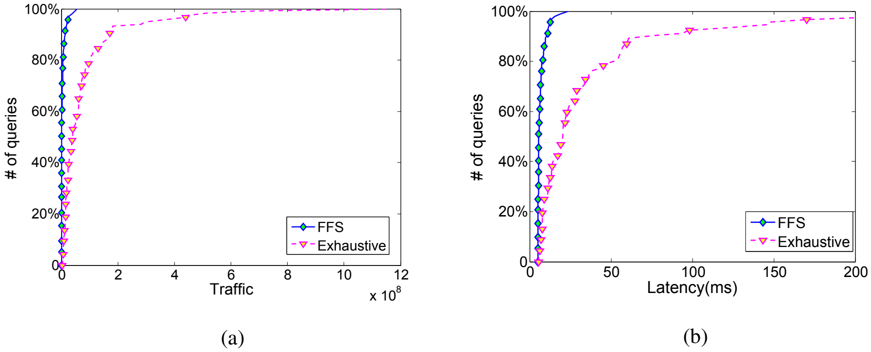


Fig. 3. Results of communication cost. (a) The CDF distribution of traffic. (b) The CDF distribution of latency.

The latency for a query is the sum of the underlying latency over each hop in the data center. It is the sum of the underlying latency for all relevant data transferred over servers. So, the latency of a query can be represented as the total data quantity divided by bandwidth.

The search accuracy is measured by two metrics, *recall* and *precision*. *Recall* is the percentage of relevant documents returned as results divided by the total number of the relevant documents. *Precision* is the percentage of relevant documents among all the results for a query presented to the user. It describes how much irrelevant material the user may have to look through to find the relevant material.

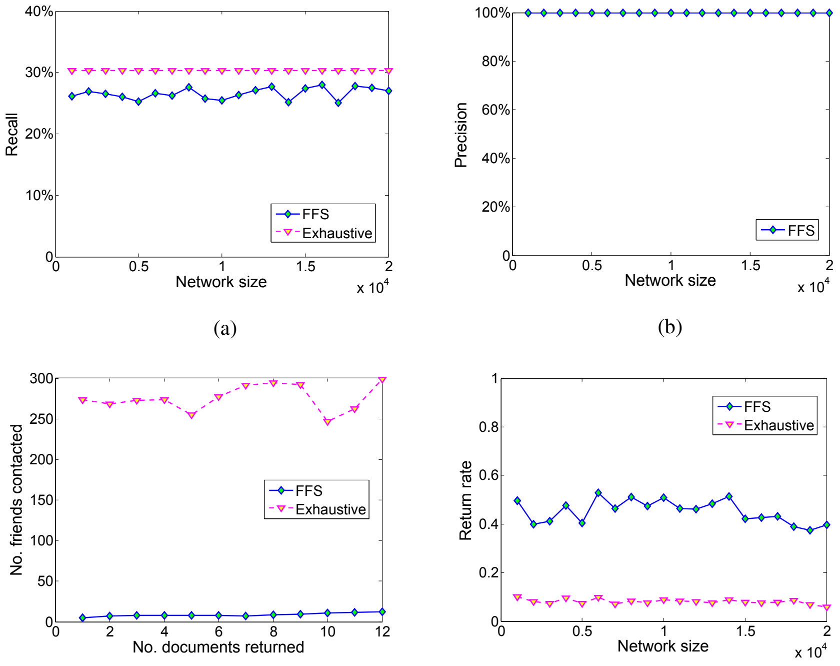
We evaluate the performance of the algorithm by comparing its recall and precision.

Figure 3(a) plots the traffic of query, where 42.7% queries using exhaustive search have traffic less than 3.1×10^7. By using our scheme, more than 97.4% of queries have such a low traffic. In the following, we present the results of our scheme as *FFS* for short. The average traffic of the query logs using exhaustive search is 8.44×10^7, while the average traffic using *FFS* is only 5.0×10^6, reducing the traffic by 94.1% significantly.

Figure 3(b) shows the latency of query, where less than 40% of the queries using exhaustive search need less than 30 milliseconds. By using *FFS*, more than 97.8% of the queries have such a short latency. The average latency of queries using exhaustive search is 78.9 milliseconds, while the average latency using *FFS* is only 13.9 milliseconds, significantly reducing the latency by 82.4%.

Figure 4(a) plots the recall over all provided queries with top 10 relevant documents returned using our scheme compared with the exhaustive search as a baseline. It can see from the figure, it performs slightly worse than the exhaustive search method. It is amazing to see that the recall of the results is almost matching using the *FFS* scheme compared with the exhaustive search.

Figure 4(b) shows the precision of the *FFS* scheme. It shows that the average precision of queries is 98.4% when the network size increases. Due to the false positives of SDBF, the system can not achieve a precision of 100%.



(a)

(b)

(c)

(d)

Fig. 4. Results of search accuracy. (a) Average recall changes with the network size. (b) Average precision changes with the network size. (c) Average number of friends needs to contact with to achieve *k* relevant documents. (d) The return rate changes with the network size.

Figure 4(c) shows the number of friends contacted when requesting different numbers of documents *k*. It shows that the *FFS* scheme has tremendously lower request than the exhaustive search.

Figure 4(d) shows the return rate of the *FFS* scheme, which means the rate between the documents returned and the number of friends contacted. It shows that the average return rate of *FFS* scheme is 0.452, while the exhaustive scheme is 0.081. Therefore, the *FFS* scheme has higher search efficiency than the exhaustive search.

5 Conclusion

In this paper, we discover the main problem of the low efficiency of keyword search over online social networks, and have proposed a lightweight summary index over two-hop friends' scale with a ranking model by extending existing Bloom filter techniques. We conduct comprehensive simulations using traces of Facebook crawled to evaluate our design. Results show it achieves high efficiency full-text search avoiding unnecessary queries over a mass of servers, and reduces the traffic and the latency significantly.

In this work, it only discusses the text-based content search. In the future, we will extend text-base content search to various content. Next, we will pay much attention to optimize the line query model in the summary index. Then, we will consider other factors into the ranking models, such as locality of interest and friends relationship strength.

References

1. http://blog.facebook.com/blog.php?post=115469877130 (2011)
2. Benevenuto, F., Rodrigues, T., Cha, M., Almeida, V.A.F.: Characterizing user behavior in online social networks. In: Proceedings of the 9th Conference on Internet Measurement. ACM (2009)
3. Bjørklund, T.A., Götz, M., Gehrke, J.: Search in social networks with access control. In: Proceedings of the Second International Workshop on Keyword Search on Structured Data, p. 4. ACM (2010)
4. Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. Communication of the ACM 13(7), 422–426 (1970)
5. Chen, H., Jin, H., Wang, J., Chen, L., Liu, Y., Ni, L.M.: Efficient multi-keyword search over p2p web. In: Proceedings of the 17th International Conference on World Wide Web, pp. 989–998. ACM (2008)
6. Chen, R., Lua, E.K., Cai, Z.: Bring order to online social networks. In: Proceedings of IEEE International Conference on Computer Communications, pp. 541–545. IEEE (2011)
7. Fan, L., Cao, P., Almeida, J.M., Broder, A.Z.: Summary cache: A scalable wide-area web cache sharing protocol. In: Proceedings of Special Interest Group on Data Communication, pp. 254–265. ACM (1998)
8. Guo, D., Wu, J., Chen, H., Yuan, Y., Luo, X.: The dynamic bloom filters. IEEE Trans. Knowledge and Data Engineering. 22(1), 120–133 (2010)
9. Hawking, D.: Overview of the TREC-9 web track. In: Proceedings of the 9th Text Retrieval Conference (2000)
10. Horowitz, D., Kamvar, S.D.: The anatomy of a large-scale social search engine. In: Proceedings of International Conference on World Wide Web, pp. 431–440. ACM (2010)
11. Joinson, A.N.: Looking at, looking up or keeping up with people?: motives and use of facebook. In: Proceedings of the Conference on Human Factors in Computing Systems. ACM (2008)
12. Kwak, H., Lee, C., Park, H., Moon, S.B.: What is twitter, a social network or a news media? In: Proceedings of International Conference on World Wide Web. ACM (2010)
13. Lakshman, A., Malik, P.: Cassandra: a decentralized structured storage system. ACM SIGOPS Operating Systems Review 44(2), 35–40 (2010)
14. Mislove, A., Gummadi, K.P., Druschel, J.P.: Exploiting social networks for internet search. In: Proceedings of the 5th ACM Workshop on Hot Topics in Networks. ACM (2006)
15. Mysore, R.N., Pamboris, A., Farrington, N., Huang, N., Miri, P., Radhakrishnan, S., Subramanya, V., Vahdat, A.: Portland: a scalable fault-tolerant layer 2 data center network fabric. In: Proceedings of Special Interest Group on Data Communication. ACM (2009)
16. Salton, G., Fox, E.A., Wu, H.: Extended boolean information retrieval. Communication of ACM 26(11), 1022–1036 (1983)
17. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communication of ACM 18(11), 613–620 (1975)
18. Vieira, M.V., Fonseca, B.M., Damazio, R., Golgher, P.B., de Castro Reis, D., Ribeiro-Neto, B.A.: Efficient search ranking in social networks. In: Proceedings of the 16th Conference on Information and Knowledge Management. ACM (2007)

GPU Virtualization Support in Cloud System

Chih-Yuan Yeh[1], Chung-Yao Kao[1], Wei-Shu Hung[1], Ching-Chi Lin[1,3], Pangfeng Liu[1,2], Jan-Jan Wu[3,4], and Kuang-Chih Liu[5]

[1] Department of Computer Science and Information Engineering
[2] Graduate Institute of Networking and Multimedia
National Taiwan University, Taipei, Taiwan
[3] Institute of Information Science
[4] Research Center for Information Technology Innovation
Academia Sinica, Taipei, Taiwan
[5] Cloud Computing Center for Mobile Applications
Industrial Technology Research Institute, Hsinchu, Taiwan

Abstract. Nowadays graphic processing unit (GPU) delivers much better performance than CPU does, and it is becoming increasingly important in high performance computing (HPC) because of its tremendous computing power. At the same time the concept of cloud computing is becoming increasingly popular. This business model suggests that GPU will be more economical because users can spend less money to rent GPUs to fit their special computing needs, rather than buying GPUs. The current practice of virtual GPU rental service is to bind a GPU to a virtual machine *statically*. As a result this static binding practice is less economical and less flexible. The goal of this paper is to design a GPU provision system that combines CUDA programs from different virtual machines and execute them *concurrently*, so as to support the concept of GPU sharing among virtual machines.

Keywords: Cloud Computing, GPU, GPGPU, GPU Virtualization, Virtual Machines.

1 Introduction

Cloud computing is becoming increasingly popular. Cloud computing users can upload their data to a data center, rent virtual machines to process the data, and retrieve the results from the data center. Cloud computing is a pay-as-you-go service, in which users only pay for the amount of services they actually used. At the end the users spend less money than buying and maintaining all the hardware and software on their own, while obtaining the same service they need. Cloud computing is also fault-tolerant – users do not need to spend money in hardware backup and maintenance.

To achieve time-sharing of resources, cloud computing uses *virtualization* techniques. Virtual machine allows users to have an illusion that they have their own stand-alone machines, and they are not aware that they are actually sharing hardware resources with others. By sharing the resources, virtualization technology improves

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 423–432, 2013.

resource utilization and reduces the cost of using resources in cloud computing. For instance, Amazon EC2 [1] provides services for users to rent virtual computers to run their applications. Google Cloud [2] provides many cloud-based applications that everyone can use.

Current technology allows CPU, memory and I/O devices to be virtualized and shared with low latency and overhead. For example virtual machine supports many users to run general applications on a physical server concurrently, sharing CPU, memory, and disk storage. Both Xen [3] and KVM [4] are convenient and efficient virtualization technologies. Xen is an open source hypervisior that provides para-virtualization to speed up the performance of virtual machines. KVM is a popular Linux kernel-based virtualization and it supports hardware assisted virtualization.

Modern *graphic processing units* (GPUs) have tremendous computing power to process a large amount of data in a short period of time. GPU use this tremendous computing power to provide extremely complicated dynamic 3D images. Modern GPUs usually have hundreds of cores, which provide extremely powerful parallel computing capabilities than CPUs. For example, AMD Radeon HD 6990 GPU reaches 5.40 Tera FLOPS (*floating-point operations per second*) single precision computing power [5] in March 2011. At the same period Intel Core i7 980 XE CPU only reaches 109 Giga FLOPS [6].

Nowadays GPUs are not only used in graphics rendering but also in high-performance computing (HPC), including molecular dynamics, protein folding, and planetary system simulation [7, 8, 9]. For example, general-purpose computing on graphics processing units (GPGPU) [10] is a methodology to use GPU in general purpose computing, not limited to computer graphics. In order to harness this tremendous computing power efficiently, NVidia [11], IBM [12], Intel [13], AMD [14] proposed new programmable languages and environment, such as CUDA [15] and OpenCL [16].

GPU is expensive and should be fully utilized. It is not economic for cloud providers to *bind* a GPU to a particular virtual machine. If we do so other virtual machines will not be able to use GPU before the virtual machine using GPU terminates. This static binding practice is *not* economic, especially to cloud service provider.

It is much easier to virtualize CPU than to virtualize GPU because CPU has a *built-in* time-sharing mechanism. CPU can easily suspend the current process and switch context to another process. In contrast GPU is much harder to virtualize because for performance GPU usually runs a single task at a time and does not switch among processes. In addition, GPU manufacturers does not provide the source code of their drivers due to business considerations, as a result GPU cannot be completely controlled by other system programs, including virtualization hypervisor.

In order to share GPU among virtual machines without doing explicit context switching among GPU processes, we propose a GPU virtualization framework that runs GPU processes in *batches*. This framework gathers all GPU kernels from each user virtual machine to a specific virtual machine that accesses the GPU directly, then recompose and compile these kernels into one GPU kernel. Finally since the NVidia Fermi [17] architecture allows concurrent kernel execution, we run this combined kernel using concurrent kernel execution and send the results back to each user virtual machine.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 describes the architecture and implementation of our GPU virtualization system. Section 4 presents and analyzes experiment results. Finally, Section 5 gives some concluding remarks.

2 Related Work

GPU has provided a better performance than CPU since 2003 [18, 19, 20]. GPU has a very large number of cores than CPU does, and it can run programs in parallel efficiently. In order to leverage the computing power of GPU, manufacturers have developed new programming languages and environments for GPUs. CUDA (Compute Unified Device Architecture) [15], AMD App (Accelerated Parallel Processing) [21], and OpenCL (Open Computing Language) [16] are the three major architectures for parallel computing with GPU.

It is evident that the development of GPU virtualization technology will introduce new economically feasible solution in high performance computing with GPU. However, restriction of proprietary software and hardware device driver prevent researchers from managing GPU at the hardware level. GPU is designed for computing a large amount of data in parallel, so it has a high data transfer bandwidth and a large number of simple cores. However, GPU is unable to handle complicated control flows. That is, GPU lacks the ability of saving process execution state, so it cannot run two or more programs in a time sharing manner. This lack of quick context switch makes it difficult to virtualize GPU.

To overcome the lack of support from vendors in GPU virtualization, researchers propose two categories of GPU virtualization – *front-end* and *back-end* techniques. Front-end techniques do *not* need to know the details of the GPU driver. Virtual machine service providers only offer a modified graphics API, which forwards requests from virtual machines to the physical machine by *remote procedure call*. For instance, Shi et al. [22] proposed a GPU virtualization architecture called *vCUDA*. vCUDA intercepts CUDA API calls from virtual machines by modifying the CUDA library. Then vCUDA redirects CUDA commands and data to a machine that has a real GPU device and this machine will perform the computations instead. vCUDA uses XML-RPC [23], a transport mechanism using extensible Markup Language [24], to pack program data and parameters of the CUDA commands to the real CUDA machine. Giunta el at. [25] proposed *gVituS* for GPU virtualization. gVituS intercepts and redirects all CUDA API calls of a virtual machine just like vCUDA does. But instead of XML-RPC, gVituS creates a TCP/IP communication to the real CUDA machine, transfers the request, and receives the execution results. Zillians [26] proposed an architecture that redirects user code to back-end physical GPU machine. Hoopoe [27] utilizes the same architecture as zillians and provides web service based

API and GUI for users to utilize GPU resources. Duato et al. [28, 29] proposed an architecture named rCUDA. rCUDA enables concurrent usage of CUDA-compatible GPUs remotely by creating virtual CUDA-compatible devices on machines without GPUs.

Back-end techniques directly associate a virtual machine with physical GPU hardware. The techniques needs support from hypervisors, such as KVM [4] and Xen [3]. These hypervisors have para-virtualization that permanently binds a physical GPU to a virtual machine. That is, a virtual machine can have the entire GPU hardware by itself. However, a virtual machine can also suffer extended waiting time due to contention with other virtual machines, so the total number of virtual machines must be restricted. Several companies, e.g., Amazon EC2 and Hoopoe [1, 27] provide GPU service by back-end techniques.

Both front-end and back-end vitalizations have their advantages. The advantage of front-end over back-end virtualization is that it is easier to practice resources sharing by modifying GPU APIs. However, it may cause significant overheads in forwarding process data than the back-end method. On the other hand, the back-end techniques have better execution performance, but may suffer long delay due to contention. Also time-sharing of GPUs is challenging for back-end techniques.

Li et al. [30] suggested that it is necessary to maintain a one-to-one mapping between CPU and GPU to avoid context switching. Li et al. [30] propose a GPU resource virtualization infrastructure that provides the concept of *virtualized units* of GPU resources. They create a virtualization layer between GPUs and CPUs. The virtualization layer will manager all the GPU resources and communicates with CPUs to answer the GPU requests. The virtualization layer will rearrange the GPU processes form different CPUs to ensure the all GPU processes can be run in parallel. They apply this approach in NVidia Fermi architecture GPU [17], and the system can support 16 concurrently running processes, and each CPU has the illusion that it its own virtual GPU.

3 System Architecture and Implement

Our system architecture consists of a *domain-U* for user virtual machines that wish to run CUDA programs, and a *domain-0* that can access the GPU. A *domain* is an executing context in which we can run processes. We can create and name a domain, except the special domain-0, in which an operating system controls the hypervisor. The hypervisor creates and controls virtual machines running in *domain-U*. The operating system in domain-0 is also responsible for executing the CUDA kernels from different virtual machines.

Our GPU virtualization architecture has three major components - a *Listener*, a *Combiner*, and an *Executor*. The Listener runs in each virtual machine of domain-U, the Combiner and Executor runs in domain-0. The system architecture is illustrated in Figure 1.

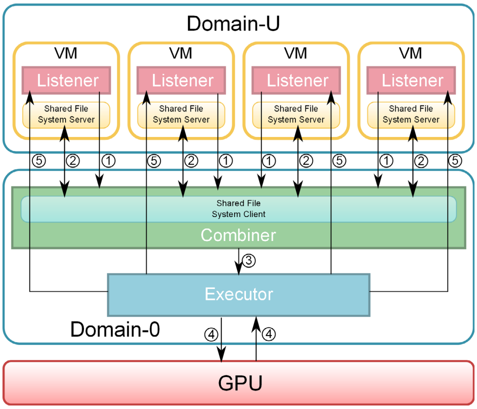


Fig. 1. WebScale system architecture

3.1 Shared File System

Our system uses a *shared file system* between domain-U and domain-0 to share GPU kernel code and data. When a user wants to execute a GPU program, the Listener passes the name of the directory containing user GPU kernel codes to Combiner so that Combiner can mount and combine all kernel codes and compile them into one executable file. Then the Executor will execute the combined executable and return the results back to each virtual machine.

We choose Network File System (NFS) as our shared file system. NFS is a distributed file system protocol, which allows users to access files stored in other machines by network. NFS is suitable for sharing a file on different virtual machines, ant it can allows different clients share a file at the same time. Every virtual machine in domain-U runs an NFS server, and the virtual machine in domain-0 runs an NFS client. Each server in domain-U exports a file system that contains the user GPU kernel code, and the client in domain-0 will mount the file systems exported by NFS servers so that Combiner can access them.

3.2 Listener

The *Listener* runs in each virtual machine of domain-U and is responsible for exporting the shared file system to the Combiner. If a user virtual machine wants to execute a GPU program, the Listener will mount the program directory in the shared file system. The user is also required to provide the memory usage of his kernel, which is used by the Combiner in order to calculate total memory usage of the combined kernel.

If a process successfully terminates, the Executor will send a termination message back to the Listener. When the Listener receives the termination message, it will notify the user that the result is ready.

3.3 Combiner

The *Combiner* runs in the virtual machine of domain-0 and is responsible for combining CUDA programs to be executed. The Combiner receives messages about what GPU kernels it needs to process from the Listener, and then processes these kernel codes through the shared file system. The Combiner then parses the source codes from each machine, creates different CUDA streams and ensures that they will not interfere with each other, and prepares the combined kernel for concurrent execution.

After mounting the file system, the Combiner chooses the kernels to execute by an FIFO policy. Kernels that were not executed immediately, either too big or too late will wait for its execution in the FIFO. To prevent starvation the Combiner sets a 10 second limit. If there are no arriving kernel for execution within 10 seconds, the Combiner will combine and compile all waiting kernels and send them to the Executor, regardless GPU memory is fully utilized or not. After deciding which kernels should be executed next, the Combiner parses and merges these kernels into one executable file. The Combiner also decides which groups of kernels should be run concurrently by checking the resource consumption of each kernel. To sum up, the Executor will run a batch of kernel if any of the following is true.

1. The combined kernel uses at least 90% of the GPU resources.
2. There are already 8 kernels ready for execution.
3. There are no incoming kernels for execution within 10 seconds.

3.4 Executor

The *Executor* runs in the virtual machine of domain-0 and is responsible for running GPU process from the Combiner. When the program terminates the Executor will save the results into files and send them back to the shared file system, then notifies the Listener that GPU process has successfully terminates. At the end, the Executor will unmount the shared file system.

3.5 System Flow

A user runs a GPU kernel in our system as follows. First the user must create and run a virtual machine in domain-U. The virtual machine will then start a Listener waiting for user commands. The detailed steps of running a GPU program are as follow.

1. The Listener receives the user request, and then notifies the Combiner to mount the directory containing GPU kernel code in the shared file system.
2. The Combiner decides which kernels will be executed concurrently, and combines these kernels into an executable file from the shared file system.
3. The Combiner notifies the executor to execute the combined GPU kernel.
4. The Executor executes the combined kernel.
5. If the combined kernel successfully terminates, the Executor saves the results into a file in the shared file system, then informs the Listener that the process has completed.

4 Experiment Results

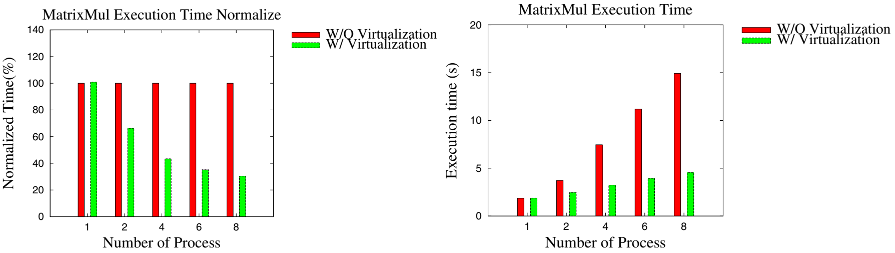
4.1 Experiment Setting

The hardware and software configuration in our experiments are as follow. We used one physical machine with an Intel Core i5-2400 Processor with 4 cores running at 3.40GHz, 8GB of memory and NVidia GTX 560-Ti for our GPU. The system uses Xen 4.1.2 as the hypervisor and Ubuntu 12.04 with Linux 3.5.0 kernel as the guest operating system in domain-0 and domain-U. Each virtual machine in domain-U has a 2 core CPU, 1GB of memory, and 20GB of disk.

4.2 Parallel Execution overhead

We conduct the first set of experiments to evaluate the parallel execution overheads of running GPU programs. In Figure 2, the execution time from 1 to 8 instances of matrixMul CUDA sample program with and without virtualization is shown in Figure 2(a). The normalized execution time using virtualization is shown in Figure 2(b).

We made the following two observations. First, when we run the programs without virtualization all program will be run serially, therefore the time will linearly increasing, as Figure 2(a) suggests. Second, we notice that the ratio between the parallel execution time to the sequential execution time decreases when we increase the currency. For example, the ratio is 100% when the number of process is 1, and it is 22% when the number of processes is 8, which is slightly larger than the theoretical bound $\frac{1}{8}$. This overhead is due to the context switching among GPU processes, and is increasing when the number of processes increases.



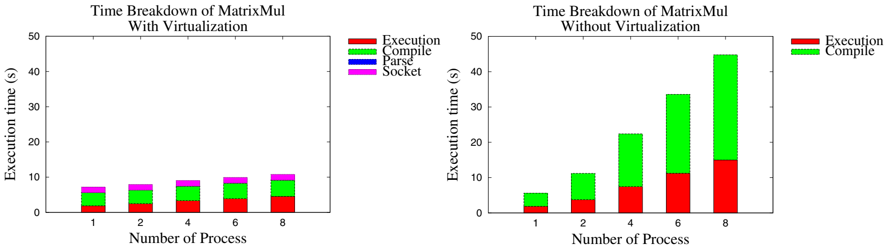
(a) (b)

Fig. 2. Total and normalized execution time of matrixMul

4.3 Dispatch System overhead

We conduct the second set of experiments to evaluate the overhead of our dispatch system in running CUDA programs. We choose several sample programs from NVidia CUDA Software Development Kit (SDK) [15] as our benchmarks. We also make two assumptions. First since we parse, compile, and then execute the program, we must consider the compilation time into our execution time. Second, we assume that programs arrive simultaneously, so we can start them together.

Figure 3(a) shows the execution time breakdown of the matrixMul benchmark with virtualization, and Figure 3(b) shows the results without virtualization. The times include compilation, network transfer, and code parsing time. Since CUDA compile its library along with the user program before version 5.0, when we compile the programs together, the common libraries are compiled only once, and the compilation time is almost a constant, as in Figure 3(a). In contrast if we do compile user programs separately, the CUDA library will be compiled multiple times, as suggested by Figure 3(b).



(a) (b)

Fig. 3. Time breakdown of matrixMul execution

4.4 Different Programs Mixture

We conduct the third set of experiments to observe the performance when we mix different CUDA programs together. We measure the execution time of the different CUDA programs with and without concurrent kernel execution. We combine four CUDA programs – Interval, vectorAdd, BlackScholes, and matrixMul together. We slightly modify matrixMul and vectorAdd to avoid security problems in the CUDA benchmark library. Figure 4 illustrates the time breakdown of running these four programs with and without virtualization. The first columns represent the time breakdown of a single programs execution, and the fifth and sixth columns represent the time breakdown of running the four benchmarks without and with virtualization. We notice that, as in Figure 2(a), the parallel execution (with virtualization) runs faster than sequential execution (without virtualization), but the performance advantage is eroded by the fact that we are combining CUDA programs could use different libraries.

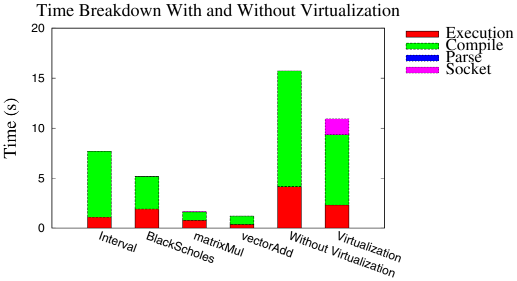


Fig. 4. Concurrent execution time of combining different programs

5 Conclusion and Future Work

We propose a GPU virtualization architecture using NVidia Fermi GPU. We show that it is efficient to achieve GPU virtualization with NVidia Fermi architecture GPU. The first advantage is that if there is sufficient GPU memory, we can merge up to 8 GPU kernels and execute them concurrently, which reduces execution time, increases system throughput, and reduces average waiting. The second advantage is that our system compiles all the necessary libraries only once and reduces compilation time.

We will study the architecture of Kepler GPU, which allows 32 CPU to access one GPU, which makes our virtualization architecture more simple and efficient. We will also improve the selection algorithm so that it makes smart decisions in choosing programs to execute concurrently.

The authors would like to thank the financial support from Cloud Computing Center for Mobile Application, Industrial Technology Research Institute, under grant number B0-B0009.

References

[1] Amazon elastic compute cloud, http://aws.amazon.com/ec2/
[2] Google cloud platform, http://cloud.google.com/
[3] Xen, http://xen.org
[4] Kvm, http://www.linux-kvm.org/
[5] Amd radeon hd 6990 graphics, http://tinyurl.com/69qxshp
[6] Intel i7-980 xe, http://tinyurl.com/86mmt37
[7] Anderson, J., Lorenz, C., Travesset, A.: General purpose molecular dynamics simulations fully implemented on graphics processing units. Journal of Computational Physics, 5342–5359 (February 2008)
[8] Chen, G., Li, G., Pei, S., Wu, B.: Gpgpu supported cooperative acceleration in molecular dynamics. In: 13th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 113–118 (April 2009)
[9] Voelz, V.A., Bowman, G.R., Beauchamp, K., Pande, V.S.: Molecular simulation of ab initio protein folding for a millisecond folder ntl9 (1-39). Journal of the American Chemical Society 132(5), 1526–1528 (210), PMID: 20070076
[10] Gpgpu, http://gpgpu.org/
[11] Nvidia, http://www.nvidia.com/
[12] Ibm, http://www.ibm.com/
[13] Intel, http://www.intel.com/
[14] Amd, http://www.amd.com/
[15] Cuda, http://www.nvidia.com/content/cuda/cuda-toolkit.html
[16] Opencl, http://www.khronos.org/opencl/
[17] Nvidia fermi architecture, http://tinyurl.com/6vdsl4q
[18] Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., Houston, M.: P.Hanrahan: Brook for gpus: Stream computing on graphics hardware. In: ACM Transactions on Graphics (TOG) -Proceedings of ACM SIGGRAPH 2004, pp. 777–786 (August 2004)
[19] Asano, S., Maruyama, T., Yamaguchi, Y.: Performance comparison of fpga, gpu and cpu in image processing. In: International Conference on Field Programmable Logic and Applications, FPL 2009, August 31-September 2, pp. 126–131 (2009)

[20] Ryoo, S., Rodrigues, C.I., Baghsorkhi, S.S., Stone, S.S., Kirk, D.B., Hwu, W.M.W.: Optimization principles and application performance evaluation of a multithreaded gpu using cuda. In: Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming. PPoPP 2008, pp. 73–82. ACM Press, New York (2008)

[21] Amd app acceleration, `http://www.amd.com/stream`

[22] Shi, L., Chen, H., Sun, J.: vcuda: Gpu-accelerated high-performance computing in virtual machines. In: IEEE International Symposium on Parallel & Distributed Processing, pp. 1–11 (May 2009)

[23] Xml-rpc, `http://xmlrpc.com/`

[24] Extensible markup language (xml), `http://www.w3pdf.com/W3cSpec/XML/2/REC-xml11-20060816.pdf`

[25] Giunta, G., Montella, R., Agrillo, G., Coviello, G.: A GPGPU transparent virtualization component for high performance computing clouds. In: D'Ambra, P., Guarracino, M., Talia, D. (eds.) Euro-Par 2010, Part I. LNCS, vol. 6271, pp. 379–391. Springer, Heidelberg (2010)

[26] zillians, `http://www.zillians.com/`

[27] Hoopoe, `http://www.hoopoe-cloud.com/`

[28] Duato, J., Pena, A., Silla, F., Mayo, R., Quintana-Orti, E.: rcuda: Reducing the number of gpu-based accelerators in high performance clusters. In: 2010 International Conference on High Performance Computing and Simulation (HPCS), pp. 224–231 (August 2010)

[29] Duato, J., Pena, A., Silla, F., Mayo, R., Quintana-Orti, E.: Performance of cuda virtualized remote gpus in high performance clusters. In: 2011 International Conference on Parallel Processing (ICPP), pp. 365–374 (June 2011)

[30] Li, T., Narayana, V., El-Araby, E., El-Ghazawi, T.: Gpu resource sharing and virtualization on high performance computing systems. In: 2011 International Conference on Parallel Processing (ICPP), pp. 733–742 (June 2011)

MGMR: Multi-GPU Based MapReduce

Yi Chen[1], Zhi Qiao[1], Hai Jiang[1], Kuan-Ching Li[2], and Won Woo Ro[3]

[1] Dept. of Computer Science, Arkansas State University, USA
{yi.chen,zhi.qiao}@smail.astate.edu, hjiang@astate.edu
[2] Dept. of Computer Science & Information Engr., Providence University, Taiwan
kuancli@pu.edu.tw
[3] School of Electrical and Electronic Engineering, Yonsei University, Korea
wro@yonsei.ac.kr

Abstract. MapReduce is a programming model introduced by Google for large-scale data processing. Several studies have implemented MapReduce model on Graphic Processing Unit (GPU). However, most of them are based on the single GPU and bounded by GPU memory with inefficient atomic operations. This paper intends to develop a standalone MapReduce system, called MGMR, to utilize multiple GPUs, handle large-scale data processing beyond GPU memory limit, and eliminate serial atomic operations. Experimental results have demonstrated MGMR's effectiveness in handling large data set.

Keywords: MapReduce, multi-GPU, atomic-free, CUDA, GPUDirect.

1 Introduction

With the stagnation of CPU's performance as well as the better programmability and performance of GPU (Graphics Processing Unit), various applications have been accelerated by GPU-related programming paradigms such as CUDA 1, OpenCL 2 and Brook+ 3. The underlying reason is that GPUs' throughput - oriented computing design closely matches the characteristics of large-scale data parallel applications.

MapReduce 4 is proposed by Google to pursue simple and flexible parallel programming paradigm. With MapReduce, users only need to write Map and Reduce functions, respectively, in order to solve problems in a parallel computing manner. The low level programming details such as the ways to handle communication among data nodes are transparent to users. Data affinity across network and fault tolerance among multiple nodes can be achieved automatically.

With the success in handling large-scale data-parallel problems, many MapReduce frameworks have been implemented on parallel platforms such as multi-core CPU systems, Cell processors and GPUs 5. However, most existing GPU-based MapReduce systems put their efforts on a single GPU in a node, neglecting the multi-GPU platforms supported by advanced techniques such as GPUDirect 6. Moreover, many such systems tend to use atomic operations in GPU global memory to handle the concurrent writes among multiple threads 910. However, such atomic operations can cause serialized access of GPU memory and decrease overall performance dramatically.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 433–442, 2013.

This paper proposes a multi-GPU MapReduce implementation, called MGMR, and makes the following contributions:

- Multiple GPUs are utilized to speed up MapReduce operation. Load balancing is achieved by distributing computations based on the capacity of all GPUs.
- Big data issue is addressed through CPU memory that normally is bigger and more extensible than GPU memory. The aggregate GPU memory is not the bottleneck anymore.
- Serial atomic operations are replaced by a parallel alternative, parallel prefix sum operation, for maximum performance gains.

The experimental results of real world applications have demonstrated that MGMR achieve significant performance gains in handling big data inputs.

The remainder of this paper is organized as follows: Section 2 briefly introduces the GPU architecture background and MapReduce framework. In Section 3, the detailed MGMR system design issues are explained for the reasons of being able to achieve the scalability. In Section 4, two real applications will be used to compare performance among CPU, single-GPU, and double-GPU MapReduce versions. Section 5 lists some related MapReduce implementations. Finally, the conclusion and future work are given in Section 6.

2 GPU and MapReduce

MGMR is developed in CUDA and based on Nvidia Fermi architecture.

2.1 Multi-GPU Architecture

Each Nvidia GPU consists of multiple streaming-multiprocessors (SMs) and can execute thousands of light-weighted hardware threads concurrently. CUDA helps map thread hierarchy onto GPU cores. Up to 512 threads are group into thread blocks that are assigned to SMs to schedule work in groups of 32 parallel threads, called warps. Extremely fast context switch with warps can help tolerate memory access latency. Each thread is assigned some registers, whereas each warp has one program counter. All threads within the same blocks can access the common shared memory. This helps synchronize threads in the same blocks, facilitate extensive reuse of on-chip data, and greatly reduce off-chip traffic.

For a machine with multiple GPUs, Nvidia GPUDirect is the technique to handle inter-GPU communication within a single system. With GPUDirect, network adapters and storage devices can directly read and write data in GPU device memory, eliminating unnecessary copies in system memory (on CPU side) to achieve significant performance improvement in data transfer. High-speed DMA engines enable this inter-GPU communication within same systems.

In Nvidia Fermi GPUs, asynchronous memory copy is another advanced feature that enables bidirectional memory copy to double data transfer bandwidth. It also helps achieve the overlapping of computation and communication 13, i.e., when GPU is busy with some calculations, DMA engines can move data around at the same time.

2.2 MapReduce Programming Model

MapReduce is widely used in various domains such as machine learning, data mining, and bioinformatics. The design goals of MapReduce include programmability, robustness and scalability. Divide-and-conquer is the basic strategy. MapReduce consists of three primary stages (*Map*, *Shuffle* and *Reduce*) where the first two can be divided further into sub-stages and the third one is indivisible. User jobs are broken apart for *Map* stage to execute. Then, *Shuffle* reorders and distributed intermediate data. Finally, *Reduce* stage merges the partial results for the final ones.

MapReduce framework tries to stay at high level and hides low-level details such as parallelism, communication, fault tolerance, and load balancing. End users only need to specify two functions: *Map* and *Reduce* for their corresponding stages. Their definitions can be abstracted as follows:

$$\text{Map: } (k_1, v_1) \rightarrow list\ (k_2, v_2) \tag{1}$$

$$\text{Reduce: (k2, list (v2))} \rightarrow \text{list (k3, v3)} \tag{2}$$

The input of *Map* is a set of key/value pairs, and the output is a list of intermediate key/value pairs. All values associated with the same key are passed to *Reduce* function that processes them for final results.

3 MGMR System Design

The target platform of MGMR is Nvidia Fermi GPU. MGMR is developed in CUDA and C++ with flexible templates. It is designed to be extensible and customizable while maintaining high occupancy of multiple GPUs. Load balancing across multiple GPUs is achieved at runtime according to hardware performance and job sizes.

3.1 Multi-GPU Utilization

All stages and sub-stages can be specified by users, and their corresponding jobs are accomplished by workers which are computers in the original MapReduce design. In MGMR, these workers will be hardware threads across multiple GPUs for load balancing.

The overview of MGMR workflow is shown in Fig. 1. The input of *Map* stage is partitioned into sets of key-value pairs, and they are assigned to workers in different GPUs simultaneously. Then, the intermediate data generated from *Map* stage are shuffled among workers across GPUs without going through CPU memory. The *Shuffle* stage incurs all-to-all communication among workers. For workers within one GPU, the communication is accomplished through commonly shared GPU global memory. For workers from different GPUs, GPUDirect enables remote GPU memory access without going through CPU memory. Performance gain is achieved there. Finally, all outputs of *Reduce* stage on multiple GPUs are copied back to CPU memory.

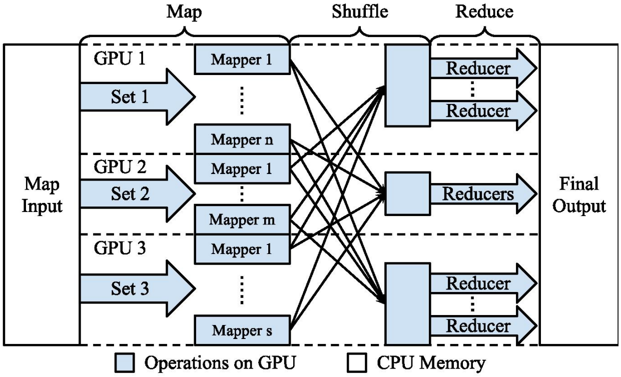


Fig. 1. MGMR workflow on multiple GPUs in single-round mode

3.2 Multi-round *Map* and *Reduce* for Big Data

Through iterative GPU activations, MGMR can handle large data set that exceeds the sum of multiple GPU memory unit sizes. However, Fig.1 only demonstrates the situation of one single round where data sets can be loaded into current GPUs. In MGMR, a data pool is allocated on CPU side in page-locked memory manner. Self-scheduling strategy is used to assign data sets onto GPUs for processing. If data cannot be processed all together by GPUs, the data pool will be used as the buffer for intermediate data in such multi-round mode. However, the single-round modes only use GPU global memory for intermediate data.

In *Map* stage, the input key-value pairs are partitioned into various sets with different sizes in the data pool. When the CPU program detects an idle GPU, it will activate one *Map* function and assign one data set over. For multi-round mode, once *Map* workers finish the work, the intermediate results will be sent back to the data pool and another group of data sets will be loaded for processing until the *Map* work is done.

In *Shuffle* stage, input/output data are placed in GPU global memory for one-round mode and in the data pool for multi-round mode.

In *Reduce* stage, *Reduce* workers will get data from GPU global memory or data pool first, and then work in self-scheduling manner. Different from Map stage, each reducer is indivisible. The input data size could be fixed or various.

3.3 *Map* Stage

The *Map* stage consists of several sub-stages: output size estimation, key-value processing and partial folder.

Output Size Estimation. MGMR estimates output size in advance to avoid memory overflow in GPU. Unlike CPU, GPU cannot dynamically allocate memory inside kernel functions. Thus, CPU program has to pre-allocate output buffer before initiating kernel execution. To reach a balance point between higher performance and shorter programs, MGMR requires users to pre-define the structure of the output values in *struct* format. Users can do the same to the output keys as well. If they do so, a comparison function for this *struct* data type should be provided so that MGMR can sort such data items correctly.

Key-Value Processing. In this sub-stage, *Map* workers fetch input data from the data pool in self-scheduling manner and execute the user-defined *Map* function. Asynchronous memory copy such as *cudaMemcpyAsync()* is used to overlap communication and computation, i.e., both GPUs and PCIe bus will be busy at the same time.

In multi-round mode, the output data sets are needed to copy back to the data pool since GPU global memory is not big enough to contain all the intermediate results. Then, MGMR takes full advantage of bidirectional memory copying in Fermi architecture since two DMA engines work in the opposite directions. Therefore, data transfer bandwidth is doubled. Communication operations are overlapped as well.

Partial Folder. If user activates this sub-stage, the intermediate output data will be folded to reduce its size. This feature gives user a way to balance between data transmission and computation overheads. I/O bound applications can use this sub-stage to reduce data transfer cost for performance gain.

3.4 Shuffle Stage

In single-round mode, if the intermediate data (output of *Map* Stage) is small enough to put in one GPU's global memory, these key-value pairs will be sorted by radix sort provided by Nvidia Thrust library 14 in *Shuffle* stage. But if they are distributed across multiple GPUs, Parallel Sorting by Regular Sampling (PSRS) 15, also called Sample Sort, is applied to incur all-to-all broadcast through GPUDirect technique. Data will be redistributed among GPUs.

If data is too big for aggregate GPU memory and multi-round mode has to be used, the input and output data of *Shuffle* will be placed in data pool (CPU side). A CPU partition schedule will use PSRS to reorder key-value pairs and build indices in the data pool. However, GPUDirect is not necessary since the data exchange does not happen among GPUs.

The MGMR version of PSRS is implemented in four steps as follows:

1. **Local Sorting.** Each GPU is assigned a contiguous set of p items out of total n key-value pairs that will be sorted locally by radix sort.
2. **Pivot Selection and Local Data Partitioning.** On each GPU, according to the number of GPUs, a certain number of pivots are selected from the local sorted list. These pivots are broadcast to other GPUs. Then, each GPU sorts all received pivots and selects certain global pivots that should be identical on all GPUs. Based on these global pivots, each GPU's local sorted set is separated into [n/p] partitions.

3. **Partition Exchange.** The *i*th GPU keeps its local *i*th partition and sends others to their corresponding GPUs. All-to-all communication occurs here. GPUDirect is used when multiple GPUs are involved in single-round mode.
4. **Merging Partitions.** Each GPU receives its [*n/p*] partitions from all others and concatenate them together for the output of this stage.

With sorted key-value pairs, MGMR removes all duplicated keys and all values associated with the same keys are stored continuously. Therefore, MGMR can refer each value list through the index of its first value and the list length. All operations are accomplished by GPUs for high performance.

3.5 *Reduce* Stage

Partition Scheduler. *Partition Scheduler* is only used if the input of *Reduce* stage exceeds the sum of all GPUs' memory sizes as in multi-round mode. Fig. 2 shows the detail of how Partition Scheduler works. After *Shuffle* stage, *Partition Scheduler* maintains all value list partitions in data pool on CPU side. The indices of these value lists have been built in advance. When a GPU is idle and its reducer workers come to ask for more *Reduce* work, the *Partition Scheduler* will assign several value lists as a combination with the consideration of load balancing and transfer it over to the designated GPU.

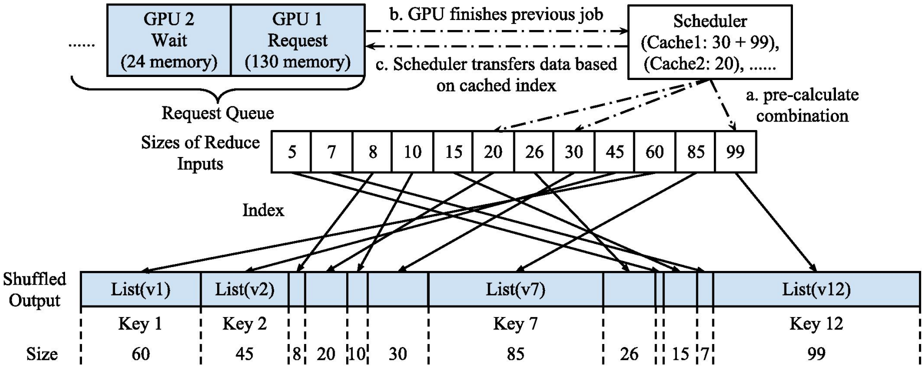


Fig. 2. The interaction between Partition Scheduler and GPUs

An approximate algorithm of subset sum problem from Przydatek 16 is used to estimate how much data can be packed into each GPU memory that will infer the workload. As shown in Fig. 2, while GPUs are busy with their reducers, multiple CPU threads concurrently calculate the possible input combinations for next round GPU execution. When a GPU finishes its work, it can get another one right from *Partition Scheduler*. Self-scheduling is applied for load balancing. However, those CPU threads get jobs ready for GPUs.

Key-Value Processing. In this sub-stage, user-defined reducer function will executed by hardware threads on GPUs. Similar to the situation in *Map* Stage, multiple GPUs' computation and PCIe's bidirectional communication capacities are exploited thoroughly to utilize the advanced features from Nvidia Fermi architecture. Atomic operations are avoided by CUDA version parallel prefix sum where data items are reduced in parallel.

4 Experimental Results

All experiments were conducted on a server containing two Intel Xeon X5660 (2.80GHz, totally 24 cores) with 24 GB RAM and two Nvidia GPUs: Quadro 6000 (1.15 GHz, 5,375 MB global memory, 64KB L1-cache/SM) and Tesla C2070 (1.15 GHz, 5,375 MB global memory, 64KB L1-cache/SM). The server is running the GNU/Linux operating system with kernel version 2.6.32. Testing applications are implemented with CUDA 5.0 and compiled with NVCC compiler in CUDA Toolkit 5.0. CPU versions are implemented with OpenMP using 24 threads to utilize all 24 CPU cores for full capacity.

Two real applications were used for experiments.

4.1 K-Means Clustering

K-Means Clustering (KMC)17 is used in data mining which aims to partition *n* observations into *k* clusters where all observations in a cluster are close to the nearest mean. The testing data is randomly generated from a 10k×10k square area with floating-point coordinates. *Map* stages finds the cluster for each point based on means and emit 〈index (cluster), point〉. *Partial Folder* is used to reduce I/O, so only the sum of the x-y coordinates of each cluster is emitted to *Reduce* stage that calculates new means. These three steps are repeated until all means stop changing. Since KMC is NP-hard, we set the test to three rounds for measurable performance. Also, the cluster number *k* is set to 24 for fair comparison between CPU and GPUs since there are totally 24 CPU cores.

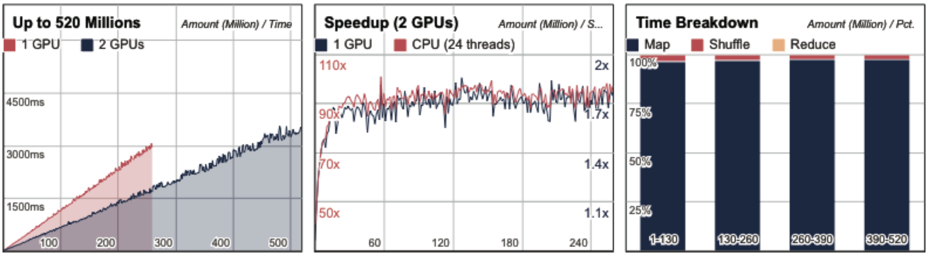


Fig. 3. Experimental results of KMC: execution time, speedup, runtime breakdown

As shown in Fig. 3, we generated up to 520 million points for the experiments. KMC is quite computationally intensive since each point needs to compare with 24 means to find the closest one. Multi-GPU version can declare clear computability advantage. Double-GPU version achieves 91.7 times speed-up over CPU version and 1.7 times speed-up over single-GPU version. *Shuffle* stage takes a small percentage of execution time because of the optimization of *Partial Folder* sub-stage. For the same reason, *Reduce* stage is very light-weighted.

4.2 Unique Phrase Pattern

Unique Phrase Pattern (UPP) can detect the most frequently used phrase patterns. Only two to three-word phrases are acceptable to our tests. The input data is randomly generated from a forty thousand-word dictionary that is pre-hashed. In MGMR, UPP is developed as a three-pass MapReduce. Therefore, no extra work is needed for allocating different sizes of buffers for different phrases. The first two passes count 2-word and 3-word phrases separately. Key-value pairs are emitted as 〈*list* (*hash* (*word1*), ...), *1*〉. Both results are used as the input for the third pass in order to sort all phrases in one mapper for their occurrence. Since UPP originally bounded by I/O, the sub-stage *Partial Folder* in *Shuffle* stage is activated in each pass to reduce the data transfer overhead.

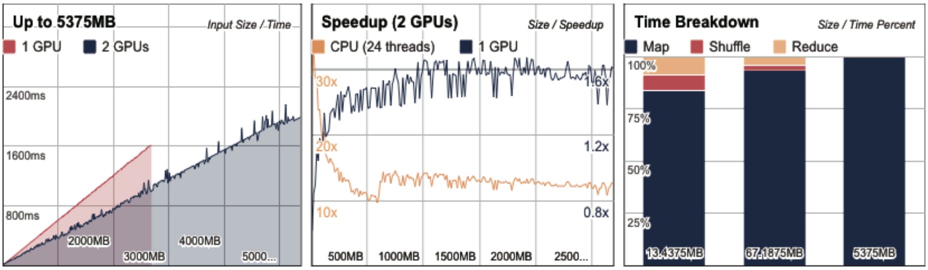


Fig. 4. Experimental results of UPP: execution time, speedup and runtime breakdown

UPP experimental results are shown in Fig. 4. Both GPU versions show very stable efficiency while the problem size increases. Double-GPU version achieves 12.6 times speed-up over CPU version and 1.5 times speed-up over single-GPU version in average. Single-GPU version is only slightly faster than double-GPU version when the input size is very small (less than 45 MB) because of the low GPU occupancy and communication overhead in double-GPU versions' (*Shuffle* stage). According to the runtime breakdown figure, *Map* stage is the most time-consuming portion.

5 Related Work

MapReduce has been implemented on many different platforms such as shared memory system, computer cluster, and GPU workstation. Each implementation has its own contributions and potential issues.

Hadoop 7 MapReduce developed by Apache Software is designed for better programmability in processing vast amount of data in cluster. Hadoop was developed in Java, but Hadoop Streaming allows users to customize their own *Map* and *Reduce* functions in other programming languages such as C and python. Phoenix 8 is a MapReduce implementation for shared-memory systems with multi-core chips and symmetric multiprocessors. Only CPU cores are utilized.

Mars 9 is the first GPU-based MapReduce system. Mars uses an atomic-free output-handling scheme on GPUs, but it has a two-step output process in order to calculate the data allocation and avoid race condition among threads. MapCG 10 designs a memory allocator to reserve buffers in GPU global memory for each warp. However, the atomic operations with global memory cause serious performance penalty. Chen and Agrawal 11 optimized MapCG by executing the Reduce function in GPU shared memory and achieved 2-60 times speedups. GPMR 12 implements MapReduce on GPU clusters to handle big data issue. Partial reductions and accumulation are used to reduce network traffic.

6 Conclusions and Future Work

In this paper, a multi-GPU MapReduce, called MGMR, is developed to tackle with big data issue. Scalability is achieved in both computational power and data size aspects. To avoid the possible communication overhead when multiple GPUs are employed, GPUDirect is applied for inter-GPU interactions without going through CPU memory. Unlike most existing GPU MapReduce systems, MGMR also considers big data input scenario. When data size is larger than the aggregate GPU memory, CPU memory is used to continue the MapReduce operation. Atomic operations are replaced by parallelize one as well. Experimental results have demonstrated MGMR's advantages over both CPU and single-GPU MapReduce in both performance and scalability aspects.

The future work includes extending MGMR to GPU clusters by using RDMA for further performance scalability, integrating it with file systems for fault tolerance, and improving its easy-to-use aspect for programmability.

Acknowledgements. This research is based upon work supported by the Industrial Strategic Technology Development Program (10041971, Development of Power Efficient High-Performance Multimedia Contents Service Technology using Context-Adapting Distributed Transcoding) funded by the Ministry of Knowledge Economy (MKE), Korea, and National Science Council (NSC), Taiwan under grant NSC101-2915-I-126-001, and NVIDIA.

References

1. NVIDIA CUDA Programming Guide 5.0, http://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html
2. OpenCL - The open standard for parallel programming of heterogeneous systems, http://www.khronos.org/opencl

3. Caylor, M.: Numerical Solution of the Wave Equation on Dual-GPU Platforms Using Brook+. Presentation, Boise State University (2010)
4. Dean, J., Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters. Communications of the ACM 51(1), 107–113 (2008)
5. Elteir, M., Lin, H., Feng, W., Scogland, T.: StreamMR: An Optimized MapReduce Framework for AMD GPUs. In: Proceedings of the 21st International Symposium on High-Performance Parallel and Distributed Computing, pp. 364–371 (2011)
6. Shainer, G., Ayoub, A., Lui, P., Kagan, M., Trott, C., Scantlen, G., Crozier, P.: The development of Mellanox/NVIDIA GPU Direct over InfiniBand a new model for GPU to GPU communications. Computer Science - Research and Development 26(3-4), 267–273 (2011)
7. White, T.: Hadoop: The Definitive Guide. O'Reilly Media, Inc./ Yahoo Press (2010)
8. Ranger, C., Raghuraman, R., Penmetsa, A., Bradski, G., Kozyraki, C.: Evaluating MapReduce for Multi-core and Multiprocessor Systems. In: Proceedings of the 2007 IEEE 13th International Symposium on High Performance Computer Architecture, pp. 13–24 (2007)
9. Fang, W., He, B., Luo, Q., Govindaraju, N.K.: Mars: Accelerating MapReduce with Graphics Processors. In: Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems, pp. 608–620 (2011)
10. Hong, C.T., Chen, D.H., Chen, Y.B., Chen, W.G., Zheng, W.M., Lin, H.B.: Providing Source Code Level Portability Between CPU and GPU with MapCG. Journal of Computer Science and Technology 27(1), 42–56 (2012)
11. Chen, L., Agrawal, G.: Optimizing MapReduce for GPUs with effective shared memory usage. In: Proceedings of the 21st International Symposium on High-Performance Parallel and Distributed Computing, pp. 199–210 (2012)
12. Stuart, J.A., Owens, J.D.: Multi-GPU MapReduce on GPU Clusters. In: Proceedings of the 2011 IEEE International Parallel & Distributed Processing Symposium, pp. 1068–1079 (2011)
13. Alam, S.R., Fourestey, G., Videau, B., Genovese, L., Goedecker, S., Dugan, N.: Overlapping Computations with Communications and I/O Explicitly Using OpenMP Based Heterogeneous Threading Models. In: Proceedings of the 8th International Conference on OpenMP in a Heterogeneous World, pp. 267–270 (2012)
14. Bell, N., Hoberock, J.: Thrust: A productivity-oriented library for CUDA. In: GPU Computing Gems: Jade Edition, pp. 359–371. Morgan Kaufmann (2011)
15. Li, X., Lu, P., Schaeffer, J., Shillington, J., Wong, P.S., Shi, H.: On the Versatility of Parallel Sorting by Regular Sampling. Journal of Parallel Computing 19(10), 1079–1103 (1993)
16. Przydatek, B.: A Fast Approximation Algorithm for the Subset-sum Problem. Journal of International Transactions in Operational Research 9(4), 437–459 (2002)
17. Yu, S., Tranchevent, L.-C., Liu, X., Glanzel, W., Suykens, J.A.K., De Moor, B., Moreau, Y.: Optimized data fusion for kernel k-means clustering. Journal of IEEE Transactions on Pattern Analysis and Machine Intelligence 34(5), 1031–1039 (2012)

DDoS Analysis Using Correlation Coefficient Based on Kolmogorov Complexity

Sung-ju Kim, Byung Chul Kim, and Jae Yong Lee

Department of Information Communications Engineering, Chungnam National University,
220, Gung Dong, Yusung Gu, Daejeon 305-764, Korea
{be3sowon,byckim,jyl}@cnu.ac.kr

Abstract. This paper describes an approach to detecting distributed denial of services (DDoS) attacks that is based on Information theory, specifically Kolmogorov Complexity. A theorem derived using principles of Kolmogorov Complexity describes that the joint complexity measure of random strings is lower than the sum of complexities of the individual strings when the strings exhibit some correlation. However, Kolmogorov complexity is not calculable, various methods exist to measure estimates of complexity. In the viewpoint of Kolmogorov complexity, we have found out the characteristics of DDoS attacks after analyzing a lot of DDoS attack cases. We propose a new method to compute the joint complexity using Deep Packet Inspection (DPI). DPI depends on string matching process and regular expression heuristics that make a thorough investigation on the packet payloads in a search for networked application signatures. As ISPs backbone links' speed and data volume increase rapidly, commodity hardware-based DPI systems face performance bottlenecks and the difficulty of scalability, which interferes on traffic classification accuracy dramatically. This paper introduces a lightweight DPI algorithm for an expeditious detection that can detect the presence of a DDoS in the Internet as quickly as possible in order to provide people accurate early warning information and possible reaction time for counteractions. Furthermore, it increases the exactitude of detecting DDoS and doesn't decrease network backbone's performance.

1 Introduction

Nowadays, Internet Service Providers (ISP) confront with a wide range of unusual events – some of which may be DDoS. DDoS has caused severe damage to servers and network. ISP should detect these DDoS early when they occur and responses appropriately as quickly as possible when they occur in order to downscale the damage. The principal challenge is how to detect DDoS in an incipient stage. ISP has been recently relying on Deep Packet Inspection (DPI) systems. For instance, there are some new software and devices capable of performing DPI, such as intelligent switching and routing, next generation firewalls, and intrusion detection and prevention systems. DPI's accuracy mostly depends on string matching process and regular expression heuristics that investigate thoroughly on the packet payloads in a search for networked application signatures. Traditionally, DDoS attacks are carried

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 443–452, 2013.

out at the network layer, such as ICMP flooding, SYN flooding, and UDP flooding. The main purpose of these attacks is to deplete the network bandwidth and deny service to legitimate users of the victim systems. Since many studies have found the characteristics of such attacks, it is not as easy as for attackers to launch new DDoS attacks based on network layer. These methods are data mining [2], bloom filter [3], outlier detection [4], nearest-neighbor methods [5], support vector machines [6], Y-means clustering algorithm [7], genetic computation [8], principal component analysis [9], Covariance Matrix [10], Kalman Filter [11] and traditional signature based DPI for detecting the latest DDoS. On July 7, 2009, major political, financial, and media websites around the world experienced new DDoS attacks. Since it was more intelligent and quiet attacks based on application layer, it could not be easily detected with the existing methods based on network layer. The emergence of new attacks changes the attack trend. It becomes more complicated to distinguish between normal traffic and DDoS traffic. In the viewpoint of a user, a zombie user behaves like a normal user. However, attackers run a massive number of queries through the victim's search engine or database query to bring the server down. A new method has been emerging such as slowloris DoS HTTP, Slow HTTP Post, HTTP GET Flooding, Cache-Control, Refresh, SQL Injection and so on. The characteristic of these attacks is that they request specific web pages and content repeatedly after connecting TCP 3way handshaking. Furthermore, they don't send excessive traffic anymore. They make the appropriate amount of sessions for the target web server to waste whole resource. A botmaster will use capacity estimation techniques like capprobe [12] to approximate the link capacity. During this process, the available bandwidth is monitored by using tools like abget [13] or pathchirp [14]. Simple scripts [13] can estimate the size of web pages. A botmaster can set a limit of the web page size between 100KB and 1MB for a bot to request [15]. A larger number of web servers will allow a sparser distribution of the attack traffic. This would make attack traffic realistic and evade detection from systems which target sudden increase in the traffic to a destination address. It makes more difficult for current techniques to detect. In this background, we propose a new DPI algorithm to find the latest DDoS early on the basis of Kolmogorov Complexity. A theorem derived using principles of Kolmogorov Complexity states that the joint complexity measure of random strings is lower than the sum of the complexities of the individual strings when the strings exhibit some correlation [1]. While it is known that, in general, Kolmogorov Complexity is not computable, various methods exist to calculate estimates of the complexity. Since we have studied many DDoS attacks, we found that the approach using packet's payload information can increase the detection accuracy. Our approach is the estimation of consecutive payload complexity in a flow. The remainder of the paper is organized as follows. Section2 states the study of various DDoS attacks. Section3 describes the concept of information complexity, specifically Kolmogorov Complexity. Section4 presents the estimation of consecutive payload complexity in a flow. Section5 proves the estimation. Finally, section6 concludes this paper.

2 The Study of Various DDoS Attacks

DDoS attacks have been evolving year by year. In the past, most DDoS exploited network level weakness. TCP SYN Flooding and UDP Flooding are representative attacks. However, application level attacks are increasing these days. There are HTTP GET Flooding, Refresh attack, SQL Injection attack, CC attack and so on. Furthermore, it is difficult to distinguish between application level attacks and normal traffics. Because, a small number of zombie PCs can damage a target web server or network link in application level DDoS attacks. On the other hand, network level DDoS attacks need a lot of zombie PCs. DDoS can be classified into resource depletion and weakness exploit on system and application. Also, resource depletion attacks can be divided into network, especially bandwidth, and host resource.

2.1 Attack on Bandwidth

- UDP/ICMP Flooding

These attacks make network link congestion or system overload by sending a lot of UDP/ICMP packets. One of ICMP Flooding examples is Smurf. This can be considered one form of reflected attack, as the flooding hosts send Echo Requests to the broadcast addresses of mis-configured networks, thereby enticing many hosts to send Echo Reply packets to the victim. Some early DDoS programs implemented a distributed form of this attack.

- DRDOS(Distributed Reflected Denial of Service)

This involves sending forged requests of some type to a very large number of computers that will reply to the requests. Using Internet Protocol address spoofing, the source address is set to that of the targeted victim, which means all the replies will go to the target.

2.2 Attack on Host Resource

- Slowloris DoS HTTP

Slowloris tries to keep many connections to the target web server open and hold them open as long as possible. It accomplishes this by opening connections to the target web server and sending a partial request. Periodically, it will send subsequent HTTP headers, adding to the request. Affected servers will keep these connections open, filling their maximum concurrent connection pool, eventually denying additional connection attempts from clients.

- Slow HTTP Post attack

This was introduced in 2010 OWASP. The attacker sends POST headers with a legitimate “content-length” field that lets the Web server know how much data is arriving. Once the headers are sent, the POST message body is transmitted at a slow speed to gridlock the connection and use server resources.

- HTTP GET Flooding attack

The infected hosts create many threads to send a large amount of requests to the victim’s website to disable it. Since these requests have legitimate contents and are sent via normal TCP connections, the server usually serves them as normal requests, and exhausts its resource finally. The attack launched by the worm Mydoom in

2004 is an example of HTTP flooding. Recently, there have been intensive DDoS attacks against major government, organization news media and financial company websites in South Korea and US around July 7, 2009

2.3 Attack on System/Application Weakness

- Ping of death

This is a type of attack on a computer that involves sending a malformed or otherwise malicious ping to a computer. A ping is normally 32 bytes in size. Many computer systems could not handle a ping packet larger than the maximum IPv4 packet size, which is 65,535 bytes. Sending a ping of this size could crash the target computer

- SQL injection attack

This is a technique often used to attack databases through a website. This is done by including portions of SQL statements in a web form entry field in an attempt to get the website to pass a newly formed rogue SQL command to the database

3 Information Complexity

Detection techniques are classified into anomaly-based detection and misuse-based detection. Anomaly-based detection looks for deviation from the pattern of normal traffic using techniques such as statistical measures. In contrast to anomaly-based detection, misuse-based detection maintains a database of signatures. Suspicious flows are matched for their signature against this database. Traditional IDS and IPS make use of a database of signatures. While these techniques have high accuracy, they have several limitations. Firstly, they are not flexible because they are typically customized for known attack patterns. Therefore, these techniques are likely to fail if a new type of packet is used or if the attack consists of a traffic pattern that is a combination of different types of packets. Secondly, a large number of detection techniques use traffic logs to identify attacks. However, traffic logs generate a large amount of data even during normal operation so it is difficult and time-consuming to scan traffic logs looking for patterns when the network is under attack. Typical detection techniques rely on filtering based on packet type and rate. These are effective to detect attacks on bandwidth. These days, it is hard to find excessive attack packets in a flow. Because they disguised as normal traffics by controlling traffic rates. With the appearance of a botnets, every flow does not send highly deviated packets. Attacks on host resource are good examples. Misuse-based detection and traditional IDS/IPS has a problem to detect these attacks. This paper describes an anomaly-based detection approach that leverages fundamentals of information complexity to provide a flexible and effective detection method. Let's take a look at the approach in detail

3.1 Kolmogorov Complexity

The DDoS attack detection algorithm makes use of a fundamental theorem of Kolmogorov Complexity that states: for any two random strings X and Y,

$$\mathrm{K(XY)} \leq \mathrm{K(X)} + \mathrm{K(Y)} + \mathrm{c} \tag{1}$$

Where K(X) and K(Y) are the complexities of the respective strings, c is a constant and K(XY) is the joint complexity of the concatenation of the strings. Proof for the above theorem is described in [16]. Simply put, the joint Kolmogorov Complexity of two strings is less than or equal to the sum of the complexities of the individual strings. The equivalence holds when the two strings X and Y are totally random, they are completely unrelated to each other. While it is known that, in general, Kolmogorov Complexity is not computable, various methods exist to compute estimates of the complexity. One of them is an entropy calculation technique for estimation of complexity. The complexity K(x) is computed using the entropy H(p) of the weight of ones ('1') in a string. Specifically, K(x) is defined in Eq. (2) where x#1 is the number of '1' bits and x#0 is the number of '0' bits in the string whose complexity is to be determined. Entropy H(p) is calculated using Eq. (3)

$$\mathrm{K(x)} \approx \mathrm{l(x)H}\left(\frac{\mathrm{x\#1}}{\mathrm{x\#1} + \mathrm{x\#0}}\right) + \log_2\big(\mathrm{l(x)}\big) \tag{2}$$

$$\mathrm{H(p)} = -\mathrm{p}\log_2 \mathrm{p} - (1 - \mathrm{p})\log_2(1 - \mathrm{p}) \tag{3}$$

The complexity estimation technique used here is not the best because empirical entropy is actually a very poor method of complexity estimation. For example, the estimate for the string 10101010 and a completely random string with equal numbers of 1's and 0's is the same under empirical entropy [1].

3.2 Correlation

The most familiar measure of dependence between two quantities is the Pearson product-moment correlation coefficient, or Pearson's correlation. It is obtained by dividing the covariance of the two variables by the product of their standard deviations. The population coefficient ${}_{X,Y}$ between two random variables *X* and *Y* with expected values μ_X and μ_Y and standard deviations ${}_X$ and ${}_Y$ is defined as:

$$\rho_{\mathrm{X,Y}} = \mathrm{corr(X,Y)} = \frac{\mathrm{cov(X,Y)}}{\sigma_{\mathrm{X}}\sigma_{\mathrm{Y}}} = \frac{\mathrm{E}\big[(\mathrm{X} - \mu_{\mathrm{X}})(\mathrm{Y} - \mu_{\mathrm{Y}})\big]}{\sigma_{\mathrm{X}}\sigma_{\mathrm{Y}}} \tag{4}$$

E is the expected value operator, *cov* means covariance, and, *corr* a widely used alternative notation for Pearson's correlation. The Pearson correlation is defined only if both of the standard deviations are finite and both of them are nonzero

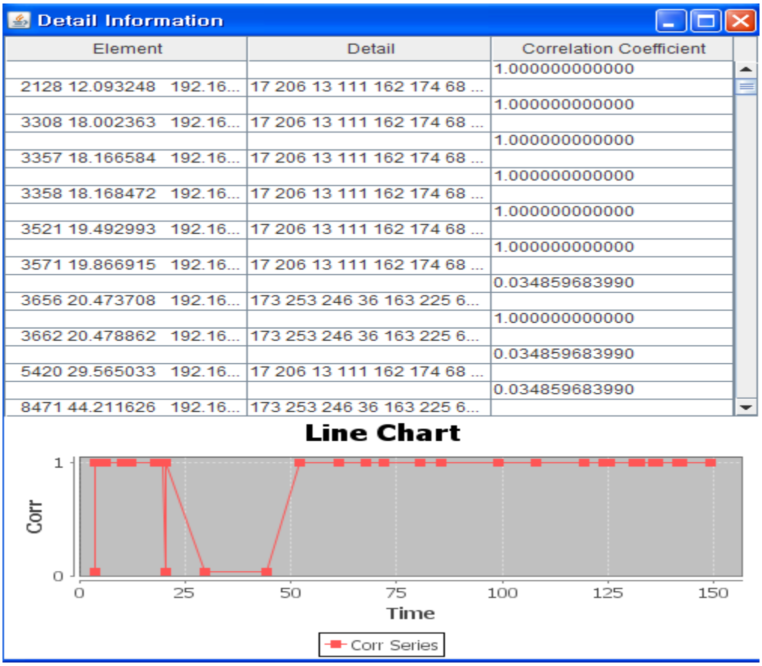
3.3 Proposal

An analysis of recently reported cases of DDoS revealed the following issues:

1. Difficulty in detection of problem
   It is difficult to determine the cause at the early stage of attack – is it by mechanical failure? Is it due to explosive connection requests? - causing delays in response.
2. Invisible enemy
   It is difficult to identify how long the heavy traffic caused by access through multiple unknown PCs will last, what the purpose of the attack is, how many more attacks will ensue, etc.
3. It is difficult to detect at an early point, as packets are sent within the normal range, rather than a PC transmitting a large volume of packets.
4. It is difficult to analyze the entire large data through real-time packing capturing.

In order to address these challenges, an algorithm that will allow effective detection of DDoS based on a sampling survey must be used. An analysis of recent attack traffic including the attacks on July 7 found that the attacks are carried out either by generating abnormal traffic or by using attack codes already hidden, and as such, the pattern of the attack is inevitably monotonous compared to normal traffic. In addition, the attack traffic transmits garbage data compared to normal data. This is because the attack traffic is not concerned about receiving a response from the victim and sends out useless data unilaterally. For this reason it is believed that there is a strong correlation between payloads of attack traffic. This characteristic is put into a numerical perspective to be used for determining attack traffic. As stated above, the concept of Kolmogorov Complexity was proved [16]. However, the estimation is inappropriate to detect the latest DDoS. Although Pearson product-moment correlation coefficient can be a good estimation for the concept of Kolmogorov Complexity, it does not aim at detecting DDoS. We propose a new correlation algorithm for the latest DDoS. We assume that an attacker performs an attack using large numbers of similar packets because zombies are controlled by a botmaster. We mainly focus on high degree of payload similarity in flows. Because the flow information such as source IP, destination IP, source port and destination port is already applied. Finally, DDoS can be defined as a series of high correlated payloads in a flow. α is defined as the number of payloads to be investigated. We extract a payload and call it as t and consecutive payloads are investigated using correlation.

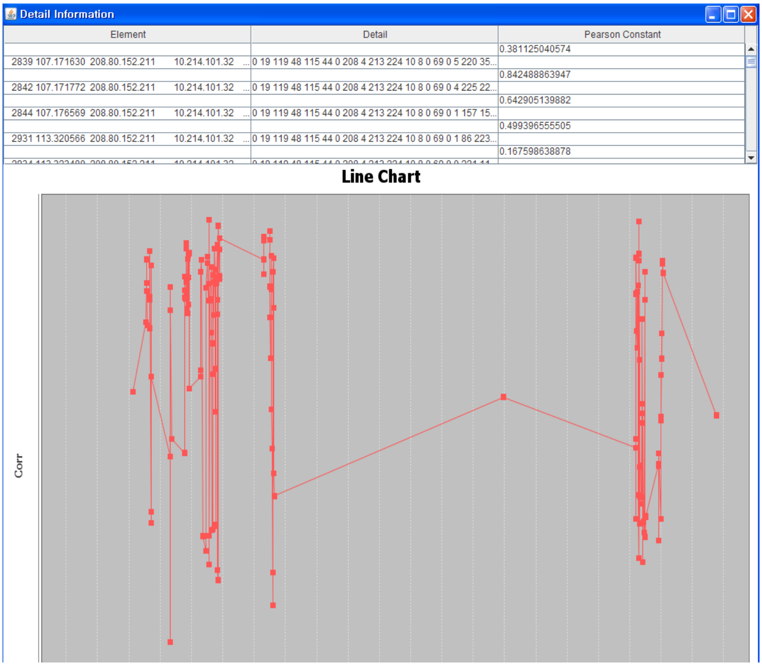
$$\stackrel{\text{def}}{=} \text{corr series in a flow} = \sum_{n=t}^{t+\alpha} \rho_{x_n x_{n+1}} / \alpha \qquad (5)$$



Detail Information

Element	Detail	Correlation Coefficient
		1.000000000000
2128 12.093248 192.16...	17 206 13 111 162 174 68 ...	
		1.000000000000
3308 18.002363 192.16...	17 206 13 111 162 174 68 ...	
		1.000000000000
3357 18.166584 192.16...	17 206 13 111 162 174 68 ...	
		1.000000000000
3358 18.168472 192.16...	17 206 13 111 162 174 68 ...	
		1.000000000000
3521 19.492993 192.16...	17 206 13 111 162 174 68 ...	
		1.000000000000
3571 19.866915 192.16...	17 206 13 111 162 174 68 ...	
		0.034859683990
3656 20.473708 192.16...	173 253 246 36 163 225 6...	
		1.000000000000
3662 20.478862 192.16...	173 253 246 36 163 225 6...	
		0.034859683990
5420 29.565033 192.16...	17 206 13 111 162 174 68 ...	
		0.034859683990
8471 44.211626 192.16...	173 253 246 36 163 225 6...	

Fig. 1. Correlation coefficient series of 7.7 DDoS dataset



Detail Information

Element	Detail	Pearson Constant
		0.381125040574
2839 107.171630 208.80.152.211 10.214.101.32 ...	0 19 119 48 115 44 0 208 4 213 224 10 8 0 69 0 5 220 35...	
		0.842488863947
2842 107.171772 208.80.152.211 10.214.101.32 ...	0 19 119 48 115 44 0 208 4 213 224 10 8 0 69 0 4 225 22...	
		0.642905139882
2844 107.176569 208.80.152.211 10.214.101.32 ...	0 19 119 48 115 44 0 208 4 213 224 10 8 0 69 0 1 157 15...	
		0.499396555505
2931 113.320566 208.80.152.211 10.214.101.32 ...	0 19 119 48 115 44 0 208 4 213 224 10 8 0 69 0 1 86 223...	
		0.167598638878

Fig. 2. Correlation coefficient series of general web dataset

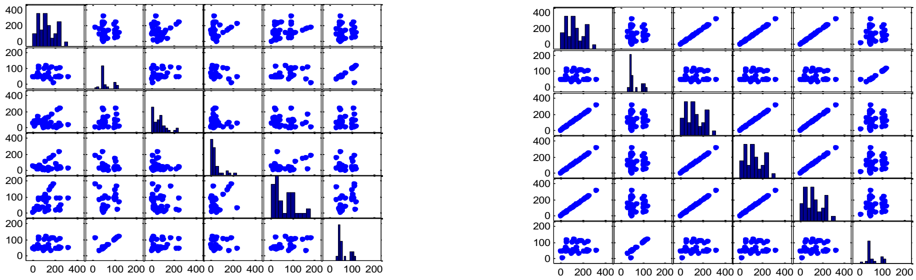


Fig. 3. Plot matrix of general web dataset vs July 7 DDoS dataset using matlab

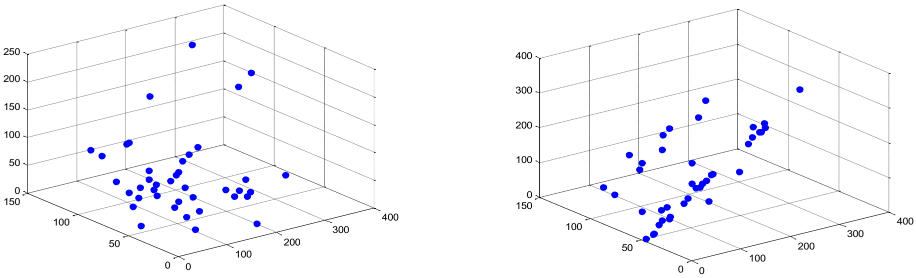


Fig. 4. 3D scatter of general web dataset vs July 7 DDoS dataset using matlab

We developed a program to calculate the correlation series in a flow using Java. Figure1 shows the analysis result of July 7 DDoS dataset and Figure2 show that of general web dataset. In case of July 7 DDoS dataset, it shows high correlation coefficient. However, in case of general web dataset, it shows low correlation coefficient. This phenomenon is appeared same when we analyze the dataset using matlab. Figure3 compares plot matrix of normal dataset and July 7 DDoS dataset using matlab. And, Figure4 shows 3D scatter of the dataset using matlab. The figure above illustrates 3D scatter of normal data and attack data. While normal data has a large distribution, attack data has regularity in the diagonal direction. From these figures, it can be said that there is lower correlation amongst normal data, whilst there is a greater correlation amongst attack data.

4 The Estimation of Payload Complexity

To validate this theory, the following null hypothesis and alternative hypothesis were established.

Ho: The payload correction coefficient of the traffic per flow by malignant code, μ is 0.7

H1: The payload correction coefficient of the attack traffic by malignant code, μ is 0.5 To validate the above hypotheses, the correlation index between successive data payloads was examined by coding the payload part of the traffic per flow and by calculating the correlation coefficient. The data analysis technique used in this paper is as follows:

1. Analysis of successive packets per flow:
   The flow has independent content from other flows. Thus, normal, successive packets cannot contain redundant content.
2. Handling of retransmission data:
   Redundant packets caused by retransmission are not collected.
3. Sampling per flow:

Surveys of packets sampled per flow can reduce the amount needed for surveys to determine a DDoS attack, while also allowing real-time analysis. A sample group to validate the hypotheses was created, as shown in the table above. The data have small-sized samples with the population standard deviation unknown, and as such, a t-test needs to be performed. Therefore, the test statistic is as follows:

$$\mathrm{t} = \frac{\overline{\mathrm{X}} - 0.7}{\mathrm{S}/\sqrt{10}} = \frac{0.855229 - 0.7}{0.177024/\sqrt{10}} = 2.772948 \tag{6}$$

Here, the test statistic t follows a t-distribution whose degree of freedom is 9 under Ho. At a 0.05 significance level, the rejection region is R: $\mathrm{t} \geq \mathrm{t}_{0.05}(9) = 1.833$. Therefore, the null hypothesis can be rejected at a significance level of 0.05.

Table 1. Sample of correlation coefficient of July 7 dataset

Number	Correlation Coefficient	Number	Correlation Coefficient
1	1.00000	6	1.00000
2	0.517430	7	1.00000
3	1.00000	8	1.00000
4	0.678287	9	0.628287
5	0.839143	10	0.839143

5 Conclusion

The study proposed a new analytical method that allows more accurate identification of DDoS attacks. In the viewpoint of data volume in the Internet, it has been exploding year by year. Thus, more lightweight and accurate method is required to investigate wide range of data in the Internet. The proposal satisfies the request.

This is based on the assumption that hackers perform DDoS attack using large numbers of similar packets. The verification shows that unlike normal traffic, DDoS attack has a high payload correlation coefficient when we applied the algorithm to 7.7 DDoS dataset. Through this, when the payload correlation is used to determine DDoS in consecutive payloads, it will not only improve the accuracy of identification but it will also reduce the volume to be surveyed.

References

[1] Kulkarni, A., Bush, S.: Detecting Distributed Denial-of-Service Attacks Using Kolmogoriv Complexity Metrics. Journal of Network and Systems Management 14(1), 69–80 (2006)

[2] Lee, W., Stolfo, S.J.: Data Mining Approaches for Intrusion Detection. In: Proc. of the 7th USENIX Security Symposium, pp. 79–84 (January 1998)

[3] Broder, A., Mitzenmacher, M.: Network Applications of Bloom Filters. Internet Mathematics 1(4), 485–509 (2003)

[4] Lu, W., Traore, I.: A novel unsupervised anomaly detection framework for detecting network attacks in real-time. In: 4th International Conference on Cryptology and Network Security, China (December 2005)

[5] Eskin, E., Arnold, A., Prerau, M., Portnoy, L., Stolfo, S.: A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. Kluwer (2002)

[6] Peddabachigari, S., Abraham, A., Grosan, C., Thomas, J.: Modeling intrusion detection system using hybrid intelligent systems. Journal of Network and Computer Applications, 114–132 (January 2007)

[7] Guan, Y., Ghorbani, A.A., Belacel, N.: An unsupervised clustering algorithm for intrusion detection. In: Proc. of the 16th Canadian Conference on Artificial Intelligence, Canada, pp. 616–617. Springer (2003)

[8] Lu, W., Traore, I.: Detecting new forms of network intrusions using genetic programming. In: Computational Intelligence, pp. 475–494 (August 2004)

[9] Shyu, M.L., Chen, S., Sarinnapakorn, K., Chang, L.: A novel anomaly detection scheme based on principal component classifier. In: Proc. of the IEEE Foundations and New Directions of Data Mining Workshop, in Conjunction with the 3rd IEEE International Conference on Data Mining, pp. 172–179 (November 2003)

[10] Jin, S., Yeung, D.S., Wang, X.: Network intrusion detection in covariance feature space. Pattern Recognition, 2185–2197 (August 2007)

[11] Soule, A., Salamatian, K., Taft, N.: Combining Filtering and Statistical Methods for Anomaly Detection. In: Proc. of IEEE INFOCOM (2006)

[12] Kapoor, R., Chen, L., Lao, L., Gerla, M., Sanadidi, M.: CapProbe: a simple and accurate capacity estimation technique. In: Proc. ACM SIGCOMM 2004, USA, pp. 67–78 (2004)

[13] Antoniades, D., Athanatos, M., Papadogiannakis, A., Markatos, E., Dovrolis, C.: Available bandwidth measurement as simple as running wget. In: Proc. PAM 2006 (March 2006)

[14] Rebeiro, V., Reidi, R., Baranuik, R., Navratil, J., Cottrell, L.: pathChirp: efficient available bandwidth estimation for network paths. In: Proc. PAM 2003 (April 2003)

[15] Shevtekar, A., Ansari, N.: Is It Congestion or a DDoS Attack? IEEE Communications Letters 13(7) (July 2009)

[16] Bezeq, R., Kim, H., Rozovskii, B., Tartakovsky, A.: A Novel Approach to Detection of Denial-of-Service Attacks via Adaptive Sequential and Batch-Sequential Change-Point Methods. In: IEEE Systems, Man and Cybernetics Information Assurance Workshop (June 2001)

An Efficient Attribute-Based Encryption and Access Control Scheme for Cloud Storage Environment

Jyun-Yao Huang, Chen-Kang Chiang, and I-En Liao

Department of Computer Science and Engineering
National Chung Hsing University, Taichung, Taiwan
allen501pc@gmail.com, s99056051@cs.nchu.edu.tw,
ieliao@nchu.edu.tw

Abstract. With the prevalence of cloud computing, many enterprise users store confidential information in the cloud servers. Therefore, the problems of data security in cloud computing are particularly important. Cloud storage service providers must offer efficient cryptography system and access control scheme to users. In recent years, some researchers proposed identity-based hierarchical key deployment model for encryption and access control in cloud computing environment. However, some of these schemes have high computing cost and do not take authentication into consideration. In this paper, we proposed a low-cost cryptography system and attribute-based access control scheme for the cloud storage environment. The simulation results and analysis show that the proposed method has lower communication and computing cost than Hierarchical Attribute-Based Encryption (HABE). Our proposed scheme can achieve the data access control via user's attribute-based rules. It also satisfies the authentication requirements by using identity-based signature in the cloud storage environment.

1 Introduction

Cloud computing is a large-scale distributed computing paradigm[1]. With the emergence of cloud computing, numerous convenient services have been provided, such as Google Gmail, Amazon EC2, Facebook and Dropbox. However, many enterprises must trust the security policies and mechanisms of cloud service providers. Cloud service providers should satisfy security requirements, such as data encryption, key management, identity authentication, and access control[2].

In recent years, a number of researchers have proposed various hierarchical architectures for the key management of cloud computing[3][4][5]. These schemes have provided some of encryption, authentication, and access control mechanisms, but not all of them. However, these proposed schemes also have a number of disadvantages. First, the computation costs of encryption and decryption are high. Second, they lack the integration of key management, encryption, authentication and access control mechanisms. Therefore, how to develop efficient encryption, authentication, and access control mechanisms is the primary issue investigated in this study.

This study proposes an efficient attribute-based encryption and access control scheme for cloud storage environments. The characteristics of the proposed scheme are described below:

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 453–463, 2013.

(1) The scheme provides a method for encryption and decryption that has comparatively lower communication and computation costs. It also satisfies access control requirements by including attribute-based rules. Users who satisfy attribute-based rules can be permitted to download and decrypt data from cloud storage servers.

(2) The scheme provides identity-based signatures for authentication as users upload data and achieves non-repudiation.

(3) The scheme provides a suitable hierarchical key management method for cloud storage environments. Using hierarchical architecture, the generation and management of public and private keys does not incur server overheads.

The remainder of this study is organized as follows. In Section 2, related work on cloud computing, key management, access control are discussed. In Section 3, the proposed scheme for key management, encryption, authentication, access control, and theoretical analysis of performance are described. The experimental results and security analysis are discussed in Section 4. Section 5 provides the study conclusions.

2 Related Work

2.1 Cloud Computing

Cloud computing is a large-scale distributed computing paradigm. The National Institute of Standards and Technology (NIST) defined cloud computing as "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [6][7]. Typically, cloud providers have their own cloud infrastructures or corresponding applications to provide services to their customers. The following three service models are commonly used for cloud computing: (1) Infrastructure as a Service (IaaS), which provides cloud computing infrastructures for customers. (2) Platform as a Service (PaaS), which provides both IaaSs and platform components such as operating systems or required libraries. (3) Software as a Service (SaaS), which provides applications on the cloud computing platform. The NIST [6][7] has also defined the following deployment models for cloud computing: public cloud, private cloud, hybrid cloud and community cloud.

2.2 Key Management in Cloud Computing

In previous studies of key management [3][4][8][9], how to distribute and compute public/private keys which are computed by user or device identities has been a significant issue.

In identity-based key management systems, the widely used mathematical property is bilinear maps based on an ellipse curve cryptography system [10]. Boneh and Franklin [11] proposed a security model for identity-based encryption and proposed a construction method using bilinear maps. For bilinear maps, they considered a large prime p and E to be the elliptic curve. Two groups exist; that is, a group over curve

E/F_p^2 with a large order q and denoted as G_q, and μq as a subgroup of F_q^*, which is also of the q order. They contended that a modified bilinear pairing ê: $G_q \times G_q \rightarrow \mu_q$ is admissible if it possesses the following three properties:

(1) *Bilinear: For all* P, *Q*∈G_q *and a, b*∈*Z, ê(*aP, bQ*)= ê(*bP, aQ*)=ê(*P, Q*)*^{ab}. *This can be restated for all* P, Q, R∈G_q, *ê(*P+R, Q*)=ê(*P, Q*) ê(*R, Q*) and ê(*P, Q+R*)= ê(*P, Q*) ê(*P, R*).*
(2) *Non-degenerate: ê(P, P)∈Fq*, is an element of order q, and a generator of* μ_q .
(3) *Computable: Given* P, Q∈G_q, *an effective method to compute ê(*P, Q*) exists.*

They proposed the identity-based encryption method developed from admissible pairing; the security of this scheme was enhanced by computational Diffie-Hellman (CDH) and Bilinear Diffie-Hellman (BDH) assumptions. The BDH assumption is explained as follows:

Given P, aP, bP, $cP \in G_1$ for unknown random a, b, $c \in Z_p^*$, computing $r = ê(P, P)^{abc}$ is difficult when G_1 is a bilinear group. However, the method developed by Boneh and Franklin is not efficient for large networks. Jeremy Horwitz et al. [12] introduced a hierarchical concept for a two-level hierarchical ID-based encryption(HIDE). Subsequently, Gentry and Silverberg [13] proposed the practical scheme for this concept.

In recent years, Hongwei Li et al. [3] proposed an identity-based hierarchical model for cloud computing (IBHMCC). In their scheme, the cloud computing environment was composed of a hierarchical structure of nodes, and authentication was conducted using bilinear pairing. However, this scheme cannot defend against replay attacks, when attackers repeatedly transmit authentication messages to the server. Liang Yan et al. [4] adopted federated identity management combined with hierarchical identity-based cryptography (HIBC) using shared secret session key without a pre-shared secret key. However, Hongwei Li et al. [3] and Liang Yan et al. [4] did not consider the key regeneration problem when lower level PKGs failed. In a previous study, we proposed a robust and low-cost identity-based encryption method for a hierarchical key distribution model by considering PKG failures[14].

2.3 Access Control

In cloud storage service, how to retain the security of data while providing valid access ability is an issue that concerns users. For public data, data owners can use the basic access control APIs provided by cloud service providers to offer access control ability. For secret data, the data owner can use encryption and add an access control table into target data to limit the users' access abilities. This study refers to these methods as "ciphertext-policy encryption".

For cloud computing, Zhu et al. [15] provided role-based access control (RBAC)[16] to offer access control abilities to target users. However, this method involves the use of the receiver's personal information to encrypt data. To submit one file to various receivers, this method entails numerous encryption procedures. Therefore, this method is obviously unsuitable for broadcasting messages.

For access control, attribute-based access control is an alternative method. Sahai et al. [17] developed a primitive attribute-based encryption model based on the method presented by Boneh and Franklin [11]. This method used user attributes as the access control permissions because each user has unique attributes. Attribute rules were transformed into ciphertext, and the users whose secret keys satisfied the rules could decrypt the ciphertext successfully. Goyal et al. [18] provided key-policy attribute-based encryption (KP-ABE) for policy based rules. Bethencourt et al [19] developed ciphertext-policy attribute-based encryption (CP-ABE); The benefit of this method is that senders are not required to know the receivers, they simply incorporate the access rules into the encrypted data; the data is then suitably protected on the distributed system with widely unknown users. Therefore, this method has been extended to cloud computing environment.

In recent years, a number of studies have consolidated CP-ABE[19] and HIDE[13] for cloud computing. In 2010, Jie Wu et al. [5] proposed using hierarchical attribute-based encryption (HABE); This reduces the overheads of key generation procedures through hierarchical architecture, and this architecture is used to manage domain access conveniently. When one user wishes to share secret data, this scheme can use an access tree structure consisting of attribute sets to verify receivers' permissions. However, this scheme only provides encryption, signature authentication functions are lacking. Additionally, the scheme had higher decryption computation and communication costs.

3 Proposed Scheme

The objective of this study was to improve HABE[5] by reducing encryption, decryption, and signature authentication costs. The architecture proposed in this study is shown in Fig. 1.

As shown in Fig. 3, each cloud computing server possesses a public key and private key generator and is considered a PKG. “Root PKG” possesses different PKGs to “Domain PKG,” and each domain PKG has users and attribute data. For security reasons, attribute data are stored on one server under the domain PKG to separate users and the attribute data.

This study features various key notations; thus, a key notations list is provided in Table 1.

In this study, unique identifiers are used for each PKG, user, and attribute. The public keys generated using these unique identifiers comprise the following concepts:

(1) *If the public key under level i for the specific domain PKG is composed of an upper PKG public key PKi-1 and self-identifier IDi, it is denoted as PKi = (PKi-1,IDi).*

(2) *When one user with his identifier IDu add to PKG under level i including public key PKi, his public key is denoted by PKu = (PKi, IDu).*

If some data attributes with identifier ID_a stored in the PKG under level i, its' public key is denoted as PK_a= (PK_i,ID_a).

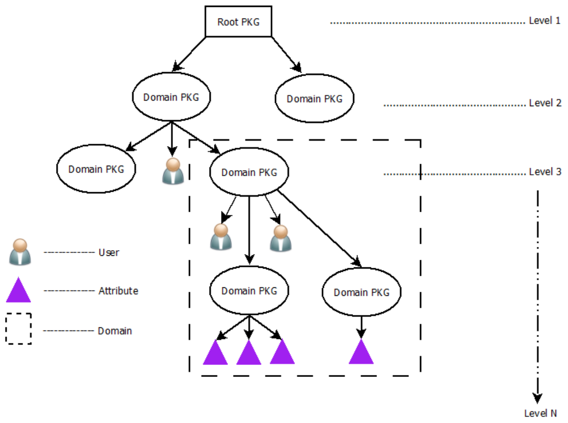


Fig. 1. System architecture

Table 1. Key notations list

Notation	Meaning	Purpose
PK_i	The public key of the *i*th domain PKG	To generate private keys for each sub-domain PKG
PK_a	The public key of various attribute data.	To generate private keys that satisfy a number of user attribute rules.
PK_u	The public key of a number of users.	To generate private keys for a number of users.
SK_i	The private key of the *i*th level domain PKG	To generate private keys for a number of users.
$SK_{i,u}$	The private key of user *u* for domain PKG under level *i*	Decryption
$SK_{i,u,a}$	The private key of user *u* for domain PKG under level *i*, which satisfies attribute *a*	Decryption

3.1 PKG Setup

Root PKG Setup: the root PKG setup is conducted using the following steps:

(1) *Select a random number* $r_0 \in Z_q$ *with a large prime* q, *two groups* G_1 *and* G_2 *with order* q, *and bilinear map* $\hat{e}$: $G_1 \times G_1 \rightarrow G_2$.

(2) *Generate* $P_0 \in G_1$ *randomly and compute* $Q_0 = r_0 P_0 \in G_1$.

(3) *Select three hash functions for encryption,* H_1*:* $\{0,1\}^* \rightarrow G_1$, H_2*:* $G_2 \rightarrow \{0,1\}^n$, H_A*:* $\{0, 1\} \rightarrow Z_q$, *where* n *is any positive integer.*

(4) *Generate system parameters* $<G_1, G_2, \hat{e}, Q_0, P_0, H_1, H_2, H_A>$ *for the lower level domain PKGs.*

Domain PKG Setup: Suppose the public and private keys of domain PKG in level i+1 are generated by the upper level domain PKG or root PKG in level *i*. Then, the key generation is proceeded by domain PKG or root PKG in level *i* using the following steps:

(1) Generate two distinct random r_{i+1} and $r'_{i+1} \in Z_q$.

(2) Compute public key $P_{i+1} = H_1(PK_{i+1}) \in G_1$ and secret $Q_{i+1}=r_{i+1}P_0 \in G_1$.

(3) Generate the private key $SK_{i+1} = Q_0+r'_{i+1}P_{i+1}$.

(4) Send system parameters $< G_1, G_2, \hat{e}, Q_0, P_0, H_1, H_2, H_A >$ to the sub-domain PKG.

3.2 User Addition

Assume that one user *u* with user public key PK_u who satisfies some data attributes *a* wants to join PKG_i (at level *i*) to access data. Then the PKG_i generate secret keys for the user using the following steps:

(1) Set $r_u = H_A(PK_u) \in Z_q$, $P_a = H_1(PK_a)$ and generate $r_i \in Z_q$.

(2) Generate two private keys $SK_{i,u} = r_i r_u P_0$ and $SK_{i,u,a} = SK_i + r_i r_u P_a$.

(3) Pass $SK_{i,u}$, $SK_{i,u,a}$ to user *u* under a secret channel.

3.3 Data Encryption and Decryption

Data Encryption: Assume that some user *u* in level *t* wants to upload file *f* to cloud servers. Then, he must proceeds the following steps:

(1) We use conjunctive clause(*CC*) to represent all attribute data in some server. Give disjunctive normal form(DNF) $A = \bigvee_{i=1}^{N} (CC_i) = \bigvee_{i=1}^{N} (\bigwedge_{j=1}^{n_i} a_{ij})$, where $N \in Z^+$ is the number of conjunctive clauses, $n_i \in Z^+$ is the number of attributes at the *i*th conjunctive clause (CC_i), and a_{ij} is the *j*th attribute at CC_i.

(2) Set $P_t = H_1(PK_t) \in G_1$ and $P_{a_{ij}} = H_1(PK_{a_{ij}}) \in G_1$, where $1 \leqq i < N$ and $1 \leqq j \leqq n_i$.

(3) Randomly select $\alpha \in Z_q$ and obtain the lowest common multiple $n_A=\{n_1,n_2,..., n_N\}$. Compute $U_0 = \alpha P_0, U_t = \alpha P_t$, $U_1 = \alpha \sum_{j=1}^{n_1} P_{a_{1j}}$, …,$U_N = \alpha \sum_{j=1}^{n_N} P_{a_{Nj}}$. Then, compute $V = f \oplus H_2\left(\hat{e}\left(\alpha Q_0, n_A P_0\right)\right)$ and $C_f = [U_0, U_t, U_1, \ldots, U_N, V]$.

(4) Output $CT = (A, C_f)$ to the target server.

Decryption: When user *u* wishes to obtain file *f* from one server in level *t*, they can pass the parameters $SK_{t,u}$ and $SK_{t,u,a_{ij}}$ to the server to decrypt *CT*. The server decryption procedure is run as follows to get file *f*:

$$V\oplus H_2\left(\frac{\hat{e}\left(U_0,\frac{n_A}{n_i}\sum_{j=1}^{n_i}SK_{t,u,a_{ij}}\right)}{\hat{e}\left(Q_t,n_AU_t\right)\hat{e}\left(SK_{t,u},\frac{n_A}{n_i}U_i\right)}\right)=V\oplus H_2\left(\frac{\hat{e}\left(Q_0,n_A\alpha P_0\right)\hat{e}\left(Q_t,n_AU_t\right)\hat{e}\left(SK_{t,u},\frac{n_A}{n_i}U_i\right)}{\hat{e}\left(Q_t,n_AU_t\right)\hat{e}\left(SK_{t,u},\frac{n_A}{n_i}U_i\right)}\right)=$$

$$V\oplus H_2\left(\hat{e}\left(\alpha Q_0,n_AP_0\right)\right)=f$$

3.4 Signature

If user *u* in level *i* wishes to add a signature to data *f*, they should use the following procedure:

(1) Compute $P_m = H_1(ID_u \parallel f)$.
(2) Use the current time string T_C to compute $t = H_A(T_C)$.
(3) Randomly generate $\beta\in Z_q$.
(4) Compute $\mu= \beta SK_{i,u} + t\beta P_\mathrm{m}P_u$.
(5) Output signature $<\mu, \beta SK_{i,u} , t\beta P_mP_0> = <\mu, A_i, B_i >$.

When the server receives the signature $s'=<\mu', A_i', B_i' >$ from user *u* and a previous signature $s''=<\mu'' , A_i'', B_i''>$ from the same user exists, the user can perform the following two steps to verify the signature:

(1) If $B_i'\neq B_i''$ proceed to Step 2 or reject the data to prevent a "replay attack."
(2) The verification is correct if $\hat{e}(P_0,\mu')=\hat{e}(P_0,A_i')\hat{e}(P_u,B_i')$.

When servers with ID_i in level *i* also create signatures, they follow similar procedures for authentication.

3.5 Theoretical Performance Analysis

The theoretical performance analysis was based on the assumption that a number of users in level *i* wish to access data on the server. To conduct clear comparisons, the notations used are explained below.

(1) C_{BM}: The time costs of bilinear map $\hat{e}$ computations.
(2) C_h: The time costs of the hash function.
(3) C_{xor}: The time costs of exclusive OR computation.
(4) $C_{parameter}$: The time costs of generating the required communicative parameters.

This study assumed that some user in some server at level *i*, uploads one file which satisfied *a* attribute rules (i.e., conjunctive clauses, CC). The computation costs of encryption and decryption are shown in Table 2 , and the required number of communication parameters is shown in Table 3.

Table 2 and Table 3 show that HABE uses iterative bilinear map operations because the generated keys are based on a hierarchical architecture. Thus, its computation costs are as high as the located levels of target data. For encryption, HABE also uses various conjunctive clauses at various levels, from root PKG to the current server, to generate permission parameters; therefore, it should generate 10 permission parameters when at level 10.

In the proposed scheme, because the keys are not generated based on hierarchical architecture, the decryption procedure requires minimal bilinear map operations and parameters.

Table 2. Comparison of computation costs of the encryption and decryption

Method	Computation	
	Encryption	Decryption
HABE	$1C_{BM} + 1C_h + 1C_{xor}+(a*i+1)C_{parameter}$	$(i+1)C_{BM}+ 1C_h + 1C_{xor}$
Proposed	$1C_{BM} + 1C_h + 1C_{xor}+ (a+2)C_{parameter}$	$3C_{BM} + 1C_h + 1C_{xor}$

Table 3. Comparison of parameters of the encryption

Method	Required number of parameters for encryption
HABE	$a * i + 1$
Proposed	$a + 2$

4 Experimental Results and Security Analysis

4.1 Experiment Environment

Under the proposed environment, the MIRACL [20] library and elliptic curve $y^2 = x^3 + 1$ were used to design the experiment. This study assumed that the number of attributes was 100 and the key size was 1024 bits. A number of emulations were adopted and the effluence of the levels of the proposed hierarchical cloud architecture was considered. Our hardware is based on AMD X4 640 processor, 4GB memory, 1TB(7200 rpm) hard disk and the software environment is based on C++ programming language and with Linux 2.6.32.

4.2 Experiment Analysis

Experiment 1. (Effects on the number of attributes): In this experiment, the effects on the number of attributes were considered. The experiment results are shown in Fig. 2.

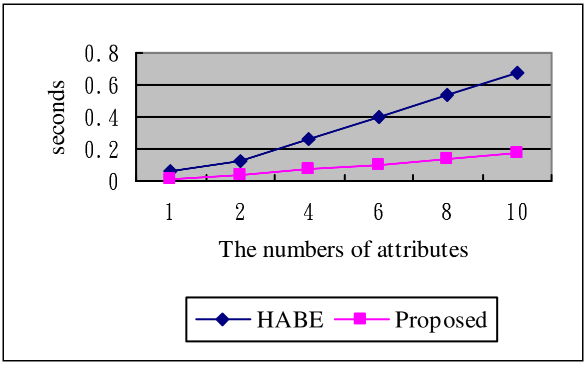


Fig. 2. Simulation result for the effects on the number of attributes

In Fig. 3, the encryption costs are significantly higher for a greater number of attributes. By contrast, the costs of the proposed scheme were lower than that of HABE, because unlike HABE, the proposed scheme does not involve iterative decryption and encryption operations, so its cost is lower than HABE

Experiment 2. (Effects on the numbers of levels): The encryption and decryption time costs for various levels were considered. The experiment results are shown in Fig. 3(a) and Fig. 3(b)

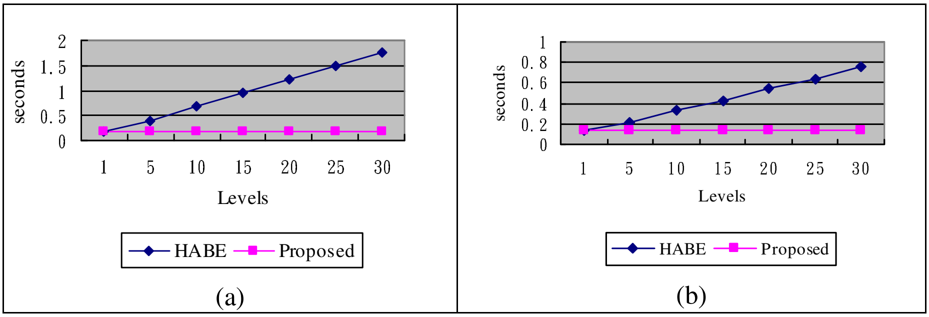


Fig. 3. (a) Simulation result for encryption at various levels (b) Simulation result for decryption at various levels

In Fig. 3(a) , as mentioned previously, HABE uses iterative key generation for encryption, resulting in higher time and communication costs; additionally, the computation costs increase with more levels. As shown in Fig. 3(b), the proposed scheme uses non-iterative key generation and considers less iterative bilinear operations; thus, the time costs do not increased with additional levels.

4.3 Security Analysis

This study considered the following general attack techniques to conduct security analysis:

(1) **Guessing Attack:** When one attacker attempts to crack the cloud server key using brute force. According to the CDH assumption[11], "given the points P, aP, $bP(a, b \in Z_q^*)$ at additive G_1 , compute abP is difficult." The key generation method of the proposed scheme was derived from $SK_{i,u} = r_i r_u P_0$ and $SK_{i,u,a} = SK_i + r_i r_u P_a$, which was also based on this assumption. Thus, the proposed method is secure.
(2) **Man-in-the-Middle Attack**: When one attacker intercepts the connection between the sender and receiver, they obtain the encrypted data and send fake data to the receiver. However, the scheme proposed in this study can prevent this attack using signature verification.
(3) **Replay Attack:** When one attacker uses the same signature and then sends data to the receiver repeatedly to busy it with verifications. However, this attack is nullified because the proposed scheme includes a timestamp in the signature and adopts the time verifier in Step 1 of signature verification.

5 Conclusions

This study proposed a low-cost cryptography system and attribute-based access control scheme for cloud storage environments. The simulation results and analysis showed that this method incurs lower communication and computing costs compared to HABE. The proposed scheme can achieve data access control through user attribute-based rules. The scheme also satisfies the authentication requirements by including identity-based signatures in the cloud storage environment.

However, this method was only implemented in a simulation environment because of resource limitations. In the future, we aim to implement this method in a true cloud storage environment for verification.

References

1. Foster, I., Zhao, Y., Raicu, I., Lu, S.: Cloud computing and grid computing 360-degree compared. In: Grid Computing Environments Workshop (GCE 2008), Austin, Texas, pp. 1–10 (2008)
2. Alliance, C.S.: Security guidance for critical areas of cloud computing version 3.0., https://cloudsecurityalliance.org/research/security-guidance/ (accessed July 20, 2012)
3. Li, H., Dai, Y., Tian, L., Yang, H.: Identity-based authentication for cloud computing. In: Proceedings of the 1st International Conference on Cloud Computing (CloudCom 2009), Beijing, China, pp. 157–166 (2009)
4. Yan, L., Rong, C., Zhao, G.: Strengthen cloud computing security with federal identity management using hierarchical identity-based cryptography. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) Cloud Computing. LNCS, vol. 5931, pp. 167–177. Springer, Heidelberg (2009)

5. Wang, G., Liu, Q., Wu, J.: Hierarchical attribute-based encryption for fine-grained access control in cloud storage services. In: Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS 2010), New York, NY, USA, pp. 735–737 (2010)
6. Grance, P.M.T.: The nist definition of cloud computing (15 ed.) National Institute of Standards and Technology (NIST), http://csrc.nist.gov/groups/SNS/cloud-computing (accessed July 20, 2012)
7. Grance, P.M.T.: The NIST Definition of Cloud Computing (Draft). National Institute of Standards and Technology (NIST), http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf (accessed July 20, 2012)
8. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
9. Ramgovind, S., Eloff, M., Smith, E.: The management of security in cloud computing. Information Security for South Africa (ISSA). University of Johannesburg, Johannesburg, South Africa, pp.1–7 (2010)
10. Miller, V.S.: Use of elliptic curves in cryptography. In: Williams, H.C. (ed.) CRYPTO 1985. LNCS, vol. 218, pp. 417–426. Springer, Heidelberg (1986)
11. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, p. 213. Springer, Heidelberg (2001)
12. Horwitz, J., Lynn, B.: Toward hierarchical identity-based encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (2002)
13. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (2002)
14. Huang, J.-Y., Liao, I.-E., Chiang, C.-K.: Efficient identity-based key management for configurable hierarchical cloud computing environment. In: IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS 2011), Tainan, Taiwan, pp. 883–887 (December 2011)
15. Tianyi, Z., Weidong, L., Jiaxing, S.: An efficient role based access control system for cloud computing. In: IEEE 11th International Conference on Computer and Information Technology (CIT), pp. 97–102 (2011)
16. Tsai, W.-T., Shao, Q.: Role-based access-control using reference ontology in clouds. In: 2011 10th International Symposium on Autonomous Decentralized Systems (ISADS), pp. 121–128 (2011)
17. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005)
18. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, New York, NY, USA, pp. 89–98 (2006)
19. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: Proceedings of the 2007 IEEE Symposium on Security and Privacy, SP 2007, Washington, DC, USA, pp. 321–334 (2007)
20. CertiVox.: MIRACL Crypto SDK, http://certivox.com/index.php/solutions/miracl-crypto-sdk/

Active One-Time Password Mechanism for User Authentication

Chun-I Fan[1], Chien-Nan Wu[1], Chi-Yao Weng[1], and Chung-Yu Lin[2]

[1] Department of Computer Science and Engineering
National Sun Yat-sen University, Kaohsiung, Taiwan, R.O.C.
[2] 2udg INC., Samoa
cifan@faculty.nsysu.edu.tw,
john-lin@2udg.com

Abstract. Cloud computing brings novel concepts and various applications for people to use computer on theInternet, where all of above-mentioned concern with user authentication. Password is the most popular approach for user authentication in daily life due to its convenienceand simplicity. However, on Internet, user's password is easier to suffer from distinct threats and vulnerability. First, for the purpose of easily memorizing, user often selects a weak password and reuses it between different service providers on websites. Without a doubt, an adversary will obtain access to more websites if the password is compromised. Next, an adversary can launch several methods to snatch users' passwords such as phishing, keyloggers, and malware, and those are hard to be guarded against. In this manuscript, we propose an active one-time password (AOTP) mechanism for user authentication to overcome two abovementioned problems, password stealing and reuse, utilizing cellphone and short message service. Through AOTP, there is no need for additional tokens, card readers and drivers, or unfamiliar security procedures and user can choose any desirous password to register on all websites. Furthermore, we also give some comparison tables to present that the proposed mechanism is better than other similar works.

Keywords: User authentication, One-time password, Password stealing attack, Passwordreuse attack, Cloud computing.

1 Introduction

Cloud computing has been developed rapidly in recent years; it offers novel concepts and innovations for presented computer environment. That is, cloud computing provides various services to people and all of those services are related to user authentication. In user authentication procedures, password is an essential factor and over the past few decades, it has been used widely by people in life, such as debit cards, cell phones and computers login codes, websites logins, and so on. Generally, password-based user authentication can withstand dictionary and brute force attacks as long as users choose strong passwords to provide plenty entropy. Nevertheless, one problem

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 464–471, 2013.

of previous solution is that it is hard for persons to memorize a long and unmeaning string, and thus most of them would select easy-to-remember passwords even though they understand those are unsafe or weak. Furthermore, investigations [6,7,10] present that, due to the customary behavior and habit, there is a high value on average of that humans register the same passwords across different service providers, called as password-reuse. Password-reuse causes that a hacker can pass through different authentication mechanisms using the same password and then obtains sensitive information stored in database of these service providers, where the personal sensitive information might be used in frauds or other illegal sides by adversaries. From above, even if the problems are caused by human factors, password is still an important method for user authentication in our life. Therefore, how to reduce the negative influence of human factors is an important issue when designing a user authentication protocol. Up to now, a number of literatures focus on reducing the negative influence of human factors in the user authentication procedure such as graphical password schemes [4, 11, 14, 18-20] and password management tools [8, 15, 21].

Except for the password-reuse problem, password stealing is another important issue. There are many password-based applications in real life, especially card-based applications, like debut card, cell phone pin code, and so on. When the passwords be revealed or stolen, an adversary can perform unauthorized payments or take personal information into frauds or illegal areas by collecting sensitive information and financial secrets [5, 9, 12, 16].

To reduce the happening possibility of above problem and provide more reliable user authentication procedure, some literatures replace password-based authentication by three-factor authentication, where three-factor authentication falls back on something you know (e.g., password), something you have (e.g., card), and something you are (e.g., biometric). Although three-factor authentication is a good mechanism against password stealing attacks, two-factor authentication is more suitable than that of being practiced since three-factor authentication needs a relative high cost [13].

1.1 Our Contribution and Paper Organization

Based on one-time password approach, we propose a user authentication mechanism, named active one-time password (AOTP), by means of cellphone and short message service (SMS) [1, 2] to solve password reuse problems and prevent password stealing attacks. For service providers, it is unopposed to integrate AOTP into original user authentication system and the combined one supplies better security of user authentication. For users, they only need a carry-on device, cellphones, helping them to execute the identity confirmation without other additional tokens, card readers, and drivers. Furthermore, they do not operate unfamiliar security procedures when logining on websites.

The rest of this manuscript is organized as follows. Section 2 reviews preliminaries including one-time password mechanism and SMS channel. Section 3 introduces the proposed mechanism, and then we discuss the advantage and security of the proposed scheme in Section 4. The final section offers concluding remarks.

2 Preliminaries

In this section, we first introduce the one-time password mechanism and SMS channel. Then, we describe a cyber system integrated one-time password and its main leakage.

2.1 One-Time Password

A one-time password (OTP) [3] is a password that is valid for only one login session or transaction, and it addresses numerous drawbacks of traditional passwords (static passwords), where the important improvement is password reuse problem (replay attack). That is, even if a potential intruder records a session OTP that a user takes it to log into a server, she/he still disables to use it to log into the server successfully since the one-time password is invalid (overdue). Therefore, in order to prevent dictionary attacks and enhance the strength of passwords, a one-time password is a random number and is difficult for users to memorize. Nowadays, a number of systems, e.g., on-line games, integrate OTPs into their login mechanisms to achieve user authentication, where Figure 1 presents an overview of authentication procedure adopting OTP. However, the disadvantage of OTP is that it requires an additional device such as dynamic keypad lock or RSA secure-id token to work.

2.2 A Cyber System Integrated One-Time Password

Before asking service for a cyber system, a user must register username/password and some personal information to the service provider in advance, where her/his phone number is a necessary item in this stage. After that, the user can login the website to access services. Figure 1 depicts a user authentication procedure in a cyber system integrated OTP and the details are described as follows.

- Step 1: When the user demands services for a service provider, she/he performs login procedure by typing individual username and password.
- Step 2: After receiving the login information from the user, the service provider submits user's phone number to an OTP generation center if the username/password checking is correct.
- Step 3: According to the receiving phone number from the service provider, the OTP generation center generates a one-time password, and sends it to the service provider and the user, respectively, where the OTP generation center texts a short message through SMS channel to the user's mobile phone.
- Step 4: After obtaining the one-time password, the user inputs it on the login-webpage with her/his username and password.
- Step 5: Finally, the service provider outputs an outcome to the user by comparing with the one-time password.

Notably, step 4 and 5 must be done in limited time.

Since the user inputs registered username, password, and one-time password on webpage through Internet, a malicious user can obtain them easily by using phishing or eavesdropping without the registered user’s mobile phone. That is, the malicious user first forges a faked login-webpage for the registered user to input the pair of username and corresponding password in Step 1, and then she/he can impersonate the latter to ask the service for the cyber system by using the pair of username and password. In Step 3, after the registered user inputs the one-time password on the forged login-webpage, the malicious user can transmits it to the real login-webpage and finally, she/he gets the service from the cyber system.

3 Active One-Time Password

In this section, we first define some assumptions of AOTP and illustrate architecture of its login process. Next, we give flowcharts to present the detail of one-time password generation and verification, respectively.

3.1 Assumptions

The assumptions of user authentication system with AOTP are described below:

- Each user has one mobile phone with unique phone number at least and she/he has responsibility to take care of it, that is, all users must register their identities to the telecommunication service provider (TSP), where the TSP is a trusted party.
- All cyber service providers (CSP) on Internet must integrate AOTP mechanism into their individual system, and between the CSP and the OTP generator, agent server (AS) is a connecter which helps CSP to communicate with the OTP generator.
- If a user loses her/his cellphone, she/he must register the lost one at the TSP and re-apply a new SIM card, where the TSP will update related information and disable the lost SIM card.

3.2 Authentication Procedure

The scenario is the same as in section 2.3. A user registers personal information to a cyber service provider (CSP) and then she/he can login the website to access services where the phone number is also a necessary item in registration stage. Figure 2 is an overview of AOTP's login procedure and the details are described below.

- Step 1: When the registered user demands services for the CSP, she/he performs login procedure by typing individual username and password.
- Step 2: After receiving the login information from the user, the CSP submits the user's phone number to the AS if the username/password checking is correct, and then the AS passes on the phone number to the OTP generator. According to the receiving phone number from AS, the OTP generator produces a one-time password and sends it to the CSP, where the OTP generator must store the pair of phone number and corresponding one-time password in database.

- Step 3: The CSP transmits the login information and the one-time password to the user.
- Step 4: When obtaining the password, the user texts it and sends the short message to a designated telecom number through SMS channel by means of the mobile phone, where the phone number is a registered one in the registration phase.
- Step 5: After obtaining the pair of phone number and OTP from the SMS platform, the AS sends the pair to the OTP generator to compare with the one which is stored in OTP generator's database. Step 6: Finally, the CSP outputs an outcome to the user according to the result from the OTP generator, where the OTP generator will keep a record for the purpose of detecting abnormal cyber users if the comparison result is false.

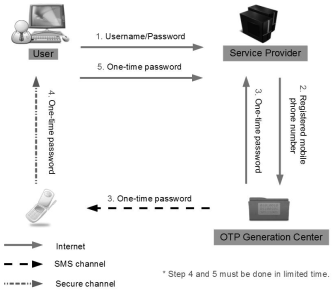


Fig. 1. User authentication procedures of a service provider system integrated OTP

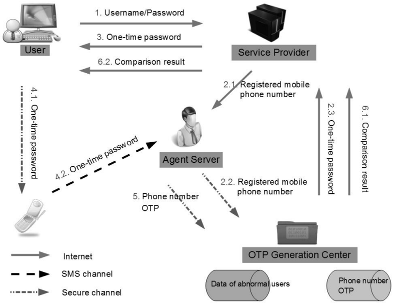


Fig. 2. User authentication procedures of a CSP system integrated AOTP

4 Discussions

A service provide on Internet integrated AOTP mechanism presents the following advantages:

1. Easy remembrance and reuse prevention of password: In login (user authentication) phase, except for the pair of registered username and password, the user also needs to text a one-time short code (password), which is from the service provider, to a designated SMS platform by using her/his mobile phone, where the phone number is registered in the service provider. Accordingly, the main security of authentication procedure is based on the confirmation of user's phone number from the telecommunication service provider. Therefore, the user can choose an easily remembering password as a register one in the service provider and even though it is stolen, an adversary still cannot login successfully since she/he has no cellphone with corresponding phone number and SIM (Subscriber Identity Module) card to text the one-time password to the agent serve. Furthermore, one-time password approach used in different logins can prevent password reuse attacks.

2. Anti-malware: Malware, e.g., keylogger, can gather sensitive information from user such as username and password. In any service provider integrated AOTP, a user logins it by entering personal username/password pair on computer and texting one-time short code via her/his cellphone, respectively. As above mention, as long as user's cellphone with SIM card are not lost, the proposed AOTP mechanism is secure and thus, it is useless for malware to gather users' username/password pairs.
3. Phishing protection: An adversary usually fakes a phishing website to steal users' usernames/passwords when they connect to the forged one. As abovementioned reason, AOTP guarantees to withstand phishing attacks.
4. Convenient device: According to [17], additional tokens, card readers and drivers, or unfamiliar security procedures are needless for the main requirements of an ideal user authentication mechanism. Hence, the above appears that cellphone is the best choice for above purpose since it is widely distributed and has high user acceptance. Furthermore, it is a carry-on device for people that they can routinely carry with one at least in daily life.

To stand out the advantages of AOTP, we present two comparison tables between AOTP and OTP, displaying in Table 1 and Table 2, where SMS OTP and telephone lock are the applications of OTP and they are widely used in login procedures of online game.

Table 1. Comparison between AOTP and SMS OTP

	AOTP	SMS OTP
1. Spam messages with malicious intention	No^{a}	Yes^{b}
2. Security problem from logging cellphone	No^{c}	Yes^{d}
3. Attack during user authentication	No^{e}	Yes^{f}
4. Non-repudiation	High^{g}	Low^{h}
5. Priority and accuracyof message transmission	High (SMS-MO)^{i}	Low (SMS-MT)^{j}
6. Value-added services	Yes^{k}	No

a: OTP is displaying on webpage

b: Each OTP must be sent to customer for all verifications

c: Using specific mobile phone and OTP to achieve user authentication

d: Only inputting OTP for user authentication

e: Only specific mobile phone can be entered user authentication after account logging and password verifying

f: Each computer can login since the account has been authenticated

g: The specific mobile phone with particular user is used to prevent counterfeit user authentication

h: Attacker can steal legal OTP to succeed user authentication

i: Mobile terminating short messages

j: Mobile originating short messages

k: Remote authorization and signature, reservation system etc.

Table 2. Comparison of authentication mechanism between AOTP and other similar works

	AOTP	Token OTP	SMS OTP	Telephone lock	Visa 3D
1.	Internet SMS	Internet Token	Internet SMS	Internet Telephone network	Internet
2.	Highest	High	Intermediate	Intermediate-low	Low
3.	Low	Low	Intermediate[a]	High[b]	High[c]
4.	High[d]	Intermediate[e]	Intermediate[e]	Low[f]	Low[g]

1: Authentication channel
2: Security level
3: Risk for recorded data or fake data
4: Non-repudiation
a: Easy to load malware into mobile phone
b: Phone number is easy to be tampered with illegal
c: Easy to load Torjan virus
d: OTP is sent by specific mobile phone
e: OTP is verified on Internet
f: Phone number for authentication can be tamped
g: Each computer could run authentication process

5 Conclusion

Password is widely applied to user authentication systems due to its convenience and simplicity, but it has higher probability to suffered from distinct threats and vulnerability on Internet because of users' misbehavior or insecure computer environment. In this manuscript, we proposed active one-time password (AOTP) mechanism for user authentication which leverages cellphones and SMS to overcome password stealing attacks and password reuse problems. The proposed AOTP can be integrated into any original user authentication system on websites without contradictory to enhance total security. For users, the login procedure can be guaranteed as well by utilizing a carry-on device, cellphone, without other extra devices. Finally, we provided comparison tables to present that the proposed AOTP is better than other similar ones.

References

1. TS 23.040: Technical Realization Short Message Service (SMS) 3GPP (Online), `http://www.3gpp.org/`
2. I. T. Report, ITU Internet Rep. 2006: Digital.Life (Online) (2006), `http://www.itu.int/`
3. One-time password, Wikipedia 2011 (2011), `http://en.wikipedia.org/wiki/One-timepassword`
4. Chiasson, S., Forget, A., Stobert, E., van Oorschot, P.C., Biddle, R.: Multiple password interference in text passwords and click-based graphical passwords3. In: Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS 2009, pp. 500–511. ACM, New York (2009)
5. Dhamija, R., Tygar, J.D., Hearst, M.: Why phishing works. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2006, pp. 581–590. ACM, New York (2006)

6. Florencio, D., Herley, C.: A large-scale study of web password habits. In: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, pp. 657–666. ACM, New York (2007)
7. Gaw, S., Felten, E.W.: Password management strategies for online accounts. In: Proceedings of the 2nd Symposium on Usable Privacy and Security, SOUP 2006, pp. 44–55. ACM, New York (2006)
8. Halderman, J.A., Waters, B., Felten, E.W.: A convenient method for securely managing passwords. In: Proceedings of the 14th International Conference on World Wide Web, WWW 2005, pp. 471–479. ACM, New York (2005)
9. Holz, T., Engelberth, M., Freiling, F.: Learning more about the underground economy: A case-study of keyloggers and dropzones. In: Backes, M., Ning, P. (eds.) ESORICS 2009. LNCS, vol. 5789, pp. 1–18. Springer, Heidelberg (2009)
10. Ives, B., Walsh, K.R., Schneider, H.: The domino effect of password reuse. Commun. ACM 47(4), 75–78 (2004)
11. Jermyn, I., Mayer, A., Monrose, F., Reiter, M.K., Rubin, A.D.: The design and analysis of graphical passwords. In: Proceedings of the 8th Conference on USENIX Security Symposium, SSYM 1999, vol. 8, p. 1. USENIX Association, Berkeley (1999)
12. Karlof, C., Shankar, U., Tygar, J.D., Wagner, D.: Dynamic pharming attacks and locked same-origin policies for web browsers. In: Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS 2007, pp. 58–71. ACM, New York (2007)
13. O'Gorman, L.: Comparing passwords, tokens, and biometrics for user authentication. Proceedings of the IEEE 91(12), 2021–2040 (2003)
14. Perrig, A., Song, D.: Hash visualization: a new technique to improve real-world security. In: International Workshop on Cryptographic Techniques and E-Commerce, pp. 131–138 (1999)
15. Pinkas, B., Sander, T.: Securing passwords against dictionary attacks. In: Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, pp. 161–170. ACM, New York (2002)
16. Provos, N., Mcnamee, D., Mavrommatis, P., Wang, K., Modadugu, N.: The ghost in the browser: Analysis of web-based malware. In: Proceedings of the 1st Conference Workshop on Hot Topics in Understanding Botnets, HotBot 2007, p. 4. USENIX Association, Berkeley (2007)
17. Sax, U., Kohane, I.S., Mandl, K.D.: Wireless technology infrastructures for authentication of patients: PKI that rings. Journal of the American Medical Informatics Association 12(3), 263–268 (2005)
18. Thorpe, J., van Oorschot, P.: Towards secure design choices for implementing graphical passwords. In: 20th Annual Computer Security Applications Conference, pp. 50–60 (2004)
19. Wiedenbeck, S., Waters, J., Birget, J.C., Brodskiy, A., Memon, N.: Passpoints: Design and longitudinal evaluation of a graphical password system. Int. J. Hum.-Comput. Stud. 63(1-2), 102–127 (2005)
20. Wiedenbeck, S., Waters, J., Sobrado, L., Birget, J.C.: Design and evaluation of a shoulder-surfing resistant graphical password scheme. In: Proceedings of the Working Conference on Advanced Visual Interfaces, AVI 2006, pp. 177–184. ACM, New York (2006)
21. Yee, K.P., Sitaker, K.: Passpet: Convenient password management and phishing protection. In: Proceedings of the 2nd Symposium on Usable Privacy Security, SOUPS 2006, pp. 32–43. ACM, New York (2006)

Hardware Acceleration for Cryptography Algorithms by Hotspot Detection

Jed Kao-Tung Chang[1], Chen Liu[1], and Jean-Luc Gaudiot[2]

[1] Department of Electrical and Computer Engineering, Clarkson University, Potsdam, NY
jchang@clarkson.edu,
cliu@clarkson.edu
[2] Department of Electrical Engineering and Computer Science, University of California, Irvine,
Irvine, CA
gaudiot@uci.edu

Abstract. Data Encryption/Decryption has become an essential part of pervasive computing systems. However, executing these cryptographic algorithms often introduces a high overhead. In this paper, we select nine widely used cryptographic algorithms to improve their performance by providing hardware-assisted solutions. For each algorithm, we identify the software performance bottleneck, *i.e.,* those "hotspot functions" or "hot-blocks" which consume a substantial portion of the overall execution time. Then, based on the percentage of execution time of a specific function and its relationship with the overall algorithm, we select candidates for our hardware acceleration. We design our hardware accelerators of the chosen candidates. The results show that our implementations achieve speedups as high as 60 folds for specific functions and 5.4 for the overall algorithm compared with the performance of the software-only implementation. Through the associated hardware cost analysis, we point to an opportunity to perform these functions in an SIMD fashion.

Keywords: cryptography, hardware acceleration, performance analysis, hotspot function.

1 Introduction

Data security is important in pervasive computing systems because the secrecy and integrity of the data should be retained when they are transferred among mobile devices and servers in this system. The cryptography algorithm is an essential part of the pervasive computing security mechanism. By performing encryption, decryption, and hash operations on transmitted data, intruders should not be able to identify the hacked cipher text without decrypting schemes, and even a minute modification of the data shall be detected with a good data integrity verification process.

However, performance is one concern regarding cryptography algorithms: these algorithms are extremely expensive in terms of execution time. To make cracking the code more difficult, many arithmetic and logical operations are bound to be executed during the encryption/decryption process. Furthermore, a huge amount of data needs

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 472–481, 2013.

to be transferred between the CPU and the memory. Using a general-purpose processor for this purpose would not be a cost-effective solution, and the performance is not that satisfying either. This paper attempts to address this issue. We first deployed *execution-based profiling* to identify the hotspot (performance-intensive) part of an application: in each benchmark, we select candidates for hardware acceleration based on certain aspects, such as the percentage of total execution time belonging to hotspot functions, the relationship between the hotspot points and the entire application, *etc*. We then implemented these hotspot points in hardware. We compared the hardware and software implementation from two perspectives: performance and hardware cost. Our ultimate goal is to provide hardware-assisted solutions along these lines, so as to improve the performance of the crypto-computations in pervasive computing systems.

The rest of the paper is organized as follows: We review related work in Section 2. We specify the cryptography algorithms in Section 3. We introduce the performance analyzer tool VTune, and describe how we employ it to identify the hotspot functions and hot-blocks across the set of benchmarks in Section 4. In Section 5, we describe the criteria for selecting the candidate algorithms for hardware acceleration. The implementation of the hardware accelerator for the chosen benchmarks is described in Section 6. In Section 7, we describe the effort to investigate the hardware cost of our hardware implementation. Finally, conclusions are drawn in Section 8.

2 Related Work

Many previous research efforts have improved different parts of the pervasive computing system. For example, Tang *et al.* [21-27] employed hardware–assisted approach to accelerate selected middleware execution, and used prefetching techniques in mobile systems. Different from them, ours is focused on addressing the issue of improving performance of cryptographic computations. With the multithreading features which have been recently added to many programming languages, the straightforward software solution is to increase the level of concurrency. For example, Bielecki *et al.* [6] has attempted to deal with the performance issue and focused on parallel programming approaches. If purely focusing on a software strategy, however, the performance improvement would be limited since a general-purpose processor is not as efficient as a dedicated cryptography coprocessor.

Many people have strived to implement cryptography operations on a single chip. Hodjat *et al.* [17] has used a dedicated cryptographic coprocessor to alleviate the load on the CPU. Bertoni *et al.* [20] had implementations at a finer granularity via instruction set extension: they implemented different stages of AES algorithm into customized instructions. The concept of applying SIMD architecture, such as graphic processing unit (GPU), in asymmetric and some modes of symmetric encryption algorithms was also employed to improve the performance of the cryptography algorithms. Harrison *et al.* [18] implemented the AES Encryption ECB mode on GPUs.

Compared with previous work targeting a specific computation-intensive part of an algorithm for hardware acceleration, we work on a set of algorithms which have certain program structures and crypto-computation operations in common. We employed *dynamic-based profiling* (also described in Chang *et al.* [28]): to profile the performance behavior when running the benchmark and identify the parts that take up the most of execution time as hotspot points. When implementing these hotspot points into hardware accelerators, not only did they have superior performance than running on general-purpose processors, but also in some cases, the hardware costs are extremely low. We can duplicate the hardware accelerators for the hotspot points so that multiple data elements can be processed on these accelerators in a parallel fashion.

3 Cryptography Algorithms

We choose nine popular cryptography algorithms as benchmarks for our study: AES [3], RSA [4], 3DES [8], RC5 [9], MD5 [10], IDEA [11], SHA1 [12], Blowfish [13], and ECC [14]. The reason for our choice is that we feel the program structures of these algorithms are quite representative of the contemporary cryptography algorithms. For example, some algorithms like 3DES, Blowfish, and RC5 use the Feistel cipher proposed by Luby *et al.* [1]; other algorithms, such as AES, IDEA, MD5, and SHA1 use the iterative cipher designed by Rijmen *et al.* [2]. ECC and RSA are public key (asymmetric) algorithms [16], which are neither Feistel cipher nor iterative cipher. Comparing with the other seven algorithms, ECC and RSA seem to be more complex. However, in these algorithms, each data can be encrypted or decrypted independently and the operations can be performed in parallel to improve their overall performance. Another common feature shared by these algorithms is the operation they use to encrypt / decrypt the data. Memory access operations are used by 3DES, AES, and Blowfish. This is because these three algorithms need to access lookup tables. Modular arithmetic operations are also heavy in modern cryptography algorithms, because they need to keep the plaintext as secure as possible.

4 Hotspot Function Identification

We profile the dynamic computation characteristics of the benchmarks and utilize VTune [5] from INTEL® as our performance analyzer to identify the hotspots. VTune analyzes the software performance on IA-32 and Intel64-based machines. It collects performance data on applications running on the host system, organizes and displays the data in an interactive way. VTune's call graph view provides a tree structure to show the call relationship among all functions along with their execution time. This would help us identify the "hotspot" functions and percentage of the hotspot functions occupying the total execution time of each benchmark.

We define *HF-Rate* as the percentage of the execution time belonging to hotspot functions of each algorithm. Figure 1 shows the *HF-Rate* for each algorithm. It should be noted that for the execution time, we only consider the crypto computation (key setup, encryption, and decryption) part of an application, excluding the file I/O or certain system calls within the dynamic-link library (DLL).

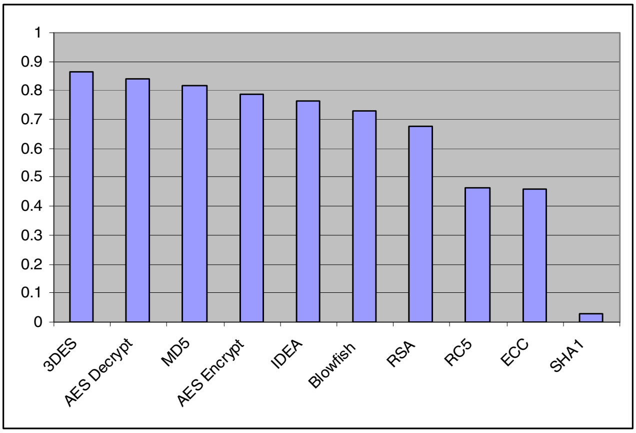


Fig. 1. HF-Rate for different algorithms

We can see that the execution time of the hotspot functions account for a majority of the total execution time for most of the benchmarks. In 3DES, MD5, and AES Decryption, the hotspot functions occupy more than 80% of the execution time of the entire algorithm. For AES Encryption, Blowfish, IDEA, MD5, and RSA, this number reaches or even exceeds 70%. However, SHA1 is an exception because most of its execution time is spent on I/O operations called by the crypto computation functions, such as reading from a file, which means that there is really no hotspot function to isolate. Overall, the function breakdown helps us identify the software bottleneck of the application.

5 Acceleration Candidate

When selecting our candidates for hardware acceleration, we need to consider two aspects of the hotspot function(s):

- Its *HF-Rate*
- The relationship between the hotspot function(s) and the overall algorithm.

The first aspect is evident: based on the Amdahl's Law, we would prefer to choose a hotspot function with a high *HF-Rate*, which is the percentage of total execution time dedicated to the hotspot function and is our target for acceleration. In the formula of Amdahl's law, *HF-Rate* corresponds to $\text{Fraction}_{\text{enhanced}}$, the percentage of the overall execution during which the performance enhancement was applied. If the *HF-Rate* is too low, the performance of the application cannot be significantly enhanced even if the performance of the hotspot point can be much improved.

$$Speedup_{overall} = \frac{1}{(1 - Fraction_{enhanced}) + \dfrac{Fraction_{enhanced}}{Speedup_{enhanced}}} \tag{1}$$

Given the first criterion, as Fig. 1 indicates, SHA1 would not be taken into consideration due to its low *HF-Rate* of the hotspot function. The second aspect is equally important: if a hotspot function is the entire process of the overall algorithm, such as the encryption/decryption part, the hardware cost would be too high, because we would need to implement many hardware instructions and correspondingly many hardware components which would occupy too large a die area. Thus, a good candidate for hardware acceleration is a hotspot function with both a high *HF-Rate* and small size.

Next let us consider the hardware implementation for specific hotspot functions. The hotspot function of RSA is a function called *Power()*, which calculates two to the power of N, accounting for 67.6% of the RSA execution time. From the profiling we know that the maximum value of *N* is 31. It takes just one 32-bit barrel shifter to finish the operation.

The encryption/decryption of AES has four stages. The stages of encryption are *SubBytes()*, *ShiftRows()*, *MixColumns()*, and *RoundKey()*, while those of decryption are *InvSubBytes()*, *InvShiftRows()*, *InvMixColumns()*, and *RoundKey()*. The hotspot function of AES encryption is *SubBytes()*, the first stage of each round. In AES decryption, the three hotspot functions are *InvSubBytes()*, *InvMixColumns()*, and *SubBytes()*, corresponding to the first, third and fourth stages of AES decryption, respectively. Among them, *SubBytes()* and *InvSubBytes()* are memory access operations, replacing a byte with another one according to a prebuilt lookup table (SBox). In recent hardware implementations, the SBox is put into the cache so that the access time could be significantly reduced, according to Mourad *et al* [19]. For *InvMixColumns()*, its instructions can be implemented by a set of AND, shift, and table lookup operations according to [3]. *SubBytes()* in AES encryption contributes 78.5% of the total execution time, while the *InvSubBytes()*, *SubBytes()*, and *InvMixColumns()* in AES decryption contribute 55.8%, 17.5%, and 10.8% of the total execution time.

For Blowfish, the hotspot function is *F()* which contributes 73.13% of the total execution time. Blowfish has 16 rounds of data transformation and *F()* is executed once in each round. *F()* takes the 32-bit input and splits it into four eight-bit inputs and access four pre-built lookup tables (SBoxes) in parallel. The outputs are then XORed and added. Again, the four lookup tables can be placed in the cache and two adders, shifters, and one Exclusive-OR gate would be sufficient to implement the rest of the function. Thus, AES, RSA, and Blowfish would be good choices for us to perform the hardware acceleration.

As for 3DES, IDEA, MD5, and RC5, our analysis points to the hotspot function being the entire function of crypto computation. If we exclude this hotspot function, the rest of the algorithm is simply the initialization, the execution time of which is comparatively negligible. As mentioned previously, these algorithms are not good candidates for hotspot function acceleration since the hardware cost would be too high.

In this situation, we need to consider reducing the granularity for acceleration. We need to look inside each function and determine whether there is a “hot-line” or “hot-block” of code which contributes a significant amount of execution time and with low hardware overhead to qualify for acceleration. We employed the sampling wizard of VTune to show the number of consumed clock cycles by each line of code. We go deeper into the functions and view the performance at a finer granularity.

For RC5, the hotspot function is the *rc5_key()* for the key expansion work. The hot-block is named *KEYXP_RC5*. *KEYXP_RC5* accounts 37.9% of the whole algorithm. For 3DES, the hotspot function is *des_crypt()*. The hot-block is named *ROUND_3DES*, which handles memory access and address translation. This hot-block *ROUND_3DES* accounts for 58.7% of the whole algorithm. The hot-block of MD5 is *P_MD5*, which consists of four rounds where each round is composed of sixteen function-based stages. The main operations for *P_MD5* are Modular additions and left rotations. *P_MD5* contributes 73.3% of the total MD5 algorithm. *MUL_IDEA* is the hot-block of IDEA. The input data is processed with the keys by the modular arithmetic and logic operations. The main operations for *MUL_IDEA* are modulo addition and multiplication operations. *MUL_IDEA* consumes 61.2% of the execution time of the entire algorithm.

6 Accelerator Implementation

Now we are ready to implement the hotspot function(s) of selected benchmarks (RSA, AES, and Blowfish) in hardware. We take the state transitions scheme for hardware implementation. First, we translate the function(s) into finite state machine. Note that although the instructions are executed in sequential order, if we find there is no data dependency among the instructions, we can execute them in parallel and put them into the same state in the finite state machine. For RSA, one 32-bit barrel shifter can shift left a number by 0 to 31 bits and we just need one state. For *InvSubBytes()* of AES Decryption and *SubBytes()* of AES Decryption and Encryption, it is only a replacement operation for a byte according to a pre-built lookup table. Some logic components would be sufficient as it is one state of memory access.

The implementation of *InvMixColumns()* using Rijindael mix columns requires six states. In the first state, the input array is read into the tentative storage used for the later states. The inverse mix column operates on the data by multiply numbers in Rijndael Galois Field, which takes four states because four dependent instructions need to be executed. In the final state, the results will be stored back to the array. Thus, a total of six states are required to implement this function.

The *F()* of Blowfish needs three states. In the first state, a 32-bit input is split into four 8-bit inputs as an index to access the lookup table. In the second state, the four lookup tables are accessed in parallel and they produce four outputs. In the third state, four outputs are combined into one using the modulo addition and XOR operations.

Next, we discuss the hardware implementation of the hot-blocks of RC5, 3DES, MD5, and IDEA. The hot-block *KEYXP_RC5* in RC5 is inside a for-loop structure. We need three states to implement the operations of mixing secret keys to the key table.

The 3DES's hot-block *ROUND_3DES* has three states. In the first state the input will be operated on using different calculations in parallel and two outputs will be generated. In the second state, the outputs are used as indices to access one of the pre-built lookup tables. Then, in the third state, the values from the second state will be used to access eight other pre-built lookup tables and to perform the XOR operation on these outputs to obtain the result.

The hot-block of MD5 is *P_MD5*. As we know, MD5 is composed of 64 function-based stages. The first, second, third, and final 16 stages are grouped together as one round with a slightly different translation functions. We thus just need to implement four stages as four hardware modules separately with different functions. During the first round, the first hardware module will be called; the second module is called in the second round, and so on. Each module just needs two states. The first state is to execute the function with the given input. The second state is to perform some simple operations based on the output of the first state.

Table 1. Hardware acceleration speedup

Hotspot Function / Benchmark	**FSM Stages**	**Function Speedup**	**Algorithm Speedup**	
Power() / RSA	1	26	2.9	
SubBytes() / AES Encryption	1	56	4.4	
InvSubBytes() / AES Decryption	1	48	1.2	5.4
SubBytes()/ AES Decryption	1	60	2.2	
InvMixColumns() / AES Decryption	6	9	1.1	
F() / Blowfish	3	2	1.7	
KEYXP_RC5 / RC5	3	4	1.4	
ROUND_3DES / 3DES	3	2	1.5	
P_MD5 / MD5	8	9	2.9	
MUL_IDEA / IDEA	4	5	2.0	

The hot-block *MUL_IDEA* of IDEA takes four states to complete. The first state is for initialization and the following three states are for set of shift, add, and multiplication operations on the input data and key.

We implemented the hardware-assisted cryptographic functions as the accelerator connected with Microblaze using the PLB bus on the Xilinx XUPV5-LX110T development system. We measured and compared the performance in terms of the number of clock cycles and of the hardware and software implementation.

Table 1 summarizes the number of stages of finite state machine needed by each hotspot function, the hardware acceleration speedups we achieved over specific functions and over the entire algorithm. The results show that for hotspot function acceleration we can achieve 2 to 60 folds of performance improvement; for hot-block acceleration, we can achieve 2 to 9 folds of performance improvement; as for the entire cryptographic algorithm, hardware acceleration can achieve performance improvements of 1.4 to 5.4 folds, depending on the program structure.

7 Hardware Cost Analysis

Given the superior performance of the hardware accelerator implementation of hotspot functions and hot-blocks of the candidate benchmarks, we now look into implementing them in an ASIC design, hardware cost being our top-most concern.

We measured the hardware resource utilization based on the number of hardware slices (*#slices*), the number of flip-flops (*#FF*), and the number of look-up tables (*#LUT*). Table 2 summarizes the hardware resource utilization of these implementations.

Table 2. Hardware Resource Utilization

Hotspot Function / Benchmark	#Slices	#FF	#LUT
InvSubBytes() / AES Decryption	5	17	14
InvMixColumns() / AES Decryption	524	226	174
SubBytes()/ AES Decryption	5	17	14
SubBytes() / AES Encryption	5	17	14
Power() / RSA	5	17	14
F() / Blowfish	26	33	96
MUL_IDEA / IDEA	58	73	102
KEYXP_RC5 / RC5	137	175	423
ROUND_3DES / 3DES	6	19	19
P_MD5 / MD5	37	74	107

Based on the criteria we listed in Section 5 in the selection of the functions for hardware acceleration, the hardware overhead incurred with the accelerator is simply minimal, as shown in Table 2. For comparison, let us consider a very simple in-order MIPS processor design [7], the resource utilization of which requires 10,450 hardware slices, 10,400 flip-flops, and 19,500 look-up tables. Thus, if we implement the hotspot functions or hot-blocks in the form of a hardware accelerator, we need much fewer hardware resources compared to modern general-purpose processors as platforms on which we normally run cryptographic applications.

From the hardware cost, we observe that the hardware accelerator implementation is not only superior in performance but also matches the low cost of modern chip design. Given the several orders of magnitude difference between the hardware and software implementations, it is worthwhile for us to consider implementing the hotspot functions in the form of special functional units. We can achieve huge data parallelism by duplicating these function units. That is, we can extend our work to build a SIMD machine by grouping multiple data elements together and performing the operations using special processing units all at once.

8 Conclusions

In this study we have designed, implemented, and evaluated hardware acceleration approaches for cryptographic algorithms. We used the VTune performance analyzer to extract the hotspot functions and hot-blocks as identified by their high *HF-Rate* and low hardware overhead. We then translated the hotspot functions and hot-blocks into finite state machines and implemented them as hardware accelerators. Compared to previous research, we used a "hardware-assisted" method, i.e., we move the computation intensive part of the cryptography application into hardware so that the general-purpose computing resource can be released to perform other useful tasks. Our results indicate that for hotspot function acceleration we can achieve 2 to 60 folds of performance improvement; for hot-block acceleration, we can achieve 2 to 9 folds of performance improvement; for overall cryptographic algorithm execution, we can achieve 1.4 to 5.4 folds of speedups, compared with the traditional general-purpose processor. Through the hardware overhead investigation, we observe that the minimal hardware overhead is incurred during the accelerator implementation. This provides us an inspiration that we may build a SIMD machine to perform the hardware acceleration simultaneously, taking advantage of the data parallelism, which will be the focus of our future work.

Acknowledgements. This work is partly supported by the National Science Foundation under grants number CCF-1065448 and ECCS-1301953. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References:

1. Luby, M., Rackoff, C.: How to Construct Pseudorandom Permutations from Pseudorandom Functions. SIAM Journal on Computing 17(2), 373–386 1988)
2. Rijmen, V.: Cryptanalysis and design of iterated block ciphers. Doctoral Dissertation, K.U.Leuven (October 1997)
3. NIST (National Institute of Standards and Technology), "Advanced Encryption Standard (AES) – FIPS Pub. 197 (November 2001)
4. Rivest, R.L., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of ACM 21(2), 120–126 (1978)
5. Intel VTune, http://software.intel.com/en-us/intel-vtune/
6. Bielecki, W., Burak, D.: Parallelization Method of Encryption Algorithms. In: Advances in Information Processing and Protection, pp. 191–204 (2008)
7. The eMips project, http://research.microsoft.com/en-us/projects/emips/default.aspx
8. Davies, D.W., Price, W.L.: Security for Computer Networks. Security for Computer Networks. Wiley (1989)
9. Rivest, R.L.: The RC5 encryption algorithm. In: Proceedings of the 2nd Workshop on Fast Software Encryption, pp. 86–96. Springer (1995)

10. Rivest, R.L.: The MD5 message-digest algorithm, Request for Comments (RFC1320). Internet Activities Board, Internet Privacy Task Force (1992)
11. Lai, X.: On the Design and Security of Block Ciphers. Hartung-Gorre Verlag (1992)
12. FIPS 180-1. Secure hash standard, NIST, US Department of Commerce, Washington D.C. Springer (1996)
13. Counterpane Systems, `http://www.counterpane.com`
14. Koblitz, N.: Elliptic curve cryptosystems. Mathematics of Computation 48, 203–209 (1987)
15. IEEE 1363: Standard Specifications for Public-Key Cryptography, `http://grouper.ieee.org/groups/1363/`
16. Schneier, B.: Applied Cryptography. John Wiley & Sons (1996)
17. Hodjat, A.: Interfacing a high speed crypto accelerator to an embedded CPU. In: Proceedings of the 38th Asilomar Conference on Signals, Systems, and Computers, pp. 488–492 (2004)
18. Harrison, O., Waldron, J.: AES Encryption Implementation and Analysis on Commodity Graphics Processing Units. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 209–226. Springer, Heidelberg (2007)
19. Mourad, O.-C., Lotfy, S.-M., Noureddine, M., Ahmed, B., Camel, T.: AES Embedded Hardware Implementation. In: Second NASA/ESA Conference on Adaptive Hardware and Systems (AHS 2007), pp. 103–109 (2002)
20. Bertoni, G.M., Breveglieri, L., Roberto, F., Regazzoni, F.: Speeding up AES by extending a 32 bit processor instruction set. In: Proceedings of the IEEE 17th International Conference on Application-Specific Systems, Architectures and Processors (ASAP 2006), pp. 275–282 (2006)
21. Tang, J., Liu, S., Gu, Z., Li, X.-F., Gaudiot, J.-L.: Hardware-Assisted Middleware: Acceleration of Garbage Collection Operations. In: Proceedings of the 21st IEEE International Conference on Application-Specific Systems, Architectures and Processors, ASAP 2010 (2010)
22. Tang, J., Liu, S., Gu, Z., Liu, C., Gaudiot, J.-L.: Prefetching in Embedded Mobile Systems Can Be Energy-Efficient. Computer Architecture Letters (February 2011), `http://doi.ieeecomputersociety.org/10.1109/L-CA.2011.2`
23. Tang, J., Thanarungroj, P., Liu, C., Liu, S., Gu, Z., Gaudiot, J.-L.: Pinned OS/Services: A Case Study of XML Parsing on Intel SCC. Journal of Computer Science and Technology (in press)
24. Tang, J., Liu, S., Liu, C., Gu, Z., Gaudiot, J.-L.: Acceleration of XML Parsing Through Prefetching. IEEE Transactions on Computers (in press)
25. Tang, J., Liu, S., Gu, Z., Li, X.-F., Gaudiot, J.-L.: Achieving Middleware Execution Efficiency: Hardware-Assisted Garbage Collection Operations. Journal of Supercomputing (November 2010), doi:10.1007/s11227-010-0493-0
26. Tang, J., Liu, S., Gu, Z., Liu, C., Gaudiot, J.-L.: Memory-Side Acceleration for XML Parsing. In: Altman, E., Shi, W. (eds.) NPC 2011. LNCS, vol. 6985, pp. 277–292. Springer, Heidelberg (2011)
27. Liu, S., Pittman, R.N., Forin, A., Gaudiot, J.-L.: Minimizing the Runtime Partial Reconfiguration Overheads in Reconfigurable Systems. Journal of Supercomputing (July 22, 2011), doi:10.1007/s11227-011-0657-6
28. Chang, J.K.-T., Liu, C., Liu, S., Gaudiot, J.-L.: Workload Characterization of Cryptography Algorithms for Hardware Acceleration. In: Proceedings of the 2nd ACM International Conference on Performance Engineering (ICPE 2011), Karlsruhe, Germany, March 14-16 (2011)

Chaotic Wireless Communication System Using Retrodirective Array Antenna for Advanced High Security

Junyeong Bok and Heung-Gyoon Ryu

Department of Electronic Engineering
Chungbuk National University
Cheongju, Korea 361-763
bjy84@nate.com, ecomm@cbu.ac.kr

Abstract. In this paper, we propose chaotic wireless communication system using digital retrodirective array antenna (RDA) for improving security and receive performance at receiver. Chaotic modulation schemes have studied for reducing the probability of interception and interrupt. As a result, chaotic communication systems have enhanced security. But the receive performance at receiver is degraded. We propose digital RDA system based on chaotic modulation for improving the reception performance while maintaining the security. Simulation results show that to overcome degradation of reception performance in CDSK, digital RDA with array elements up to five is required compared to BPSK modulation scheme.

Keywords: DCSK**,** CDSK, Security, tent map, retrodirective array antenna.

1 Introduction

Recently, security problem is become an important research topic due to popularization of device such as smart phone and tablets. Many studies about chaotic wireless communication have been studied to enhance security. There has been studied about efficient coding schemes for improving security applying chaotic signal due to the chaotic signal whit irregular phenomena.

Especially, differential chaos shift key (DCSK) and correlation delay shift keying (CDSK) are well known as chaotic modulation schemes. These modulation techniques have enhanced security because of non periodic properties. But the receive performance is degraded due to non periodic characteristic compared to QAM modulation [1-2].

There have been studied diversity technique and beam forming technique in order to improve the receive performance. Especially, one of beam forming techniques, digital retro-directive array (RDA) technique is able to transmit signal toward source with no a priori knowledge of the arrival direction [3]. Retrodirective array has more simple structure than smart antenna technique and it is possible to do automatically beam tracking. Also, retrodirective system has merit such as high link gain, easy interference elimination, and high energy efficiency.

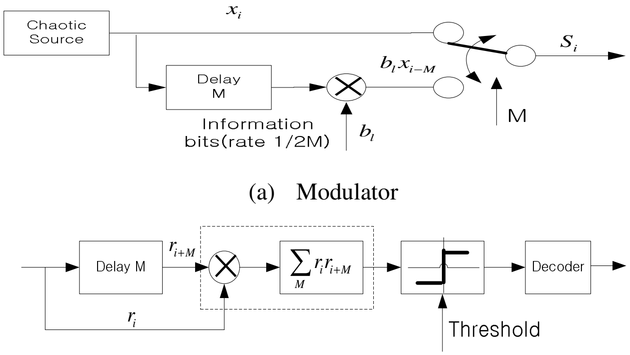
J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 482–490, 2013.

In this paper, we propose digital retrodirective array antenna system (RDA) based on CDSK for improving security without receive performance degradation. The proposed system can improve the receive BER performance of communication system compared to without RDA and kept security of whole communication system.

2 Chaotic Modulation Techniques

2.1 DCSK Modulation

DCSK is well knows as chaotic modulation techniques. Fig. 1 shows modulator and demodulator for DCSK. Multiply chaotic signal x_i with length M and information bit $b_l = \pm 1$ together and we get transmitter signal. In other words, information bit is spreading throughout chaotic signal with length M. Transmission signal s_i can be expressed by



(a) Modulator

(b) Demodulator

Fig. 1. Block diagram of modulator and demodulator for DCSK.

$$s_i = \begin{cases} x_i & ,0 < i \le M \\ b_l x_{i-M} & ,M < i \le 2M \end{cases} \tag{1}$$

where b_l is information bit and x_i is chaotic signal. For demodulation of DCSK, received signal r_i is multiplied by delayed signal r_{i+M} at the receiver. As a result, we can receive average signal by spreading factor M. Finally, the output signal of the correlator can be expressed by

$$S = \sum_{i=1}^{M} r_i r_{i+M} \tag{2}$$

where S is the output signal of correlator.

If we consider AWGN noise, the output of correlator can be expressed by

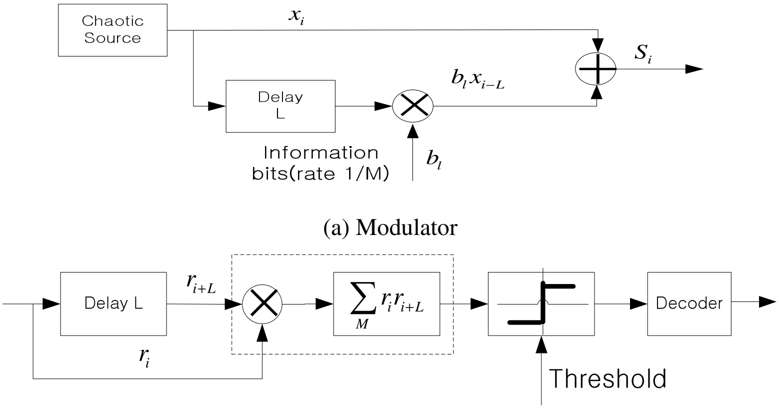
$$\begin{aligned} S &= \sum_{i=1}^{M} (s_i + n_i)(s_{i+M} + n_{i+M}) \\ &= \sum_{i=1}^{M} (b_l x_i^2 + x_i(n_{i+M} + b_l n_i) + n_i n_{i+M}) \\ &= b_l \sum_{i=1}^{M} x_i^2 + \sum_{i=1}^{M} x_i(n_{i+M} + b_l n_i) + n_i n_{i+M}) \end{aligned} \tag{3}$$

where n_i is noise signal at i th.

The output of correlator is separated the first term as the desired signal and other term as interference and noise. Average and variance of AWGN are 0 and σ^2 because of interference and noise with Gaussian distribution. DCSK scheme has more enhanced security than generally modulation methods such as PSK and QAM. But, bandwidth efficiency is degraded because the same chaotic signal send repeated twice.

2.2 CDSK Modulation

DCSK modulation do not use half of the symbol period for carrying information. To solve bandwidth reduction problem correlation delay shift keying (CDSK) modulation was proposed by Sushchik. This paper, we use CDSK modulation with good spectral efficiency for security communication.



(a) Modulator

(b) Demodulator

Fig. 2. Block diagram of modulator and demodulator for CDSK

Fig. 2 shows modulator and demodulator of correlation delay shift keying (CDSK) schemes. CDSK modulation method is made of the sum of the signals multiplied by chaotic signal x_i and binary information bit $b_l = \pm 1$. CDSK modulation did solve the bandwidth problem of DCSK as using adder instead of DCSK with switch. The output of correlator of CDSK can be expressed by

$$\begin{aligned} S &= \sum_{i=1}^{M} (x_i + b_l x_{i-L} + n_i)(x_{i-L} + b_{l-1} x_{i-2L}) \\ &= b_l \sum_{i=1}^{M} x_{i-L}^2 + \sum_{i=1}^{M} \eta_i \end{aligned} \tag{4}$$

$$\begin{aligned} \eta_i = x_i x_{i-L} + b_{l-1} x_i x_{i-2L} + b_1 b_{l-1} x_{i-L} x_{i-2L} + x_i n_{i-L} + \\ b_1 x_{i-L} n_{i-L} + x_{i-L} n_i + b_{l-1} x_{i-2L} n_i + n_i n_{i-L} \end{aligned} \tag{5}$$

where L is delay length, η_i is noise and interference term.

As you can show the equation (4), the first term is designed signal and other term the noise of correlator and interference of different chaotic signal. The degradation of BER performance in CDSK occurs due to additional interference terms in shown (5) compare to DCSK modulation. The bet error rate equation (BER) of CDSK modulation can be expressed by [4].

$$BER = erfc(\sqrt{\frac{E_b}{8N_0}(1+\frac{19}{20M}\frac{E_b}{N_0}+\frac{M}{4}\frac{N_0}{E_b})^{-1}}). \tag{6}$$

3 Chaotic Communication System Using Digital Retrodirective Array Antenna

Chaotic communication system provides enhanced security. But the receive BER performance is degraded compared to the BPSK because of no periodically properties. In order to improve the receive performance we propose chaotic wireless communication system using digital RDA. Digital RDA can retransmit information data from digital RDA to source without prior information about direction of receive. Digital RDA has some merits that it is does not need complex signal processing and it can have array gain through beam.

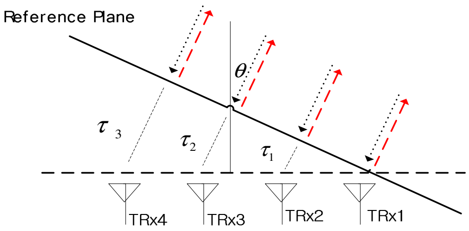


Fig. 3. Concept of digital RDA

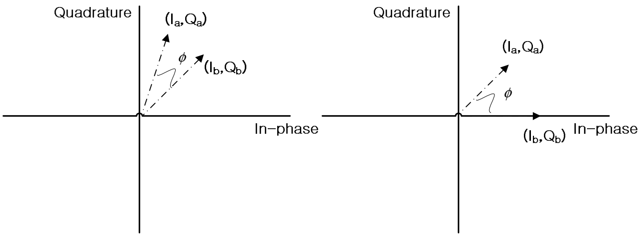
The received signal through each of the receiver has different phase delays $0,\ \varphi,\ 2\varphi, 3\varphi$ due to the different time delays $0,\ \tau_1,\ \tau_2,\ \tau_3$ when incidence angle has θ shown in Fig. 3. In other words, when the number of antenna elements is Nth, received signal has a different phase delay by following equation

$$\triangle(n-1)\varphi = 2\pi f \frac{d}{c}\sin\theta, \quad n=1,\cdots,N \tag{7}$$

where $\Delta\varphi$ is phase delay between adjacent elements like reference element and second element. f is center frequency, c is light speed, d is distance between adjacent element, and θ is incidence angle. To retransmit information data toward incidence angle, we can retransmit received data after processing phase conjugation at received signal. The following condition must be satisfied in order to satisfy these conditions.

$$\varphi_{T_x} = -\varphi_{R_x} \tag{8}$$

where is φ_{R_x} is phase delay and φ_{T_x} is phase delay for retransmission from digital RDA to source. It is important to detect phase delay for processing phase conjugation at received signal.



(a) QPSK (b) DCSK or CDSK

Fig. 4. Phase detector based on baseband

Fig. 4 shows phase detection based on baseband for digital RDA in case of QPSK and chaotic modulation such as DCSK and CDSK. We can represent as received signal (I_b, Q_b) through reference element and received signal (I_a, Q_a) of adjacent element in case of QPSK shown in fig. 4(a). Received signal through reference antenna element has phase delay=0 and N=1 (7).

Phase delay φ by adjacent antenna elements is given by following equation

$$e^{j\varphi} = e^{j(\phi_a - \phi_b)} \tag{9}$$

Equation (10) is given by rearranging equation (9) as follows

$$\begin{aligned} e^{j\phi} &= \frac{I_a}{\sqrt{I_a^2+Q_a^2}}\frac{I_b}{\sqrt{I_b^2+Q_b^2}} + \frac{Q_a}{\sqrt{I_a^2+Q_a^2}}\frac{Q_b}{\sqrt{I_b^2+Q_b^2}} \\ &+ j(\frac{Q_a}{\sqrt{I_a^2+Q_a^2}}\frac{I_b}{\sqrt{I_b^2+Q_b^2}} - \frac{Q_b}{\sqrt{I_b^2+Q_b^2}}\frac{I_a}{\sqrt{I_a^2+Q_a^2}}) \end{aligned} \tag{10}$$

In case QPSK modulation, the amplitude signal has $\sqrt{2}$. In generally the phase delay of adjacent antenna element is small 30 degree ($|\phi| < 30^0$), the equation (10) can be approximated as follows

$$\phi \simeq \sin\phi = \frac{1}{2\sqrt{2(I_a^2 + Q_a^2)}}(I_b Q_a - I_a Q_b) \tag{11}$$

When we ousted phase information about equation (11), can be obtained final expression as follows

$$\phi = (I_b Q_a - I_a Q_b) \tag{12}$$

Chaotic wireless communication is generated by chaotic maps such as Logistic map, Tent map, Henon map, and Bernoulli shit map. The chaotic signal generated is multiplied by bit information. As a result, chaotic signal has only in-phase component in contrast QPSK modulation. In other words, Imaginary part of received signal has 0 ($Q_b = 0$).

In the case of use chaotic modulation, phase delay φ between adjacent elements can be expressed as

$$\phi = (I_b Q_a) \tag{13}$$

Digital RDA in the case of chaotic modulation technique can detect and retransmit phase delay and data information by using equation (13) and phase conjugation.

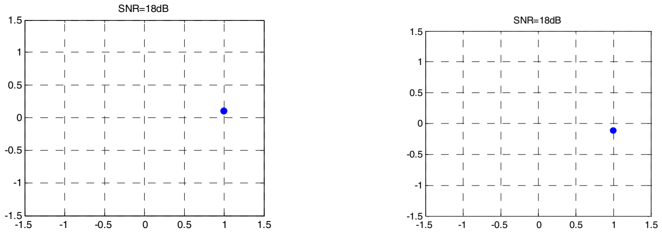
4 Simulation Results

In this paper, we propose a digital retrodirective array antenna (RDA) to improve receive performance of CDSK modulation which has enhanced security. Digital RDA technique can retransmit toward direction of source without prior information about receive direction. Table 1 shows simulation parameters.

Table 1. Simulation Parmeters

Parameters	Value
Chaotic map	Tent map
Spreading factor(M)	10, 20, 50, 100
Modulation	CDSK
Delay(L)	7

We assume that array element of Digital RDA is two. If element is two, first element which has phase delay 0 degree is reference plane and second element has a phase lag φ. We can calculate the phase lag between reference element and adjacent element by using equation (13). Fig. 5(a) shows the output of phase detector, when phase delay φ is 10 degree. We can know the relationship between the incidence angle θ and phase delay φ through equation (7).



(a) The output of phase detector (b) The output of phase conjugator

Fig. 5. The output of phase detector and phase conjugator

Fig. 5(b) shows the output of phase conjugator. Fig. 5(b) and Fig. 5(a) is related by conjugation. We confirm that digital RDA can retransmit to direction of source by using the block of phase detection and phase conjugation. As a result, Fig. 5(a) and Fig 5(b) shows the output phase detector and phase conjugator when phase delay is 10 degree and SNR of AWGN is 18dB.

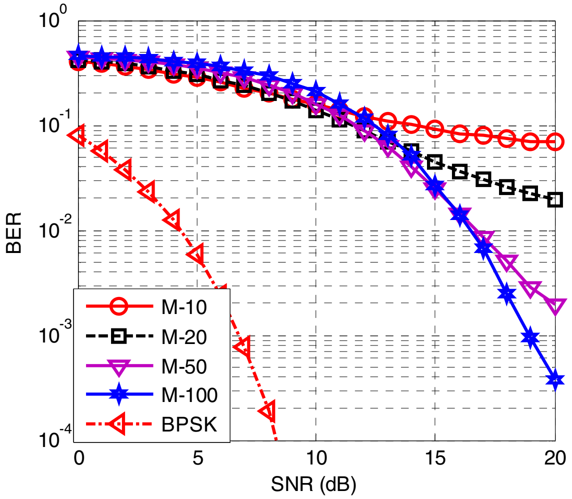


Fig. 6. Comparison of BER performance in CDSK by size of M

Fig. 6 shows the BER performance of CDSK modulation according to changing spreading factor M. CDSK modulation has higher security compared with BPSK modulation because it has not a periodically specific character. But, CDSK is degrade receive BER performance. When spreading factor increases, we can see that both interference and energy an error is reduced through equation (6) and simulation in Fig. 6. Depending on the choice of spreading factor in interference environment, the effect of interference is different. The BER performance of CDSK is not good compared to that of BPSK modulation.

Fig. 7 shows comparison BER performance with and without RDA. We analyze the BER performance of proposed system according to changing the number of RDA elements. We use CDSK modulation and digital RDA antennas. The proposed system has enhanced security and no degradation of BER performance in the case of CDSK when the number of array elements is five compared to BPSK modulation. Digital RDA can automatically make beam toward direction of transmitter. In this case beam of digital RDA provides additional antenna gain. Simulation results show the number of element increases, the BER performance of CDSK improves due to array gain.

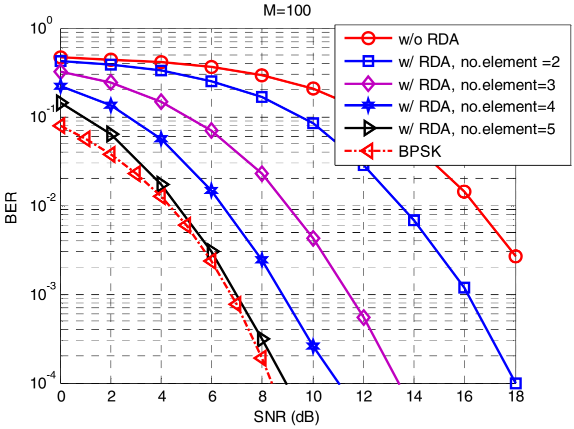


Fig. 7. Comparison of BER performance according to changing the number of RDA element

5 Conclusion

In this paper, we design and analyze the receive performance of CDSK based on digital RDA to improve receive BER performance compared to only CDSK modulation. We propose chaotic wireless communication system enhanced security by using CDSK modulation and digital RDA. The proposed system get a better receive BER performance than that without RDA. Simulation results show we use digital RDA with array element up to five in CDSK to overcome to receive BER performance is degraded compared to BPSK modulation schemes.

Acknowledgment. "This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2012017339)"

References

1. Arai, S., Nishio, Y.: Noncoherent Correlation-Based Communication Systems Choosing Different Chaotic Maps. In: ISCAS 2007, New Orleans, LA, May 27-30, pp. 1433–1436 (2007)
2. Wren, T.J., Yang, T.C.: Orthogonal chaotic vector shift keying in digital communications. IET Communications 4, 739–753 (2010)
3. Sun, J., Zeng, X., Chen, Z.: A Direct RF-undersampling Retrodirective Array System. In: Proceedings of IEEE Radio and Wireless Symposium (RWS 2008), pp. 631–634 (January 2008)
4. Jin, C.-H., Ryu, H.-G.: Performance Evaluation of Chaotic CDSK Modulation System with Different Chaotic Maps. In: ICTC 2012, Jeju lsland, Korea, October 15-17, pp. 603–606 (2012)

Policy-Based Customized Privacy Preserving Mechanism for SaaS Applications

Yuliang Shi[1], Zhen Jiang[1], and Kun Zhang[2]

[1] School of Computer Science and Technology, Shandong University, Jinan, China
[2] School of Information science and Engineering, University of Jinan, Jinan, China
liangyus@sdu.edu.cn, jizh1234@gmail.com, ise_zhangk@ujn.edu.cn

Abstract. In the SaaS (Software as a Service) model, the sensitive data of tenants are in danger of leakage. Meanwhile there are different privacy requirements for different tenants. This paper presents a policy based customized privacy preserving mechanism which realizes the preserving of tenants' sensitive data. Based on the requirements of the tenants and the transactions of SaaS application, we build the policy of tenants' customized privacy preserving and fragment tenants' sensitive data through the Related Attributes Model(RAM). Finally we realize the effective combination of unencrypted privacy preserving and SaaS application's transaction. To avoid the leakage of tenants' privacy policy, this paper presents a trusted third party model to manage the policy of tenants' customized privacy preserving. The experiment certified it's an effective and practical privacy preserving mechanism.

Keywords: SaaS, Hybrid Fragmentation, Data Privacy, Customization.

1 Introduction

Software as a Service is a new-style software service model which has many advantages such as low cost, fast deployment and scalability. In the SaaS model tenants' sensitive data are deployed by the service provider who manages and maintains it. As the service provider is unreliable, tenants' sensitive data face the risk of leakage. For example, they may break the commodity's quote information to the competitors.

There have been some mature solutions to preserving privacy such as data encryption [1, 3, 5, 6, 7, 9] and data confuse [14, 15, 16], etc. Data encryption can effectively prevent the leakage of privacy data, but the efficiency of the ciphertext processing is low. Data encryption can also be realized by hardware. Based on the anti-attack capability of the password collaborative processor, it can use the password collaborative processor deployed on the unreliable port as the reliable processor to realize encryption and deciphering in its interior. Unlike encryption, data confusion keeps some features of data and could make some computing on confused data, but the efficiency of data processing need to be enhanced. In SaaS model, the traditional data encryption technique or data confusion technique reduces the efficiency of data processing. The password collaborative processor increases the complexity of system design. Based on the transaction of SaaS application, the combination of the privacy preserving and

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 491–500, 2013.

performance should be realized. Meanwhile the customized privacy preserving requirements of different tenants ought to be considered.

The paper adopts a privacy preserving mechanism based on policy which ensures that the tenant sensitive data are protected without encryption. The mechanism uses the policy to describe tenants' customized privacy requirements and protects the tenants' sensitive data based on policy through hiding the relation of sensitive data. Privacy is deployed on trusted three-party to ensure the security of the tenants' privacy policy, it is necessary to build the secure association model between the SaaS application, the reliable third party and the tenants. Through the model, it can prevent unauthorized tenants obtaining the privacy policy and leaking the tenants' sensitive data.

This paper is further organized as follows. Section 2 gives related work. In section 3 we describe the method of privacy data fragment. Section 4 introduces the three-party interactive model. In section 5, we explain our experiments and analyze the results. Section 6 concludes this paper.

2 Related Work

In the methods of data privacy protection, both encryption and confusion are relatively common, which have obtained a wide range of application in the data outsourcing, data dissemination, data analysis and other fields.

Encryption is an effective method to protect individual privacy, but the encrypted data often lose the operability, and therefore improve the processing efficiency of the cipher-text and the retrieval speed of the cipher-text data has become the hot spot in the privacy research. In [1], researchers have analyzed the characteristics of cloud computing, then propose a keyword searching model of privacy protection, which gives a support that the ISP can participate in some of decryption work in order to reduce the burden of clients. Besides, this model can realize the keywords search above the encrypted data in order to protect the tenant data privacy and user query privacy. In [2], Sadeghi et al. have designed a credible software token bound with a security function authentication module in order that many function operations for outsourcing sensitive data can be implemented on condition that no information is lost.

In [3] the researchers have constructed a data security service module CSS based on the public cloud infrastructure platform to ensure user data privacy through encryption and token service. In [4], researches on the cloud data privacy protection respectively got the solutions about the cipher-text retrieval. In order to provide privacy in the cloud computing environment, [5] has designed a calculable encryption scheme CESVMC based on matrix and vector operations. [6] has adopted the multi-layer encryption to protect data privacy in a relational database. However, these studies belong to the scope of the Database as a Service without considering the scene of the multi-tenant applications.

Homomorphic encryption supports the processing and computing in the cipher-text data. Traditional homomorphic encryption only supports a certain type or a specific operation, such as state multiplication RSA [7], the homomorphic addition Paillier [8] and so on. The use of homomorphic encryption can outsource the specific calculation to non-trusted third party, such as encryption or digital signatures [9]. At present, the researchers have made deep research on this, however, the homomorphic encryption technology efficiency is very low [10].

Encryption methods have a big impact on the performance of data processing, and researchers have proposed other ways to prevent the disclosure of privacy. Munts-Mulero etc [11], have discussed the existing privacy processing technology, including K anonymous, Figure anonymous and data pretreatment, and proposed some solutions of the problems faced as a large-scale release data. [12] has proposed a privacy protection method based on lossy decomposition. [13] has proposed a multidimensional data packet technology according to sensitive property privacy. However, these studies are mainly for data dissemination in the field of privacy protection issues and the field of data dissemination. The operation of the data values has been less involved in. Contrarily, in the cloud multi-tenant scenarios, due to the needs of the multi-tenant, the tenants' data is changing dynamically and constantly. These privacy protection methods can't fully resolve issues on cloud data security and privacy protection for multi-tenant applications, but they can provide a good reference for the cloud data protection.

In summary, the researchers have launched data privacy protection studies in the cloud computing, but these studies mainly aim at the data privacy protection realized through encryption. Data processing efficiency needs to be further improved, and these studies are not suitable for the scene of the multi-tenant SaaS applications.

3 Privacy Preserving Mechanism

This section introduces policy-based customized privacy preserving mechanism. Different from encrypt single attribute privacy constraint formerly [14], we split singleton attribute. And for the combine attribute privacy constraint we adapt the pattern of privacy fragment.

3.1 Customized Privacy Preserving

For described tenant's privacy preserving requirements, we first define some defination [14]. Singleton attribute is the attribute itself can leakage tenants' privacy. Attribute combination is the attributes can leakage tenants' privacy in together.

Definition 1: Non-Compatible Constraint (NCC) is used to describe single attribute privacy constraint and combine attribute privacy constraint. For attributes combination $A_c\{A_1, A_2,\dots ,A_n\}$, *NCC* $\{A_c\}$ describes the attribute in attributes set A_c can't be put together. For singleton attribute A_s, *NCC*$\{A_s\}$ described it. *NCC* $\{A_s, A_c\}$ describes both the Singleton attribute and attributes combination.

Definition 2: Related Attributes Set (RAS) is made up of attributes set and the weight of this set. It is expressed as *RAS ({A}, W)*. *{A}* describes attributes which a SQL statement contains and *W* stands for weight which describes this SQL's proportion in all the SQLs.

For example, *RAS {(A, B, C, D), 25}* indicates this SQL statement contains four attributes which are A,B,C,D and it's accounts for 25% in all the SQL statements.

We use *RAS set* to guide the data fragment. We count tenant's SQL statements in SaaS application including attributes set and its proportion, as is shown in Table1.

Table 1. SQL Statistics

SQL	RATIO
select D,E,F from T-A where E>e	**20%**
select A,B,C from T-A where B>b	**30%**
select A from T-A where B<b and C>c	**20%**
select A, I from T-A where I>i	**10%**
select D from T-A where E<e	**10%**
select E,F from T-A where D>d	**10%**

We preserve attributes in the same table and remove the attributes from other tables. We make pretreatment for every SQL that makes attributes set as *{A}* and proportion as *W*. We analyze all the *SQL* and make merger for the *SQL* which has inclusion relationship as a RAS. Then we get *RAS ({A}, W)*, as is shown in the right part of Figure 1.

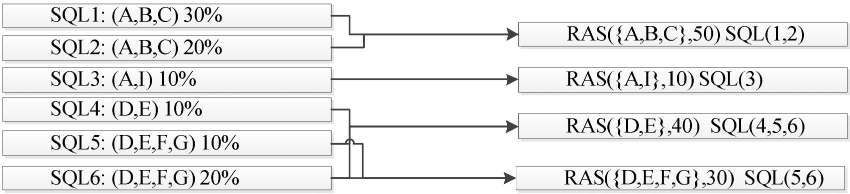


Fig. 1. SQL Combination

3.2 Data Fragment

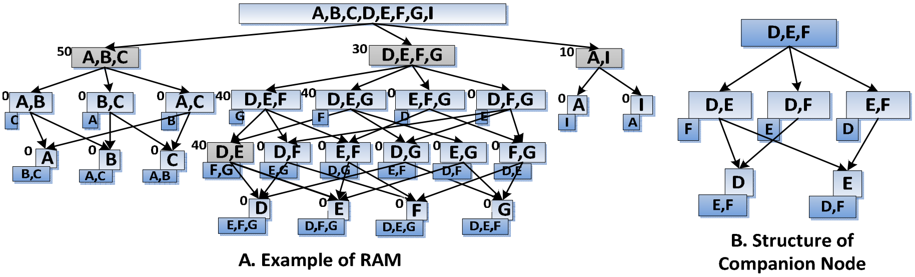
Related Attributes Model. We find optimal fragments for every *RAS* by building related attributes tree model based on *RAS set*.

Definition 3: Related Attributes Model (RAM) is described as *RAM(N, E)* in which *N={R, RAS, C, T}* is all the node set of the tree. The R stands for root node, *RAS* stands for *RAS* node, and C stands for companion node and T stands for explore node. $\mathrm{E} \subseteq \mathrm{N} \times \mathrm{N}$ is edge set of the node. Constraint $\mathrm{RAS} \subset \mathrm{T}$ means *RAS* node can convert to be explore Node. Constraint $(\mathrm{R}, \mathrm{RAS}) \in \mathrm{E}$ means that root node's children node can be only RAS node. Constraint $(\mathrm{T}, \mathrm{C}) \in \mathrm{E}$ means that companion node is only linked with explore node. *RAS* node and companion node can extend related explore node.

We build related attributes tree as follows:

Firstly, singleton attribute is disposed before building related attributes model by splitting the singleton attribute to N parts and adding *NCC*$(I_1, I_{2,} \ldots I_n)$ to *NCC*$\{A_c\}$.In this way we can handle singleton attribute as attributes combination and see every split singleton attribute as a new attribute.

Secondly, we create node for the model tree. The root of the tree contains all attributes that appear in the *RAS set*. Each *RAS* in *RAS set* are distributed to a node where preserves the *RAS*'s attributes and weight. Then we build explore node as *RAS* node's children node which removes an attribute of *RAS* respectively and the weight of explore node is 0. At last we create companion node for every explore node. The companion node can extend explore node same as *RAS* node (Figure 2-b). If a *RAS* node exists in another *RAS's* explore branch, this *RAS* node becomes explore node and assigns this *RAS*'s weight to all the further explore node in the branch except that the node has already had weight value(not 0),e.g. the DE node in Figure 2-a. And each *RAS* node's explore node set have no relevance.



A. Example of RAM

B. Structure of Companion Node

Fig. 2. Related Attributes Tree Model

Privacy Fragment Algorithm. After the related attributes tree has been created, we sort all the children of root by ascending order and traverse these children in sequence. For each RAS node we get the node in the highest layout which don't against *NCC* $\{A_c\}$ and regard it as the optimal fragment node. Then we traverse the optimal fragment node's companion node as the same. If there are many nodes in the highest layout, we choose the node with highest weight. For example, we traverse the DEFG node's branch and get DEF node as optimal fragment node, then we traverse DEG's companion node and get node F. The worst-cost of the algorithm is traversing all the node of the tree, so the time complexity of this algorithm is O (n).

Algorithm 1. Privacy Fragment Algorithm

```
input: RAS- TREE,NCC{A_c}
output: optimal node
sort children of root by RAM child's ratio in asc order
for each child N of root
    currentNode←null, currentRatio←0, currentLevel←n, tempLeve←0
    TestNode(N)
```

TestNode

```
input: Node
output: PFS
if (NCC ⊄ N) Then currentNode←N
else if (N ⊃ NCC)
    Then currentNode←TraversalNode (N) //get N's optimal fragment node
add optimal node into PFS
get companyNode cn of currentNode
    TestNode(cn)
```

Definition 4: Privacy Fragmentation policy (PFS) for the table R and related *RAS set* and *NCC*(A_c), R can be split to many parts F_1, F_2 …F_n by *NCC* and *RAS set.* F_i is subset of R, $F_i \cap F_j = \emptyset$ and $R=F_1 \cup F_2 \cup \ldots \cup F_n$.

In this paper, the evaluation criterion of the optimal fragment policy is the higher weight of RAS would have higher integrality for which ensure the transaction proportion with higher make less join. And above all should under the premise of without departing from the NCC. Algorithm 1 which we present would prior traverse RAS node with highest weight and joins the result in the Privacy Fragmentation Strategy (PFS). In this way we can ensure the RAS node with lower weight can't affect the RAS node with higher weight.

Algorithm 2. Add Node Into PFS

input: *optimal Node from privacy fragment algorithm*

output: *PFS{F_1, F_2…F_n}*

```
attr$\leftarrow$N.attribute
Sort F by common attributes in PFS in asc order
 i$\leftarrow$0
 for each F$\in$PFS
     attr$\leftarrow$attr-{attribute|attribute$\in$PFS}
     if (NCC$\subset$(attr$\cup$F)) Then continue
     else if(!NCC$\subset$(attr$\cup$F) Then  F$\leftarrow$attr$\cup$F, i$\leftarrow$1,break
if (i==0) Then F$\leftarrow$attr, PFS$\leftarrow$PFS$\cup$F
```

The way to put fragment node to *PFS* is as follows: First we sort fragment policy by common attribute with added node ascending in *PFS* and remove the added node's attribute which has contained in *PFS*. Second we combine added node attributes and *PFS*'s fragment policy respectively and check whether the combined attributes are against *NCC*. If they are not against, we replace old policy with the combined attributes. If all the checks are failed, we create a new policy in *PFS*.

4 Three-Party Interactive Model

We have fragmented tenant's privacy data according to tenant's requirement. But if the fragment policy of tenant saves at the service provider, it also exists possibility that the provider get integrate data by the policy. This section mainly introduces how to ensure the safety of tenant's customized privacy preserving policy.

In SaaS model, storing and processing the sensitive data of tenants are operated on the untrustworthy platform of service provider. As service provider is unreliable, tenants' sensitive data face the risk of leakage. To realize the preserving of tenant's policy, this section introduces the trusted third-party and builds a privacy preserving model for the data of SaaS application. This model consists of three roles which are tenant (T), unreliable service provider (SP) and trusted third-party (TTP). TTP is responsible for managing tenant's privacy preserving policy and preventing SP from obtaining the policy illegally through identity authentication for the tenant.

4.1 Three-Party Interactive Entity and Process

The paper designs the process of three-party interactive process. The process involves three interactive entities which are tenant(T) of SaaS application, unreliable service provider(SP) and trusted third-party(TTP) which is responsible for tenant's identity authentication, saving and managing the privacy policy.

The memory of SaaS server can't be lookup by SP is the pre-condition of the three part interaction model. 1) When tenants rent SaaS services from SP, 2) SP will guide tenant to register at TTP and tenant will get an authentication to identify from TTP. 3) After register succeed, tenant can present requirement of privacy to privacy preserving model in SP's memory. Privacy preserving model will call the algorithm to get privacy policy and fragment tenant's data by the policy. 4) The privacy customized Model will send tenant's policy to TTP's privacy policy management model to store and manage. 5) SP's privacy preserving model will get accredit when tenant accesses SaaS application and need to use data. 6) And then the privacy preserving model will get the tenant's policy to combine the fragment data.

All above steps are processed in memory and don't be persistent, so the interaction is secure.

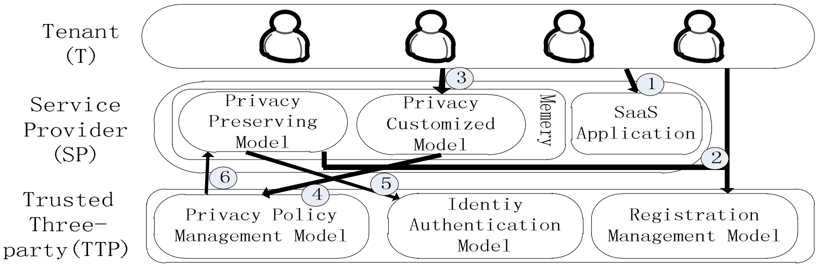


Fig. 3. Three-party interactive process

4.2 Three-Party Interactive Security Verification

This paper adopts identity-based authenticated mechanism to make authentication of SP and tenant. The identity-based authenticated mechanism generate public key by tenant's identity information. Private Key Generator (PKG) according to tenant's public key generate tenant's privacy key and sent it to tenant through secure channel. The identity-based authenticated mechanism need PKG is reliable to ensure the security of system's privacy key, and mean while need authenticate tenant's identity before send the privacy key.

The three-party interactive including three phases: initialization phase, registration phase and access phase.

Initialization phase: firstly we build two group with q2 order G_1, G_2, q is a big prime, G_1 is consist of point in elliptic curve, P is generator of G_1, G_2 is a multiplication cyclic group in $F_{q^2}^*$.Select bilinear map $e: G_1 \times G_2 \to G_2$.PKG select a random number $s \in Z_q^*$ as system's privacy key, select two hash function $H_1: \{0,1\} \to G_1$, $H_2: \{0,1\}^* \times G_2 \to Z_q^*$. Finally ensure public system parameter $params = \langle q, P, G_1, G_2, e, H_1, H_2 \rangle$, privacy key s saved by PKG.

Registration phase: SP sent register information to TTP to register. TTP generate SP's public key $Q_{sp} = H_1(ID_{sp})$ and SP's privacy key $d_{sp} = s \cdot Q_{sp}$ by SP's identity

ID_{sp}. Then send SP's privacy key to SP in secure channel. Tenant T rent SaaS application of SP, SP will guide tenant register to TTP. T send message M_1 which have T's ID_T, SP's identity ID_{sp}, register requirement R and $H_2(R, g_1)$. $Q_T = H_1(ID_T)$, d_{sp} is SP's privacy key, $g_1 = e(Q_T, d_{sp})$, $H_2(R, g_1)$ is authorization from SP which allow T register to TTP. When TTP get M_1 and generate $d_T = s \cdot H_1(ID_T)$, $g_1' = e(d_T, H_1(ID_{sp}))$, then test and verify if $H_2(R, g_1')$ equals $H_2(R, g_1)$,the test passed it equal. TTP send T's privacy key d_T to T through secure channel after register.

Access phase: SP need download privacy policy from TTP when T access SaaS application. T send message M_2 to TTP which including ID_T, ID_{sp}, ask time T, download requirement D and $H_2(D\|T, g_2)$. TTP will judge $\Delta T = T' - T$ when received M_2, T' is the time received the message, if ΔT is in set time TTP accept the message. Then TTP generate $d_{sp} = s \cdot H_1(ID_{sp})$ and $g_2 = e(Q_T, d_{sp})$, test if $H_2(D\|T, g_2')$ equals $H_2(D\|T, g_2)$,if it is equal TTP allow SP download T's privacy policy to its privacy preserving model.

5 Experimental Evaluation

5.1 Experiment Environment

For SaaS data privacy protection, we check the practicability of privacy protection fragment algorithm by simulation experiment. The experiment is made with MySQL 5.1.22 and Eclipse-SDK-3.4.3-win32 in the Windows7 Professional Service pack 1. The CPU is Inter(R) core(TM) i5-2400, 3.1GHz.and the memory is 3G.

In our experiment, we get the RAS and NCC in random from the array set in advance.

5.2 Privacy Data Fragment Experiment

To check the performance of the fragment algorithm with different privacy requirements, we design different privacy requirements as Table 2. All the *RAS* and *NCC* attributes are selected on the same table.

Table 2. different types of privacy requirement

Type	*NCC* number
Type A	20
Type B	40
Type C	60

For the different types of *NCC* number, we get the cost with 100, 200, 500 and 1000 *RAS* number respectively.The experimental result is as Figure 4.

The experimental results show that the cost is increased with the increase of *NCC* number. The cost and the transaction of SaaS application present linear growth relationship under the same *NCC* number. The cost of three types tenant's privacy requirements is within acceptable limits.

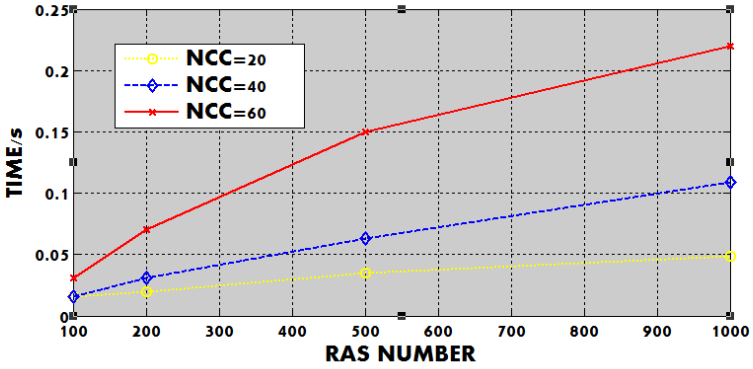
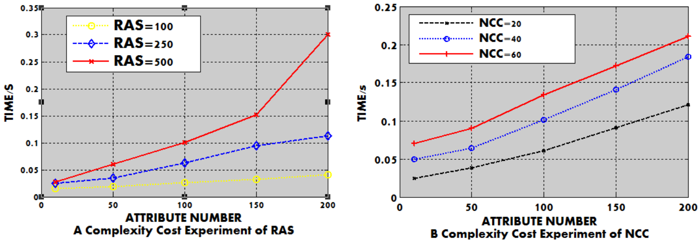


Fig. 4. Privacy Data Fragment Experiment Result

5.3 Complexity Cost Experiment

To check the cost of algorithm with different meta-data number, we design an experiment with 10, 50, 100 and 200 mete-data in a table and simulate the cost.

Firstly we simulate that *NCC* is 20 with *RAS* number 100, 250 and 500 respectively. The results are shown as in Figure 5-A. From the Figure we can see the cost increases with the meta-date increasing; the more the *RAS*, the faster data increases. We then simulate that *RAS* is 250 with *NCC* number 20, 40 and 60 respectively. The results are shown as in Figure 5-B. And both two results are within the acceptable limits.



A Complexity Cost Experiment of RAS

B Complexity Cost Experiment of NCC

Fig. 5. Complexity Cost Experiment

6 Conclusions

With the development and the ripeness of SaaS technique, it has become the main issue for SaaS model to protect the tenants' privacy. The paper puts forward policy-based Customized Privacy Preserving Mechanism for SaaS Applications. The Mechanism can describe the requirements of tenants' Customized Privacy Preserving based policy, and succeed in combining the privacy preserving with the performance of SaaS application processing through the data-blocking-based policy. It builds three-party secure interaction model to make the security of the privacy preserving policy.

Acknowledgement. We thank the anonymous reviewers of GPC for their feed-back on previous versions of this paper. This research was supported by the National Natural Science Foundation of China under Grant No. 61272241, the Natural Science Foundation of Shandong Province of China under Grant No. ZR2010FQ026, Independent Innovation Foundation of Shandong University under Grant No.2012TS074.

References

1. Liu, Q., Wang, G., Wu, J.: An Efficient Privacy Preserving Keyword Search Scheme in Cloud Computing. In: Proceedings of the 2009 International Conference on Computational Science and Engineering, vol. 02, pp. 715–720 (2009)
2. Sadeghi, A.-R., Schneider, T., Winandy, M.: Token-based cloud computing: secure outsourcing of data and arbitrary computations with lower latency. In: Acquisti, A., Smith, S.W., Sadeghi, A.-R. (eds.) TRUST 2010. LNCS, vol. 6101, pp. 417–429. Springer, Heidelberg (2010)
3. Kamara, S., Lauter, K.: Cryptographic cloud storage. In: Proceedings of the 14th International Conference on Financial Cryptograpy and Data Security, pp. 136–149 (2010)
4. Ananthi, S., Sendil, M.S., Karthik, S.: Privacy preserving keyword search over encrypted cloud data. In: Abraham, A., Lloret Mauri, J., Buford, J.F., Suzuki, J., Thampi, S.M. (eds.) ACC 2011, Part I. CCIS, vol. 190, pp. 480–487. Springer, Heidelberg (2011)
5. Hu, H., Xu, J., Ren, C., Choi, B.: Processing Private Queries over Untrusted Data Cloud through Privacy Homomorphism. In: Proc. the 27th IEEE International Conference on Data Engineering, ICDE 2011 (2011)
6. Cao, N., Wang, C., Li, M., Ren, K., Lou, W.: Privacy-preserving multi-keyword ranked search over encrypted cloud data. In: 2011 Proceedings IEEE INFOCOM, pp. 829–837 (2011)
7. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM 21(2), 120–126 (1978)
8. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of the 17th International Conference on Theory and Application of Cryptographic Techniques, pp. 223–238 (1999)
9. Hohenberger, S., Lysyanskaya, A.: How to securely outsource cryptographic computations. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 264–282. Springer, Heidelberg (2005)
10. Smart, N.P., Vercauteren, F.: Fully homomorphic encryption with relatively small key and ciphertext sizes. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 420–443. Springer, Heidelberg (2010)
11. Muntés-Mulero, V., Nin, J.: Privacy and anonymization for very large datasets. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 2117–2118 (2009)
12. YuBao, L., Zhilan, H., Jian, Y., Weic, F.: Harm decomposition- based data privacy protection method. Journal of Integrative Plant Biology 46(7), 1217–1225 (2009)
13. Xiaochun, Y., Yazhe, W., Bin, W.: Multi-sensitive faced privacy protection method. Chinese Journal of Computers 31(4), 574–587 (2008)
14. Zhang, K., Li, Q., Shi, Y.: Data privacy preservation during schema evolution for multi-tenancy applications in cloud computing. In: Gong, Z., Luo, X., Chen, J., Lei, J., Wang, F.L. (eds.) WISM 2011, Part I. LNCS, vol. 6987, pp. 376–383. Springer, Heidelberg (2011)

QoC-Aware Access Control Based on Fuzzy Inference for Pervasive Computing Environments

Yao Ma, Hongwei Lu, and Zaobin Gan*

School of Computer Science and Technology,
Huazhong University of Science and Technology,
Wuhan 430074, China
mayaobox@smail.hust.edu.cn, luhw@hust.edu.cn,
zgan@mail.hust.edu.cn

Abstract. In Pervasive Computing Environments (PCE), existing Context-Aware Access Control (CAAC) models mainly extend the RBAC model to realize the context awareness and take into account the uncertainty of the imperfect context information by excessive constraints of the QoC (Quality of Context) parameters or degrees. To solve the problems, we present a novel QoC-Aware Access Control (QAAC) model which applies fuzzy inference to make authorization decisions. The QoC-awareness is reflected in the modification of the provided context and the fuzzy inference process. Compared with existing work, the proposed model has a more comprehensive utilization of static and dynamic attributes, better adaption to the context dynamicity, feasibility in QoC-awareness and semantic expressiveness.

Keywords: Pervasive computing environment, Access control, Context-aware, Quality of context, Fuzzy inference.

1 Introduction

With the rapid development of the personal computing devices and sensors, peoples are surrounded with various computing devices (e.g., laptops, PDAs, and smart phones) and sensors (e.g., RFID readers, cameras, and infrared sensors). These devices and the related context-aware applications not only facilitate peoples' daily life, but also pose a challenge to the access control mechanism. The heterogeneity and dynamicity of the sensor-rich PCE call for the Context-Aware Access Control (CAAC) that can dynamically adjust permissions to the current changing *contexts* [1].

However, the existing context-aware access control models mainly extend the RBAC (Role Based Access Control) model [2] with context constraints. The RBAC model is initially designed for closed systems with static sets of known users and resources, which cannot meet the requirements of CAAC in PCE. Meanwhile, the QoC (Quality of Context) has a profound impact on the decisions of context-aware applications. In existing work, QoC is generally considered by constraints of the QoC parameters or degrees, which needs plenty of work to determine. Furthermore, it is infeasible

* Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 501–508, 2013.

to always satisfy all these constraints simultaneously, which makes the access control fails to give a definite authorization decision response.

Thus, based on the idea of Attribute Based Access Control (ABAC) [3], we further classify the attributes of entities into Static Attributes and Dynamic Attributes in order to have a comprehensive utilization of them and propose a novel QoC-Aware Access Control (QAAC) model that applies fuzzy inference to make authorization decisions.

The rest of the paper is organized as follows. Section 2 overviews the related work in CAAC. Section 3 presents the QoC-Aware Access Control model based on fuzzy inference. Section 4 gives a use case and the comparison. Finally, Section 5 concludes the paper and outlines the future work.

2 Related Work

Much research has been widely done in the field of context-aware access control for PCE. Some work tried to extend the RBAC model with context-relative roles or constraints. Park et al. [4] introduced proposed the Context-Role Based Access Control (CRBAC) which considers the activation and revocation of context roles as well user roles. Kulkarni et al. [5] tried to apply context constraints in role admission, validation and permission etc. Jung and Joshi [6] proposed a Community-centric Role Interaction Based Access Control model (CRiBAC) that utilizes a range of contexts from public ones such as ambient contexts to private ones belonging to individuals to guarantee dynamic and fine-grained access control at runtime. Nevertheless, these models lack dynamicity and flexibility for PCE due to the nature of RBAC.

Moreover, the importance of considering QoC in CAAC is recognized gradually. The Cerberus framework [7] associates confidence levels with different authentication mechanisms. Filho et al. [8] proposed a generalized QoC-Aware Context-Based Access Control model that takes into account the requestor's, owner's, and resource's context together with constrained QoC indicators of them to make access control decisions. The Proteus [9] exploits QoC as a filtering principle to discard context data with insufficient QoC and select applicable access control polices. These QoC-aware access control research introduces the QoC-awareness by constraining the QoC parameters or degrees with specified thresholds, which needs plenty of work to determine them in large number of policies. Meanwhile, it is rigid to meet the QoC requirements of every context in a policy, which makes the enforcement less infeasible.

3 QoC-Aware Access Control Based on Fuzzy Inference

3.1 Terminologies

Static Attributes and Dynamic Attributes

Based on the attribute classification in the ABAC model [3], we further classify the attributes of entities into Static Attributes and Dynamic Attributes according to their timeliness.

Static Attributes are the time-continuous security-relevant characteristics of entities such as the user's name and birth date. Static Attributes have a determinate lifetime, and they are issued by Attribute Authorities by attribute credentials. Attribute credentials can be acquired from the access requestor in push mode and/or the directory systems in pull mode and verified by the Source of Authority (SOA).

On the contrary, *Dynamic Attributes* are transient security-relevant characteristics of entities, such as the user's location and heart rate which are unpredictable and require frequent measuring and/or inferring. The *Dynamic Attribute* is considered synonymous with *Context* [1] in this paper because of the similarity. The Dynamic Attributes are provided by the Context Information Service Providers (CISP).

QoC Parameters and QoC Degree

In [10], *Quality of Context* is defined as any information that describes the quality of information that is used as context information. In order to show the worth of context information for an application, a quality measurement in terms of these QoC parameters is needed and noted as QoC degree. The different parameters have different weights in the calculation of the QoC degree.

Let *o* be a context object that has the context information about a real world entity. The context object *o* is deduced from the raw information collected by sensors which can be physical sensors or logical ones. We denote the value of the i^{th} QoC parameter as $QP_i(o)$ ($QP_i(o) \in [0,1]$) and the corresponding weight as $W_i(o)$ ($\sum_{i=1}^{n} W_i(o) = 1$), *n* is the number of QoC parameters. The QoC degree of *o*, noted as $QD(o)$, can be computed by $QD(o) = \sum_{i=1}^{n} W_i(o) \cdot QP_i(o)$.

3.2 QAAC Based on Fuzzy Inference

Mamdani's fuzzy inference method [11] is applied to make authorization decisions based on fuzzy categories of the static attributes and the dynamic attributes according to the access control policies in form of IF-THEN fuzzy rules. It is feasible to utilize the high-level context information in form of fuzzy sets because the approaches of fuzzy logic and probabilistic logic etc. are broadly used to deduce higher-level contexts or situations [12]. We present the QoC-Aware Access Control (QAAC) model and describe the basic elements as follows:

Static Attributes (*Sa*) and *Dynamic Attributes* (*Da*): the sets of the static attributes and the dynamic attributes of the related principals, resources and environments. For any attribute $attr \in Sa \cup Da$, it is provided as $\mu_{attr}(x)$. $Attrcat_{attr}$ is the set of fuzzy categories of the attribute. Each fuzzy category $attrcat_{attr} \in Attrcat_{attr}$ implies a specific membership function $\mu_{attrcat_{attr}}(x)$, where *x* is the value of the attribute *attr* mapped on the universe of discourse *X* ($x \in X$).

Resources (*Res*): the set of resources that are under the protection of access control. A resource can be data (e.g. files) or a service.

Operation (*Oper*): the set of operations that can be executed on the resource. Operations can be various, such as read and write.

Policy ($Policy_{ResOper}$): the set of policies that describe how the resource is protected against illegal access requests in a specified application scenario.

Authorization Decision ($Authdecs_{(res,oper)}$): the set of possible authorization decisions responding to the request $(res, oper)$. Each authorization decision $authdec \in Authdecs_{(res,oper)}$ implies a specific membership function $\mu_{authdec}(x)$.

In the QAAC model, under the specified resource operation combination circumstances $ResOper$ ($ResOper \in 2^{Res \times Oper}$), any requested resource operation combination $(res, oper) \in ResOper$ is protected by one access control policy $policy_{(res,oper)}$. Since $\forall (res, oper) \in ResOper : \exists | \, policy_{(res,oper)} \in Policy_{ResOper}$, the goal of the QAAC model is to deduce the authorization decision responding to the request $(res, oper)$ based on *Sa*, *Da* and $policy_{(res,oper)}$ by fuzzy inference.

The $policy_{(res,oper)}$ can be further described as:

$$policy_{(res,oper)} = (Rule_{(res,oper)}, Authsetting_{(res,oper)})$$

where the $Rule_{(res,oper)}$ is the set of IF-THEN fuzzy rules for the request $(res, oper)$, and $Authsetting_{(res,oper)}$ is the authorization setting of the authorization decisions.

$Rule_{(res,oper)}$ can be described as:

$$Rule_{(res,oper)} = \{ (ant(attrcats_i), cons)_i \}$$

where i indexes the rules ($1 \le i \le m$, m is the number of rules in the policy), $attrcats_i \in 2^{Attrcat_{attr_1} \times Attrcat_{attr_2} \times \ldots \times Attrcat_{attr_n}}$ $(attrcats_i \ne \emptyset)$ and $n = | Sa \cup Da |$. Each rule comprises an antecedent and a consequence. The antecedent $ant(attrcats_i)$ is an *AND*-conjunction of category constraints with optional negation operator *NOT*. The consequence *cons* is an authorization decision.

$Authsetting_{(res,oper)}$ is the authorization setting that comprises all the membership thresholds for each authorization decision for the request $(res, oper)$ and described as:

$$Authsetting_{(res,oper)} = (Authdecs_{(res,oper)}, Mt_{(res,oper)})$$

where $Mt_{(res,oper)}$ is the set of the membership thresholds assigned for each authorization decision and can be described as:

$$Mt_{(res,oper)} = \{ mt_{authdec} \mid 0 \le mt_{authdec} \le 1, authdec \in Authdecs_{(res,oper)} \}$$

Preprocessing of Static and Dynamic Attributes

Preprocessing of Static Attributes

Static Attributes are issued by Attribute Authorities in form of attribute credentials and of no uncertainty. To deal with the static attributes together with context information in fuzzy rules, the membership function of the constraint of static attribute can be preprocessed as

$$\mu_{attrcat_{attr}}(x) = \begin{cases} 1 & x \in X_{attrcat_{attr}} \\ 0 & otherwise \end{cases}$$

where $X_{attrcat_{attr}}$ is the set of values of *attr* belonging to $attrcat_{attr}$ mapped on the universe of discourse X ($X_{attrcat_{attr}} \subset X$). For example, the judgment of adults by age can be: $attrcat = 'adult' \in Attrcat$, $X_{adult} = \{ age \mid age \ge 18 \}$.

Preprocessing of Dynamic Attributes

Suppose that the context object o that has the context information ($attr \in DA$) is provided by a CISP as fuzzy set $\underset{\sim}{o}$ in form of membership function. Factoring in the QoC degree $QD(o)$, the modification function $M(QD(o), \mu_{\underset{\sim}{o}}(x))$ can be described as:

$$\overline{\mu}_{\underset{\sim}{o}}(x) = M(QD(o), \mu_{\underset{\sim}{o}}(x))$$

To give a specific operational modification method, we obtain the modified $\overline{\mu}_{\underset{\sim}{o}}(x)$ by scaling down $\mu_{\underset{\sim}{o}}(x)$ by $QD(o)$ in this paper and denote it as $\overline{\mu}_{\underset{\sim}{o}}(x) = QD(o) \cdot \mu_{\underset{\sim}{o}}(x)$

Thus, the value field of $\overline{\mu}_{\underset{\sim}{o}}(x)$ is $[0, QD(o) \cdot \max(\mu_{\underset{\sim}{o}}(x))]$. The changes of the physical proximity context and the corresponding QoC are illustrated in Fig.1 and Fig.2 by the solid red and blue curves. The universe of discourse for any context information discussed in this paper is normalized to the value field $[0,1]$.

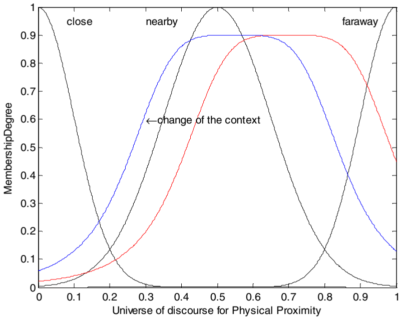


Fig. 1. Change of the context with constant QoC

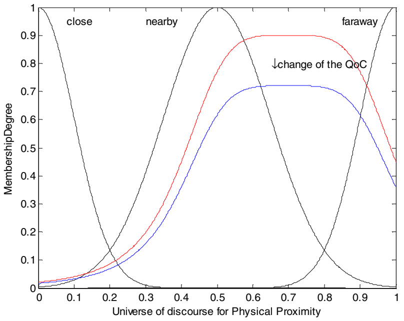


Fig. 2. Change of the QoC with constant Context

Fuzzy Inference Based on IF-THEN Rules

As the most common rule of composition, *MAX-MIN* composition is used here. The inferred output of each rule is a fuzzy set chosen from the minimum firing strength. For $rule_i$, the fuzzy output $\mu_{\underset{\sim}{d}_i}(y)$ can be represented as

$$\mu_{\underset{\sim}{d}_i}(y) = \alpha_i \wedge \mu_{authdec_i}(y)$$

where α_i is the "firing strength" and can be described as

$$\alpha_i = \min_{attrcat_{attr} \in attrcats_i} \left(\max_{attrcat_{attr} \in attrcats_i} (\mu_{attrcat_{attr}}(x) \wedge \mu_{attr}(x)) \right)$$

where $\mu_{attrcat_{attr}}(x)$ and $\mu_{attr}(x)$ are the membership functions of the category $attrcat_{attr}$ and the attribute $attr$ respectively. So the aggregated fuzzy output $\mu_{\underset{\sim}{d}}(y)$ on the universe of discourse for decision tendency can be represented as

$$\mu_{\underset{\sim}{d}}(y) = \bigvee_{i=1}^{m} \mu_{\underset{\sim}{d}_i}(y) = \bigvee_{i=1}^{m} [\alpha_i \wedge \mu_{authdec_i}(y)]$$

where $\vee$ and $\wedge$ are the *MAX* and *MIN* operators in fuzzy inference.

Determination of the Authorization Decision

To determine the authotiation decision, the aggregated fuzzy output $\mu_{\underset{\sim}{d}}(y)$ will be defuzzificated first. As a popular technique to defuzzificate the fuzzy value, the Center of Gravity (COG) is used to get the crisp value of the decision tendency y^* by $y^* = \int_{\min}^{\max} y \cdot \mu_{\underset{\sim}{d}}(y) / \int_{\min}^{\max} \mu_{\underset{\sim}{d}}(y)$. Suppose $Authdecs_{(res,oper)} = \{permit, deny\}$, the algorithm of determining authorization decision is illustrated as follows:

```
Input: the defuzzificated output $y^*$ , the authorization setting $Authsetting_{(res,oper)}$
Output: the authorization decision $authdec$
/*default authorization decision can be set to permit or deny
$default\_authdec \leftarrow 'deny'$
$mt_{permit} \leftarrow get\ from\ Authsetting_{(res,oper)}$
$mt_{deny} \leftarrow get\ from\ Authsetting_{(res,oper)}$
if $\mu_{permiy}(y^*) \geq \mu_{deny}(y^*)$ and $\mu_{permit}(y^*) \geq mt_{permit}$ then
    return $authdec \leftarrow 'permit'$ end if
if $\mu_{permit}(y^*) < \mu_{deny}(y^*)$ and $\mu_{deny}(y^*) \geq mt_{deny}$ then
    return $authdec \leftarrow 'deny'$ end if
return $authdec \leftarrow default\_authdec$
```

4 Case Study and Comparison

We consider the following scenario. Alice and Bob are not familiar with each other and their mobile devices have a Photo Album Service (PAS) application which allows users to store, view and organize their digital pictures. Bob is more willing to share pictures with the people physically closer to him when the available power of the laptop is more abundant. Moreover, as a Photography Club member, he has few constraints on the users belonging to the club. The constraints of the IF-THEN fuzzy rules for the resource operation combination $(picture, download)$ are listed in Table 1.

Table 1. Access control policies in form of fuzzy rules

ID	Antecedent				Consequent
	Connection	***Club Member***	***Physical Proximity***	***Available Power***	***Authorization Decision***
1	AND	(none)	Close	Much	Permit
2	AND	(none)	NOT Close	Little	Deny
3	AND	(none)	Faraway	Medium	Deny
4	AND	Satisfied	Close	Medium	Permit
5	AND	Satisfied	Nearby	Much	Permit

To give a visualized understanding of the above rules, we plot the defuzzificated Decision Tendency surfaces (the surface of y^*) in Fig. 3 and Fig.4, where the inputs of the dynamic attributes are represented as crisp values to realize visualization. It shows that the requestor tends to be granted a *permit* decision when he has the club credential.

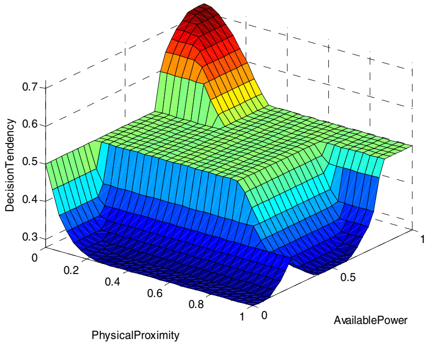


Fig. 3. Defuzzificated decision tendency surface without club credential

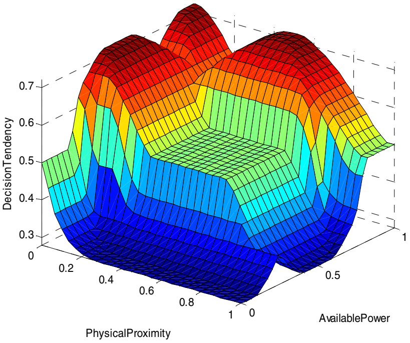


Fig. 4. Defuzzificated decision tendency surface with club credential

Compared with existing work, the advantages of our work are listed as follows:

- *Utilization of static and dynamic attributes*. The works [4]-[6] based on RBAC are inapplicable for PCE due to the nature of RBAC. In this paper, we comprehensively classify the static attributes and dynamic attributes from the concept of ABAC and distinguish the two manners of attributes acquirement and verification, which makes the access control flexible and fine-grained.
- *Dynamicity and uncertainty of context*. In [8][9], the contexts are in form of crisp values which looses the uncertain information of context. Constraints on crisp contextual value may make the privilege granted intermittent when at least one concerned context value floats up and down around the marginal value. We utilize high-level contexts in form of fuzzy sets with fuzzy rules, which can avoid the ignorance of uncertainty information and make the decision adapt smoothly to the change of context and QoC.
- *Feasibility in QoC-awareness*. In [7]-[9], it needs much work to determine the QoC degree or QoC parameter constraints in vast policies. Meanwhile, it is rigid to meet all the QoC requirements in a policy simultaneously. In our work, the QoC-awareness is realized by two parts: the modification of the context membership function considering the QoC degree; the COG of the aggregated fuzzy output, which makes the setting and enforcement of more feasible.
- *Semantic expressiveness*. The contexts in form of crisp value are constrained by specific domains in existing work. These constraints are lack of semantics and not reusable for different policies. The fuzzy linguistic categories in our work can describe the ambiguous constraints of attributes and be reusable.

5 Conclusions and Future Work

In this paper, we present a novel QoC-Aware Access Control (QAAC) model, where the QoC-awareness is reflected in the modification of the provided context and the extent to which the defuzzificated decision tendency support the authorization decision. The comparison with existing work shows that our work has a more comprehensive utilization of static and dynamic attributes and better feasibility in context-awareness and QoC-awareness. For future work, we plan to consider the uncertainty caused by the trustworthiness of the attribute credential issuers and CISPs, and evaluate the performance of the work for a large number of users and policies.

Acknowledgment. This work is supported by National Natural Science Foundation of China under grant No. 61272406.

References

1. Dey, A.K.: Understanding and Using Context. Personal and Ubiquitous Computing 5(1), 4–7 (2001)
2. Ferraiolo, D.F., Sandhu, R., Gavrila, S., et al.: Proposed NIST Standard for Role-based Access Control. ACM Trans. on Info. and Sys. Sec. 4(3), 224–274 (2001)
3. Yuan, E., Tong, J.: Attributed Based Access Control (ABAC) for Web services. In: Proc. of the IEEE Intl. Conf. on Web Services, pp. 561–569. IEEE CS, Washington (2005)
4. Park, S.H., Han, Y.J., Chung, T.M.: Context-role based access control for context-aware application. In: Gerndt, M., Kranzlmüller, D. (eds.) HPCC 2006. LNCS, vol. 4208, pp. 572–580. Springer, Heidelberg (2006)
5. Kulkarni, D., Tripathi, A.: Context-aware Role-based Access Control in Pervasive Computing Systems. In: Proc. of the 13th ACM Symp. on Access Control Models and Technologies, pp. 113–122. ACM, New York (2008)
6. Jung, Y., Joshi, J.B.D.: CRiBAC: Community-centric Role Interaction Based Access Control Model. Computers & Security 31, 497–523 (2012)
7. Al-Muhtadi, J., Ranganathan, A., Campbell, R., Mickunas, M.D.: Cerberus: a Context-aware Security Scheme for Smart Spaces. In: Proc. of the 1st IEEE Intl. Conf. on Pervasive Computing and Communications, pp. 489–496. IEEE CS, Los Alamitos (2003)
8. Filho, J.B., Martin, H.: A Generalized Context-based Access Control Model for Pervasive Environments. In: Proc. of the 2nd SIGSPATIAL ACM GIS 2009 Intl. Workshop on Security and Privacy in GIS and LBS, pp. 12–21. ACM, New York (2009)
9. Toninelli, A., Corradi, A., Montanari, R.: A Quality of Context-aware Approach to Access Control in Pervasive Environments. In: Bonnin, J.-M., Giannelli, C., Magedanz, T. (eds.) Mobilware 2009. LNICST, vol. 7, pp. 236–251. Springer, Heidelberg (2009)
10. Buchholz, T., Küpper, A., Schiffers, M.: Quality of Context: What it is and Why We Need it. In: Proc. of the 10th Intl. Workshop of the HP OpenView University Association, Geneva, Switzerland (2003)
11. Mamdani, E.H., Assilian, S.: An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. Intl. Journal of Man-Machine Studies 7(1), 1–13 (1975)
12. Bettini, C., Brdiczka, O., Henricksen, K., et al.: A Survey of Context Modelling and Reasoning Techniques. Pervasive and Mobile Computing 6(2), 161–180 (2010)

Per-File Secure Deletion Combining with Enhanced Reliability for SSDs

Yi Qin[1], Dan Feng[1], Wei Tong[1,*], Jingning Liu[1], Yang Hu[2], and Zhiming Zhu[1]

[1] Wuhan National Laboratory for Optoelectronics
[1] School of Computer Science and Technology
[1] Huazhong University of Science and Technology, Wuhan, China
qinyihust@gmail.com, dfeng@hust.edu.cn,
{weitong, j.n.liu}@163.com, zhuzhiming1210@126.com
[2] China ship development and design center, Wuhan, China
yanghu@foxmail.com

Abstract. Flash memory based Solid State Drives(SSDs) become an indispensable part in mobile computers. To protect confidential data from being leaked, user have to run secure deletion software to erase the confidential files. However the traditional secure deletion software may report success on SSDs, but do not work at all. To solve the problem, we proposed a per-file secure deletion method to clean up the sensitive data without erase the whole SSD. In addition, RAID technique is also employed to enhance the reliability and eliminate the potential risk caused by secure deletion.

Keywords: secure deletion, reliability, SSD.

1 Introduction

As the storage medium of SSDs, NAND Flash memory is organized by chip, die, plane, block and page. The operation pattern is that read and write in pages, but erase in blocks. In general, the write operation should not be processed until the erase operation changes all the bits in the block from 0 to 1. Also, every page in a block can only be written once. Moreover, the erase count of a block is limited (normally 10000 for MLC flash, and 100000 for SLC flash). So, SSDs usually employ the flash translation layer (FTL, [1, 2, 3, 4]) to shield the complex characteristic of flash memory from host system.

Known as out-of-place update, FTL in all SSDs write update data of any logical sector to a new allocated page. At the same time, the page with old data will be marked as "invalid" and remain unchanged. After that, the upper file system cannot directly visit that physical page again. Such mechanism is a hidden danger for the secure deletion process.

On the one hand, when the users runs any file erase software to secure erase a confidential file, things will different from expected. The file erase software covers the

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 509–516, 2013.

confidential data several times with some certain data. However, the new data is written to other free pages in the flash memory, and leave confidential data unchanged. Afterwards the file erase software may finish work and report "success". This will lead to grave consequences that the users believe the erasing is successful. They will never be aware of the full remanence of confidential data.

On the other hand, when the users edit a confidential file, some parts of the file may be modified. Because of out-of-place update, the corresponding pages that contain old data in flash memory will be abandoned. In later file erase process, these pages will not be mentioned. These will cause leaking of partial confidential data.

To solve these problems, Redundant Arrays of Inexpensive Disks (RAID, [5]) technique should be deployed. We purpose a secure deletion scheme combining with RAID-5 architecture for SLC flash based SSDs, which can fully ensure both reliability and security. In such method, intelligent partition detection is deployed to make a difference between disk partitions. Afterwards, the reprogram attached update mode is only applied to the specified partition to reduce performance penalize. In this paper, we first introduce the exist methods of secure deletion, and then propose the scheme of secure deletion combining with RAID-5 architecture. At last, we evaluate the cost on performance of secure deletion with enhanced reliability.

2 Related Works

In order to eliminate the effect of out-of-place update on secure deletion, Wei [6] proposed a new update method—scrubbing, which reprogram the physical page from 1s to 0s. A number of SLC and MLC chips have been tested for data scrubbing. The result shows that scrubbing causes no error for SLC chips. For MLC chips, pages must be scrubbed in pair, and scrubbing causes different data error according to the number of reprograms in a block. However, this method has not been proved reliably even for SLC chips. There may be some certain errors that can only be observed in some particular environments over a long period of time.

In 2012, Diesburg [7] present TureErase, a per-file secure deletion in full storage data path. To prevent data leaking, it design and implement secure deletion from application layer down to hardware layer. A Secure-deletion user model is added between application and VFS layer. Then, a type/attribute propagation module is embedded in kernel space, to gain information from VFS layer and file system in time. Also, the storage-management layer is enhanced to implement secure deletion for storage medium.

With the rapid development of high integration density of flash memory, reliability has become an important element for SSDs. Texas Memory Systems [8] expounds the reliability situation of NAND flash based SSDs. It summarizes the reliability from chip level to board level, and data center architecture level. As its viewpoint, in chip level, NAND flash memories suffer from the severe limited endurance, especially MLC Flash. In system level, RAID architecture of flash memory is effective for data reliability for SSDs.

Greenan [9] present a multilevel architecture, which employ data redundant in page level, block level and board level. The proposed multilevel RAID can ensure the data reliability at a high level.

More recently, Im [10] present a RAID-5 architecture to enhance the reliability of flash memory SSDs. They put forward the delayed parity update and partial parity caching techniques, which can reduce the cost of parity updates. The partial parity caching technique can also help when recovering the failed data without full parities.

3 Methodology

In this section, a per-file secure deletion scheme combining with RAID 5 architecture for SSDs is proposed. In our scheme, update process in the hardware layer of SSDs is improved. A reprogram operation is attached to a normal update process, in order to eliminate data leaking.

As shown in Fig. 1, the abandoned page in every update operation in flash memory is reprogrammed to all zeros.

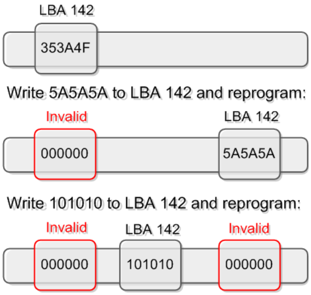


Fig. 1. Reprogram attached update mode

When reprogram attached update mode is applied to SSDs, the main potential danger of data leaking is disappeared. First, the update operation no longer leaves behind outdated confidential data as "invalid page" in flash memory. Second and the most important, traditional secure deletion software are able to shred confidential file completely. Thus, user can efficiently secure delete confidential files by running traditional secure deletion software.

Usually, a general SSD contains a flash array with dozens of chips of flash memory. (Normally a flash memory package includes 1 to 4 chips of flash memory.) SSDs make interleaving access to the flash memory array from independent channels. It is inevitable for flash memory that, some physical units accidentally damaged. For example, some pages are not accessible, or block erase is failed, etc. As small probability event, a whole chip may be failed to work. When such unit failure happens, a lot of data must be damaged.

Moreover, the reprogram operation obviously increases the program duty of physical pages in flash memory. This may exercises subtle influence on endurance and lifetime of flash memory. In such situation, we should take measures to eliminate the potential risk. Thus, a RAID-5 architecture to enhance the reliability should be taken into consider.

In this section, a hybrid architecture for SSDs, which includes reprogram attached update and RAID-5 architecture, will be purposed to achieve secure deletion combining with enhanced reliability. To offer data recovery from one point failure, the RAID-5 architecture generates parity page for each data strips, which consist of corresponding logical pages in each channel. As shown in Fig. 2, when a data page is updated, the old physical page is reprogrammed to all zeros. Besides, the corresponding parity page is updated.

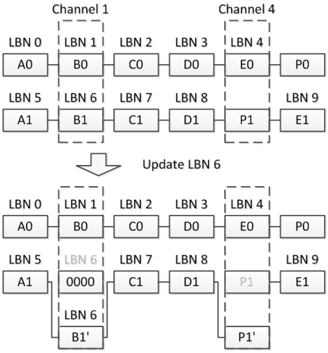


Fig. 2. Secure deletion scheme with enhanced reliability

Suppose there are six channels in a SSD, then there are six physical pages in a strip. The parity page is calculated by XOR all the pages in the corresponding strip, and stored equally in six channels by turns. Generally, P_n=$A_n \oplus B_n \oplus C_n \oplus D_n \oplus E_n$. In the expression, P_n represent for parity page of strip n, A_n means physical page number n of channel 0, and B_n indicate the physical page number n of channel 1, etc.

The redundant parity architecture takes effect when any unit failure occurs. If a parity page failure takes place, the SSD may continue regular work and rebuilds the affected parity page in background. Else if a data page failure occurs, the original data can be recovered from the associated pages in the same strip. For example, if E_n is broken accidentally, data can be recovered by XOR operations: E_n= $A_n \oplus B_n \oplus C_n \oplus D_n \oplus P_n$. Afterwards, the recovered data will be written to other good pages. In this architecture, the SSD can recover from continuous page or block failure, except two pages in a strip failed in the same time.

3.1 Intelligent Partition Detection

Reprogram attached update mode is efficient in reliability but obviously adverse to performance. Because it doubles each write operation. To decrease such negative effects, intelligent partition detection is employed to FTL in SSDs.

In a general way, the space of a hard drive is divided to several partitions. The first sector of hard drive holds the MBR (Master Boot Record), which contains the DPT

(Disk Partition Table). As shown in Fig. 3, there are four records in the DPT, each of which indicate the type and address of a primary partition (or extend partition). In extend partition there are several logical partitions, in head sector of which contains a partition table and indicate the address of next partition. By monitoring the DPT in the first sector and the subsequent partition tables, it is feasible for FTL to detect the logical range of all partitions of SSDs. Then the users can define and mark a partition as "confidential partition". Afterwards, the SSD can only apply reprogram attached update mode to the specified partition. In this way, the performance penalize can be minimized.

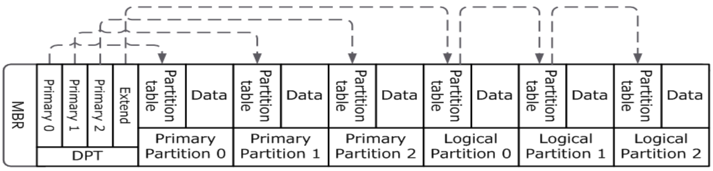


Fig. 3. Partition table detection

3.2 Parity Cache

Once a logical page in a strip is updated, the specified parity page should also be updated. In some hot area of file system (such as FAT, File Allocation Table), the logical page may be frequently updated. As a result, the corresponding parity page may be updated for dozens of times in a second. If the SSD directly write the latest parity pages to flash memory every time they are updated, the endurance of flash memory will be exceeded in no time. In consideration of this, a parity cache is employed to reduce the program access to flash memory.

In this scheme, an NVRAM is deployed and act as parity cache to buffer parity updates. Each node of parity cache contains a parity page and some metadata. The metadata contains strip index, LRU pointer, binary tree pointer and other flags. The nodes of parity cache are chained by both LRU (Least Recently Used) queue and AVL tree.

When a parity page is updated, the parity cache node will be created. Then it is added (or updated) to the corresponding point of the AVL tree (search by strip index) and the head of the LRU queue. If the parity cache is full at the same time, a swap out will be taken place. The node in the tail of the LRU queue will be removed from parity cache, and be written to flash memory.

3.3 Background Operation

In this scheme, the parity writing operation and reprogram operation are time-costing. To reduce the influence to the normal I/O operation, the two operations can be running in background. In that way, the parity writing and reprogram operation can be processed asynchronous. This mode may obviously take effect on reducing the write penalize.

4 Evaluation

To evaluate the influence to the performance, we implement our secure deletion scheme on an open-source simulator—SSDsim [11, 12]. In the simulator, we suppose the SSD is consisted of 8 data channels. The configuration of flash memories in each channel is shown in table 1.

Table 1. Configuration of flash memories in each channel

Entry	Configuration	Entry	Configuration
Chips per channel	4	Blocks per plane	2048
Dies per chip	2	Pages per block	64
Planes per die	2	Page capacity	2KB

To simulate the work load in real environment, three disk I/O traces is employed to simulate the I/O request to the SSD—Financial1, ExchangeAM, MSN [13]. The Financial1 and ExchangeAM trace are collected at a financial institution and a network service situation respectively. And MSN is collected on Microsoft's MSN Storage servers.

The platform is simulated in six modes—no parity and no reprogram, normal parity and no reprogram, normal parity and normal reprogram, background parity and intelligent reprogram, background parity and background reprogram, background parity and background intelligent reprogram. We call them NPNR, PNR, PR, PbRi, PbRb, PbRbi for short respectively.

To evaluate intelligent partition detection, we suppose the first partition, which capacity is set to a quarter of the total logical sector range of each trace, has been set to "secure partition". For traces of which logical sector range exceeds the SSD's total capacity, the secure partition capacity is set to a quarter of the SSD's capacity. In intelligent reprogram mode, the reprogram attached update mode is only applied to the "secure partition".

Beyond all, the read performance of the secure deletion scheme is evaluated for SSDs. Fig. 4 shows the read response time for the SSD under six modes for three different traces. The result is distinct that the proposed secure deletion scheme shows little impact on the read performance of SSDs. Only under the ExchangeAM trace, the read response time rises about 4%.

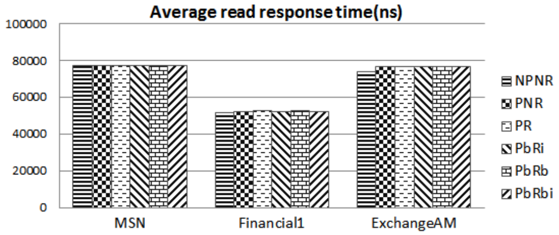
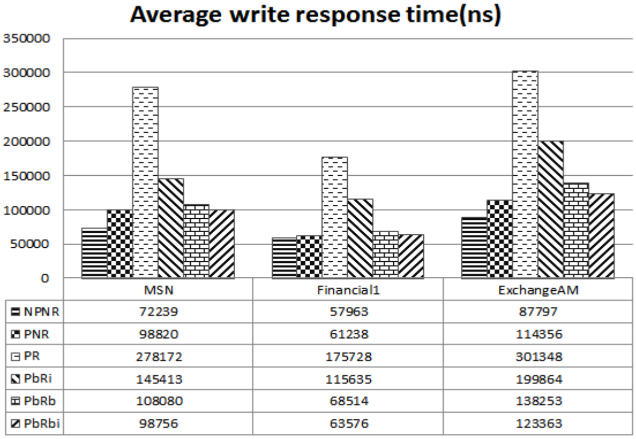


Fig. 4. Average read response time for three traces

The write penalty of six modes of our secure deletion scheme is shown in Fig. 5. The trend of the write response time in six modes is similar under three traces. The write response time in RP mode is the highest, which is almost 3 times on the NPNR mode. Compare to PR mode, the PbRb mode reduce the write response time by above 60% under MSN and Financial1 trace, and 54% under ExchangeAM trace. While, the PbRi mode reduce it by about 50% under MSN trace, and above 30% under the other two traces. This contrast makes clear that both background operation and intelligent reprogram can greatly reduce the write penalty. Finally, in PbRbi mode, the write response time is almost the same as the PNR mode. In a sense, the write performance of our secure scheme is on the same level of normal RAID-5 architecture. In other words, the negative effect of reprogram is shielded by background operation and intelligent partition detection.



	MSN	Financial1	ExchangeAM
▤ NPNR	72239	57963	87797
▦ PNR	98820	61238	114356
▭ PR	278172	175728	301348
▧ PbRi	145413	115635	199864
▥ PbRb	108080	68514	138253
▨ PbRbi	98756	63576	123363

Fig. 5. Average read response time for three traces

5 Conclusion

To secure delete confidential file from SSDs, we implement an efficient per-file secure deletion scheme. A reprogram attached update mode is proposed to prevent data leaking. And that user can process common erasing software to secure delete their confidential file. Furtherly, RAID-5 architecture is employed to enhance the reliability and eliminate the negative effect of reprogram on flash memory. In the meantime, the performance of our secure deletion scheme is attractive. Give the credit to background operation and intelligent partition detection, the performance is controlled at the level of normal RAID-5 architecture.

Acknowledgements. This work is supported by the National Basic Research 973 Program of China under Grant No.2011CB302301, the National Science Fund for Distinguished Young Scholars No.61025008, the National Natural Science Foundation of China No. 60933002, No.61232004, and No.61173043, the National High

Technology Research and Development Program ("863"Program) of China under Grant No.2013AA013203 and No.2012AA012403, the National Key Technology R&D Program No.2011BAH04B02, and the Electronic Development Fund of Information Industry Ministry.

References

1. Hu, Y., Jiang, H., Feng, D., Tian, L., Zhang, S., Liu, J., Tong, W., Qin, Y., Wang, L.: Achieving page-mapping ftl performance at block-mapping ftl cost by hiding address translation. In: Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST 2010, pp. 1–12 (2010)
2. Gupta, A., Kim, Y., Urgaonkar, B.: Dftl: a flash translation layer employing demand-based selective caching of page-level address mappings. In: Proceedings of the 14th International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2009, pp. 229–240 (2009)
3. Shin, J.Y., Xia, Z.L., Xu, N.Y., Gao, R., Cai, X.F., Maeng, S., Hsu, F.H.: Ftl design exploration in reconfigurable high-performance ssd for server applications. In: Proceedings of the 23rd International Conference on Supercomputing, ICS 2009, pp. 338–349 (2009)
4. Lee, S.W., Park, D.J., Chung, T.S., Lee, D.H., Park, S., Song, H.J.: A log buffer-based flash translation layer using fully-associative sector translation. ACM Trans. Embed. Comput. Syst. 6(3) (July 2007)
5. Chen, P.M., Lee, E.K., Gibson, G.A., Katz, R.H., Patterson, D.A.: Raid: high-performance, reliable secondary storage. ACM Comput. Surv. 26(2), 145–185 (1994)
6. Wei, M., Grupp, L.M., Spada, F.E., Swanson, S.: Reliably erasing data from flash-based solid state drives. In: Proceedings of the 9th USENIX Conference on File and Stroage Technologies, FAST 2011, p. 8. USENIX Association, Berkeley (2011)
7. Diesburg, S., Meyers, C., Stanovich, M., Mitchell, M., Marshall, J., Gould, J., Wang, A., Kuenning, G.: Trueerase: Per-file secure deletion for the storage data path. In: ACSAC 2012. ACM, Orlando (2012)
8. Hutsell, W., Bowen, J., Ekker, N.: Flash solid-state disk reliability. Texas Memory Systems White Paper (2008)
9. Greenan, K., Long, D.D.E., Miller, E.L., Schwarz, T., Wildani, A.: Building flexible, fault-tolerant flash-based storage systems. In: Proceedings of the Fifth Workshop on Hot Topics in System Dependability, HotDep 2009 (June 2009)
10. Rossi, D., Metra, C., Riccò, B.: Fast and compact error correcting scheme for reliable multilevel flash memories. In: Proceedings of the Eighth IEEE International On-Line Testing Workshop, IOLTW 2002 (2002)
11. Hu, Y., Jiang, H., Feng, D., Tian, L., Luo, H., Zhang, S.: Performance impact and interplay of ssd parallelism through advanced commands, allocation strategy and data granularity. In: Proceedings of the International Conference on Supercomputing, ICS 2011, pp. 96–107 (2011)
12. Hu, Y.: SSDsim (2012), `http://storage.hust.edu.cn/SSDsim/`
13. SNIA: Block I/O traces, `http://iotta.snia.org/tracetypes/3`

An Energy-Aware Secured Routing Protocol for Mobile Ad Hoc Networks Using Trust-Based Multipath

Isaac Woungang[1], Sanjay Kumar Dhurandher[2], and Michael Sahai[1]

[1] Department of Computer Science
Ryerson University, Toronto, Ontario, Canada
iwoungan@scs.ryerson.ca, msahai@ryerson.ca
[2] Division of Information Technology
Netaji Subhas Institute of Technology, University of Delhi, India
dhurandher@gmail.com

Abstract. Message security in multi-hop infrastructure-less networks such as Mobile Ad Hoc Networks has proven to still be a challenging task. A number of trust-based secure routing protocols have recently been introduced which comprise the traditional route discovery phase and a data transmission phase. In the later, the action of relaying the data from one mobile node to another relies on the peculiarity of the wireless transmission medium as well as the capability of source nodes to keep their energy level at an acceptable and reasonable level, posing another concern which is that of energy efficiency. This paper proposes an Energy-aware Trust Based Multi-path secured routing scheme (E-TBM) for MANETs, based on the dynamic routing protocol. Results show that our E-TBM scheme outperforms the Trust Based Multi-path (TBM) secured routing scheme - chosen as benchmark - in terms of energy consumption in the selected routing paths, and number of dead nodes, chosen as performance metrics.

1 Introduction

A mobile ad-hoc network (MANET) is a collection of highly wireless mobile nodes organized to create a temporary connection between them to forward the data, without any pre-established network infrastructure or extraneous hardware to assist in this communication. To fulfill this capacity, some form of collaborative or corporately multi-hop strategy is required to happen between the mobile nodes, which may not necessary prevail since misbehaving nodes could be part of the current set of MANET nodes. Therefore, securing the message delivery in MANETs is a key concern.

Typically, the routing mechanism involves two steps, namely the route discovery phase and the actual data transmission phase using the discovered secured route. The former relies on the underlying targeted routing protocol (in this case, we use trust-based multi-path DSR). On the other hand, the later involves investigating the peculiarities of the wireless transmission medium used as well as determining the required battery level of the source nodes involved in the data transmission process. Indeed, when performing the data transmission, it is essential that the nodes (here referred to as battery operated computing devices) that carry the operation be energy conserving

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 517–525, 2013.

so that their individual battery life can be prolonged, and the maximum lifetime of the network be achieved. These facts have led to the consideration of energy-efficiency as another important design aspect that should be taken into account in the routing decision, the goal being to achieve secure routing while lowering the network overall power consumption and number of dead nodes; where a dead node is defined as a node which has completely depleted its power level. This paper adds energy considerations into our recently proposed message security scheme in MANETs (so-called Trust Based Multi-path message security (TBM)) [1], in order to strengthen its design. Typically, the route discovery and selection algorithm in [1] is substantially modified to take into consideration the energy level of the selected routing paths while maintaining their security and trust levels, resulting to our so-called Energy-Aware Trust Based Multi-path message security protocol (so-called E-TBM). The modification consists in assigning a power-aware metric [2] to each node involved in the selected routing paths so as to quantify the amount of energy consumed by the node, thereby determine the energy consumption necessary to maintain an acceptable level of message security in the network. Our E-TBM approach consists of a combination of trust assignment mechanism, soft-encryption technique, and multi-path DSR-based routing, where the decision on the routing selection paths is energy constrained.

The rest of the paper is organized as follows. In Section 2, we present some related work. In Section 3, the proposed E-TBM approach is described in-depth. In Section 4, simulation results are presented. Finally, Section 5 concludes our work.

2 Related Work

Secured routing protocols for MANETs have been the subject of interest to the research community in the recent years. These protocols have been designed to satisfy the primary principles of network security, i.e. confidentiality, integrity, and availability, each having its own dynamics for achieving such goal. Approaches that have been proposed include: credit-based schemes; cryptographic-based methods; reputation-based schemes; methods specifically designed to protect the route discovery process; message security schemes based on trust-based multi-paths using conventional routing protocols, and others [1].

In this paper, our focus is on message security schemes based on trust-based multi-paths routing, where energy constraint is directly embedded in the design approach. Apart from relying on the proper selection of hardware, such approach must also involve the study of coupling among layers of the system since energy consumption does not occur only through transmission, but also through processing. Following this trend, representative energy-aware secured routing schemes for MANETs follows.

In [3], Sheng et al. introduced a DSR-based energy efficient routing protocol for MANETs (called NCE-DSR) which uses the number of times that a node sends messages as a parameter for deciding on the inclusion of this node in the selected routing path. A routing cost function is designed for determining the choice of the routing path. However, the overhead generated from this method is not revealed. In [4], Vadivel and Bhaskaran proposed an energy-efficient and secured routing protocol (called Intercept detection and correction (IDC)) for MANETs. The IDC algorithm identifies the malicious nodes by recognizing the selective forwarding misbehavior from the normal channel losses by means of a residual energy parameter. However, no clue is provided as to how this energy related parameter is determined. In [5], Babu proposed an energy-based secure authenticated routing protocol (called EESARP) for MANETs. The EESARP scheme uses an attack resistant authentication combined with hop-by-hop signatures to mitigate the routing misbehavior of potential malicious nodes while improving the reliability of the route request packet. In [6], Taneja and Kush proposed an energy-efficient and authentic routing protocol (called EESSRP) for MANETs which incorporates security (by means of hash key generation and Diffie-Hellman protocol) and power features in its design. In [7], Banerjee et al. proposed a trust based multipath OLSR routing protocol for MANETs (called ESRP) where trust is established by means of a signed acknowledgement based on asymmetric key cryptography. Unlike these schemes, our proposed E-TBM scheme is a mimic of our recently TBM scheme [1], where energy consumption at each node is incorporated within the route selection phase to decide on the secure route to transfer the message.

3 The Energy-Aware Trust Based Multi-path Message Security Scheme

Assuming that a source node, say S, wishes to transmit a message, say *m,* to a destination node, say D, our E-TBM approach follows the same steps as the TBM approach [1] to securely send the message. The method consists of a combination of message encryption, message routing using DSR, and message decryption as follows.

A. Message Encryption

At node S, the message *m* is segmented into four blocks *a, b, c,* and *d* and encrypted using soft-encryption. Typically, a XOR operation on bits is used, producing the message parts *a', b'*, *c'*, and *d'* as follows [1]:

$$a' = a\ XOR\ c,\ b' = b\ XOR\ d,\ c' = c\ XOR\ b,\ and\ d' = d\ XOR\ a\ XOR\ b \qquad (1)$$

B. Message Routing Using DSR

This step combines a trust mechanism and an enhanced DSR-based routing technique to securely transfer the encrypted parts a', b', c', and d'. The details are as follows.

Trust Mechanism: A node observes each of its neighbors to which its packets can be transferred then it assigns a discrete trust value in the range [-1, 4] to every neighbor based of the acknowledgements of the packets that it received and the trust recommendations from its peers [1]. These values are taken into account when making the decision to route the packets using DSR. When doing so, the trust defined strategy consists of the policy that a node with a certain trust assigned level *t* can be given the right to read and forward at most *t* parts of the message.

Routing Strategy: When a source node needs to route a message to a destination node, a route request (RREQ) packet is broadcasted. If a neighbor node that replies to the RREQ has the route to the destination or if the packet reaches the destination node, a route reply (RREP) is sent back to the source node acknowledging a successful delivery. In the packet header, the RREP message and trust levels of the previous nodes involved in the packet forwarding are sent backwards along the routing path selected by DSR. The current battery level (energy) of a node (computed as shown in Equation (2) – obtained from [2]) is added to the packet header:

$$Ej(t) = Ej(0) - \left(\sum_{t=0}^{Gj(t)} Cp(\tau) + CT(\tau)\right) - \left(\sum_{t=0}^{Xj(t)} (CR(\tau) + Cp(\tau))\right) - \left(\sum_{t=0}^{Rj(t)} [CR(\tau) + Cp(\tau) + CT(\tau)\right) \tag{2}$$

where *Gj(t)* is the number of packets generated by node j up to time *t; Xj(t)* is the number of packets received by node j up to time *t; Rj(t)* is the number of packets relayed by node *j* up to time *t; Cp(*τ*), CT(*τ*), and CR(*τ*)* are the processing power cost, transmitting power cost, and receiving power cost of packet τ respectively. The E-TBM algorithm finds the secure routes from a set of given routes as follows:

1. When a new route is found, these routes are arranged in the increasing order of their hop count. Two counters are set, one to keep track of the selected nodes in the routing paths, the other to keep track of nodes energy values.
2. The first route is selected and it is assumed that the maximum number of message parts that can be routed through it have been routed. No actual routing is done at this step.
3. The next route is selected and it is assumed that the maximum number of message parts that can be routed via have been routed. If all the parts of message can be routed securely, the actual routing is done by using the selected paths.
4. If four paths have been selected out of all possible combinations of paths, arrange these paths by the energy it would be required to send the data
5. Select the path that has the smaller energy path value. Out of the remaining paths, use the next lowest path energy, and so on.
6. Repeat this process until secured routes are found.
7. If no secured routes are found, the algorithm is repeated by starting at Step 2, by selecting second route as the first route.
8. This algorithm is repeated until all the combinations of the paths are exhausted. If no secured route is found, the algorithm waits for another route. If all routes have been found or a specific time interval has expired, it is assumed that the algorithm has failed.

The above process for selecting the secured routes is captured in Fig. 1.

```
Arrange the paths $P=P_1, P_2,..., P_n\}$ in increasing order of path length
Initialize Count $C_j$ for all nodes = 0
Initialize Count $E_j$ for all nodes Energy to 0
Select the smallest path from P {
            Select the next smallest path
                 if(for all selected nodes j, $C_j <= T_j$){
                        if( four paths selected){
                          if(for all selected nodes j,
               $E_j \leq Threshold\_E_j$){
                                     /* $Th\_E_i$ is a threshold on the battery power of the node. $E_i$ is cal-
                                     culated using Equation (2) */
                                     Select path with smaller energy value
                          }
             Select next smallest path with lowest
                          energy
               else
                                     continue;}
         if(all paths are exhausted)
                        Wait for another path
   }
   if (no paths left)
   Print("Not possible to route securely")
```

Fig. 1. Algorithm to select secure routes

C. Message Decryption

At the destination node D, the encrypted message parts a', b', c', and d' are decrypted to recover the original message m as follows [1]:

$$a = b' \text{ XOR } d', \; b=a' \text{ XOR } b' \text{ XOR } c' \text{ XOR } d',$$
$$c = d' \text{ XOR } b' \text{ XOR } d', \; d = d' \text{ XOR } c' \text{ XOR } d' \quad (3)$$

4 Performance Evaluation

A. Simulation Tool and Parameters

To compare the E-TBM scheme against the TBM scheme, we use the GloMoSim simulation tool [10], where our soft encryption using multiple message parts is implemented at the application layer. We also assume that the trust levels of nodes are available to the source nodes. The remaining simulation setup is given in Table 1.

Table 1. Simulation parameters

Parameter	Setting
Terrain dimension	2000 m x 2000 m
Number of nodes	Variable and placed uniformly throughout the terrain dimension
MAC protocol	IEEE 802.11
Radio transmission power	Variable and depends on the number of nodes used.
Traffic Type	CBR
Simulation Time	600 s
Initial battery power of each node	5000 Joules

The following performance metrics are considered: (1) *Route selection time* – i.e. the total time required for the selection of a routing path, and (2) *trust compromise* – i.e. the sum of access violation in all the paths selected for routing. The access violation at a node *n* is defined as the difference between $n_{parts,}$ the number of encrypted message parts that *n* has received and $T_{n,}$ the trust level of *n* if $n_{parts} \geq T_n$, i.e. if N_p is the set of nodes in a routing path *p*, the trust compromise for path *p* is:

$$TrustCompromise_p = \Sigma_{n \in Np} (n_{parts} - T_n), \qquad (4)$$

wherever $n_{parts} \geq T_n$. and T_{n} is the trust assigned to node *n* and $\mathrm{n}_{\mathrm{parts}}$ is the number of encrypted message parts received by node *n* from all the paths. The aggregate trust compromise is calculated for all the paths selected for routing. It has been demonstrated [1] that the trust compromise of the selected paths in the T-EBM scheme is always equal to zero; (3) the *number of dead nodes:* a dead node is defined as a node which has completely depleted its power level. When a node is drained of all its available power, it no longer plays a role in the route selection process; (4) The *total energy consumed by the selected routing paths*: This is the energy consumed by the nodes that are chosen to be part of the selected routing paths.; (5) The *total energy consumed in the network*: This is the energy consumed by all the nodes in the network, regardless of their involvement in the route selection process.

B. Simulation Results

The trust compromise for the E-TBM and TBM schemes are presented in Fig. 2. As expected, regardless of the number of nodes, the total trust compromise of both schemes is equal to 0. This result is in agreement with that obtained in [1]. This is due to the fact in both schemes, the routing paths are selected according to the policy that no node in such path can receive more encrypted message parts than its trust level would permit.

Next, we compare the route selection times for the two algorithms. The results are depicted in Fig. 3. In Fig. 3, it can be observed that the route for the E-TBM scheme has increased overall compared to that of the TBM scheme. This can be attributed to the fact that in the E-TBM scheme, more computation and time are required in selecting the paths with the least amount of energy while maintaining the secure route. We also compare the total energy consumed (in Joules) by the nodes that are selected for the secure transmission in both schemes. The results are captured in Fig. 4. In Fig. 4, it can be observed the energy consumed in the case of the TBM algorithm is significantly higher compared to that of the E-TBM algorithm. This constitutes a justification of taking the energy required to transmit a packet into account when designing secured routing protocols for MANETs.

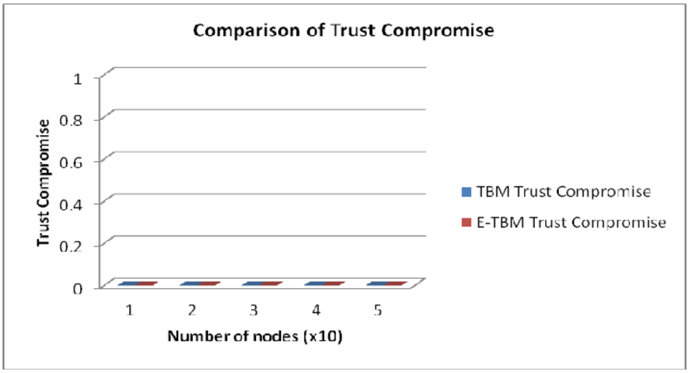


Fig. 2. Total trust compromise of E-TBM vs. TBM schemes

Next, we compare the total energy (in Joules) consumed in the network. The results are shown in Fig. 5. In Fig. 5, it can be observed that for E-TBM scheme, the overall energy consumption required for multiple paths to be selected securely and for messages to be sent down those multiple paths is much lower than that experienced with the TBM scheme.

Our simulation is started with each node having 5000 Joules of power, which decreases according to the type of routing operation being performed and which involves that node. In Fig. 6, it can be observed that by the end of the simulation, there were fewer nodes that had depleted their power (dead nodes) in the E-TBM scheme compared to the TBM scheme. This result is a direct correlation to the decreased total energy observed in the case of E-TBM. Since the total energy consumption is lower, nodes will survive longer, thus, the lifetime of the network will be increased.

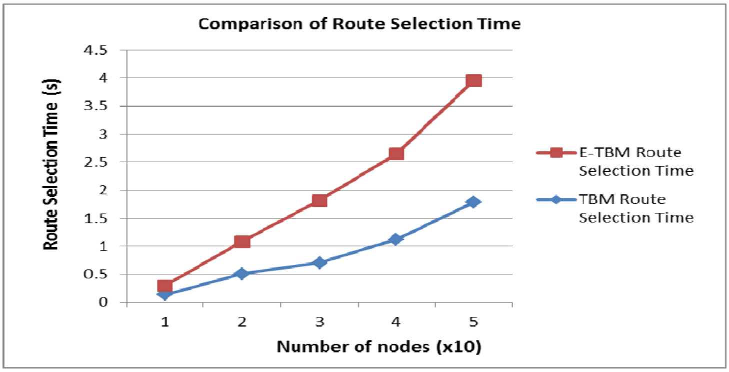


Fig. 3. Route selection time for E-TBM vs. TBM schemes

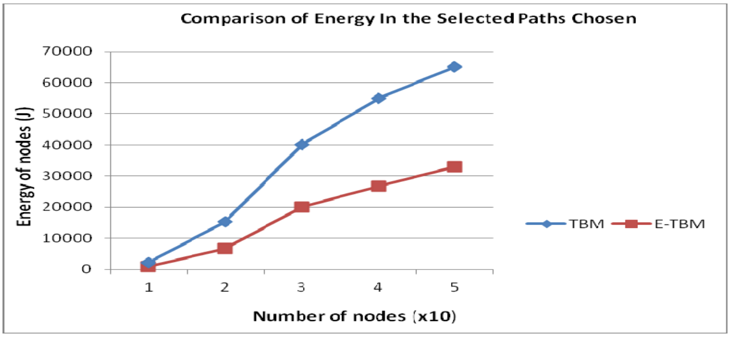


Fig. 4. Total energy consumed in the selected routing paths for E-TBM vs. TBM schemes

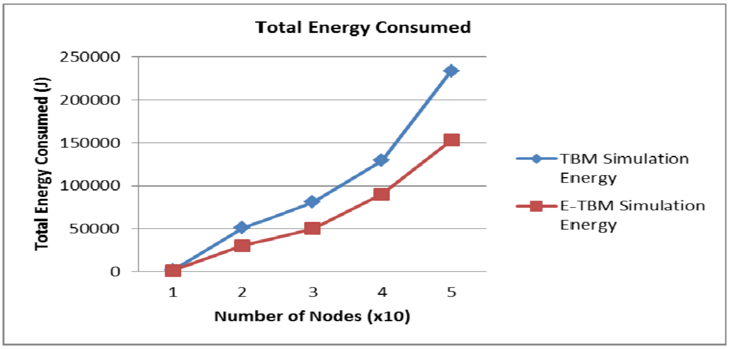


Fig. 5. Total energy consumed for E-TBM vs. TBM schemes

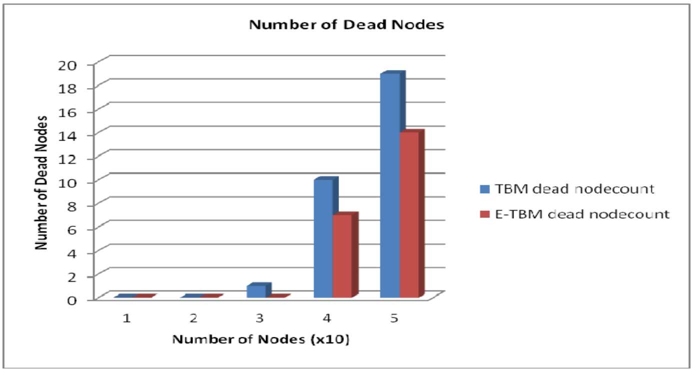


Fig. 6. Number of dead nodes in the E-TBM vs. TBM schemes

5 Conclusion

We have proposed a DSR-based secured routing scheme for MANETs and proved that it uses an energy efficient secure paths selection mechanism which minimizes the number of dead nodes, hence maximizes the network life time compared to the TBM scheme. We also observed that there is a compromise between message security (trust compromise) and routing time for both schemes. In future, we intend to compare our scheme against other known energy-aware secured routing protocols for MANETs.

References

1. Narula, P., Dhurandher, S.K., Misra, S., Woungang, I.: Security in mobile ad-hoc networks using soft encryption and trust-based multi-path routing. Computer Commuications 31(4), 760–769 (2008)
2. Roux, N., Pegon, J.-S., Subbarao, M.W.: Cost Adaptive Mechanism to Provide Network Diversity for MANET Reactive Routing Protocols. In: Proc. IEEE MILCOM (2000)
3. Singh, S., Woo, M., Raghavendra, S.: Power-aware with Routing in Mobile Ad Hoc Networks. In: Proc. of ACM/IEEE Intl. Conference on Mobile Computing and Networking (MobiCom 1998), Dallas, TX, USA (1998)
4. Vadivel, R., Bhaskaran, V.M.: Energy Efficient with Secured Reliable Routing Protocol (EESRRP) for Mobile Ad-Hoc Networks. Procedia Technology, 703–707 (2012)
5. Babu, M.R.: An Energy Efficient Secure Authenticated Routing Protocol for Mobile Adhoc Networks. American Journal of Scientific Research (9), 12–22 (2010) ISSN 1450-223X
6. Taneja, S., Kush, A.: Energy Efficient, Secure and Stable Routing Protocol for MANET. Global Journal of Computer Science and Technology, Network, Web and Security 12(10), Version 1.0 (May 2012)
7. Banerjee, A., Bhattacharyya, A., Bose, D.: Power and Trust Based Secured Routing Approach in MANET. Intl. Journal of Security, Privacy and Trust Management (IJSPTM) 1(3/4) (2012)
8. Zeng, X., Bagrodia, R., Gerla, M.: Glomosim: A library for the parallel simulation of large-scale wireless networks. In: Proc. of the 12th Workshop on Parallel and Distributed Simulation, Banff, Alberta, Canada, pp. 154–161 (May 1998)

A Grid-Based Approximate K-NN Query Processing Algorithm for Privacy Protection in Location-Based Services

Miyoung Jang and Jae-Woo Chang*

Dept. of Computer Engineering, Chonbuk National University,
567 Baekje-daero, deokjin-gu, Jeonju-si, Jeollabuk-do, South Korea
{brilliant,jwchang}@chonbuk.ac.kr

Abstract. Location-Based Services (LBSs) are becoming popular due to the advances in wireless networks and positioning capabilities. Providing user's exact location to the LBS server may lead revealing his private information to unauthorized parties (e.g., adversaries). There exist two main fields of research to overcome this problem. They are cloaking region based query processing methods which blur a user's location into a cloaking region and Private Information Retrieval (PIR) based query processing methods which encrypt location data by using PIR protocol. However, the main disadvantages of existing work are high computation and communication overheads. To resolve these problems, we propose a grid-based approximate k-NN query processing algorithm by combining above two methods. Through performance analysis, we have shown that our scheme outperforms the existing work in terms of both query processing time and accuracy of the result set.

Keywords: LBSs, Query processing, K-NN query, Location privacy preserving query processing, Cloaking region based query processing.

1 Introduction

Location-Based Services (LBS) such as Telematics and Navigation Systems (i.e., devices equipped with Global Positioning System) are being popular due to their application in our daily life. The LBS consist of two main components, LBS server and query user. The user gains Point-of-Interest information by forwarding a query to LBS server along with his exact location. The LBS server processes the user request and returns the POI information as a query result. However, in this process the user's exact location can be overheard by adversaries and the user's private information may revealed and misused [1,2]. Therefore, the user's location information must be protected while using LBS applications.

To address this privacy concern, there are two types of research existed in the field of privacy preserving query processing. They are location cloaking [3,4,5,6] method

* Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 526–535, 2013.

and Private Information Retrieval (PIR) based method [7,8]. The location cloaking method generates a cloaking region which encloses k-1 other users to blur a user's exact location. Based on the cloaking region, the server processes the user's query and returns a candidate set of POIs to him/her. However, the main disadvantage of this approach is that the LBS server returns many unnecessary candidate POIs so that the database may leak its important information to the user. On the other hand, PIR based method [9,10] can process a given query without revealing the exact location of the user through computationally complicated encryption method. But, it suffers from high computation and communication cost.

To resolve these problems, Ghinita et al.[9] proposed a hybrid two step approach. In the first step, like in the case of traditional cloaking methods, user generated cloaking region and a query are sent to the LBS server. In the second step, this approach implied the PIR protocol to control the amount of disclosed data to the user. This method has two main problems. First, this algorithm splits the cloaking region based on the system parameters, in order to control the number of disclosed POIs. This may lead the decline of query result accuracy. Second, since the hybrid two-step approach only supports an approximate Nearest Neighbor (NN) search, it cannot be applied to search k-Nearest Neighbor (k-NN).

Therefore, in this paper, we propose a new k-NN query processing algorithm by combining existing two methods: location cloaking and the PIR protocol. In our scheme, a user generates a cloaking region in order to hide his exact location and sends a query with this cloaked area to the LBS server. In addition, to process the query efficiently, we adapt PIR protocol considering only the Points of Interest (POIs) within the cloaked area unlike the original PIR proto-col considering the whole database of POIs. Moreover, to support k-NN query processing, we propose a new POI density based k-NN search algorithm. By indexing the whole database into a gird structure, we can retrieve k-NN POIs and expand the cloaking region efficiently. Also, we introduce an overlapping k-d index structure which controls the number of returned POIs increasing the accuracy of result set.

The rest of the paper is organized as follows. In section 2, we present related work. Section 3 is devoted to introduce overall system architecture, our overlapped index structure and an approximate k-NN search algorithm. To show the efficiency of our approach, performance evaluations are provided in section 4. Finally, section 5 concludes this research with some future work.

2 Related Work

There have been a lot of research existed in the field of user privacy protection and privacy preserving query processing in LBS. This research can be classified into two groups, location cloaking based query processing and Private Information Retrieval (PIR) based methods. First, location cloaking methods protect users' location information by generating a cloaking region that hides the users' exact locations [5,6,7,8]. The cloaking methods outline various types of user requirements, such as

K-anonymity, L-diversity, so that they can provide high level of Quality of Service (QoS). Moreover, query processing algorithm at the server side finds a candidate set of POIs which includes the exact result or approximate result to the user. The main disadvantage of these methods is that they return many unnecessary candidate POIs. As a result, the database may leak important information of the system to the user.

Second, G. Ghinita et al. proposed a privacy-aware NN query processing algorithm [10] by using a computationally private information retrieval (cPIR). The primary idea of their work is that a client set two large prime numbers and N which is the multiplication of the prime numbers. Then, the client determines the quadratic residue (QR) and quadratic non-residue (QNR) numbers of N. Since the database is represented by bit values (i.e., 0 and 1) the user sends an index query to the server. In the pre-processing phase, the server calculates the Voronoi cells of all POIs and stores the information of cells. When a user sends a query, the client sends a query index by setting his located cell to be QNR and other cells are set to be QR. The server processes the query by using the complex computation and returns an index. In the end, the user computes the area of QNR in the result index. Although this method guarantees searching the exact result without revealing user's location, the communication cost is greatly increased and the cost of query response time is high.

To resolve these problems, Ghinita et al. proposed a hybrid two step approach [11]. They provide protection for both users and the database of the LBS server by combining location cloaking and cPIR methods. First, both the user generated cloaking region and his query are sent to the LBS server. Second, the server retrieves R*-tree in order to search the POIs which intersect the cloaking region. Third, with the found POI information, the server generates k-d index which encloses k number of POIs in each partition. In order to find out where the user is located among the k-d index partitions, the authors also devised a Homomorphism-based cryptographic protocol that privately evaluates whether a point is enclosed inside a rectangular region. By privately evaluating the difference between user coordinates and the boundary coordinates of each partition, the server retrieves the user located rectangle in the k-d index. Finally, by using the PIR protocol the server returns the desired POI information of user-located partition to the user. However, there are two main problems of this method. First, this algorithm split the cloaking region in order to control the number of disclosed POIs which leads the decline of query result accuracy. Second, since the hybrid two-step approach only supports an approximate Nearest Neighbor (NN) search, it cannot be applied to search k-Nearest Neighbor (k-NN). Since the hybrid two-step approach only supports an approximate NN search, it cannot be applied to search approximate k- NN.

3 Grid-Based Approximate k-NN Query Processing Algorithm

In this section, we devise a grid-based approximate k-NN query processing algorithm for both user privacy protection and reduced POI disclosure.

3.1 Overall System Flow

Fig. 4 depicts overall system flow of our k-NN query processing algorithm. There are two main components in this system: a query issuer and LBS query processing server. We assume that a user generates a cloaking region either by peer to peer communication or by using the third party anonymizer to blur his exact location before sending a query to the server. Then, the LBS query processing server performs the query and returns a result to the user. Similar to the existing privacy aware system architecture, our query processing server is embedded inside the LBS location database server to deal with cloaked spatial region rather than the exact user's location. After receiving the candidate set, the user evaluates his exact query result from the received candidate POI set. Our system process an approximate k-NN query through two rounds of communication between the query issuer and query processing server. In the first round, the query issuer generates a cloaking region, then sends a query with the cloaking region and encrypted location E(x, y) to the server (Fig. 4-①). The location server processes the Grid-based POI search methods in order to retrieve candidate k-NN POIs for the cloaking region (Fig. 4-②,③). After this step, the server execute k-d overlapped index of expanded cloaking region which contains restricted number of POIs in each partition while allowing overlapping between partitions determined by parameter α (Fig. 4-④). The query processing server privately evaluates the enclosure condition of the user's location with encrypted location E(x,y) among the partitions in the manner of the private evaluation of point-rectangle enclosure method proposed by Ghinita et al. [11]. Finally, the server returns the encrypted result to the user (Fig. 4-⑤). In the second round, the user requests the POI information of his located partition through PIR protocol and filters out unnecessary results after receiving the POI information (Fig. 4-⑥,⑦).

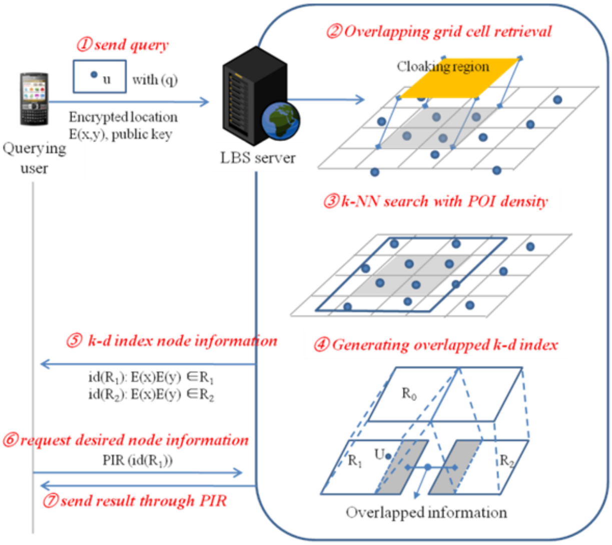

①send query
u
with (q)
Encrypted location
E(x,y), public key
Querying user
LBS server
②Overlapping grid cell retrieval
Cloaking region
③k-NN search with POI density
④Generating overlapped k-d index
R_0
R_1
U
R_2
Overlapped information
⑤ k-d index node information
id(R_1): E(x)E(y) ∈R_1
id(R_2): E(x)E(y) ∈R_2
⑥request desired node information
PIR (id(R_1))
⑦send result through PIR

Fig. 1. Overall system flow

3.2 k-NN Search Method Based on POI Density

To retrieve k-NN POIs from the given cloaking region, we present a k-NN search method by utilizing POI density of the given area. Because our method initiates expansion from the highly POIs dense region, it can reduce the number of expanding cells and enclosing the number of POIs. This improves the performance of k-d index generation overhead. Our k-NN search method consists of three steps. They are searching grid index, calculating POI density and expanding cloaking area.

1) Searching Grid Index

After receiving a query and a cloaking region from a user, the server retrieves the grid cells which intersect the cloaking region and POIs.

2) Calculating POI Density

In this step, we calculate the POI density to search k-NN candidates. By expanding the cloaking region based on the POI density, we can reduce the unnecessary expansion of gird cells. The algorithm relatively expands the cloaking region from the edge of the cloaking region which has high POIs density. The POI density of cloaking region is calculated as below.

$$density(d) = \frac{\#\ of\ POIs}{\#\ of\ cells} \tag{1}$$

3) Expanding Cloaking Area

Based on the POI density, the algorithm performs cloaking area expansion. Since the user can be located at any point in the cloaking region, the algorithm retrieves all possible k-NN POIs from each edge of the cloaking area. In order to reduce the number of expanding cells, the expansion initiates from the edge whose intersecting number of cells is greater than others. The number of expanding cells is calculated as below. In this expression, k is the number nearest neighbor POIs to be searched, which is commonly known as k-NN POIs.

$$\#\ of\ expanding\ cells = \frac{k}{d} \times \frac{1}{2} \tag{2}$$

3.3 Overlapped k-d Indexing Method

Before describing our indexing method, we explain the importance of area partitioning method for protecting database of the LBS. When a user sends a query to the LBS server with his cloaking area, the LBS server retrieves POIs within the expanded cloaking region and generates overlapped k-d index with them. After the user's located partition is evaluated, the LBS returns all POIs of the partition, whereas the existing cloaking based query processing algorithms return the information of all POIs within the cloaking region. This can reduce the number of revealing POIs of the system while protecting user's privacy.

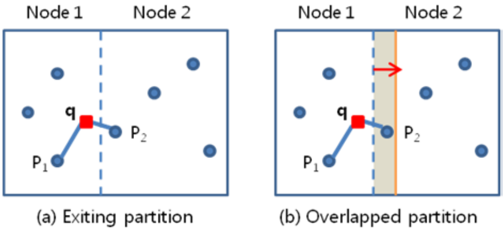


Fig. 2. Advantage of overlapped partitioning

With this premise in mind, we propose an overlapped k-d indexing method which provides approximate k-NN query processing and higher result accuracy to users.

Our overlapped k-d indexing method adopts the concept of adjusted median split heuristic from [11]. Furthermore, we enhance the heuristic by allowing overlapping between partitions in order to support k-NN query and increase the accuracy of the query result. In Fig. 6, there are two partitions (i.e., Node1 and Node 2) which are split by dotted line and they have 3 and 4 number of POIs respectively. If a query user q is located in Node 1 as shown in Fig. 6-(a), the existing work returns all POIs in Node 1 as the NN query result. Therefore, the user would assume that P1 is the NN from his location whereas the exact NN is P2 which has not been considered since P2 is located near to the split point. In our scheme, overlapped index scheme allows area overlapping between split nodes so that the exact POIs near to the query user can be retrieved. As shown in Fig. 6-(b), P2 is the NN result to the query user since it is located in the overlapped area and duplicated to Node 1.

Our overlapped k-d index structure is basically working on an adjusted median split [11] that controls the cardinality of leaf nodes. First, this scheme partitions the given space recursively based on the number of POIs in each partition. Given the node cardinality F of the current partition, our index structure ensures that all partitions contain F number of POIs except the last one which may contains up to 2F-1 POIs. Obviously, to support k-NN query, the F should be greater than average k, the number of POIs the user wants to find. However, during maintenance of the index structure, more than one node (partition) may have 2F-1 POIs. In this case, it is required to split the current overflow partition. There can be several candidate sets to split the current partition. Therefore, the index structure splits the partition into two candidate partitions with F and measures the sum of perimeters of two partitions for the minimum bounding rectangles (MBRs) of points. The candidate set with minimum sum of perimeters is chosen. This split technique considers both X-axis and Y-axis, and chooses the split with the minimum sum of perimeters of two partitions. Moreover, before storing the partitioned nodes information, our overlap index structure expands each division with overlapping parameter α from the split point, where α is the percentage of the duplicated area of neighboring node. If α is 10, each partitioned node expands its area by covering 10% area of the neighboring node and duplicates the POI information. This enhances the accuracy of query result while reducing the number of returned POIs. Since the LBS server always returns POIs of the partition in which the user is located, the finally revealed number of POIs is less than 2F+β where β is the number of overlapped POIs. Thus, the communication overhead of the server side is significantly reduced. Fig 9 present our proposed algorithm that efficiently retrieves approximate k-NN POIs.

```
Input: Cloaking Region CR, E(x), E(y), public key E
Output: Encrypted partition information E(id, x, y)
System Parameter: Node Cardinality F, Overlap area
STEP 1 Grid Cell search
 1:  search grid cells which intersect the CR
 2:  return POIs in CR
STEP 2 Expand cloaking area
 3:  calculate POI density in the CR
 4:  expand cloaking area by grid cell units considering POI density
 5:  return MBR of expanded area
STEP 3 Generate overlapping index
 6:  sort POIs in P increasingly according to X-coordinates
 7:  add up the perimeters of all partition
 8:  compare the sum of perimeters of partitions
 9:  if (SumLeft split < SumRight Split)
10:    split CR from Right side
11: else split CR from Left side
13:  repeat step(line 6-12) for Y-axis and choose the smallest sum of
           partitions
14:  expand partitions with overlap parameter
15:  store partition information
STEP 4 Perform enclosure condition of user among partitions
16:  perform the function Private Point-Rectangle Enclosure
17:  encrypt the result with public key E
18:  send the encrypted result to user
End Algorithm
```

Fig. 3. Approximate k-nearest Neighbor Search Algorithm

4 Experimental Evaluation

We experimentally compare the effectiveness of our Grid-based k-NN query processing algorithm against existing hybrid NN scheme [11] under various settings. Since the existing work only support 1-NN query processing, we intuitively applied density-based k-NN search algorithm which proportionally expands the cloaking region based on the area size. We developed both algorithms in VS C++ 2006 and ran the experiments on Window XP with an Intel Xeon 3.0GHz with 2GB RAM. In our experiment, we used the real data with 120,000 POIs which have postal addresses of northern California, USA and three synthetic data sets with 100,000 generated by using the Generate Spatio-Temporal Data (GSTD) algorithm [12], with uniform, Gaussian and skewed distribution. However, because of the space constraint, we illustrate Gaussian and Real data set results. We used the average value over 100 query results in terms of query processing time and query result accuracy. The query result accuracy is calculated by counting the number of result sets containing the exact query result. Table 1 shows the experimental parameters.

Table 1. Experimental Parameters

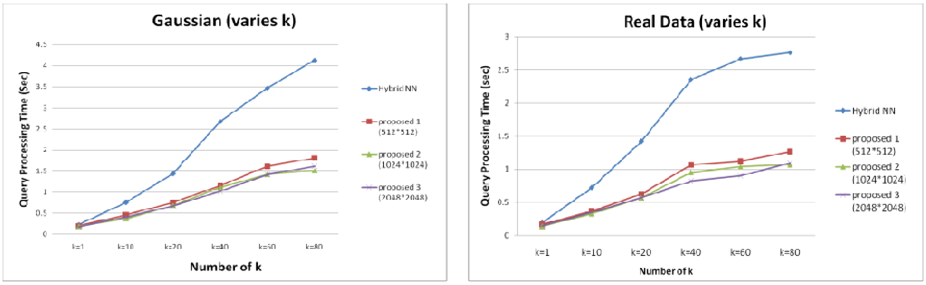
Parameter	Range	Default
Cloaking Area Size	1%, 2%, 5%, 10% of whole	5%
Grid Size	512*512, 1024*1024, 2048*2048	-
k-NN (k)	10, 20, 40, 60, 80	10
Node Cardinality(F)	varies from k	-
Overlapping parameter(α)	5%, 10%, 15%, 20%	10%

4.1 Performance Evaluation on Query Processing Time

We experimentally evaluate the proposed K-NN query processing algorithm and existing hybrid NN algorithm when k is varies from 1 to 8. With increasing number of k, the default setting of the node cardinality F is also increased from 40 to 1600. Fig. 11 shows the effect of different k on the query processing time of existing method and our algorithm. Both methods require more query processing time when k is increased. However, for all the cases, our method outperforms the existing method. In the real data distribution, the query processing time of the existing method is 0.19656 whereas out method with grid size = 2048*2048 requires 0.14284 second when k=1, F=40. It is also shown that the grid index structure of our method guarantees better performance in terms of k-NN query processing time. This is because the existing work retrieves R*-tree in order to search POIs, but our method utilizes the grid-index for retrieving POIs.

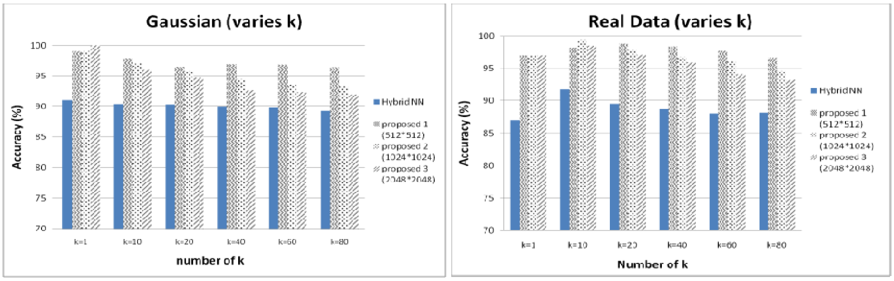
4.2 Performance Evaluation on Query Result Accuracy

Recall that one of the goals of our query processing algorithm is to enhance the query result accuracy by allowing area overlapping between partitions. As shown in Fig. 12, for all tested case, our method achieves much higher query result accuracy than the existing method. In the real data distribution, our method with grid size=512*512 achieves 98% query result accuracy when k=40 whereas that of existing method provides 88%. In terms of query result accuracy, our method with grid size=512*512 shows the best performance among three different grid sizes. This is because when the grid size is larger, it relatively retrieves greater number of POIs when expanding cells. Hence, the number of candidate POIs is increased which affect the size of result set. Therefore, when the grid size of our method is 512*512, the query result accuracy is the best among three different grid sizes whereas it shows the worst performance in terms of the query processing time



(a) Gaussian distribution (b) Real data distribution

Fig. 4. Query Processing Time

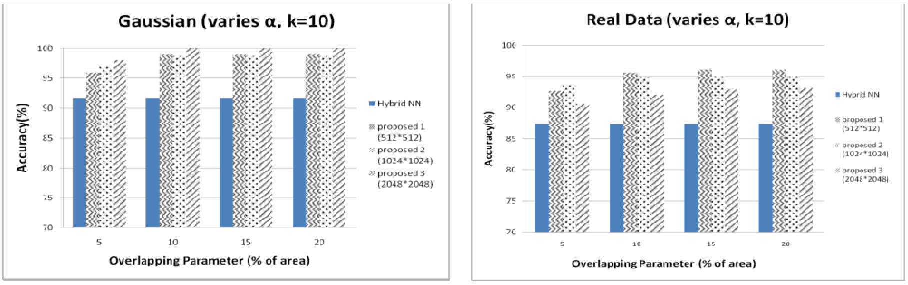


(a) Gaussian distribution (b) Real data distribution

Fig. 5. Query Result Accuracy

Also, we measure the query result accuracy of the existing method and our method when the overlapping parameter α ranges from 5% to 20%, when k=10, F=200. Fig. 13 presents the query result accuracy under different overlapping parameter α. From the result, it is proven that with increasing number of α, the query result accuracy is also increased. In the real data set, the existing method retrieves only 87% of genuine query result, whereas our method with grid size=1024*1024 finds 96%, 97%, 97% and 98% of actual query result, respectively, when α is 5% , 10%, 15% and 20%.

From the overall experimental evaluation, we verified that our approximate k-NN query processing algorithm outperforms the existing NN query processing algorithm in terms of query result accuracy and query processing time.



(a) Gaussian distribution (b) Real data distribution

Fig. 6. Query result accuracy with vary α

5 Conclusion

This paper introduces a hybrid scheme to process an approximate k- nearest neighbor query considering both user privacy protection and controlled POI disclosure. To achieve this goal, we propose a POI density based k-NN search algorithm and devise an overlapped k-d indexing. To the best of our knowledge, this is the first k-NN query processing algorithm which considers both the privacy of user location and controlled disclosure of POIs. Through performance analysis, we have shown the effectiveness of the proposed method. The query processing time of our k-NN query processing algorithm supports 3 times faster than the existing method in maximum, and the query result accuracy outperforms the exiting method by 10%. As future work, we have a plan to extend our research for supporting exact k-NN queries.

Acknowledgment. This research was supported by Basic Science Research program through the Na-tional Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number 2010-0023800).

References

1. Foxs News. Man Accused of Stalking Ex-Girlfriend With GPS (2004), http://www.foxnews.com/story/0,2933,131487,00.html
2. USA TODAY News, GPS System used to stalk woman (2002), http://www.usatoday.com/tech/news/2002-12-30-gps-stalker_x.html
3. Ku, W., Chen, Y., Zimmermann, R.: Privacy Protected Spatial Query Processing for Advanced LBSs. Wireless Personal Communications 51(1) (October 2009)
4. Mokbel, M., Chow, C., Aref, W.: The New Casper:Query Processing for Location Services without Compromising Privacy. In: Proc. of the International Conference on Very Large Data Bases, pp. 763–774 (September 2006)
5. Chow, C.Y., Mokbel, M.F., Liu, X.: A Peer-to-Peer Spatial Cloaking Algorithm for Anonymous Location-based Services. In: Proc. of the ACM International Symposium on Advances in Geographic Information Systems, pp. 171–178 (November 2006)
6. Bamba, B., Liu, L.: PRIVACYGRID: Supporting Anonymous Location Queries in Mobile Environments. Research Report in National Technical Information Service (2007)
7. Kushilevitz, E., Ostrovsky, R.: Replication is NOT Needed: SINGLE Database, Computationally-Private Information Retrieval. In: FOCS (1997)
8. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.L.: Private Queries in Location Based Services: Anonymizers are not Necessary. In: Proc. of ACM SIGMOD International Conference on Management of data (2008)
9. Ghinita, G., Kalnis, P., Kantarcioglu, M., Bertino, E.: A Hybrid Technique for Private Location-Based Queries with Database Protection. In: Mamoulis, N., Seidl, T., Pedersen, T.B., Torp, K., Assent, I. (eds.) SSTD 2009. LNCS, vol. 5644, pp. 98–116. Springer, Heidelberg (2009)
10. Theodoridis, Y., Silva, J.R.O., Nascimento, M.A.: On the Generation of Spatiotemporal Datasets. In: Güting, R.H., Papadias, D., Lochovsky, F. (eds.) SSD 1999. LNCS, vol. 1651, pp. 147–164. Springer, Heidelberg (1999)

Density-Based K-Anonymization Scheme for Preserving Users' Privacy in Location-Based Services

Hyunjo Lee and Jae-Woo Chang

Dept. of Computer Engineering, Chonbuk National University,
567 Baekje-daero, deokjin-gu, Jeonju-si, Jeollabuk-do, South Korea
{o2near,jwchang}@chonbuk.ac.kr

Abstract. Due to the explosive growth of location-detection devices, such as GPS (Global Positioning System), a user' privacy threat is continuously increasing in location-based services (LBSs). However, the user must precisely disclose his/her exact location to the LBS while using such services. So, it is a key challenge to efficiently preserve a user's privacy in LBSs. For this, the existing method employs a 2PASS cloaking framework that not only hides the actual user location but also reduces bandwidth consumption. However, it suffers from privacy attack. Therefore, we, in this paper, propose a density-based k-anonymization scheme using a weighted adjacency graph to preserve a user's privacy. Our k-anonymization scheme can reduce bandwidth usages and efficiently support k-nearest neighbor queries without revealing the private information of the query initiator. We demonstrate from experimental results that our scheme yields much better performance than the existing one.

Keywords: component, Privacy threat, location-based services (LBS), location privacy, cloaking, bandwidth, k-anonymity, weighted adjacency graph.

1 Introduction

Location-based services allow users to connect with others based on their current locations. In most cases, people use their positioning devices (i.e., iPhone, Android, Blackberry) to find out his/her location like restaurants, bars and stores that they visit. However, frequent and continuous accesses to the services expose users to privacy risk. Due to an increasing awareness of privacy risks, users might desist from accessing LBSs, which would prevent the proliferation of these services [1, 2].

Current research aligns on developing techniques to elaborate on k-anonymity [3-18, 20] that preserve a user's privacy during the access of LBSs. In the existing k-anonymization schemes(i.e., cloaking methods), they blur a user's location among k-1 users. Most of the k-anonymization techniques enclose non-result objects with a real object due to the achievement of a user's privacy. As shown in Figure 1, the user sets range with his/her accurate location and requests for the nearest clinic. The LBS server returns non-result objects (clinics) C1, C2 and C3 with the actual object C to the user. However, larger result set size preserves more privacy, but consumes more network bandwidth and device battery as well. Furthermore, the crucial matter is how to

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 536–545, 2013.

minimize the number of non-result objects during cloaking period. Some researches [3, 6-12] indirectly minimize the bandwidth by minimizing the size of cloaking region for different privacy metrics. The location cloaking approach called 2PASS (2-Phase Asynchronous Search) [21] has been proposed to minimize the bandwidth usages as well as preserve user privacy. By using Voronoi diagram, 2PASS is able to save the bandwidth usages compared to the Range Nearest Neighbor (RNN) [22] approach. Although 2PASS optimizes the bandwidth usages, it suffers from privacy attack.

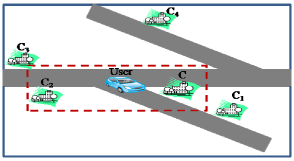


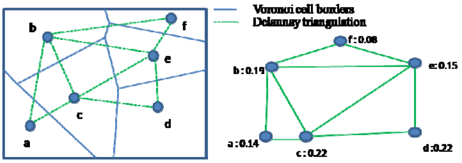
Fig. 1. Range Nearest neighbor Query

In this paper, we propose a weighted adjacency graph [21] based k-anonymous cloaking technique that can reduce bandwidth usages and provide protection to user. We follow user-cloaking-server model [3-12] where the trusted third party (location cloaker) performs location cloaking for the user. Our algorithm computes a KNN query in three phases. In the first phase, the user requests the location cloaker (LC) and LC requests the WAG information corresponding to the query. LC selects objects to request in the second phase and returns the actual result to the user in the third phase. We also include k-anonymity property for enhancing users' privacy while accessing LBSs.

The rest of this paper is organized as follows. In Section 2, we discuss related cloaking methods. We discuss our detailed system architecture and propose a density-based k-anonymization scheme using a weighted adjacency graph in Section 3. Section 4 is devoted to experimental results. Finally, we conclude our work with future direction in Section 5.

2 Related Work

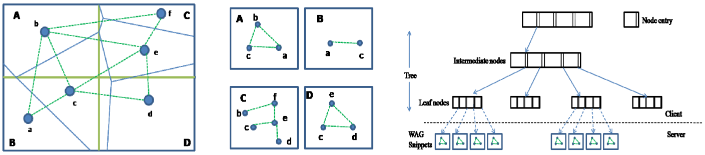
To the best of our knowledge, there exists only one research on result-aware location cloaking approach (called 2PASS), proposed by H. Hu and J. Xu [21]. 2PASS is based on the notion of voronoi cells and each cell contains one object that is the nearest neighbor of any point in its cell. For example, Figure 2(a) shows an example of voronoi diagram with 6 objects such as a, b, c, d, e and f. To access the voronoi cell information, they develop a weighted adjacency graph (WAG). WAG is a weighted undirected graph that stores the voronoi diagram and Delaunay triangulation. For example, in Figure 2(b), each vertex in this graph denotes an object, and each edge denotes a line in the Delaunay triangulation. Each vertex is also assigned a nonnegative weight. The specialty of this graph is to notify that the WAG vertices are weighted based on voronoi cell area size. User can compute out the objects to request from the server by using WAG-tree index. The criteria of object selection are a combination of the following: a) the sum of the areas of Voronoi cells from the selected objects must exceed τ; b) the genuine nearest neighbor o* must be selected; and c) these Voronoi cells must be connected, i.e., no cell is isolated from the rest of the cells.



(a) Voronoi Diagram (b) WAG

Fig. 2. Voronoi diagram and its WAG

To reduce computational overhead, 2PASS [21] also proposes to partition the entire WAG into WAG snippets of reasonable size so that the user receives only the snippets surrounding the query location. For example, in Figure 3(a), the four snippets are obtained by partitioning the space into four sub-spaces A, B, C and D of equal widths and heights and computing their WAG's, respectively. The weight of an object in a WAG snippet is set to its voronoi cell area that resides in this subspace. WAG snippets can be joined to become the WAG of the union of these subspaces. The join is done by merging the vertices corresponding to the same object and assigning its new weight as the sum of the weights of these vertices. WAG-tree follows a top down recursive fashion. For each node, the algorithm maintains objects whose voronoi cells in the whole space overlap this sub-space. Figure 3(b) shows a WAG-tree and snippet pointed by it. 2PASS is able to save bandwidth usage compared with others by returning less number of non-result objects. However, it suffers from privacy risk.



(a) WAG snippets (b) WAG-tree

Fig. 3. WAG snippets and WAG-Tree

3 Density-Based K-Anonymization Scheme Using Weighted Adjacency Graph

In this section, we first will present system architecture of our work. In addition, we introduce our proposed work, a density-based k-anonymization scheme using weighted adjacency graph (Density i-WAG). We first describe our system architecture that is based on client-cloaker-server architecture. Our approach to address the range nearest neighbor queries is based on the weighted adjacency graph (WAG) that encloses voronoi diagram. Using WAG we propose a density-based k-anonymization scheme that adopt trusted third party model. Figure 4 shows the system architecture of our method. In the first phase, user sends a query to location cloaker (LC) and LC requests LBS for iWAG (Improved WAG) information, where the weight of a vertex is based on the area of the corresponding voronoi cell and the number of users on that cell. In the second phase, LC selects objects from the iWAG (e.g. three restaurants Pizza Hut, MacDonald and KFC) and requests them for their complete contents (e.g., customer reviews and reservation status etc). In third phase, LC sends the exact answer to the user. In our system architecture, LC is responsible for cloaking procedure instead of user. Thus, the LBS server may not be used to infer a query issuer user.

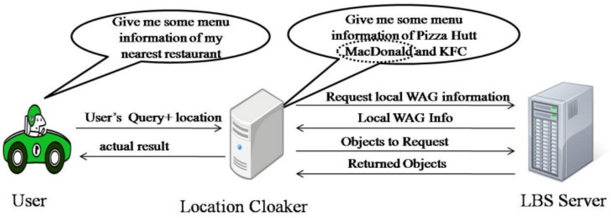

Give me some menu information of my nearest restaurant
Give me some menu information of Pizza Hutt MacDonald and KFC
User's Query+ location
actual result
Request local WAG information
Local WAG Info
Objects to Request
Returned Objects
User
Location Cloaker
LBS Server

Fig. 4. System Architecture

We propose a density-based k-anonymization scheme using weighted adjacency graph(Density i-WAG) to solve the problems of the 2PASS [21]. In our Density i-WAG, user sends a query with privacy requirement to location cloaker (LC) and LC requests the objects (including the genuine nearest neighbor (NN) together with other non result objects) based on the Voronoi cell information to satisfy the privacy requirement on the cloaked region. Our work is unique in that the LC controls what objects to request from the server so that their total number (i.e., the overall bandwidth) is minimized. To minimize the object number while still meeting the privacy threshold τ and k-anonymity requirement, the criteria of object selection are a combination of the following: (ì) the sum of the areas of voronoi cells from the selected objects must exceed τ and the number of users ≥ k on that cell; (ìì) the genuine nearest neighbor o* must be selected; and (ììì) these voronoi cells must be connected, i.e., no cell is isolated from the rest of the cells. The last criterion guarantees that the cloaked region is a single region, which is a common assumption in all existing location cloaking approaches. Besides, the single-region assumption not only adapts to most location-based services which readily accept a single location as the input, but also alleviates some security problems. For example, a single region is more resilient than isolated regions against background or domain knowledge attacks. With the iWAG, the object selection is equivalent to finding a sub graph that satisfies the following criteria: ì) the sum of the weights of vertices in the sub graph must exceed τ and the number of user ≥ k on that cell; ìì) o* must be in the sub graph; and ììì) this sub graph must be a connected component. Now, we describe the iWAG generation procedure. For this, we give the weight (w) for each object (Vw) based on voronoi cell area size (Va) and number of user (Un) on that voronoi cell. We set the priority for voronoi cell area size and the number of user in that cell. For example, if we consider the total priority, p = (α + β) = 1, then the preference of the number of user (β) is get priority than the preference of voronoi cell area size (α). Therefore, the following equation holds true,

$$V_w = (V_a \times \alpha) + (\frac{U_n}{total \ \ U_n} \times \beta) \tag{1}$$

Figure 7(a) shows voronoi diagram with eight users. We calculate objects weight based on equation (1). For example, if we consider object a, then weight of aw =0.206. By this, we get all of objects' weight as shown in Figure 7(b). We consider the total vertex weight is 1. Our objective is to find out the valid weight connected component based on iWAG. For this, we follow approximate minimum valid weight connected component (MVWCC) algorithm [21].

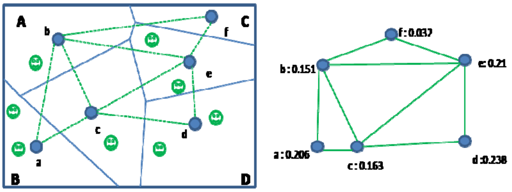
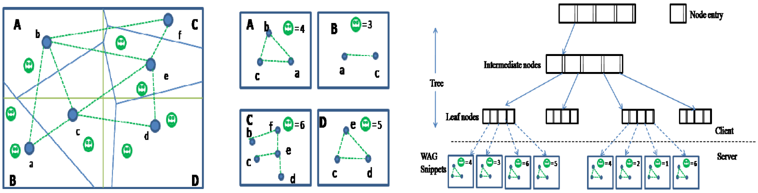


Fig. 5. (a) Voronoi Diagram and (b) iWAG with user

We divide the entire iWAG into iWAG snippets (i.e., as like [21]) of reasonable sizes so that the LC receives only the snippet(s) surrounding the query location. For example, eight users and the four snippets are obtained by partitioning the space into four subspaces A, B, C and D of equal size and computing their iWAG's, as shown in figure 6. It is noteworthy that an object being outside of a subspace can still appear in the iWAG snippet of this subspace, as long as the Voronoi cell of this object in the iWAG of the entire space overlaps this subspace, e.g., objects a and c in snippet A. The weight of an object in a iWAG snippet is set to its Voronoi cell area and the number of users that resides in this subspace. iWAG snippets can be joined to become the iWAG of the union of these subspaces. The join is done by merging the vertices corresponding to the same object and assigning its new weight as the sum of the weights of these vertices. In order for the LC to know which snippet(s) to request in the first phase of the query, we build a hierarchical index called iWAG-tree construction algorithm like a quad tree. This index recursively partitions the space into quadrants until a certain criterion is met. Each entry in its leaf node points to a WAG snippet. Figure 6(b) shows the iWAG-tree and snippet pointed by it.



(a) iWAG snippet (b) iWAG-tree

Fig. 6. iWAG Snippets and iWAG-tree

Finally, query processing of Density i-WAG is as follows. First, the whole *iWAG* tree is sent to LC during the system initialization time. Secondly, based on the kNN, the LC traverses the *iWAG* tree and finds out the snippet that contains the query point. Thirdly, the LC matches the privacy requirements (k-anonymity, the area size (τ) etc) that are sent by the user. If the area of this snippet is still smaller than the user specified requirements, the user will locate the lowest-level child node of this snippet whose area exceeds privacy requirements. Then the user requests all snippets rooted at this node, called host snippets. At last, the LC adds the received host snippets into a single *iWAG* and calculates the minimum valid connected components by using MVWCC algorithm [21]. In this process, LC does cloaking procedure instead of user and LC does not provide any location information or privacy requirements of user to the server.

4 Performance Analysis

In this section, we present the performance results of our location cloaker and query processing algorithm. We implemented our cloaking technique and query processing algorithms to evaluate the performance of our approach. Our main objectives are to observe the influence of performance factors on the system and to test the feasibility of our technique.

4.1 Experimental Setup

For the performance evaluation, We use the real data set of Northern East America (NE) that contains 119,898 point of interest (POIs). For easy presentation, the coordinates of these objects are normalized to a unit square. The query load consists of 70,000 queries that are uniformly distributed in the unit square. We compare our work with the exiting approach 2PASS [21] in terms of response time and bandwidth size. Table 1 represents the experiment environment. The parameter settings are summarized in Table 2.

Table 1. Experimental Environment

CPU	Intel® Xeon® CPU 2.00 GHz
Memory	2 GB
Simulator	Visual Studio 2010
OS	Windows XP

Table 2. Simulation Parameter

Parameters	**Range**
Total User	239,668
Query Number	70,000
Granularity Threshold (τ)	0.000001, 0.00001, 0.0001, 0.001
Maximum Area of WAG Snippet	0.001
K-Anonymity	2, 4, 6, 8, 10
Average Number of User in each Cell	2

4.2 Query Processing Time

We vary the user specified threshold (τ) value from 0.000001 to 0.001 and the threshold value reflects the response time that means query processing time. Here, the response time can be defined as how quickly the server returns the result set after receiving the query. As for query performance, the query processing time of 2PASS and our scheme are near to similar when τ is equivalent to small value. But, the query processing time of 2PASS is much higher than our scheme when τ becomes greater value. It is mainly due to the more number of objects it request. As a consequence, 2PASS also consumes more bandwidth than our scheme. Figure 7 demonstrates the query processing time with different τ value. Since a bigger threshold value usually contains more underlying network area, it takes longer to process. As shown in Figure 7, the increase of former metric is quite moderate until τ ≤ 0.0001. On the other hand, the latter metric linearly increases as τ grows. We also test the impact of varying the number of k-anonymity with query processing time. We alter k-anonymity range from 2 to 10. As shown in Figure 8, the query processing time remains the same when we raise k-anonymity from 2 to 10. That means the proposed work fulfills the k-anonymity requirement without affecting the query processing time.

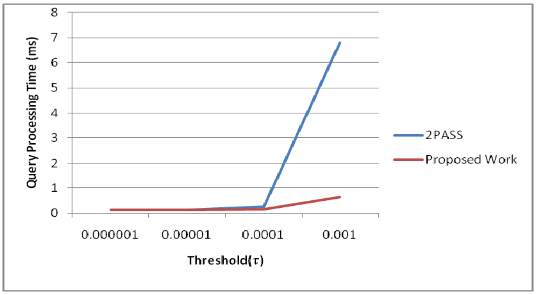


Fig. 7. Query Processing Time vs. τ

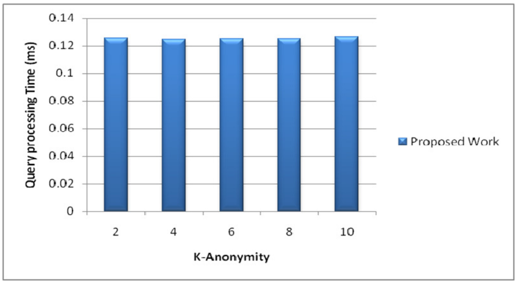


Fig. 8. Query Processing Time vs. k-anonymity

4.3 Bandwidth

The response time of 2PASS is larger than that of our scheme, which is mainly due to the more number of objects in our scheme. As a consequence, 2PASS also consumes more bandwidth than the proposed one. Here, the bandwidth can be defined as page size that encloses objects. We calculate the bandwidth size based on average number of objects returned by varying with different threshold value (τ). Figure 9 depicts the result set size with different τ. We observe that a bigger τ value generates a larger candidate result set. Figure 10 shows the result set size with different k-anonymity. As expected, our scheme returns more candidate result set as k-anonymity increases. In fact, the increase of average number of result is quite similar until k-anonymity equals 25. We observe that the result set increases linearly when we raise k from 30 to 60.

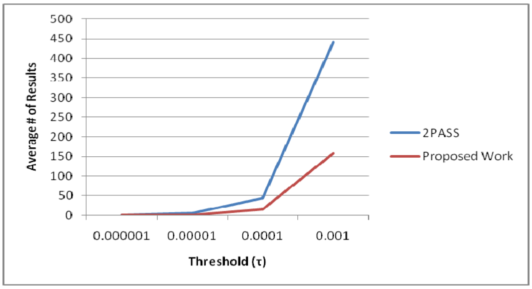


Fig. 9. Average Number of results vs. τ

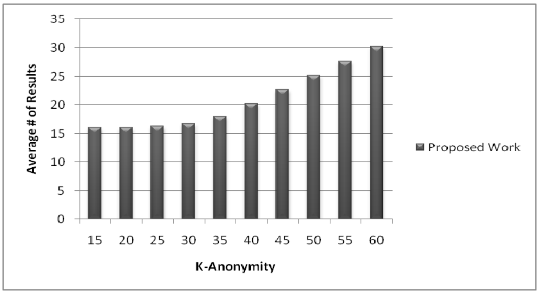


Fig. 10. Average Number of results vs. k-anonymity

5 Conclusion

We identify the limitations of privacy-ware data access in location-based services. We propose a density-based k-anonymization scheme using a weighted adjacency graph that allows k-anonymity property for providing the location privacy of all users in the network. Our technique follows third party based approach where the location cloaker handles cloaking process instead of user. We also minimize the bandwidth consumption by using iWAG-tree index from which the location cloaker can compute out the objects to request from the server. Through our experimental performance evaluations, we have shown that our cloaking method is much more efficient in terms of both response time and bandwidth consumption than the 2PASS.

In the future work, we plan to extend our work beyond a larger geographical area. We also plan to conduct the extensive performance evaluations by our experiments with various data sets and study the behavior of our work when the user issues a series of requests within a short period.

Acknowledgment. This research was supported by Basic Science Research program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number 2010-0023800).

References

1. Privacy concerns a major roadblock for location-based services say servay (2007), http://www.Govtech.com/gt/article-es/104064
2. Muntz, W.R., Barclay, T., Dozier, J., Faloutsos, C., Maceachren, A., Martin, J., Pancake, C., Satyanarayanan, M.: IT Roadmap to a Geospatial Future. The National Academics Press (2003)
3. Gruteser, M., Grunwald, D.: Anonymous usage of Location-Based Services Through Spatial and Temporal Cloaking. In: Proceedings of the First ACM/USENIX International Conference on Mobile Systems, Application and Services (MobiSys), San fransisco, CA, USA (2003)
4. Gedik, B., Liu, L.: Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms. IEEE Trans. Mobile Computing 7(1), 1–18 (2008)
5. Schilit, B.N., Hong, J.I., Gruteser, M.: Wireless Location Privacy Protection. IEEE Computer 36(12), 135–137 (2003)
6. Schilit, B.N., Hong, J.I., Gruteser, M.: Wireless Location Privacy Protection. IEEE Computer 36(12), 135–137 (2003)
7. Mokbel, M.F., Chow, C.Y., Aref, W.G.: The new casper: query processing for location services without compromising privacy. In: Proceedings of the International Conference Very Large Database (VLDB), pp. 763–774 (2006)
8. Mokbel, M.F.: Towards Privacy-Aware Location Based Database Servers. In: Proceeding of the 22nd IEEE International Conference on Data Engineering (ICDE) Workshop, Atlanta, Georgia, USA (2006)
9. Kalnis, P., Ghinita, G., Mouratidis, K., Papadias, D.: Preventing Location-Based Identity Inference in Anonymous Spatial Queries. IEEE Transactions on Knowledge and Data Engineering 19(12), 1719–1733 (2007)
10. Bamba, B., Liu, L., Pesti, P., Wang, T.: Supporting anonymous location queries in mobile environments with privacygrid. In: Proceeding of the International Conference World Wide Web, pp. 237–246 (April 2008)
11. Wang, T., Liu, L.: Location Privacy Over Road Networks. In: The International Conference Very Large Database, VLDB (2009)
12. Hossain, A., Hossain, A.A., Chang, J.W.: Spatial Cloaking Method Based on Reciprocity Property for Users' Privacy in Road Network. In: IEEE 11th International Conference on Computer and Information Technology (CIT), pp. 487–490 (2011)
13. Chow, C.Y., Mokbel, M.F., Liu, X.: A Peer-to-Peer Spatial Cloaking Algorithm for Anonymous Location-Based Service. In: Proc. Ann. ACM Int'l Symp. Advances in Geographic Information Systems, GIS (2006)
14. Ghinita, G., Kalnis, P., Skiadopoulos, S.: PRIVÉ: Anonymous Location-Based Queries in Distributed Mobile Systems. In: Proc. of the International Conference World Wide Web (WWW), pp. 371–380 (2007)

15. Monjur, M., Ahamed, S.I., Chowdhury, S.H.: ELALPS: A Framework to Eliminate Location Anonymizer from Location Privacy Systems. In: 33rd Annual IEEE International Computer Software and Applications Coference (2009)
16. Hashem, T., Kulik, L.: Safeguarding Location Privacy in Wireless Ad-Hoc Networks. In: Krumm, J., Abowd, G.D., Seneviratne, A., Strang, T. (eds.) UbiComp 2007. LNCS, vol. 4717, pp. 372–390. Springer, Heidelberg (2007)
17. Zhong, G., Hengartner, U.: Toward a distributed k-anonymity protocol for location privacy. In: Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society, CCS, pp. 33–38 (2008)
18. Solanas, A., Martínez-Ballesté, A.: Privacy protection in location-based services through a public-key privacy homomorphism. In: López, J., Samarati, P., Ferrer, J.L. (eds.) EuroPKI 2007. LNCS, vol. 4582, pp. 362–368. Springer, Heidelberg (2007)
19. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.-L.: Private Queries in Location Based Services: Anonymizers Are Not Necessary. In: Proc. ACM SIGMOD (2008)
20. Sweeney, L.: k-anonimity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge Based Systems 10(5), 557–570 (2002)
21. Hu, H., Xu, J.: 2PASS: Bandwidth-Optimized Location Cloaking for Anonymous Location-Based Services. IEEE Transactions and Parallel on Distributed Systems (2010)
22. Hu, H., Lee, D.: Range Nearest Neighbor Query. IEEE Trans. Knowledge and Data Eng. 18(1), 78–91 (2006)
23. Beresford, A., Stajano, F.: Location privacy in pervasive computing. IEEE Pervasive Computing 2(1), 46–55 (2003)
24. Kido, H., Yanagisawa, Y., Satoh, T.: An anonymous communication techniques using dummies for location based services. In: Proceeding of 2nd ICPS, pp. 88–97 (2005)
25. You, T., Peng, W., Lee, W.: Protect Moving Trajectories with Dummies. In: Proceeding International Workshop Privacy-Aware Location-Based Mobile Services (2007)
26. Suzuki, A., Iwata, M., Arase, Y., Hara, T., Xie, X., Nisho, S.: A user location anonymization method for location based services in a real environment. In: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (2010)
27. Berg, M., Kreveld, M., Overmas, M.: Computational Geometry: Algorithm and Applications. Springer, Heidelberg (1997)
28. `http://www.mailshell.com/` (2009)

A Routing Mechanism Using Virtual Coordination Anchor Node Apply to Wireless Sensor Networks

Chih-Hsiao Tsai[1], Kai-Ti Chang[2], Cheng-Han Tsai[3], and Ying-Hong Wang[4]

[1] Takming University of Science and Technology Infromation Technology, Taiwan, R.O.C.
chtsai2104@gmail.com
[2,3,4] Tamkang University, Dept. of Computer Science and Information Engineering, Taiwan, R.O.C.
funkyhome@gmail.com, {699410014@s99,inhon@mail}.tku.edu.tw

Abstract. In recent years, wireless sensor networks are widely used in many areas. Collecting information efficiently and deliver to base station reliably among the hot topics in wireless sensor networks. Most of the previous studies used the Geographic routing to resolve this problem. Sensors have to know not only their location information, but also one hop neighbor and destination location information. Generally, the Global Positioning System (GPS) provides location and time information, but it will increase the cost, power consumption, and reduce the lifetime of wireless sensor network. In this paper, we propose a Virtual Coordinate System (VCS). With the VCS, WSNs can find the four extreme nodes in the scene as virtual anchor nodes. Then, a shortest path between virtual anchor nodes and sink for transfer data in the random distribution network is created. In this approach, establish a low-power, extend the network lifetime, efficient and fault-tolerant routing mechanism.

Keywords: anchor node, routing, virtual coordinate system, wireless sensor networks.

1 Introduction

Due to the booming network technology, wireless network come into existence as another choice other than cable network, providing people more and more multiplication of network type and making wireless network become a hot issue in recent years. In order to meet different requirements, types of wireless specification and technologies have been developed, especially Wireless Sensor Networks (WSNs) [1], which is a successful example of combine sensor and wireless network.

In WSNs, routing mechanism need to achieve separate data transport, power saving and ensure data can be complete sent to sink. The main function of the routing mechanism is to transfer data separately, saving power, and to ensure data can be completely sent to the sink. However, many researches proposed before [2-4] used wireless sensor node with GPS as an anchor node to localization. However, high power consumption must be considered. When the anchor node power is exhausted, other nodes could not replace the original one, resulting in the interruption of data transmission.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 546–555, 2013.

Therefore, under the premise of reducing the power consumption, GPS device is not applied in this research. On the other hand, since most the objectives of the many applications of WSNs are in a static environment, the mobility of the wireless is not the first concern and calculation ability is not strong. In this regard, power consumption and cost are expected to be minimized. In this paper, we propose a novel routing mechanism "A Routing Mechanism Using Virtual Coordination Anchor Node". With this approach, sensors are randomly deployed and the data is transferred through the specially designed routing mechanism, which is expected to expand the lifetime of WSNs, achieve the efficient transmission.

2 Related Work

Due to the blooming of microelectromechanical systems in recent years, the size of sensor nodes becomes smaller and smaller. How to effectively use the limited battery power to achieve the highest efficiency is an important issue. Since the main factor of power consumption is hardware and data transmission path, to reduce hardware and select a valid path is the focus of this paper research. The current routing algorithms, can be basically divided into four directions: Flooding [6] , Chain [7] , Clustering [8] , and tree [9].

Low-Energy Adaptive Clustering Hierarchy (LEACH)

LEACH[10] use the clustering architecture, it also has the characteristics of the active routing protocol. This method will divide wireless sensor network into many different cluster regions, member node only can communicate each other in the same region. The whole process of long-distance transfer requires the following steps: Firstly, every region will choose a node as cluster head and collect the data collected in the region; then, it will send the date to the base station or data sink, shown as figure 1.

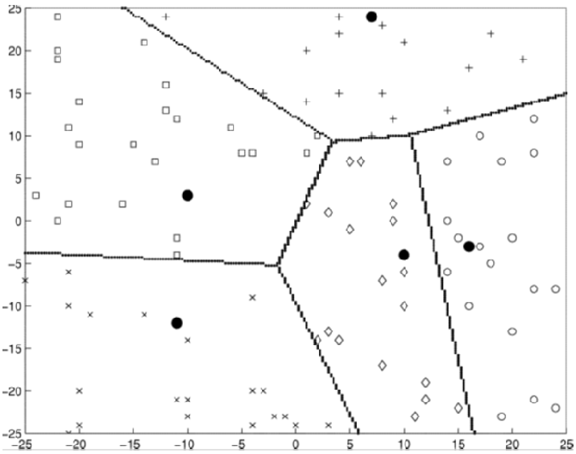


Fig. 1. EACH clustering architecture diagram

The choice of cluster head is using random method self-generated and power consumption of cluster head will be higher than other nodes. In order to prevent the death of the previously-selected cluster, a new cluster head will be re-elected after each round of data transfer.

Each round will compare and select a new cluster head, but it will shorten the lifetime of WSNs.

Two-Tier Data Dissemination(TTDD)

TTDD[11] is based on grid, it will create virtual grid in network environment. This method will create virtual grid in Advertisement phase as to find relative path. There will be many small grids, knows as "cell", in the grid structure. Data source will be the first Dissemination Nodes, then grids cross node as other Dissemination Nodes. Each grid node will be linked to a Dissemination Node and each Dissemination Node knows its upstream and downstream Dissemination Node. When Dissemination Node needs data, it will send a query message and this message will be transferred by Dissemination Node to data source. Then, data source transfers sensing data to Dissemination Node in reverse, shown as figure 2.

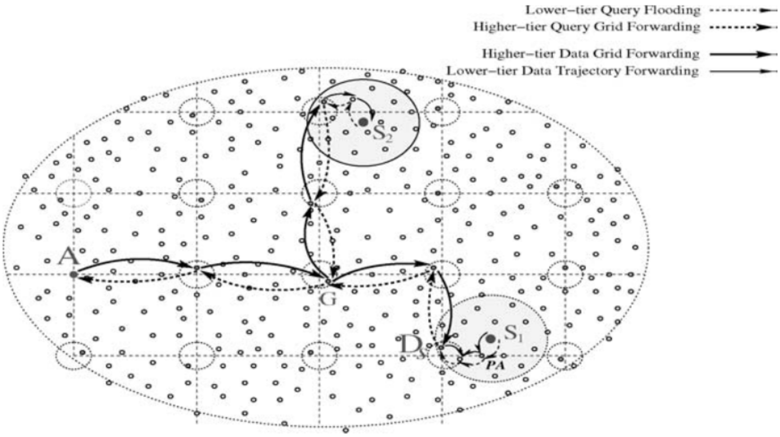

Lower–tier Query Flooding
Higher–tier Query Grid Forwarding
Higher–tier Data Grid Forwarding
Lower–tier Data Trajectory Forwarding
S2
A
G
D
S1
PA

Fig. 2. TTDD grid architecture diagram

Before the grid is created, position it needed and highly cost. Each time when there is a need to recreate grid or node as Dissemination Node, these nodes will cause the increase of power consumption. Overall, this method can be achieved fairly stable in transmission success rate.

3 A Routing Mechanism Using Virtual Coordination Anchor Node Apply to Wireless Sensor Networks.

This chapter will discuss the routing mechanism proposed above to overcome the problem caused by holes of random deployment and unknown boundary in WSNs.

Network Environment Settings

In this paper, Each node has its own Node_ID and transmission range to be recognized. In our architecture, R is the maximum transmission range; m is the unit of meter. The whole range are divided into four zones, each has its sending priority.

Create Virtual Coordinate and Find Anchor Node Phase

When wireless sensor nodes settle down, sink node starts to create virtual coordinate, then it will find the four directions of the entire network environment as the virtual anchor nodes. Therefore, sink node will create a special ANS packet. As shown in table 1, the packet has many different fields for sensor nodes to record coordinate status.

Table 1. ANS packet

Node_ID	Rec_ID	Pre_Coor	Self_Coor	Next_Coor	Status	Left_Node	Up_Node	Right_Node	Down_Node
						A0(X0,Y0)	A1(X1,Y1)	A2(X2,Y2)	A3(X3,Y3)

Node_ID: Unique identify number of node.
Rec_ID: Unique identify number of destination node.
Pre_Coor: Coordinate value of preview node.
Self_Coor: Coordinate value of current node.
Next_Coor: Coordinate value of next node.
Status: Current state, represent by Null, Warning, Bad, and NA.
Direction_Node: The utmost edge coordinator of the virtual anchor in the network

After sink node creates ANS packet, the Self_Coor will be set as (0,0) and according to the priority of sensor node to send beacon toward the priority1 direction, shown in figure 3.

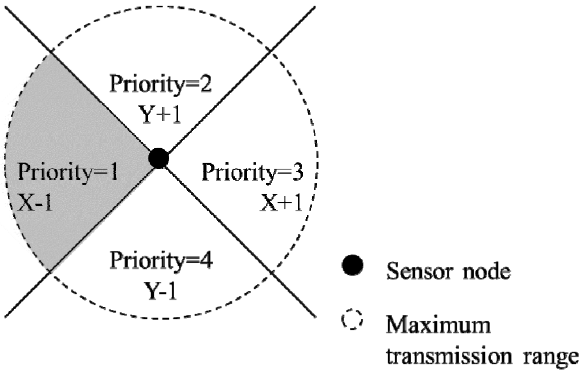


Fig. 3. Send beacon toward the priority1 direction

The Sender will choose the nearest node as the next ANS packet, and according to the definition of priority1 X-1, take out (0,0) from Self_Coor then execute X-1 to get (-1,0). Then, set Next_Coor of ANS packet as (-1,0) then send the fastest reply node.

When sensor nodes received ANS packet, it will copy Self_Coor from ANS packet to Pre_Coor as the coordinate of previous node, and copy Next_Coor to Self_Coor as the current coordinate to save to memory of node. We do this to ensure the link between each sensor node is correct. Then, we take Self_Coor respectively compare with the Left_Node, the Up_Node, the Right_Node and the Down_Node to see if that should Self_Coor replace anchor nodes of four corner or not.

If no ack packet response, it means priority1 direction doesn't have any sensor nodes, or the status of sensor nodes is bad, shown in figure4. Sensor node A doesn't sense any node from priority1 direction; likewise, sensor node A doesn't receive ack packet.

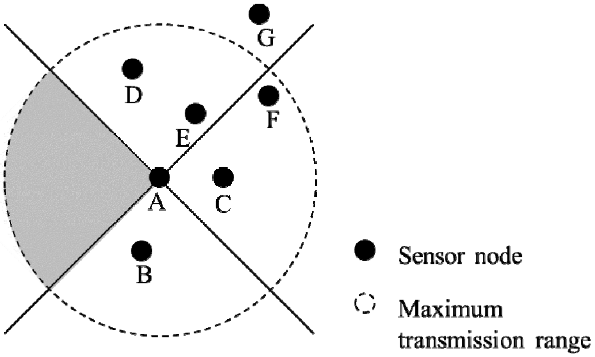


Fig. 4. Node A couldn't find other nodes in priority1 direction

At this phase, the status of node A will be set as Warning, then it will seek the node in sequence according to its priority until it find the next node.Then, it will find node D and node E.

Then it will seek according to the priority. The purpose of this mechanism giving the priority1 direction a chance again owning to the random distribution of sensor node, it is possible that the current direction temporary doesn't have any nodes and it will revise its direction by switching to the next priority, finding the sensor node again.

If sensor node in priority1 direction can't find any sensor and status is W, then priority will change to the next set priority, and clean the value of status, re-find nodes from priority1 direction, the switching sequence of the Priority group is from top-left to bottom-right, Shown as Figure 5.

Establishing virtual coordinates and searching of virtual anchors will last repeatedly until the ANS packet is received repeatedly by wireless sensor node. Those wireless sensor node receiving ANS packet repeatedly will produce Final Coor according to the packet it has received. The repeatedly-received packet will be abandoned after the production of Final Coor.

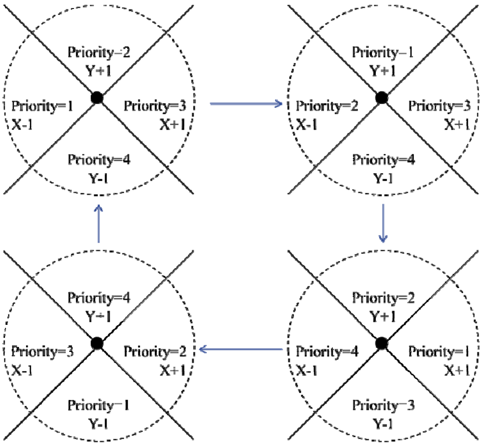


Fig. 5. Switching of Priority group

To Establish the Routes of the Virtual Anchors and Sink Node

In this stage, the establishment of routing begins with the four virtual anchors in Final Coor. If the virtual nodes are Left_Node or Right_Node, Pre_Coor and Next_Coor in ANS packet of the exact virtual node will be applied to search for the Y coordinate equal/lager than that of Self_Coor until it reaches the virtual node when Y coordinate is 0. If the virtual node is Up_Node or Down_Node, Pre_Coor and Next_Coor in the ANS packet of the exact virtual node will be applied to find the X coordinate which is smaller or equal to that of Self_Coor until it reaches the virtual node when X coordinate is 0.

Search for the Substitute Virtual Anchor or Route

When the electric quantity of a wireless sensor node—which serves as a virtual node or a route node—is lower than 20% as the sensor node A in Figure 6, the node will send Low Energy Message to its One Hop and inform the neighboring nodes.

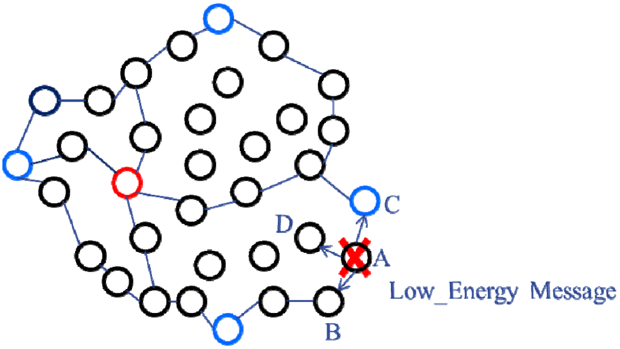


Fig. 6. power-exhausted Node Sending Low Energy Message

Meanwhile, it will regard itself as NA and will merely transfer data but will not serve as a virtual node.

When the node receives a Low Energy Message, it will make sure if it is the one on the border or the route.

After that, check if it has receive the same message before and check if the messages refer to the identical node. If confirmed, it will send a New_Anch_Chk Packet to the node which is searching for a new one and to create a new route.

4 Comparison and Analysis of Simulation

In this chapter, our approach will be programmed and stimulated in C++ and compared with Flooding, TTDD, LEACH, and Anchor Node Based Sink Location Dissemination Scheme for Geographic Routing. As Simulation parameter: 50 sensors with 10M transfer range under 100M X 100M, transfer a packet every 10 seconds.

Analysis and Comparison of Results

In the beginning, time is taken as the parameter to see its impact to the packet control by the wireless sensor node. The stimulating time ranges from 20 seconds to 200 seconds, shown as Figure 7.

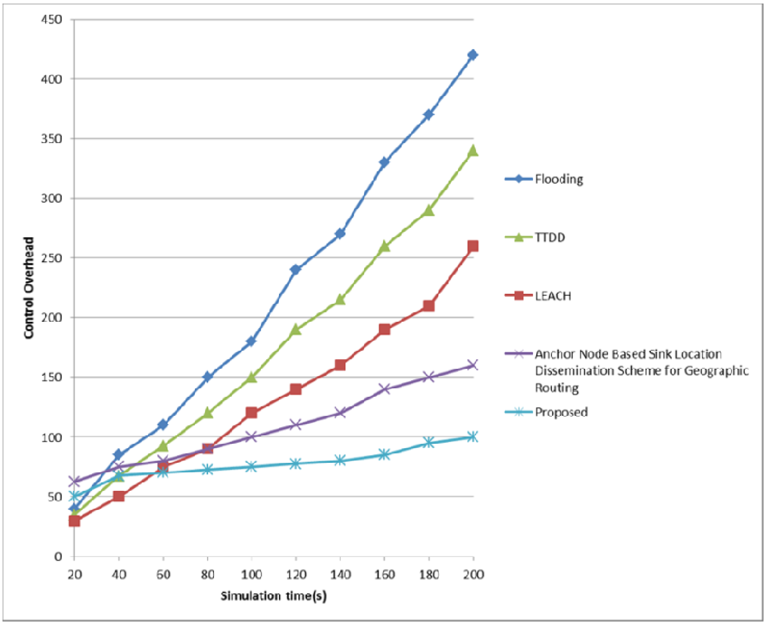


Fig. 7. Diagram of the number of the controlled packet and time

From the result, it is clear to find that the number of the controlled packet is much higher than that in other routing, mainly owing to reason that each wireless sensor node needs to participate in Flooding. For TTDD, a grid is established by the wireless sensor nodes which serve as the sources. The grid need to be maintained regularly, therefore the number of its controlled packet is lower than that in Flooding. In each round of LEACH, cluster head need to be compared and re-elected, resulting in the number of controlled packet lower than the previous two kinds of mechanisms. In this paper, a sink location announcement packet and a sink location query packet are sent along two respectively on the sink node location query route and sink location announcement route, so two routes are required in the query. Since there is no repair mechanism, an enormous amount of controlled packet is seen when the anchor break down.

ANS packet produced by the Sink Nodes will be transmitted along the border nodes and send back the anchor data along the same route. In this way, Flooding is not applied; instead, a substitute mechanism is available and substitute node is expected to be found once the route node is dead or breaking down.

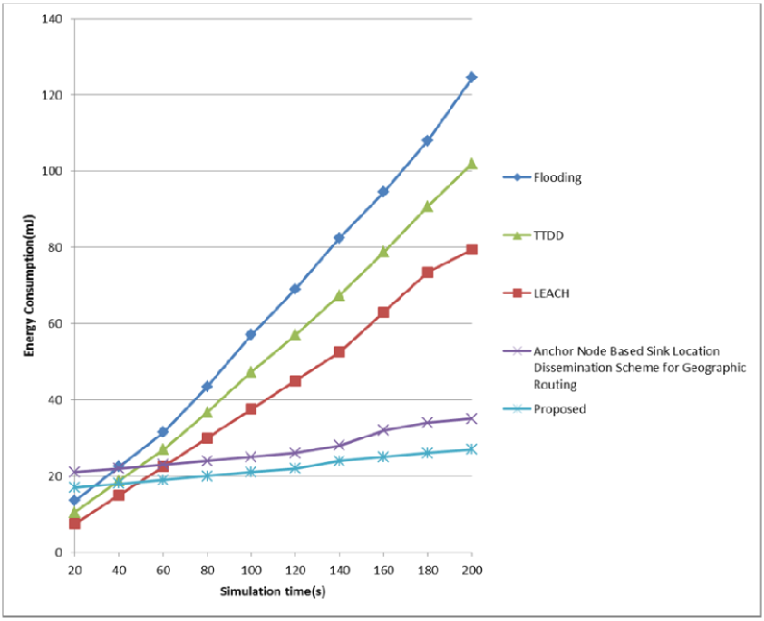


Fig. 8. Diagram of relation of time and energy consumption of wireless sensor node

From the diagram, it is revealed that both of the approach in this paper and that in "VCS" system share higher level of energy consumption in the beginning compared to other three routing since both approaches require more energy in initialization which involves the coordinates of a Pre_Node, a Next_Node and the coordinates of four border nodes. However, GPS devices is not necessary in this paper, therefore it causes lesser energy consumption than that in "VCS" system.

From two sets of stimulating experiment, it is apparent that “A Routing Mechanism Using Virtual Coordination Anchor Node” in this paper is a sufficient routing which can save energy, expand the lifetime of wireless sensor network, and reach a high delivering rate. Finally, the purpose of the research is achieved.

5 Conclusion and Future Research

From the result of the stimulation, it is obvious that the number of controlled packets as well as energy consumption is reduced under the routing protocol even the density of transferring packets is high; and the success rate of date transmission is raised as the survival rate of each node rises, which further ensures the function of the network and expand the lifetime of wireless sensor network. Meanwhile, power-saving mechanism is anticipated to be involved in the future research to lower the energy consumption in the initializing level of the network. In addition, cluster routing or tree routing protocol are wished to join the working of the network so the efficiency of the Internet could be maximized.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Computer Networks 38, 393–422 (2002)
2. Liu, P., Zhang, X., Tian, S., Zhao, Z., Sun, P.: A Novel Virtual Anchor Node-Based Localization Algorithm for Wireless Sensor Networks. In: Proceedings of the Sixth International Conference on Networking, p. 9 (2007)
3. Fucai, Y., Younghwan, C., Sang-Ha, K., Euisin, L.: Anchor Node Based Sink Location Dissemination Scheme for Geographic Routing. In: Proceedings of IEEE Vehicular Technology Conference, VTC 2008, pp. 2451–2455 (2008)
4. Fucai, Y., Younghwan, C., Soochang, P., Euisin, L., Ye, T., Minsuk, J., Sang-Ha, K.: Anchor Node Based Virtual Modeling of Holes in Wireless Sensor Networks. In: Proceedings of IEEE International Conference on Communications, ICC 2008, pp. 3120–3124 (May 2008)
5. Shankarananda, B.M., Saxena, A.: Energy efficient localized routing algorithm for Wireless Sensor Networks. In: Proceedings of 3rd International Conference on Electronics Computer Technology, ICECT 2011, pp. 72–75 (April 2011)
6. Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed diffusion: a scalable and robust communication paradigm for sensor networks. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, Massachusetts, United States, pp. 56–67 (2000)
7. Dressler, F., Awad, A., Gerla, M.: Inter-Domain Routing and Data Replication in Virtual Coordinate Based Networks. In: Proceedings of IEEE International Conference on Communications, ICC 2010, pp. 1–5 (May 2010)
8. Yaling, T., Yongbing, Z.: Hierarchical flow balancing protocol for data aggregation in wireless sensor networks. In: Proceedings of Computing, Communications and Applications Conference, January 11-13, pp. 7–12 (2012)

9. Kuo, T.-W., Tsai, M.-J.: On the construction of data aggregation tree with minimum energy cost in wireless sensor networks: NP-completeness and approximation algorithms. In: Proceedings of 31th Annual IEEE International Conference on Computer Communications, INFOCOM 2012, pp. 2591–2595 (March 2012)
10. Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor networks. IEEE Transactions on Wireless Communications 1, 660–670 (2002)
11. Luo, H., Ye, F., Cheng, J., Lu, S., Zhang, L.: TTDD: two-tier data dissemination in large-scale wireless sensor networks. Wirel. Netw. 11, 161–175 (2005)

Effect of Genetic Parameters in Tour Scheduling and Recommender Services for Electric Vehicles[*]

Junghoon Lee, Gyung-Leen Park, Hye-Jin Kim, Byung-Jun Lee,
Seulbi Lee, and Dae-Yong Im

Dept. of Computer Science and Statistics,
Jeju National University, 690-756, Jeju Do, Republic of Korea
{jhlee,glpark,hjkim82,eothsk,gwregx,dlaeodyd123}@jejunu.ac.kr

Abstract. This paper assesses the performance of a tour scheduling and recommender service for electric vehicles, aiming at verifying its effectiveness and practicality as a real-life application. The tour service, targeting at electric vehicles suffering from short driving range, generates a time-efficient tour and charging schedule. It combines two computing models, one for user-specified essential tour spots as the traveling salesman problem and the other for service-recommended optional spots as the orienteering problem. As it is designed based on genetic algorithms, this paper intensively measures the effect of the population size and the number of iterations to waiting time, tour length, and the number of visitable spots included in the final schedule. The experiment result, obtained through a prototype implementations, shows that our scheme can stably find an efficient tour schedule having a converged fitness value both on average and overloaded set of user selection.

Keywords: Electric vehicle, tour scheduler, genetic algorithm, waiting time, visitable node.

1 Introduction

Not just for energy efficiency, but also for the reduction of greenhouse gas emissions, the market penetration of EVs (Electric Vehicles) is encouraged in modern and future transportation systems. Especially on tour places having a bunch of natural attractions, clean air is more important. In those places, EV rent-a-cars are considered to be a promising business model. However, it is well known that the driving range of EVs is too short and their batteries must be charged more often [1]. It takes about 6 ~ 7 hours to fully charge an EV with slow chargers, and a fully charged EV can drive at most 150 *km*. Moreover, terrain and climate effect can further reduce the driving range. As the daily driving distance of ordinary vehicles is less than this range, overnight charging is enough. However, EVs rented for a tour can possibly drive beyond this range, and they need to be charged during the tour, wasting the tour time.

[*] This work (Grants No. C0026912) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded by Korea Small and Medium Business Administration in 2012.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 556–562, 2013.

The constraint in the driving range is sure to affect the tour schedule. Even if tourists want to visit a famous tour spot, they can't go there provided that the distance between the spot and the last charging facility is longer than the driving range on the way to the spot. Moreover, waiting time during the tour will be different according to visiting sequences. As a result, for EV rent-a-cars, sophisticated tour scheduling is more important to deal with the driving range constraint. Just like other facilities and services in the smart grid, EV rent-a-cars can benefit from the intelligence of the computing algorithms. Hence, our previous work has developed a tour scheduling scheme which creates the visiting order capable of reducing the waiting time for EV charging for the given set of user-selected attractions [2]. It can further recommend additional tour spots having chargers to avoid time waste in a tour.

For the integration of such a service into the real-life product, it is necessary to evaluate and verify its performance and reliability. It includes many execution variables including the omission degree, charging facility probability, and stay time distribution. Basically, as this scheme is built on top of genetic algorithms, the performance behavior according to the genetic parameters is the first to be investigated. In this regard, this paper measures the performance of the EV tour scheduling and recommender system based on a prototype implementation. The performance metrics are consist of waiting time, tour length, and the number of visitable tour spots according to the population size and the number of iterations. With the performance data collected by the experiment, its practicality as a commercialized service will be assessed.

2 Background

For the given set of nodes, deciding a visiting order is a typical TSP (Traveling Salesman Problem), and there have been many researches and applications for them. However, this application belongs to non-polynomial complexity problems and has different cost functions. Sometimes, two or more goals may conflict. In the case of tour schedulers, each tour spot is associated with a profit, or degree of user satisfaction and it is not necessary to visit all spots. In addition, every time a tourist moves from one spot to another, travel cost is added. According to the survey of [3], one of the most efficient methods to solve such a problem is to define an object function which gives precedence to profit maximization, while taking the travel cost as a constraint. This problem type is called an orienteering problem, and a genetic algorithm has been designed for it [4]. In its encoding, a vertex will be removed by the omission probability and not every vertex will be included in each chromosome.

Our previous work has designed a tour scheduling and recommender service for EVs to enrich EV rent-a-car business by computational intelligence [2]. For the set of user-selected tour places, it finds the visiting order and where to charge the EV, considering the inter-spot distance as well as tracing battery remaining. However, if a tourist wants to visit a series of spots far away from each other, he or she must stop by a charging station and wait for his or her EV to be charged. Instead, our service recommends additional spots in which the tourist can take another tour activity while the EV is being charged, even though they are not selected at first. In this design, genetic operations are tailored to create a tour plan consisting of essential selected and optional recommended places by means of combining legacy traveling salesman problem and orienteering problem solvers. Its encoding scheme represents a visiting order by a fixed-length integer-valued vector, while the fitness function estimates time waste for a tour route.

3 Service Scenario

For EV-based tours, the renters select the set of tourist attractions they want to visit and the tour planner or recommender helps them to decide the visiting order [5]. It has the time and space complexity of O(n!), where n is the number of attractions. So, computerized selection will be better than human calculation. Genetic algorithms investigate just a part of the whole search tree to meet the time constraint [3]. That is, the schedule must be created within user-tolerable response time. If a tour spot has a charging facility, the EV can be charged during the renters take a tour. Battery remaining increases in proportion to the stay time at the place [6]. On the contrary, when no charging facility is available, the EV gains no battery charging. This makes the visiting sequence more critical to waiting time.

Waiting time can be serious enough to make travelers feel severely inconvenient, if they cannot do anything while their EVs are being charged. Instead, they can avoid this waste, if they visit a place where both tour activities and charging facilities are available, even though the places are not selected at first. The list of recommendable spots is available in tour information services and can be retrieved through spatial queries. While all the places selected by the tourists must be included in the final tour plan, the recommendable places don't always have to be included. After all, the route planner for the EV-based tour is a combination of the legacy TSP solver for the user-selected places and the orienteering problem solver for recommendable places. The planner pursues the reduction of waiting time and the enhancement of tourists' satisfaction. It can be quantified by the number of visitiable places or the sum of satisfaction degrees for visitable spots.

For a genetic algorithm-based design, each schedule is encoded to an integer-valued vector. Its length coincides with the total number of both user-selected and service-recommended spots. In this fixed size vector, some of recommended spots are omitted and marked by -1. Next, the fitness function calculates the cost for a schedule. As this service focuses on the waiting time for EV charging, it is necessary to follow the sequence to find out where and how much charging is required, considering the stay time usually available in tour statistics. Finally, the genetic operators are run generation by generation. In the mean time, after crossover operations, some essential entries appear more than once while others become absent. Hence, the duplicated entries are replaced by disappearing ones. On the contrary, for optional spots, duplicated entries are replaced by other optional ones.

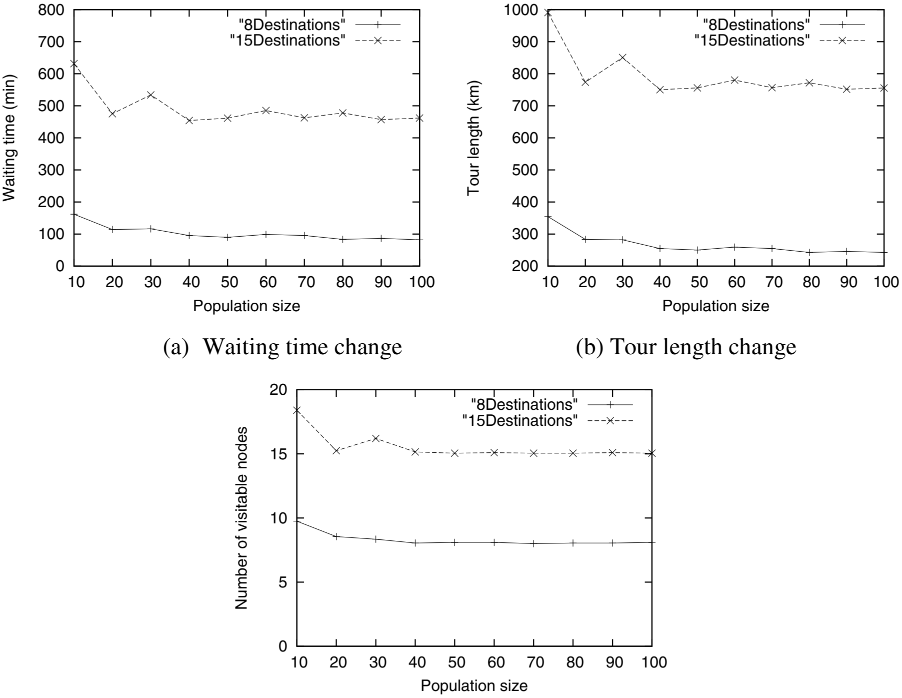
4 Experiment Result

Before the experiment on the genetic parameters, the scheduler-specific parameters need to be chosen, mainly considering the target tour environment. In our experiment, each tour spot is located randomly in the map. The inter-spot distance exponentially distributes with the average of 20 *km*. For two destinations *A* and *B*, the distance from *A* to *B* is different from that from *B* to *A*. They are not symmetric, but the difference is less than 5 %. Next, stay time also distributes exponentially with the average of 20 *minutes*. But, its lower and upper bounds are 20 *minutes* and 3 *hours*, respectively. The probability that a tour spot has a charger is set to 0.8 and the omission degree is to 0.8. Finally, an EV is assumed to start a day trip with it battery charged enough to go 90 *km*, considering overnight charging. The experiment generates 20 parameter sets for each parameter setting, and averages the results.

In genetic algorithms, the population size affects the diversity of chromosomes. However, if it is large, execution time will also increase. For applications having a constraint in the maximum response time, a large population does not always find a better solution, as the number of iterations can be limited. Particularly, each genetic loop sorts or at least partitions chromosomes in the population, so its time complexity can be approximated to be O(*p log p*), where p is the population size, or the number of chromosomes. Next, with more iterations, we can expect to get better solutions. However, in most cases, the genetic loop converges to a reasonable solution very quickly in the early stage of the whole generations. Thus, the fitness value remains unchanged in the subsequent iterations. After all, it is important to find an efficient parameter selection capable of obtaining an acceptable schedule within the given time bound.

Each experiment is conducted for the cases of 8 and 15 destinations, respectively. The first accounts for the most common tour pattern and the second for the overloaded condition, in which genetic algorithms can possibly fail to converge to an acceptable schedule. Hence, they can check the practicality and the stability of our tour scheduling and recommender service, respectively. Here, the number of recommended destinations is set to 15. Hence, the tour scheduler takes either 23 or 30 destinations in total. It cannot be calculated with exhaustive search methods and genetic algorithms can investigate only the part of the whole search space. Actually, recommended spots are supposed to be retrieved from the spatial database. However, as the charging facility map is not available yet, our experiments locate them randomly. As a result, in spite of the increase in the number of recommended destinations, waiting time does not always get better, as they can be located far away from the feasible routes.

The first experiment measures the effect of population size to waiting time, tour length, and the number of visitable destinations, respectively, and the results are shown in Figure 1. In this experiment, we change the population size from 10 to 100. As the fitness function mainly calculates the waiting time for each schedule, a large population can reduce waiting time by the improved diversity. In the case of 8 destinations, the waiting time is 161.7 *minutes* when the population size is 10 and gets improved to 82.55 *minutes* when the population size is 100. It corresponds to 42.9 % reduction as shown in Figure 1(a). In the case of 15 destinations, we can see 26.9 % improvement. Even if waiting time is reduced thanks to the increase in the population size, it hardly gets better after the population size of 50. Further increase in the population size just leads to the extension of execution time.



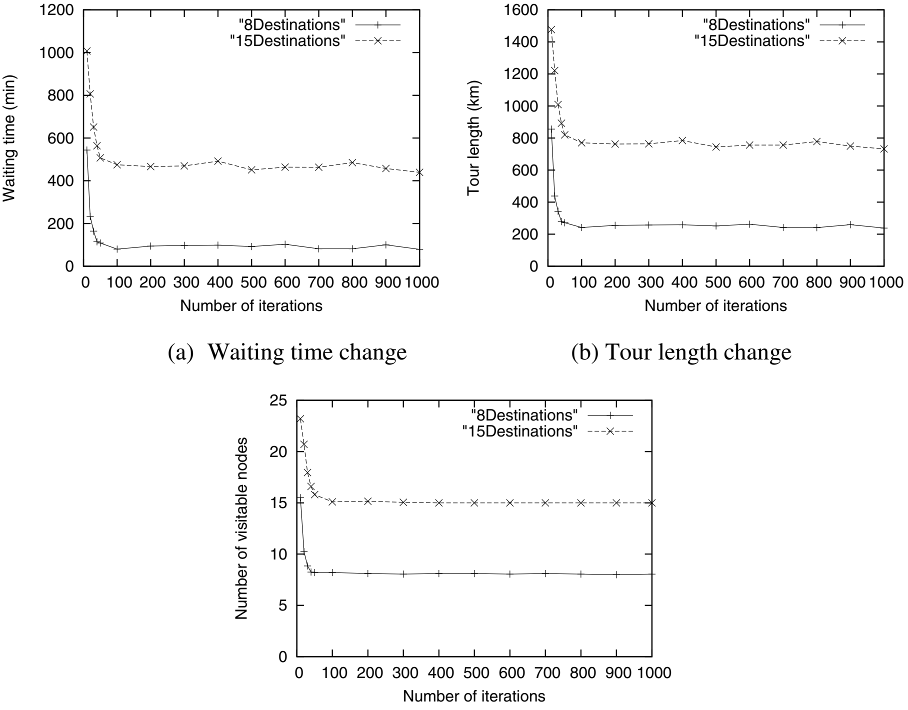
(a) Waiting time change

(b) Tour length change

(c) Change in the number of visitable nodes

Fig. 1. Effect of population size

Tour length is closely related to waiting time, so they show a similar pattern. If tour length increases, the tourists are likely to charge their EV for longer time. The increase in the population size from 10 to 100 results in the reduction of tour length by 31.6 % in the case of 8 destinations and 23.7 % in the case of 15 destinations, respectively, as shown in Figure 1(b). On the contrary, the number of visitable spots decreases according to the increase of the population size. Actually, the reduction in waiting time tends to remove non-essential destinations in the tour schedule. A recommended destination can contribute to the reduction of waiting time when it is located between two stations unreachably far away from each other with average battery remaining. Hence the number of visitable spots is reduced from 9.75 to 8.1 with the improvement in the efficiency of the tour schedule, as shown in Figure 1(c).



(a) Waiting time change

(b) Tour length change

(c) Change in the number of visitable nodes

Fig. 2. Effect of the number of iterations

The next experiment measures the effect of the number of iterations to waiting time, tour length, and the number of visitable destinations, respectively, and the results are shown in Figure 2. In this experiment, we change the number of iterations from 10 to 1,000. Genetic iterations improve the fitness of the solution generation by generation. Our main concern lies in how many iterations are usually needed to get the converged result. In the case of 8 destinations, waiting time is 544 *minutes* at first and cut down to 78.7 *minutes* with 1,000 iterations, showing 85.6 % improvement, as shown in Figure 2(a). In addition, in the case of 15 destinations, waiting time is reduced from 1,007 *minutes* to 439 *minutes*, showing 56.4 % improvement. As the experiment generates the destination set randomly, each set has a different optimal schedule. Hence, even with more iterations, waiting time may increase as respective fitness values are averaged.

As in the case of the experiment on population size, tour length shows a similar pattern as waiting time. It is shown in Figure 2(b). Tour is length is less sensitive to the number of iterations, compared with waiting time, as the reduction is 72.1% in the case of 8 destinations and 50.4 % in the case of 15 destinations, respectively. In addition, just like waiting time, tour length reaches a stable value within 100 iterations in most cases, and then rarely changes. Finally, as for the number of visitable spots, it is reduced according to the enhancement of waiting time. Particularly, in the case of

15 destinations, no recommended spot survives in the final tour schedule after 500 iterations. This result indicates that the locations of recommended spots are important and accurate spatial information is essential to this service. Additionally, in the case of 8 destinations, the number of visitable spots is cut down from 15.5 to 8.05.

5 Conclusions

Due to their eco-friendliness, EVs are encouraged in many tour cities which have many natural attractions, not just for personal ownership but also car sharing and rent-a-car services. However, the short driving range of EVs is the main obstacle and inconvenience factor in EV rent-a-car services. In this paper, we have assessed the performance of a tour scheduling and recommender system, focusing on the effect of genetic algorithm-specific parameters such as the population size and the number of iterations to waiting time, tour length, and the number of visitable tour spots included in the final schedule. Combining the traveling salesman problem and the orienteering problem, this service generates a visiting sequence for user-specified essential and service-recommended optional tour spots. The measurement result shows that our scheme can stably find an efficient tour schedule having a converged fitness value both on average and overloaded set of user selection. Moreover, waiting time can be managed below 1 *hour* for the given tour scenarios.

References

1. Bessler, S., Grønbæk, J.: Routing EV users towards an Optimal Charging Plan. In: International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium (2012)
2. Lee, J., Park, G.: A Tour Recommendation Service for Electric Vehicles Based on a Hybrid Orienteering Model. Submitted to ACM SAC (2013)
3. Giardini, G., Kalmar-Nagy, T.: Genetic Algorithm for Combinational Path Planning: The Subtour Problem. Mathematical Problems in Engineering (February 2011)
4. Tasgetiren, M., Smith, A.: A Genetic Algorithm for the Orienteering Problem. In: Proc. Congress on Evolutionary Computing. pp. 1190–1195 (2000)
5. Ferreira, J., Pereira, P., Filipe, P., Afonso, J.: Recommender System for Drivers of Electric Vehicles. In: Proc. International Conference on Electronic Computer Technology, pp. 244–248 (2011)
6. Lee, J., Kim, H.-J., Park, G.-L.: Integration of Battery Charging to Tour Schedule Generation for an EV-Based Rent-a-Car Business. In: Tan, Y., Shi, Y., Ji, Z. (eds.) ICSI 2012, Part II. LNCS, vol. 7332, pp. 399–406. Springer, Heidelberg (2012)

Enabling Massive Machine-to-Machine Communications in LTE-Advanced

Kyungkoo Jun

Dept. of Embedded Systems Engineering, University of Incheon, Korea
kjun@incheon.ac.kr

Abstract. The worldwide markets for Machine-to-machine (M2M) communications over cellular networks are expected to grow for the time being. However, since the M2M communications has the characteristics that are quite different from existing human-centered wireless communications, it poses significant challe. In this paper, we address the challenges in facilitating M2M call admission and scheduling over LTE-A networks and propose a call admission method that improves the call drop probability. Such improment is made possible by adjusting the maximum number of the QoS classes. The simulation results show that the proposed method is superior to other fixed QoS class schemes in terms of the call drop probability.

Keywords: M2M communications, LTE-Advanced, call admission control, delay constraint, QoS.

1 Introduction

Machine-to-machine (M2M) communications over cellular networks has the characteristics that are quite different from existing human-centered wireless communications. For example, those are a large number of devices, small-sized data transmissions, and diverse ranges of applications. It means that M2M communications pose significant challenges which have not been experienced in current communications systems. Since current wireless solutions are originated from general packet radio service, they cannot be a good fit for the M2M systems. Thus, more recent cellular wireless systems such as long-term evolution (LTE) and LTE-Advanced (LTE-A) are expected to accommodate the requirements of the M2M systems. However, such advanced systems still require the improvements in resource allocation and scheduling in order to deal with increased signaling overhead and meet the wide span of QoS requirements.

It is expected that the worldwide markets for M2M communications over cellular networks would grow for the time being. To this end, the cellular systems which are already deployed or will be in near future provide solid foundations to build up the M2M infrastructure because of their cost-competitiveness and proved technologies. According to recent surveys [1][2], more than 500 million devices will be connected through the M2M cellular communications by the end of 2015. It implicitly means that there will be a massive number of M2M devices that one cell should handle and it

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 563–569, 2013.

presents a significant challenge. On the other hand, it is hard to imagine the boundary of M2M applications. The span of the applications includes smart metering, monitoring related with ubiquitous health care and sensing for intelligent transportation systems. To enable such applications, the standardization bodies such as 3GPP and IEEE have began to evolve current systems in order to facilitate the development, deployment and operation of such applications.

Current cellular systems provide limited support for realization of the M2M communications. For example, the short message service is one of the ways that enable the M2M communications. However, it is only possible under the condition that the number of the participating M2M devices is relatively small. In other words, the capacity of the cellular systems is a major huddle to deploy the M2M systems. Even though the cell capacity is determined by several factors such as the partitioning and the reuse patterns, it is quite smaller compared with the requirement for the scenario in which one cell hosts more than thousands of the M2M devices. In addition, the fact that the cellular networks of today mostly support the terminal-initiated communication setup is another obstacle [3].

Today, 3GPP LTE is considered to provide the most promising technologies for the realization of the M2M communications. It provides higher capacity and improves radio resource management in terms of flexibility compared with existing cellular systems. However there exists a huge gap between the design initiatives of LTE and that of M2M. While LTE is engineered to support broadband applications, the M2M applications are mostly narrowband, i.e. sending and receiving small amounts of data. In addition, the requirements of low power consumption of the M2M applications do not go well with the design consideration of LTE. To this end, several work groups of 3GPP have been devoted to conduct study on these issues and come up with feasible solutions [4]. These efforts deal with various issues such as energy efficiency, meeting diverse QoS requirements and coexistence with current communications infra as well as accommodating a vast number of the M2M devices.

LTE improves its flexibility in resource allocation by exploiting the two dimensional resources consisting of time and frequency and fine-grained scheduling unit of physical resource blocks (PRBs). Nevertheless, to support the M2M communication in addition to user equipment (UE) communications, LTE is required to work on a faster time scale such as transmission time interval (TTI). It is then able to deal with more dynamic and diverse channel and traffic changes. On the other hand, more complex and efficient signaling mechanisms are required in order to carry not only scheduling information but also additional information that facilitates scheduling such as channel quality.

In this paper, we present the challenges in facilitating M2M call admission and scheduling over LTE-A networks and propose a call admission method that satisfies the delay requirements. This paper is organized as follows. Section 2 descries the M2M challenges and discusses the proposed method in the context. Section 3 presents the simulation results and Section 4 concludes the paper.

2 Challenges of M2M communications in LTE and Proposed Call Admission Control

Most traffic in the M2M communications is delivered in the uplink direction, because the applications are mainly built upon monitoring and sensing. Regarding the uplink communications, the scheduling decision is made at the base station which is called eNodeB in 3GPP, and then it is notified to UEs via control channels.

As the first step to request the uplink allocation, an UE sends a scheduling request to the eNodeB via PUCCH. And each UE is assigned one PUCCH for this purpose. It means that a large number of M2M devices may cause the shortage of PUCCH resources. Such shortage aggravates more when the M2M devices report the per-device channel quality information to the eNodeB through the uplink channel. In addition, as the number of the M2M devices increases, so does the uplink signaling load.

Now, consider the second step of the uplink allocation. Upon receiving the scheduling requests, the eNodeB determines the grant of physical resource block (PRB) to UEs at each TTI and inform the UEs of the grant information via physical downlink control channel (PDCCH). Again, the shortage of PDCCH may occur because maximum ten UEs can be allocated with PDCCH in a single subframe. Besides PDCCH, LTE provides a random access transport channel for the uplink. However it is still not sufficient. The random access transport channel can be used for the uplink scheduling request in a contention-based way while the PUCCH operates in a contention-free way. To occupy the random access transport channel, the UEs randomly choose one of orthogonal sequences and send the request by using the sequence. Since there are only 64 sequences available per cell, collisions may occur if more than one UE selects a same sequence. If we imagine the scenario that thousands of the M2M devices compete for the uplink allocations at the same time, the current scheduling scheme cannot support it. Thus, modifications or new design are required for LTE to support the M2M communications.

Today, the scheduling in LTE is performed in two domains. The first is related with time domain. For the current time frame, a set of UEs is selected for transmission opportunities. And the second is about frequency domain. The selected UEs are allocated with the PRBs. The scheduling decision is mainly based on QoS requirements. To this end, the 3GPP specifications introduce a set of nine QoS classes which are denoted by QoS class identifiers (QCI), priority, packet delay budget, tolerable packet loss rate and radio bear types such as guaranteed bit rate (GBR) or non-GBR.

However, the M2M communications complicate the scheduling of LTE because of the broad span of QoS requirements. For example, the delay budgets range from tens of milliseconds to minutes, in some cases, up to several hours, and the tolerable packet loss rate also widens its boundaries. Therefore, the idea of the QoS classes does not work well in these situations. Furthermore, the scheduling needs to collect and analyze traffic and channel status from the M2M devices before making the allocations. It means that the habitation of a massive number of the M2M devices complicates this work by increased delay and pressure on the limited storage of the eNodeB. Thus, it is a significant challenge to integrate the support for the M2M communications while minimizing negative impact on standard LTE services.

A group-based scheduling [5] proposes to reduce complexity for managing radio resources and scheduling of the M2M communications by forming the groups or clusters of the M2M devices. Each group is associated with a predefined QoS profile. Then the eNodeB needs to only control the groups while the M2M devices connect to the eNodeB transparently in the context of their belonging groups. Under this scheme, the eNodeB determines the scheduling priority to the groups, not individual M2M device, the overhead and complexity can be reduced. The QoS profile consists of data arrival rate, tolerable delay and jitters, etc.

A semi-persistent scheduling [6] is proposed to decrease the complexity by reducing the scheduling frequency. It makes an allocation decision for an extended time period which is longer than TTI, even though the scheduling per TTI is optimal in terms of system utilization and performance. It relieves the eNodeB of the burden of informing the M2M devices on a TTI basis

To this end, we propose a call admission control scheme for the M2M communications in LTE. It is a hybrid of the group-based scheme and the semi-persistent method. Although the group based scheme decreases the signaling overhead, it brings about another issue because of the need to specify the QoS classes that cover the requirements of the M2M devices. Considering the broad range of the M2M requirements, the number of the different classes would be infinite. It means that the categorization of the classes should be designed to capture the characteristics of the M2M applications. In our proposed scheme, the QoS class is defined by the data inter-arrival time and the tolerable maximum delay. Moreover, we support the dynamic creation of the QoS classes according to the presence of the M2M devices of which traffic characteristics cannot be captured by the existing classes. Such dynamic group formation is appropriate given than the M2M traffic patterns and the characteristics are not a priori known.

However, we limit the maximum number of the QoS classes, *N* that are managed by the eNodeB. Otherwise, the overheads such as the signaling increase beyond the capacity of the eNodeB. Considering the effect of the grouping granularity on system performance, it is important to find an optimal number of the QoS classes depending on situations and applications. In [7], the percentage of the M2M devices that violates their delay constraints is compared between using only the standard QCI classes and using additional classes that capture the whole span of the requirements. It shows that the case of using the additional classes is superior to the other case. It implicitly means that the system performance is linearly proportional to the number of the supported classes unless we consider the overheads.

In our scheme, we adjusts *N* according to the average population of the M2M devices that belong to a same group as follows.

$$N = N \cdot \left(\frac{\sum_{i=1}^{N} N_i}{N} \cdot \frac{1}{N_T}\right) \tag{1}$$

where N_i is the number of the M2M devices that belong to group i, N_T is a predefined threshold. As *N* increases, new groups with finer-grained QoS profile are created, while existing groups are coalesced under a group with a coarse-grained profile as N decreases.

Fig. 1 shows the algorithm of the proposed call admission control scheme. An M2M device asks the call admission by sending its QoS requirement to the eNodeB. If there already exists a group that can satisfy the requirement and the group has available resources during its granted time interval (GTI), the M2M device is admitted and joins to the group. Otherwise the creation of a new group is attempted. Using Eq. 1, if N is calculated and is bigger than current N, whether the creation of a new group affect the satisfaction of the QoS requirements of the existing groups is checked. It is achieved by checking a sufficient condition of Eq. (3) for all the existing groups including a new group as follows [8].

$$w(i) = \sum_{k=1}^{i} \tau \cdot \left\lceil \frac{p_i}{p_k} \right\rceil \tag{2}$$

$$w(i) \leq d_i, for\ all\ i = 1, \cdots, N \tag{3}$$

where τ is the duration of GTI, p_i and p_k are the data inter-arrival time of group i and k, respectively, and d_i is the tolerable maximum delay of group i.

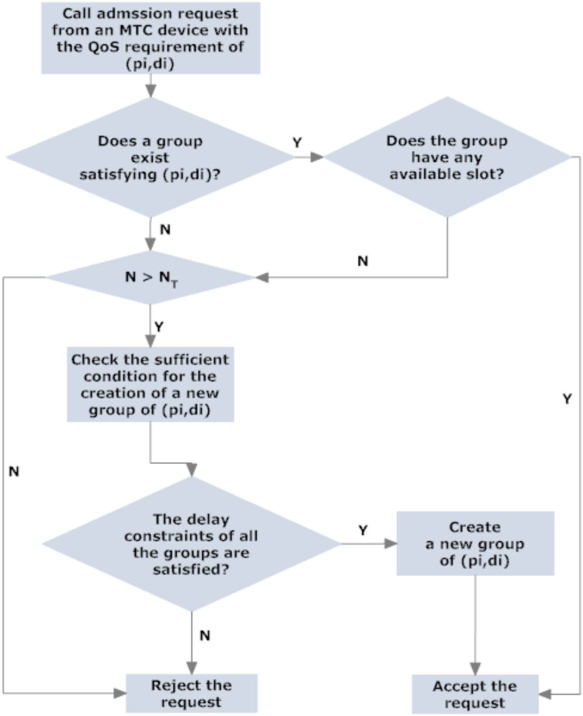

Call admssion request from an MTC device with the QoS requirement of (pi,di)
Does a group exist satisfying (pi,di)?
Y
Does the group have any available slot?
N
N > N_T
N
Y
Check the sufficient condition for the creation of a new group of (pi,di)
Y
N
The delay constraints of all the groups are satisfied?
Y
Create a new group of (pi,di)
N
Reject the request
Accept the request

Fig. 1. Call admission control algorithm of the proposed method

3 Performance Evaluation

We evaluate the proposed method by simulations. The Simulations assume that there is one eNodeB and its bandwidth is 20 MHz. One GTI consists of 100 PRBs and its duration τ is 1 ms. Thus one GTI can support up to 20 M2M devices if the data amount of the devices fit into 5 PRBs.

In the simulations, the M2M devices are created one by one with the creation interval that follows the exponential distribution with mean 1 ms. Once created, the M2M device choose randomly its QoS requirement that consists of data inter-transmission time and tolerable delay. And the M2M device, if it is admitted by the eNodeB, has the call duration that follows the exponential distribution with mean 15 ms. After completing its call, the M2M device dos not attempt more calls. Each simulations runs around 1000 sec. We average the results from a set of the repeated simulations.

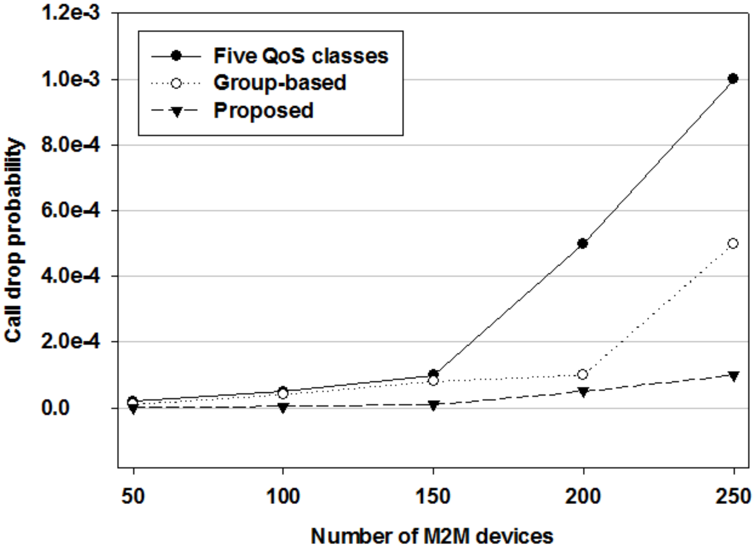


Fig. 2. Comparison of call drop probability

Fig. 2 depicts the performance of the proposed in comparison with those of the group-based scheme and the standard LTE call admission scheme. We assume that the standard scheme employs five QoS classes which is noted as 'Five QoS classes' in the figure and the group-based scheme can use five more classes in additional to the standard classes of LTE. Our scheme can increase the number of the QoS classes up to 20 including the standard classes. It can be seen that, using the flexible number of groups, the probability of the call drop of the M2M devices is lower than the ones using the fixed number of groups.

4 Conclusions

Machine-to-machine (M2M) communications over cellular networks has the characteristics that are quite different from existing human-centered wireless communications. It means that M2M communications pose significant challenges which have not been experienced in current communications systems. In this paper, we address the challenges in facilitating M2M call admission over LTE-A networks and propose a call admission method that satisfies the delay requirements. The proposed method improves the call drop probability by adjusting the maximum number of the QoS classes. The simulation results show that the proposed method is superior to other schemes that employ the fixed number of the QoS classes in terms of the call drop probability.

References

1. Machine-to-machine (M2M), the Rise of the Machines. Juniper Networks White Paper (2011)
2. Vodafone: RACH Intensity of Time Controlled Devices. 3GPP Tech. Rep., R2-102296 (2010)
3. Martsola, M., Kiravuo, T., Lindqvist, J.: Machine to Machine Communication in Cellular Networks. In: Proc. of 2nd Int. Conf. Mobile Technology, Applications and Systems, p. 6 (2005)
4. 3GPP. System Improvements for Machine Type Communications, Rel. 11. TR 23.888 (2011)
5. Lien, S., Chen, K., Lin, Y.: Toward Ubiquitous Massive Accesses in 3GPP Machine-to-machine communications. IEEE Comm. Mag. 49(4), 66–74 (2011)
6. Jiang, D., Wang, H., Malkamaki, E., Tuomaala, E.: Principle and Performance of Semi-Persistent Scheduling for VoIP in LTE system. In: Proc. of Int. Conf. Wireless Communications, Networking and Mobile Computing, pp. 2861–2864 (2007)
7. Lioumpas, A., Alexiou, A.: Uplink Scheduling for Machine-to-Machine Communications in LTE-based Cellular Systems. In: Proc. of IEEE Globecom Conf., pp. 353–357 (2011)
8. Jun, K.: Call Admission Control satisfying Delay Contraint for Machine-to-Machine Communications in LTE-Advanced. In: Proc. of FGNC, pp. 48–55 (2012)

Enhancements for Local Repair in AODV-Based Ad-Hoc Networks

Hyun-Ho Shin[1], Seungjin Lee[2], and Byung-Seo Kim[2]

[1] Dept. of Telecommunication System, SamSung Electronics, Korea
[2] Dept. of Computer and Information Communications Eng. Hongik University, Korea
{hyunho1986,seungjin1230}@gmail.com, jsnbs@hongik.ac.kr

Abstract. Route recovery process of Ad-hoc On-demand Distance Vector (AODV) protocol has been extensively studied. However, the recovery process still requires long delays and overheads. In this paper, an enhanced method to perform quick local recovery process is proposed. In the proposed method, when a link is broken, a node detecting a link-break asks to neighbor nodes who can be a substitute for a node causing the link– break. If there is such a node, then the recovery is quickly and locally completed. The proposed method does not increase overhead to find the substitute comparing to the conventional AODV protocol. This paper provides only the idea at this time, but the performance evaluations for the proposed method will be provided in the upcoming works.

Keywords: WRP, AODV, Ad-Hoc, Route-Recovery.

1 Introduction

In a past decade, wireless ad-hoc networks have been extensively researched be-cause they have capabilities of self-configurable and self-healing, flexibility and scalability. As a results, applications based on wireless ad-hoc networks remarkably increase, for example, vehicular networks, Machine-to-Machine communications (M2M), internet of things, future tactical networks, public safety networks and so on [1][2]. The wireless ad-hoc networks enable nodes to communicate over wireless multi-hop distances without any infrastructures. In order to implement this capability, the networks require Wireless Routing Protocols (WRPs) to find the optimal multi-hop path from the source to the destination. One of the well-known WRPs is Ad-hoc On-demand Distance Vector (AODV) routing protocol. While the protocol uses routing tables like routing protocols for wired networks, it searches the route only when it is needed, so that it reduce the overhead maintaining unnecessary route information [3].

Unlike conventional routing protocols used in the wired networks, WRPs are critical to the overheads and channel conditions. Since the channel conditions and network topologies in the wireless networks have time-varying nature, the built routes are frequently broken and recovered. Therefore, in WRPs, how fast to detect and to recover the broken links are essential research area for WRPs as shown in [4]-[10]. The link breaks in [4] are detected by a data transmission failure in Medium Access

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 570–576, 2013.

Control (MAC) layer. The method in [5] proposes the detection based on the quality of wireless channel measured from a physical layer. Unlike [4] and [5], the patent application in [6] proposes detection method by a node itself causing a link break. The method will be explained in detail in Section 2. The studies in [7]-[10] propose the enhanced local repairs. The enhanced repair methods will be introduced in detail in Section 2. This paper also proposes a way to quickly recover the broken link. In particular, we focus on the enhancement on the local repair which is one of route repair method defined in AODV protocol [3].

Section 2 illustrates not only the local repair of AODV protocol and a link-break detection method that is our previous work. In addition, prior arts on the local repair are introduced. In Section 3, after introducing the motivation of this paper, the proposed method is described in detail. After the proposed method is logical compared with the prior arts in Section 4, finally, the conclusion is made in the last section.

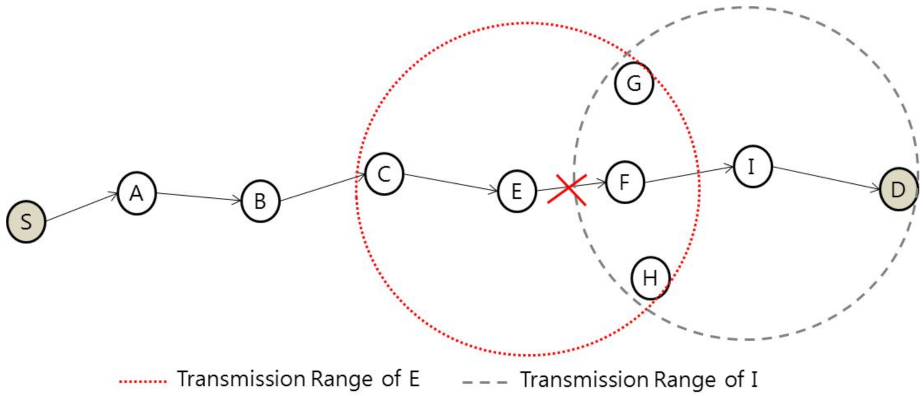


Fig. 1. An example of a target network

2 Local Repair Process and Self-Link-Breakage Detection

AODV protocol defines two types of route repair processes. One way is to find the whole new route from the source to destination, called source repair hereinafter. The other method, called Local Repair, is to find a new route from the upstream node of the broken link to the destination. For example, in the networks shown in Fig. 1, the on-going route from a source, Node S, to a destination, Node D, is currently set to Node S->A->B->C->E->F->I->D as the arrow indicates in the figure. If a link between Node E and F is broken and the number of hops from Node E to the destination is shorter than that from Node E to the source, Node E broadcasts RREQ message to find a new route to Node I instead of sending Route ERRor (RERR) message to the source to initiate a new whole route discovery process. However, in certain cases, the local repair causes the performance degradations as described in our previous works [11]. If the local repair is not successfully completed, the source repair process needs to be started, so that the delay to recover the broken link increases more. In particular, when the route to be locally repaired is long, the performance degradation increases because other links in the upstream route can be broken during performing the local repair. Moreover, when Address Resolution Protocol (ARP) is operated with AODV protocol, the loss of ARP request packet causes the failure of the local repair described in [11].

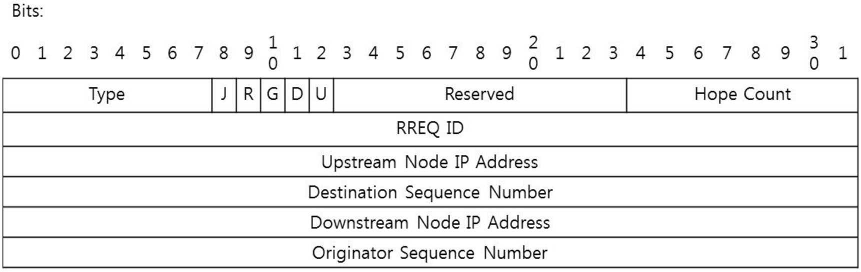
As mentioned in Section 1, there are some studies on the local repairs as shown in [7]-[10]. In [7], Proximity Approach To Connection Healing (PATCH) has been proposed. PATCH replaces the broken one-hop link with two-hop link by finding a new node between the end nodes of the broken link. Enhanced Local Repair AODV (ELRAODV) proposed in [8] repairs the broken link by replacing the other new node. For example, in Fig. 1, if the link between Node E and F is broken, ELRAODV makes new route following Node E->G->I or E->H->I. For this, ELRAODV uses a unicast RREQ message and all nodes need to exchange nodes' neighbor information.

In [9], OverHearing On-Demand (OHO) method is proposed. OHO recovers the broken link by finding the alternative node like ELRAODV. When a link is broken, OHO broadcasts Helper REQuest message (HREQ) message containing information of the broken node address (Node F in Fig. 1), initiated recovery node address (Node E in Fig. 1), destination address (Node D in Fig1.) including destination sequence. When a node receives the HREQ message and its neighbors are upstream and downstream nodes of broken node, it will replace the broken node by sending Help REPly (HREP) message. The method in [10] is also similar to ELRAODV. It requires all nodes to periodically exchange their one-hop neighbor information and to maintain neighbor table. If a link is broken, a unicast message is forwarded based on the neighbor table and find new route.

To make quick detection on the link break, the study in [6] proposes a self-link breakage detection method. In the method, while most detection method is performed by the neighbor nodes around a node causing the link break, the paper proposes a way for a node causing the link break to declare the upcoming link break. To perform the proposed method, the system proposed in [6] utilizes sensors to detect any sudden environments changes, so that a node expects the communication disabilities of itself. When a node expects any upcoming communication disability, it broadcasts a built-in message to all one-hop neighbors so that the neighbors on the route start immediately route repair process. However, even though this method provides quick detection method on the link break, it also has the aforementioned issues on the repair process itself.

3 Proposed Protocol

We revisit Fig. 1 for the better explanation. The proposed protocol operates when a link is broken, for example, Node F moves out over the radio range of Node E or Node F lost communication capability due to power extinction or hardware damage or so on. In the scenario, Node E and Node I are named as Upstream Node and Downstream Node, respectively. In the conventional AODV protocol, each node maintains routing table which has only one-hop away node information. However, in this proposed protocol, all participating nodes maintain two-hop away node information as well as one-hop away node information by overhearing transmissions of the next hop node. For example, Node G and H in Fig. 1 have the IP and MAC addresses of Node E, F, and I. Node E also has the address information of Node F and I.


Bits:

0 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 20 1 2 3 4 5 6 7 8 9 30 1

Type	J	R	G	D	U	Reserved	Hope Count
RREQ ID							
Upstream Node IP Address							
Destination Sequence Number							
Downstream Node IP Address							
Originator Sequence Number							

Fig. 2. Format of ANR message

When a link is broken, the process with the proposed protocol is as follows:

- When a node detects a link break to the next hop node (hereinafter the node detecting the link-break is named as a detector) and a node causing the break (hereinafter the node is called a lost-node) or a node expects it causes a link-break described in [6], they broadcast Alternative Node Request (ANR) message. The format of ANR message is same as the format of RREQ shown in Fig. 2. However, for representing ANR message, *Type* field is set to 5. Instead of *Destination IP Address*, and *Originator IP Address* fields used in the RREQ format, ANR message has *Upstream Node IP Address* and *Downstream Node IP Address* fields. In the example networks shown in Fig. 1, *Upstream Node IP Address* and *Downstream Node IP Address* are Node E's and Node I's IP addresses, respectively. ANR message is not flooded over the networks. Therefore, when a node receives an ANR message, it does not forward the message to the next hop. Therefore, *Destination Sequence Number* and *Originator Sequence Number* fields in the message format are set to 0.
- A node receiving an ANR message check their neighbor table as shown in Fig. 3. The table includes the MAC and IP addresses of neighbor nodes and link states. The node checks if it has neighbors whose addresses are the addresses of the detector and the next upstream nodes. If it has both addresses in the table, then it prepares to send the RREP to the detector.
- In the table, "Life Time" indicates how long the neighbor information is maintained. The life time is periodically checked so that if the value in Life Time is longer than certain threshold, the neighbor information is deleted. Whenever a node receives a packet from its neighbor, Life Time is reset to 0 and elapsed as time goes by.
- Even though ARP protocol is operating over wireless routing protocol, it may not operate in this method because the neighbor table has MAC address mapped with the target IP address. Therefore, RREP is sent right away without sending ARP request message unlike conventional systems.

- It is possible there are multiple nodes receiving the ANR message and having the information of the detector and the next upstream node. In this case, those try to send their RREP and it may cause the collisions. In addition, it is not guaranteed that the node having the best link reliabilities with the detector and the next upstream node sends its RREP earlier than others having the less reliability.

Neighbor IP address	Neighbor MAC address	Link State	Life Time
	• • •		

Fig. 3. Neighbor Table

To give the priority to the node that has the better link reliability, the candidates, that may send RREP message, uniformly choose their back off time slots between 0 and 2^n and sends RREP message after waiting the chosen time slots. The n is obtained as a function of Signal-to-Noise Ratios (SNRs) of the links of Upstream Node/Candidate and Downstream Node/Candidate as follows:

$$n = N - \left\lfloor \frac{\gamma - SNR_{\min}}{STEP_{SNR}} \right\rfloor, \quad if \quad \gamma > SNR_{\min}, \tag{1}$$

where N is the maximum number of n, $STEP_{SNR}$ is $(SNR_{max}\text{-}SNR_{min})/N$, and SNR_{min} and SNR_{max} are the minimum and maximum $SNRs$ required in the system, respectively. γ is defined as $\gamma = \alpha \cdot SNR_{Up} + (1-\alpha)SNR_{Down}$ where α is system design parameter, SNR_{Up} and SNR_{Down} are SNRs of Upstream Node/Candidate and Downstream Node/Candidate, respectively.

- When the detector and the next upstream node receive the RREP, then they update their routing table. That is, the lost node's information is replaced by the information of the new node which sends RREP message successfully.
- If No RREP is not received within Feedback-Time period, the detector begins the conventional local or source recovery process.

If a conventional node receives the ANR message, it will just ignore the message because it does not understand a message setting Type field to 5 and the upstream node will start the conventional route recovery process.

4 Thoughts on the Proposed Method

In this section, the proposed method is logically evaluated with prior arts. The proposed method is similar to the methods in [8]-[10]. However, there are some unique differences. The differences are described below.

- Unlike [8][10], it does not require any additional information exchanges between nodes before the link is broken, so that no overhead increases.
- Unlike [9], it is feasible even though the alternative node does not know who cause the link break.
- It provides how the alternative node sends a reply message on the RREQ message from a detector, which is not mentioned by [8]-[10]. It sends the reply message after a back off period calculated based on the link qualities as shown in (1).
- When there are multiple possible alternative nodes, it gives a priority to an alternative with the better link quality. No methods in [8]-[10] provides solutions on this issue. Without this mechanism, recovered rout may be broken in a short time later.
- It is compatible with a system described in [6] unlike methods in [8]-[10].
- It solves a ARP issue over WRPs mentioned in Section2, which is not considered by [8]-[10]. Since the proposed method maintains the MAC address mapped with IP address, it removes the necessity in the use of the ARP.
- It is simplest method comparing the other methods.

5 Conclusion

The one of the representative ad-hoc routing protocols is AODV protocol. The protocol has been extensively researched, but there are still issues to be resolved. This paper also studies about one of well-known AODV issues which is the route recovery process. An enhanced method to resolve the issue is proposed in this paper. When a node detects a link-break or expects an upcoming link–break caused by it itself, it broadcasts a message to one-hop neighbors that can substitute the node causing the link-break. If there is such node, then the route will quickly be recovered with the substitute. For the method, nodes needs to overhear any packets transmitted by one-hop neighbors and records their MAC address, IP address, and link status. The proposed method does not increase any overhead comparing the other methods and is backward compatible with the conventional systems. Unfortunately, the performance evaluations through simulations or mathematical analysis are not provided in this paper at this time. Therefore, as a future works, the proposed method will be evaluated through the simulations and proved its efficiencies.

Acknowledgements. This research was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (2012-0003609) and in part by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency" (NIPA-2012-(H0301-12-2003)).

References

1. Park, J.-S., Gerla, M., Lun, D.S., Yi, Y., Medard, M.: Codecast: A network-coding-based ad hoc multicast protocol. IEEE Wireless Commun. Mag. 13(5), 76–81 (2006)

2. Camp, T., Boleng, J., Davis, V.: A survey of Mobility Models for Ad-Hoc Network Research. Wireless Communications & Mobile Computing: Special Issue on Mobile Ad-Hoc Networking: Research, Trends and Applications 2, 483–502 (2002)
3. Perkins, C., Belding Royer, E., Das, S.: Ad Hoc on demand distance Vector (AODV) Routing, IETF RFC 3561 (2003), `http://www.ietf.org/rfc/rfc3561.txt`
4. Ashraf, U., Abdellatif, S., Juanole, G.: Route Maintenance in IEEE 802.11 Wireless Mesh Networks. Comput. Commun. 34, 1604–1621 (2011)
5. Macintosh, A., Ghavami, M., Siyau, M.F., Ling, S.L.: Local Area Network Dynamic (LANDY) Routing Protocol: A Position based Routing Protocol for MANET. In: 18th European Wireless Conference (2012)
6. Park, J.D., Ryu, H.Y., Lee, S.S., Kim, B.-S.: Communication Node and Method of Processing Communication Fault Thereof. U.S. Patent Application Number 12887785 (2010)
7. Liu, G., Wong, K.J., Lee, B.S., Seet, B.C., Foh, C.H., Zhu, L.J.: PATCH: a novel local recovery mechanism for mobile ad hoc networks. In: 2003 IEEE Vehicular Technology Conference, vol. 5, pp. 2995–2999 (2003)
8. Singh, J., Singh, P., Rani, S.: Enhanced local repair AODV (ELRAODV). In: International Conference on Advances in Computing, Control, and Telecommunication Technologies, pp. 787–791 (2009)
9. Sirilar, J., Rojviboonchai, K.: OHO: overhearing on-demand route repair mechanism for mobile ad hoc networks. In: 2010 International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology, pp. 66–70 (2010)
10. Chuang, P.-J., Yen, P.-H., Chu, T.-Y.: Efficient Route Discovery and Repair in Mobile Ad-hoc Networks. In: 26th IEEE International Conference on Advanced Information Networking and Applications, pp. 391–398 (2012)
11. Shin, H.-H., Kim, B.-S.: Performance Evaluations on Local-Repair of AODV Protocol over IP-Based Ad-Hoc Networks. In: Lee, G., Howard, D., Kang, J.J., Ślęzak, D. (eds.) ICHIT 2012. LNCS, vol. 7425, pp. 57–63. Springer, Heidelberg (2012)

Smart Watch and Monitoring System for Dementia Patients

Dong-Min Shin[*], DongIl Shin, and Dongkyoo Shin

Department of Computer Engineering, Sejong University, Seoul, Korea
gentletiger@gce.sejong.ac.kr, {dshin,shindk}sejong.ac.kr

Abstract. Monitoring information on the behavior of dementia patients could improve their health and safety, and thus quality of life. To monitor daily activities, dementia patients require portable and wearable monitoring device. Various sensor technologies are currently used to monitor emergency situations such as falling down and wandering activities as a result of memory and cognitive impairment. Therefore, in this research paper, a watch-type device (Smart Watch), server system, and step detection algorithm utilizing a 3-axis acceleration sensor are developed. The suggested step detection algorithm showed an accuracy of 96% in verifying normal steps.

Keywords: 3-axis accelerometer, u-health, step number detection algorithm.

1 Introduction

In modern society, increases in the older population due to the development of medical technology and birth-rate decreases creates challenges for caretakers. Dementia refers to cognitive impairment that makes functioning in daily life more difficult. Early symptoms of dementia experienced by patients include memory loss, which gradually affects everyday activities. Typically, from a few months to several years, the first symptoms are mild but develop slowly, gradually leading to serious memory loss. In addition, patients recognize the family and complicated task, it becomes difficult to do. Wandering symptoms of patients decrease ability causes. More than 73% of dementia sufferers experience going missing [1].

In this study, we develop a Smart Watch System for dementia patients to improve their health and safety. The Smart Watch System includes a wristwatch-type device and server system. The device includes a built-in GPS, ambient light sensor, acceleration sensor, and to communicate with the server system. The server system functions include the creation of a personal profile for patients and monitoring the location, motion through the amount of light and step count

The system developed in this paper avoids the risk of a dementia sufferer going missing or experiencing wandering symptoms, using GPS technology. In addition, the number of steps and amount of light are used to record motion activity information to identify the exact amount of exercise, which can be used as medical data.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 577–584, 2013.

This paper is composed of the structure of the Smart Watch System for dementia patients and step detection algorithm development using acceleration sensors for measurement of the exact amount of exercise undertaken by the patient.

2 Related Work

2.1 Smart Care Service

Spain's company Keruve medical services for dementia patients. This study used bracelets with built-in GPS and a PSP-type device. The GPS watch features precise location detection using triangulation, even if the patients are in the room [2].

KT in Korea has developed a location-tracking system using GPS and CDMA. Gangnam-gu developed the Gangnam u-safe system. The service began in May 2009 using USN (Ubiquitous Sensor Network) technology and GPS. It is an ultra-compact device featuring an emergency alarm service that is used for safety purposes with socially vulnerable individuals, such as those with autism, intellectual disabilities, and children [3-4].

Existing medical devices for patients with dementia are focused on location tracking using GSP. The system developed in this paper suggests the addition of a variety of new services. It includes GPS, as well as the ability to measure motion activity and the amount of light. These features enable the creation of a profile that contains the location of the patient, and records the amount of exercise. These additional services should contribute to advances in the existing research.

2.2 Steps Detection Algorithm Using 3-Axis Accelerometer

Studies on steps detection and behavioral patterns have been carried out in various ways. Jun Yang has proposed an algorithm for motion recognition using a motion-tree developed using the acceleration features of a mobile phone [5]. The motion detection algorithm is one of the basic methods for detecting the number of steps [6-8]. The experiment distinguishes users' movements by a pattern recognition algorithm and has developed a way of extracting various motions from basic motion patterns and feature vectors of humans. This function reads normal and abnormal movements. For example, sitting, standing, and falling down, as well as the number of steps [9-11].

In this paper, we contribute to the current research by collecting acceleration information extracted from the Smart Watch, a portable device. The watch is used to monitor health information for dementia patients.

3 Development of Smart Watch System

3.1 Structure of Smart Watch System

The system developed in this paper consists of a Smart Watch (portable device) and server system. The Smart Watch includes a GPS, 3-axis accelerometer and ambient light sensor. The Smart Watch, which is worn on the patient's wrists, periodically transfers the patient's activity information derived from his/her location and amount of light to detect sun exposure through communication with the server. The protector and doctor can monitor the patient's health condition through the webpage.

The server identifies the location through the patient's data transferred from the Smart Watch and measures the patient's activity information through the step number detection algorithm and creates a profile about the patient's health information, together with the amount of light to detect sun exposure.

3.2 Development of Portable Devices

Smart Watch's location tracing functions using a GPS sensor can monitor the present location and migration route of the patient. The ambient light sensor measures the size of sunlight exposed to the watch, and records it. Thus, the sensor can measure the time the patient is exposed to the sunlight and amount of sunlight. The 3-axis acceleration sensor records the value of the x, y and z-axis in real time. The server can get the number of patient's steps through the step number detection algorithm.

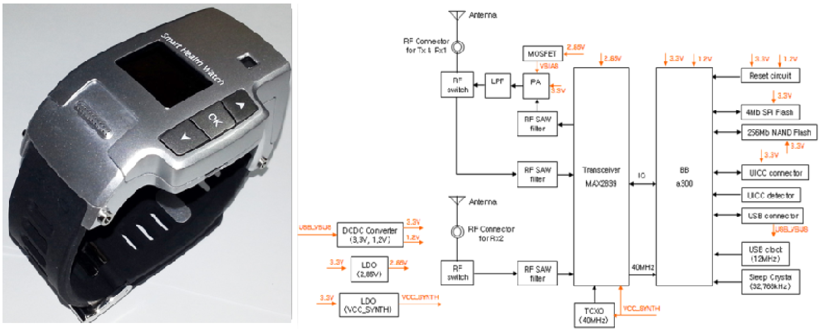


Fig. 1. Smart Watch and Block diagram

The values of the sensors are obtained through the real time transfer of the data through TCP/IP communication using the CDMA network. After connection to the server through SMS such as Server Open SMS and Transmission Close SMS for transfers, the values of the sensors exchange data with each other. At this moment, the transmission time transfer of the data by contacting the server according to the regular cycle defined by the user. The server can inform the protector or patient by alarm in the case of special events such as injection time, safety scope excess of patient, and so on. The Smart Watch system developed in this paper is designed to be worn easily using the form factor of a wrist watch. And because the Smart Watch is held in position by a clamp, it can prevent a patient from taking off or losing it. Thus, if a demented patient experiences an emergency or loitering symptoms, any problems can be quickly dealt with.

3.3 Development of Smart Watch server

The server system is divided into a receiver module for transmitting data and a health management module for analyzing data, as well as a webpage to perform management functions and patient monitoring.

First, the receiver module manages the watch's connection through the SMS receiver while waiting for the Smart Watch's SMS. The receiver module with the Connection SMS receives the saved period data in the watch as per the defined protocols after assigning a socket and a thread using TCP/IP communication.

The health management module plays a role in generating the profile by analyzing the transferred data.

The GPS signals check whether the user moves out of the scope of designated safety or not. And the ambient light sensor converts data into a percentage from 0 to 100, by accounting for exposure time and exposure state at the point at which the amount of sunlight is not zero. Finally, a 3-axis acceleration sensor measures the step number corresponding with the period monitored through the step number detection algorithm.

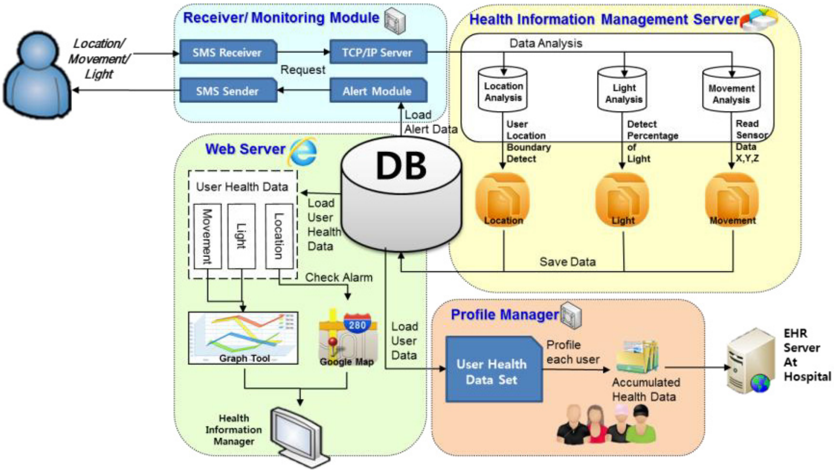


Fig. 2. System operational scenario

The patient's data obtained by the health management module is separately saved in the database. The data in the database is recorded in the profile of each patient, and can be monitored through the webpage.

The webpage plays a monitoring role as a tracing location and the health information of the patient obtained from the DB. First of all, a protector can set up a communication period, and the scope of the safety schedule of the Smart Watch through the settings. The Smart Watch is connected to the communication module by the server in a designated period on the basis of the settings information. The server indicates whether the traced patient's location is within the scope of safety or not. The scope of safety guarantees the patient's safety because the present location and the scope of the safety of the patient would be marked in a circle on the map. The amount of sunlight indicates the exposure state hourly as the time-axis and exposure-axis through the graph.

The activity mass also expresses the number of walk hourly through graph.

The health information can preserve the patient's health and safety because it monitors the patient's state through activity listed by time order, amount of sunlight, and location of the patient measured during outdoor activities, as follows.

4 Development of Step Detection Algorithm

In addition to the location tracking service for dementia patients, the Smart Watch system developed in this paper provides accurate step detection for use in health care.

The step detection algorithm uses a 3-axis accelerometer to accurately detect a user's steps and further analyzes the patient's activities.

4.1 Experiment Design

The experiment suggested in this paper uses the Smart Watch to compare the actual steps counted in 30 ~ 60 seconds to the value detected by the accelerometer under the same conditions.

Eight experimenters created 170 data of 3 types of steps – fast steps, normal steps, and slow steps, every day. Each data is categorized in the database by experiment date, time, and the number of steps. Stored results are preprocessed into energy values for peak picking and analysis of distinctive features of the walk.

Analyzed features are used to distinguish the step and non-step activities and the measured number of steps is then compared to the actual number of steps counted.

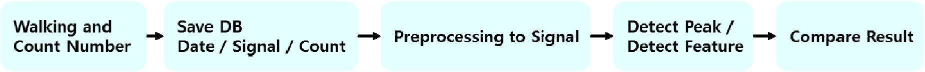


Fig. 3. Preprocessing of Accelerometer data

4.2 Preprocessing Data

Acquired x, y, z axis data are each in 8 byte double data types, recorded 80 times per second. Therefore, using the SVM (Signal Vector Magnitude) values makes the calculation more efficient than using 3 values simultaneously for each calculation. SVM in this experiment is expressed as the following equation.

$$\mathrm{SVM} = \sqrt{x_i^2 + y_i^2 + z_i^2} \tag{1}$$

The accelerometer records 80 times per second and it even catches subtle movements. Therefore, even if the patient is standing still, the accelerometer will be recording constantly changing values. These subtle noise signals might result in errors when measuring the number of steps. In this paper, we have used the Moving Average Filter to filter out these noises, preventing errors. The Moving Average Filter has low pass filter properties and it can be expressed as the following equation.

$$\mathrm{T[n]} = \frac{1}{5}(svm[n-2] + svm[n-1] + svm[n] + svm[n+1] + svm[n+2])$$

$$= \frac{1}{5}\sum_{m=-2}^{2} svm[n-m] \tag{2}$$

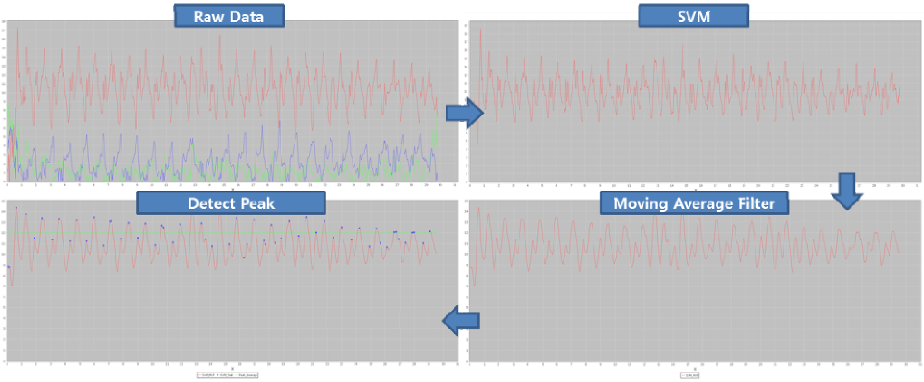


Fig. 4. Result of preprocessing

4.3 Step Detection Algorithm

The step detection algorithm being suggested in this paper finds the peaks from the preprocessed data, then counts the number of peak values that are over the threshold value, which is calculated from the data.

First, to pick out the peaks, we find the wave's mean gradient by computing the average of the gradient of two bundles of data intervals. If this value is greater than the threshold value, it is considered the start of the peak, and when the mean gradient becomes a negative value, this point is put into the peak point candidate. It can be expressed as the following equation.

$$G_n = \frac{svm_{n+1} - svm_n}{T_{n+1} - T_n} \tag{3}$$

$$\text{Average of } G_n = \frac{G_n + G_{n+1}}{2} \tag{4}$$

The peak candidate includes waveform errors or noise errors. In this paper, we used the following method to clear out the errors and find the genuine peaks. First, we find the peak candidates with a time interval of less than 0.3 seconds. Collected data are acceleration data for detecting the number of steps, so the movements must show regular intervals of high peak and low peak. Therefore, peak candidates in the low period are noise values from the wrong movement. Then, we store the candidate with high SVM values as the actual peak and drop the values considered as errors.

Detected peak values are affected by the patient's footsteps and the height of the swinging of arms, so the values include individual differences. However, every waveform of walking has a large amplitude followed by a small amplitude. Therefore, in this paper, we use the following feature to derive a threshold value with the mean amplitude over 1 second, and collect the peaks over the threshold value.

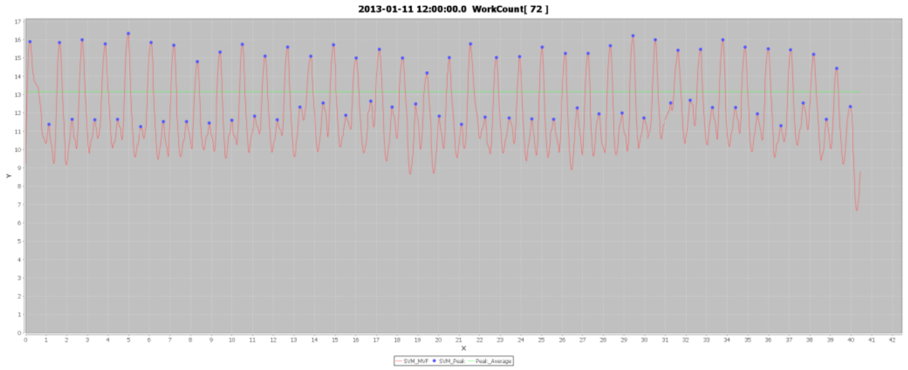


Fig. 5. Result of detection step peak

4.4 Result of Experiment

The suggested algorithm is tested with the Smart Watch with an embedded accelerometer using a 80Hz sample rate, attached to experimenters' wrists, and tested on fast steps, normal steps, and slow steps.

Table 1. Experiment Result

Lab no.		58	71	72	83	99	110	112	150	Total	Accu-racy
Fast Step	**U. C**	117	111	111	109	116	118	117	105	904	93.03%
	R	138	120	119	117	121	123	120	109	967	
Slow Step	**U. C**	33	35	40	32	39	31	32	35	277	96.02%
	R	33	36	41	33	44	31	34	36	288	
Nor-mal Step	**U. C**	71	77	71	72	68	66	66	68	559	96.77%
	R	77	75	66	72	75	61	73	78	577	
Total Mean(%)										1832/ 1740	94.71%

To measure the accuracy of the suggested algorithm, we compared the actual sum of steps and the detected sum of steps derived with the algorithm. The results of this method showed 94.7% accuracy in total, 93% in fast steps, 96.7% in normal steps, and 96% in slow steps.

As the pace gets faster, the gradient of SVM tends to grow larger, and the phase interval narrows, resulting in higher error rates. However, in cases of normal and slow steps in which the amplitude is gradual, results have a higher rate of finding the peaks correctly, showing a closer value to the actual number of steps.

Table 1 shows the analyzed data from the 8 experimenters.

5 Conclusion

In this paper, we developed a Smart Watch for dementia patients, and a server system to monitor patients. The server system can not only monitor patients' locations, but also can help manage patients' health by determining patients' activity according to the data derived with the step detection algorithm, along with the ambient light sensor and accelerometer. According to the results of our experiments, normal steps have a 96% accuracy in detection, and on average, showed 94% accuracy.

Medical services for dementia focused mainly on tracking the patients' location to prevent a patient from going missing. The system developed in this paper provides and monitors further health information of the patients. If the results of this paper are applied to medical facilities such as hospitals and nursing homes, a better medical service based on the amount of a patient's lighting and exercise can be provided.

Further research based on this work could include a more comprehensive analysis of a patient's activities such as running or sitting.

Acknowledgment. This research is supported by Seoul R&BD Program (SS110008).

References

1. Tabert, M. H., Liu, X., Doty, R.,L., Serby, M., Zamora, D., Pelton, G.H., Marder, K., Albers, M.W., Stern, Y., Devanand, D.P.: A 10-item smell identification scale related to risk for Alzheimer's disease. Annals of Neurology 58(1), 155–160 (2005)
2. Company Keruve (2008), `http://www.keruve.com`
3. u-safe Gang-nam (2009), `http://www.gangnam.go.kr`
4. kt i-search (2009), `http://www.kt.com`
5. Yang, J.: Toward Physical Activity Diary: Motion Recognition Using Simple Acceleration Features with Mobile Phones. In: IMCE-2009, pp. 1–10 (2009)
6. Bao, L., Intille, S.S.: Activity Recognition from User-Annotated Acceleration Data. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 1–17. Springer, Heidelberg (2004)
7. Baek, J., Lee, G., Park, W., Yun, B.-J.: Accelerometer Signal Processing for User Activity Detection. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) KES 2004. LNCS (LNAI), vol. 3215, pp. 610–617. Springer, Heidelberg (2004)
8. Ravi, N., Dandekar, N., Mysore, P., Littman, M.L.: Activity Recognition from Accelerometer Data. In: Proceeding of the National Conference on Artificial Intelligence, vol. 20(3), pp. 1541–1546 (2005)
9. Yoo, H.-M., Suh, J.-W., Cha, E.-J., Bae, H.-D.: Walking Number Detection Algorithm using a 3-axial Accelerometer Sensor adn Activity monitoring. Korea Contents 8(8), 253–260 (2008)
10. Shin, S.H., Park, C.G.: Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors. In: IEEE Sensors Applications Symposium, San Diego, California, USA, pp. 1–5 (2007)
11. Noh, Y.-H., Ye, S.-Y., Jeong, D.-U.: System Implementation and Algorithm Development for Classification of the Activity States Using 3 Axial Accelerometer. J. KIEEME 24(1), 81–88 (2011)

A Lesson from the Development of Surveillance and Reconnaissance Sensor Networks Systems

Daesik Kim, Seongkee Lee, and Mirim Ahn

Cyber Technology Center, 2nd R&D Institute, Agency for Defense Development,
Republic of Korea
{woodburn,seongkeel,mirimahn}@hanmail.net

Abstract. Recently there has been much research on fulfilling tasks such as surveillance and reconnaissance coupled with a sensor networks system. It is possible to realize the system because of many technical progresses in the area of sensor networks such as signal processing, wireless networks, sensor deployment, etc. To construct sensor network systems effectively and efficiently, lots of considerations such as user requirements, sensor capabilities, and signal processing technologies should be reviewed. Consistent and unambiguous architecture for the development process coupled with the technologies should be built. This paper presents a verifiable logical architecture with petri net.

Keywords: Sensor Networks, Systems Engineering, Systems Architecture, Wireless Sensor.

1 Introduction

There are lots of studies in the area of sensor networks, but few are for practical use. So while we were developing the sensor networks system, we experienced lots of trials and errors. When reviewing the problems, they can be roughly aggregated. The first problem is to define the operational concept of sensor networks for surveillance and reconnaissance. The other problem is to secure the technologies needed for implementation of them.

Lots of studies are focused on network protocol. We tried to design a homogeneous hardware in every application in sensor networks. Unfortunately, current sensor technology couldn't meet the requirements of applications. To meet the requirements, we made additional efforts to improve the signal processing capability of sensors.

Also, we made our efforts to define the common and agreeable terminologies such as detection rate, false alarm rate, classification rate used in the applications such as the classification, and the tracking of the targets in the field of the sensor networks.

Even though we set our project based on the series of procedures in System Engineering concepts to prepare against the risks forecasted, while we were processing the project, we experienced so many trials and errors when solving the problem. To meet the purposes and requirements of surveillance and reconnaissance through many experiments, we concluded that the just one hardware type is not enough for lots of

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 585–593, 2013.

applications. Therefore, we categorized the applications and built the purpose oriented sensor networks for each application.

We tried to maintain the project with an objective view. To find the procedure that would reduce errors, we considered the concept of systems architecture. It is commonly known as an area of system engineering, but it has a very strong advantage in its ability to share the partakers' understandings with the products of systems architecture development. If there is a lack of sharing in the system architecture, the whole procedure of system engineering development is not well developed.

It shows the views of the partakers' concept for the system. At first these kinds of architectural views don't have to be the same, but it makes all people understand and reduces the differences by evolving the processes. By repeating the process, they will make the same architecture for their system with various views. System architect coordinates can manage the whole series of processes until the architecture is done from logical architecture to physical architecture.

In this paper, based on the experiences of the project, we are going to explain the necessary items and procedures for the systems architecture theory. In Chapter 2, Surveillance and Reconnaissance Sensor Networks (SRSNs) is introduced. Chapter 3 describes the system architecture theory and application of petri-net theory. In Chapter 4 we present the logical architecture of SRSNs. Finally we conclude this paper in Chapter 5.

2 SRSNs

SRSNs consist of sensor nodes, relay nodes, and a C2 (command and control) terminal, and is operated as shown in Figure 1.

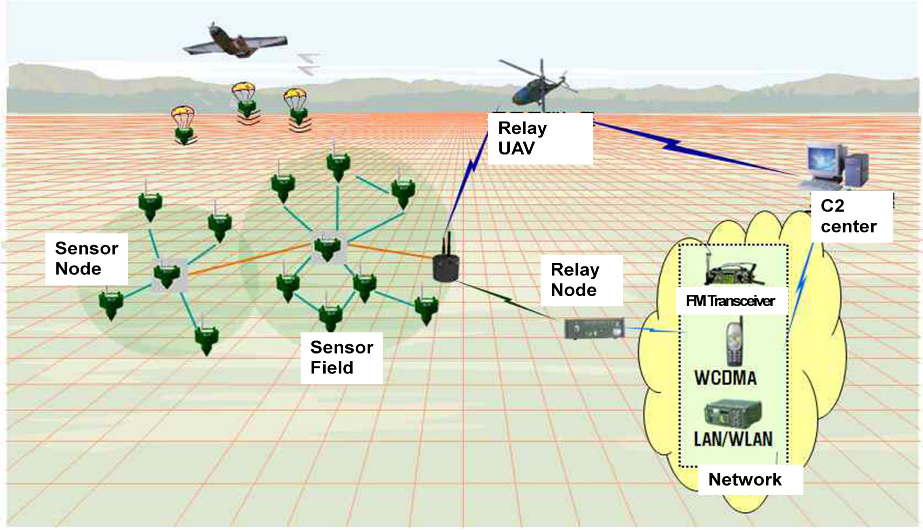


Fig. 1. Operational Concepts Diagram for SRSNs

The sensor nodes have sensors capable of detecting magnetism, seismicity, acoustics, light intensity, and heat. These networks have a self-organization capability that is used to build a network in unfriendly areas, including enemy territory. They also have a self-healing capability to deal with sensor fault or damage.

The relay node communicates with the sensor nodes and other relay nodes. The relay node has extended transmission capability as it has more energy resources than the sensor node.

The C2 terminal is deployed at the monitoring site in order to control the sensor network in general and has a user-friendly interface for a human operator. Detected data is synthesized, analyzed, and displayed on the C2 terminal.

In accordance to their own purposes each node deployment and operation methodologies are different. In case of reconnaissance, to track the intruder moving one way through the trail, the vertical deployment is mainly concerned. In case of surveillance, to enclose the facility with double or triple deployment strategy, the horizontal deployment is mainly concerned.

To develop SRSNs there are specific considerations such as sensor signal processing area, RF communications, wireless ad-hoc networks, network management control, and display. The importance of considerations is different for each situation.

For example, if the sensor node is deployed at long distance, a wireless ad-hoc network is required frequently. In this case, the sensor node is adjusted nearer than before. To communicate with the node from far away, the output power must be increased. This leads to an increase in battery consumption. There must be a trade-off between two deployment strategies.

As a matter of fact, in SRSNs, our experience shows one hop communication is optimal and effective while ad-hoc communication is useful only for fault tolerant purposes.

The performance of Sensor networks is also affected by the climate, terrain, vegetation. The details are beyond our concern.

3 System Architecture and Petri Nets

The systems architecture theory is a branch of the systems engineering. Defense and industrial areas have especially brought lots of concerns. They have large systems to be solved. System architecture is constructed ahead of building real systems. Lots of problems are discovered and improved while system architecture is built.

In reality, system engineering, especially in system architecture areas, is not helpful for building a system. It is unnecessary to fit the regulations of government.

System architecture sometimes may not useful for building a small system, but when the size grows at some amount, without system architecture, lots of errors happened naturally. I am going to explain the necessary items and share the experiences (Figure 2)

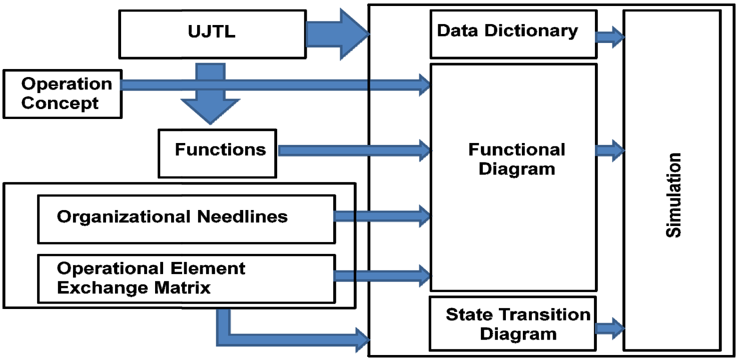


Fig. 2. Relationship between System Architecture Items

You may have experienced that it is very hard to try to build a totally new system. To find what a system needs, benchmarking the conventional systems or experiences is important. Securing the fundamental technologies is also important. As a result of these activities, Universal Joint Task List (UJTL) is defined.

In the view of system, first of all, the main user should be defined among partakers. At the beginning stage, the approach of development should be decided between researcher and serviced users. In the view of researcher, the purpose can be to obtain a new technology. If so, the completeness of system is not his/her concern anymore. In the view of serviced user, the procurement of system itself can be the goal. Without control of these problems at the first stage, the cost to solve the troubles in the middle of project for each step will be drastically increased.

If the viewpoint of system is defined, the purpose of system is required: who is going to use the system, what kinds of effects should be brought. With a good definition of system, the system volume, participants, technologies, and development periods can be defined well. To do this the requirements of user is very important and should be considered and reflected.

Organizational Needline Diagram and Operational Elements Exchange Matrix for organization and subsystem are needed. To operate the system, it should be defined such that the actual user of each components and should be identified the information exchanged. If each item is specified, a logical model should be composed. It doesn't consider physical items of the system. To secure the expected function and to classify the expected performance, a model has to look over them with a global view. Logical models can be very simple. In accordance to the scale of system, it can be very complex. In case of a complex system, they can approach the system in accordance with understanding for their own interest. With the logical model, we can understand the role of the system model.

When developing a system model top-down or bottom-up methodologies are used. Generally, in case of development with well-known technology, a top-down approach is used. In case of development without specifying the technology, a bottom-up approach is used. In real situation many research development is set with top-down approach. However, in the middle of development bottom-up approach is partially adopted.

To apply the new technologies and modifications during the process, evolutionary development methodology is used. Evolutionary development is progressed with the step called "build." Typically with 2~3 level of "build", a project is developed. System models should be satisfied with every function of a logical model. This paper doesn't implement a system model. It just explains the relationship between a logical model and a system model.

A logical model is composed of Functional Diagrams (Activity Diagram or Operational Diagram in DoD Architecture Framework)[1], which describes the relationship between functions, State Transition Diagrams, which shows the dynamics of the system, and a Data Dictionary. These models should be consistent and unambiguous.

A Petri net is a mathematical concept. It represents the matrix MxN to the diagram. With the firing matrix (transposed 1xN) mechanism, the current status can be monitored, so when the system is represented by the matrix, the current status and future status should be calculated while the status can be monitored with the Petri net. Modeling with a petri net can specify the problems of the system. Logical models implemented by a petri net can detect and avoid the model logical inconsistency and system deadlock problems.[2,3]

4 Logical Architecture of SRSNs

SRSNs are a kind of surveillance and reconnaissance system. The concept is shown as Figure 1. The organizations such as the sensor field (sensor node), relay node, and C2 center involved in SRSNs cooperate with each other by exchanging information. In Figure 3, the relationship between organizations is shown. The information exchange in Figure 3 is depicted in Table 1.

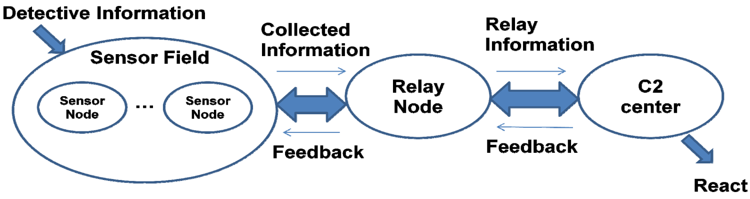


Fig. 3. Needline Diagram of SRSNs

Table 1. Operational Elements Exchange Matrix

From / To	Signal Generation	Sensor Field Detection	Transceiver Relay Node	C2 Center Display	Superior Level.
Signal Generation	N/A	Detective Info	N/A	N/A	N/A
Sensor Field Detection	N/A	N/A	Collected Info	N/A	N/A
Transceiver Relay Node	N/A	Feedback	N/A	Relay Info	N/A
C2 Center Display	N/A	N/A	Feedback	N/A	React
Superior Level	N/A	N/A	N/A	N/A	N/A

As a user in the C2 center, the information from the sensor field and effective information for results are recognized as detective information. The detective information is measured by comparing it with the previously defined learning knowledge. The result can lead to further action. Functional Diagram is composed with IDEF0 format.

The advantage of IDEF0 is their ability to decompose the function into sub functions. Very complex models can be represented by hierarchy. [4]

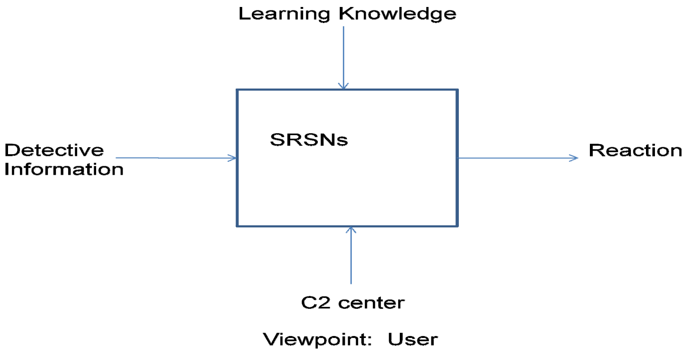


Fig. 4. Diagram of SRSNs

As shown in Figure 5, the role of SRSNs is composed of the roles such as “detect in the sensor field”, “receive transmission of relay node”, and “display situation.”Collected information is acquired by the aggregation of detective information from several sensor nodes. Through relay nodes, collected information is converted as relay information. The C2 center displays this information and determines the reaction.

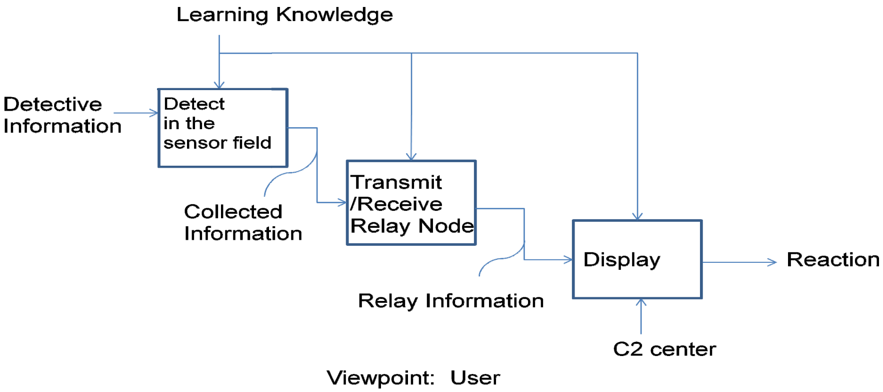


Fig. 5. Diagram of SRSNs

The three roles of SRSNs can be divided in accordance with the complexity of the model. The status of this model is one of the statuses in Figure 6.

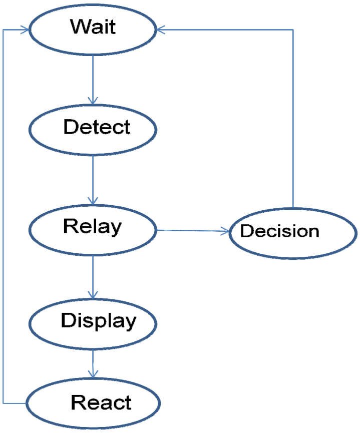


Fig. 6. Transition Diagram of SRSNs

Data Dictionary used in SRSNs has a structure shown in Figure 7. Signal information can be separated as acoustic, seismic, magnetic, PIR, and microwave. Conditional information is the distributed values of weather, climate, and vegetation. Detective information is the sum of signal information and conditional information. Learning knowledge is the aggregation of detective threshold and condition controllable data. As for reactions, there is “report” and “counterattack”.

```
Detective Information = Sig(Acou|Seis|Mag|PIR|Micro)*
+ Cond(weather+climate+vegetation)
Learning Knowledge= Detective Threshold
+ Conditional Controllable Range
Reaction = Report | Counterattack
```

Fig. 7. Dictionary of SRSNs

Logical models can be developed with the operational concepts, various diagrams, and data sets. With this model, as shown in Figure 8, interactive simulation is possible to show the performance of the system. We developed the simulation model with the simulation tool named CPN Tools [2]. When you see the simulation model, you will find that the structure of the model looks just like the functional diagram. We simulate the time required to reach the C2 center when the detective signal is detected with the time interval 4 times per 1 second.

Detective signals take 5 seconds from sensor node to relay node. From relay node to C2 Center, it takes 5.6 seconds. It takes 3 seconds to display information. We know that the total time is 13.6 seconds.

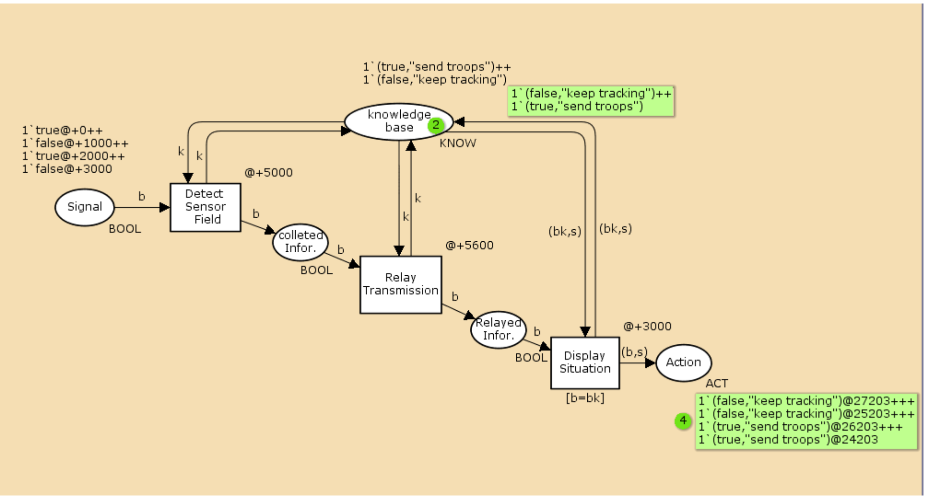
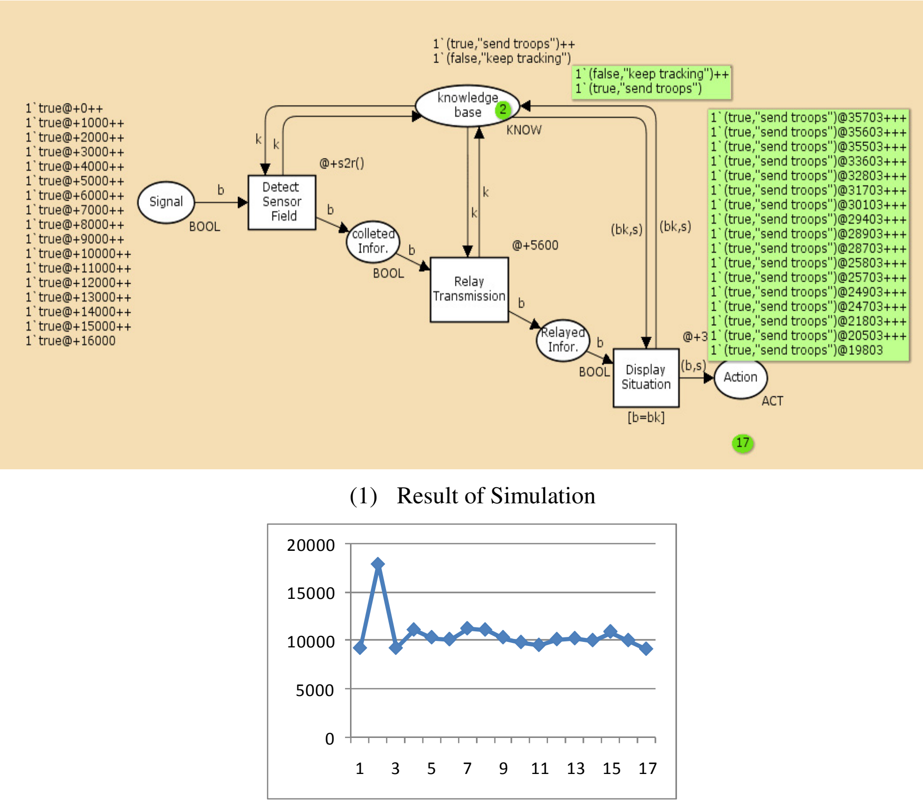


Fig. 8. Simulation of Logical model for SRSNs



(1) Result of Simulation

(2) Graph of Simulation

Fig. 9. Simulation with sensor node transmission time per hop as 300ms (17 times)

15 seconds from the sensor field to the C2 center may be fast enough, but it may be too slow for the purpose of some sensor networks. If the adequate time is not met in the simulation, it means thatthe problem is based on the technology itself, not on the system. It is easy to find the part that can be improved and to put effort in improving that part through the logical model.

For example, if we improve the transmission time per hop as 300 ms, it takes 9.2~17.9 seconds. The average is 10.6 seconds. Through the simulation shown in Figure 9, we can see some progress.

Otherwise, by the improvement of transmission speed between relay nodes or by the improvement of displaying time, the total access time will be improved. Improvement for more effectiveness part is better. Also, simulation through logical models would be desirable.

5 Conclusion

We studied the project SRSNs, and we secured the technologies needed and did our best to develop these technologies. Now we are in the stage of maturing SRSNs technology to manage stable operation.

To improve this project, more stability, reduction of trials and errors, and the development of more complicated architecture models to consider lots of situation is needed. To do that, research of the system level should be required.

Acknowledgement. This work was supported by the Dual Use Technology Program (UM09021RD1).

References

1. DoD Architecture Framework Version 2.02 (August 2010)
2. CPN Tools Version 3.2.2 (October 2011), http://www.cpntools.org/
3. Girault, C., Valk, R.: Petri Nets for Systems Engineering: A guide to Modeling, Verification, and Application. Springer (2003)
4. IDEF0, http://www.idef.com
5. Karl, H., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. Wiley (2005)
6. Wagenhals, L.W., Shin, I., Kim, D., Levis, A.H.: C4ISR Architectures II: A Structured Analysis Approach for Architecture Design. Systems Engineering 3(4) (Fall 2000)

A Classifier Algorithm Exploiting User's Environmental Context and Bio-signal for U-Home Services

HyunJu Lee[1], DongIl Shin[1,*], Dongkyoo Shin[1], and SooHan Kim[2]

[1] Department of Computer Engineering, Sejong University, Seoul, South Korea
nedkelly@gce.sejong.ac.kr, {dshin,shindk}@sejong.ac.kr
[2] Visual Display Div. R&D Team,
SAMSUNG Electronics Co.HQ, Suwon, South Korea
ksoohan@samsung.com

Abstract. U-Home is a home-service through an interaction between human and object. Smart-home-middle-wear provides its users with services needed through interactions between users and home equipment. In this study, users' conditions in four rooms with Smart-home-middle-wear using had been sent through EG sensor device and they were then classified by emotion-perceiving-agent-system adapting an algorithm. The emotions, which were experimented, had been divided into eight categories; Normal, Happy, Surprise, Fear, Neural, Joy, Stress(Yes) and Stress(No). In this study's experiments, modified Decision Tree algorithm was adapted and it extracted over 90% of results totally.

1 Introduction

U-Home is a home service to achieve human-based U-life perceiving "the meaning of spatial, social aspects and digital communities", which can realize the interaction among human, computer and home equipment based on 'Ubiquitous' [1]. U-Home provides the best fitted home-service-environment through home-middle-wear which can make people, devices and spaces interact with each other without any limitations on time and space. Smart-home suggests the progressive experiments in terms of the effort that delete obstacles between human and housing through Human-Computer interaction and Ubiquitous-Computing [2].

Thus, in this study's experiments, the purpose was on the presentation of Smart-home-middle-wear in order to provide intelligent home services. The presented Smart-home-middle-wear has a function - predicting expectable home services which its users are supposed to have. And, in this study, patterns of brain signals were analyzed to find out that how selected home-services affected on users' stress and emotion through a kind of human bio-signals, ECG (Electrocardiogram). Although, in previous analysis, SVM algorithm had used, the modified algorithm of DECISION TREE was adapted to pursue improvement of increased accuracy rate of the pattern analysis in this study.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 594–601, 2013.

2 Smart-Home-Services and Context Aware

In this competitive society, people usually back to their house with exhausted body condition by stress and exhaustion. Thus, for modern citizens, home has to play a healing role as an invisible assistant to make its owners take a rest as much as possible while the people stay in [3, 4].

Smart-home should detect users' health condition, emotion, environmental circumstance, preference and so on automatically and provide a function which can balance digital environment around the user [5, 6]. Also, Smart-home can comprehend its' users' statuses using the function of 'Context aware'.

This process can be defined providing users with proper information or services related to users' works as "Context", Ranganathan and Campbell presented middle-wears in terms of the agent of 'Context aware' in Ubiquitous environment. These middle-wears were investigated to detect situations switched by agents which gathering of information of the situations was conducted by the agent as well.

Since 'Context Aware' had had difficulty on classification and identification different from ordinary experimented data, Context toolkit was made to exceed this difficulty [7]. The toolkit provided a library package which assisted making programs easily through defining circumstances. In this study's experiments, the tests on users' context aware were performed in four different rooms totally; living room, bed room, study room and DVD room. And, by suggesting 'Context model', users' pattern of behavior in home environment was predicted. Fig 1 indicates Context model consisted of Two-layers.

Layer 1 defines 7 context data inputs that are acquired from the user, the home environment, and the home appliance: pulse(P), body temperature(BTP), facial expres sion value(FV), room temperature(RTR), time(T), user location(UL), and user motion(UM). Layer 2 consists of the person's widget(User State), the environment widget(Environment State), and the device widget(Device State); each widget manages context in an HHIML (human-home interaction mark-up language) based XML tree structure.

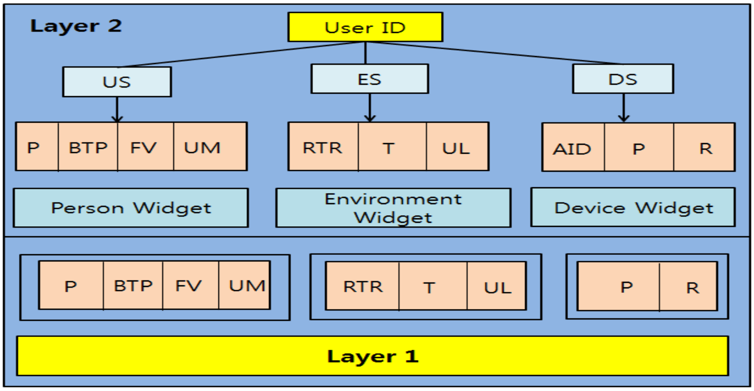


Fig. 1. The structure of two-layer

3 Smart-Home-Middle-Wear and Emotion-Perceiving Agent

3.1 Users and Smart-Home-Middle-Wear

In this study's experiments, configured Smart-home was composed of totally four rooms including living room, bed room, study room and DVD room. In each room, there were cameras to observe users' location.

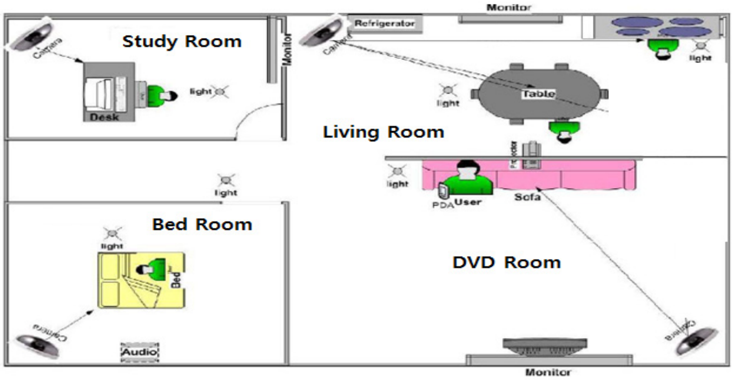


Fig. 2. Structure of Smart Home

Smart-home actively collected a variety of situation data for analysis of emotion and stress according to users' behavioral patterns. Based on the location-observing system (basically invented to draw on UWB), it also provided services to users. A series of situations when Smart-home provides appropriate information and services related to item users' works or activities. In DVD room, users could watch TV, movies and listen to music which they want to enjoy. In bed room, users could use certain music and lamps to sleep. Smart home provided such services automatically according to the data on the analysis of users' behavioral patterns and stress without passive orders from users. And it could receive users' signal pulses, body temperatures and ECG signals based on bio-signals according to users' patterns through ECG sensor device and Bluetooth.

In order to perceive Facial expression, WinCE PDA using WINCE OS was set to equip Facial Expression Recognition module and WiFi was used in their communication. Users' motions were detected by cameras at each side corners in the rooms and sensor devices which could measure temperature were installed to measure inside temperature in each room. TCP/IP through Landline LAN was used in communication among UWV location observing system, network server and cameras and Serial port was used in the communication with sensor device of temperature measurement. Also, every device equally connected to each other with integrated module and every communication was controlled by Context Manager.

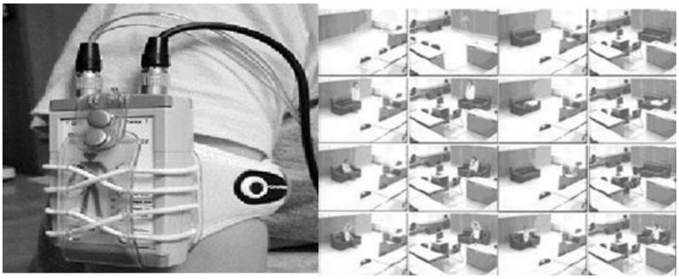


Fig. 3. ECG sensor device and its location on users' body

3.2 Emotion Detect Agent

Experiments of emotion classification were conducted using Emotion detect Agent system, which dealt with conveyed signals from ECG sensor stated above paragraph. The Agent system analyzed and generalized conveyed users' ECG signals from Smart home to prepare experiments on classification of emotions. Though the generalized data, it found out featuring-points and saved the figures of featuring-points in database.

Feedback is a figure generated from users who were allowed to select an emotion that they had felt during the experiment step. Emotion learner mainly dealt with learning, the learning step was divided into two steps. First, it leant emotions from ECG data and users' feedback which initially came from the figures of featuring-points in database. And it then learnt users' feedback emotions and ECG data at the same time. For this learning, Decision Tree algorithm was adapted as classifier.

Emotion Predictor predicts emotions after it learnt certain emotions in User state and Data Manager saves figures of the featuring-points generated from generalization step. At last, emotion analyzer conducts comparative analysis the rate of emotion prediction at pre-prediction step.

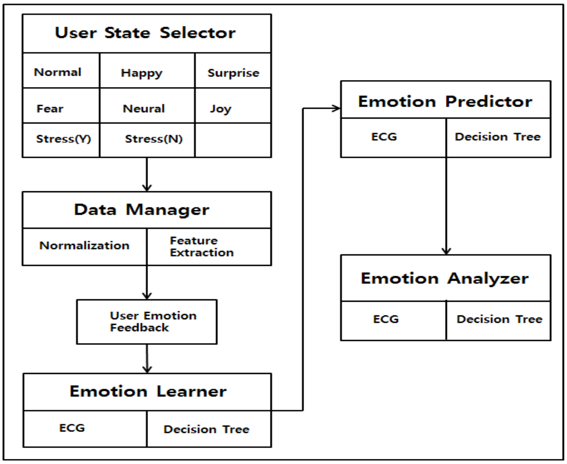


Fig. 4. Agent system on emotion perception

The agent of system on Emotion perception classifies Normal, Happy, Surprise, Fear, Neural, Joy and Stress.

4 Experiment

4.1 ECG Signals

ECG (Electrocardiogram) is an electric signals released by heart activities, which is used as a reference that can identify conditions and diseases of the heart [8]. ECG consists of five ripple marks; P, Q, R, S and T, which verify signals according to height of ripple marks and features of interval. In this study's experiments, measured ECG signals were experimented to classify emotional conditions based on situation perceiving with classification algorithm. Feature extraction of signals was conducted to extract R-R interval. When the changes of the intervals of R-R interval were analyzed, the activated rate of sympathetic nerve and parasympathetic nerve, which consists of autonomic nerve system, could be comprehended. They have been fully investigated and understood that they are one of the most sensitive factor reflecting condition of stress in human body [9, 10]. Therefore, R-R interval of ECG was extracted and used in this study's experiments.

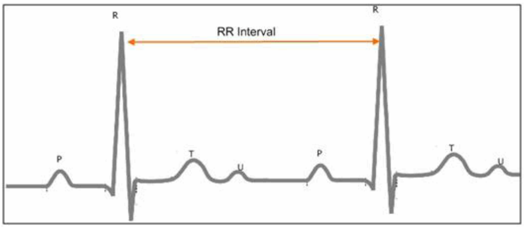


Fig. 5. R-R interval

4.2 Modified Algorithm of Decision Tree

Decision Tree is the method that fulfills Classification and Prediction with them schematizing. In other words, it learns Decision Tree from Training and Tuple in Class label. There are several indications according to each sector: Internal node of Decision Tree conducts examination in terms of attributes, each Branch indicates results of the examination and Left node reveals the distribution of Class label.

It is Root node that is placed on the highest position. Internal node and Left node are illustrated quadrangle and ellipse in each. Decision Tree is adapted in experiments of the classification. If Tuple X which is unknown in Class label will be given, the attribute value of the Tuple is examined by Decision Tree and the predicted class of the certain Tuple is gained with the path chased from Root node to Left node [11].

Thus, if the Decision tree is constructed once, it is significantly simple to distribute one of the examination records. The process is started from Root node and follows proper Branch according to results of the examinations applying conditions of the examination into the records [12].

After above process, new conditions of the examinations will be applied or arrive Left node when it reaches another External node. When it reaches Left node, the attribute of Class labels related to each node is configured in the record.

Therefore, Decision Tree classifier shows relatively higher accuracy than other algorithms. In this study, the reinforcement of the accuracy was sought with Decision Tree algorithm modifying. In this study, FP-Tree algorithm was applied into Decision Tree. FP-Tree algorithm is bottom-up method, which generates frequent-sector-set in FP-Tree exploring tree [12]. Since it reduce overfitting, it also contribute to improve the accuracy of the algorithm. Fig 7 represents that FP-Tree was applied into this study's experiments and the classified emotions were Normal, Neural, Fear, Surprise, Happy, Joy, Stress(Yes) and Stress(No) in this experiments. Applied FP-Tree had firstly divided Normal at once and Neural and Surprise were then divided next. When Fear was appeared under Neural, this tree started to analyze Stress(Yes) generated on the bottom side of Surprise. And Joy located under of Fear was comparatively analyzed with Stress(No) located under Happy which had been classified from the other Node of Neural.

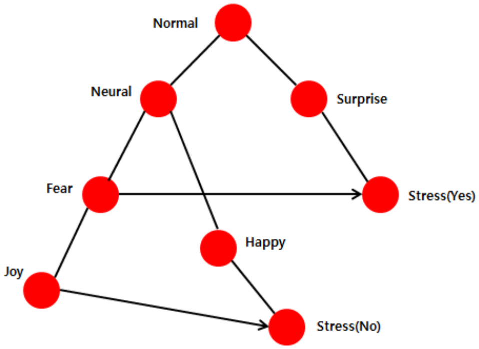


Fig. 6. Modified Algorithm

4.3 The Results of the Experiment

The experiments were conducted in the agent of emotion perception. Participants for the test of this study's algorithm were composed of 10 numbers of male and female students in an university. Each participant tested the services provided by Smart-home system in four rooms such as DVD-room and the signal data were collected in ECG signal conveyed from ECG sensors. The sensors-ECG- were equipped on participants' both of two wrists and their left legs. The results of the experiments were comparatively analyzed with established researches.

Table 1. The results of the comparative analysis on Decision Tree and K-means

Emotion	K-means(Pre-Study) Accuracy (%)	Decision Tree Accuracy (%)
Normal	71.7	97.36
Happy	59.3	99.2
Surprise	52.1	98.8
Fear	55.8	99.58
Neural	75.1	96.57
Joy	67.5	99.26
Stress(yes)	83.2	98.33
Stress(no)	83.2	97.5
Average	68.49	98.32

The experiments using K-means were the results of established researches in the past. In the results of K-means, Normal, Happy and Surprise were Juyeon[13]'s results and Fear, Neural, Joy. Stress(Yes) and Stress(No) were Choi[14]'s research results. When the results were compared to this study's results, Decision Tree could generate better results on Accuracy since it has shown over 90% of Accuracy.

5 Conclusion

In this study, Smart-home-middle-wear based on Ubiquitous-Home technology had been presented and the experiments on emotion perception using bio-signals according to the situation perception of Smart-home system were conducted.

In emotion perception section, emotion classification was done adapting the agent system and Decision Tree was adapted as the algorithm for the experiments. The results were then comparatively analyzed with the results of established researches.

Through situation perception Smart-home-middle-wear automatically provides its users with environments which the people want to enjoy. Thus, in order to measure users' statuses, the extraction of bio-signals should be needed and sensor devices, UWB location detector and communication methods for network system were also necessary to convey the collected data.

Therefore, further research which deals with sensors that could detect users' status more sensitive and location detecting system should be needed continuously. Moreover, in the emotion sector, the number of emotions and improved algorithm for analysis in agent system should be investigated more since there are more number of emotions excepting the eight emotions which were tested in this study.

Acknowledgments. This research is supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2009.

References

1. Park, Y.C.: A Study on Device Interoperability Communication Protocol for Self-integrated Control of U-Home Appliances. MS Thesis. Sejong University (2010)
2. Choi, J.H., Hwang, D.J., Shin, D.I., Shin, D.K.: Robot System Embedding Smart Home Middleware Aware of Human Being. KIISE (The Korean Institute of Information Scientists and Engineers) 26(4), 22–29 (2008)
3. Sherif, M.H.: Intelligent Homes: a new challenge in telecommunications standardization, Communication Magazine. IEEE 40(1), 8 (2002)
4. Das, S.K., Cook, D.J.: Guest Editorial Smart Homes. IEEE Wireless Communications 9(6), 62 (2002)
5. Ranganathan, A., Roy, H., Campbell: A middleware for context-aware agents in ubiquitous computing environments. In: ACM/IFIP/USENIX International Middleware Conference (2003)
6. Fablet, R., Bouthemy, P.: Motion recognition using nonparametric image motion models estimated from temporal and multiscale co-occurrence statistics. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(12), 1619–1624 (2003)
7. Dey, A.K., Abowd, G.D.: The context toolkit: aiding the development of context-aware applications. In: Proceedings of the Workshop on Software Engineering for Wearable and Pervasive Computing (June 2000)
8. Park, K.S., Cho, B.H., Lee, D.H., Song, S.H., Lee, J.S., Chee, Y.J., Kim, I.Y., Kim, S.I.: Hierarchical Classification of ECG Beat Using Higher Order Statistics and Hermite Model. Kor. Soc. Med. Informatics 15, 117–131 (2009)
9. Kim, K.T.: A Study on Standardization of Measuring Time for Heart Rate Variability. MS Thesis. KyungHee University (2006)
10. Sakaribara, H.J.: Accuracy of assessment of cardiac vagal tone by heart rate variability in normal subject. Am. J. Cardiol. 67, 199–204 (1991)
11. Han, J., Kamber, M.: Data Mining + Concepts & Techniques, 2nd edn. Elsevier Inc. (2007)
12. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. 1st edn. Addison-Weseley (2006)
13. Lee, J.Y.: Research on the Emotion Recognition Agent based on Biometrics. MS Thesis. Sejong University (2008)
14. Choi, J.H.: The Smart Home Middleware based on Pattern Recognition of Physiological and Environmental Context. DR Thesis. Sejong University (2008)

DNA-S: Dynamic Cellular Network Architecture for Smart Communications

Taegyu Lee and Gi-Soo Chung

Korea Institute of Industrial Technology (KITECH), Ansan, Korea
{tglee,gschung}@kitech.re.kr

Abstract. To meet increasing smart device services, wireless networks have smaller cellular architecture compared to previous that. Nevertheless, future communication systems will need more communication channel capacity than current one. The systems will have the problems that they cannot be avoid the concentrated bottleneck by monotonous communication path through base station or mobile switching center.

To overcome these bottleneck problems, we propose a new dynamic cellular architecture called DNA with *dynamic cell*, which can be created or deleted by service servers or mobile hosts. Thus, the DNA architecture can support various communication channels among base stations and smart devices. Also, the proposed architecture saves channel capacity for accepting more mobile hosts.

Keywords: Network architecture, cellular architecture, channel allocation, handover.

1 Introduction

Now many wireless network technologies have been tried to meet increasing mobile clients such as smart devices in wireless communication environments. Smart users ask for large data transfer services including text, voice, and video messages [1][2]. Thus wireless networks require more and reliable channels. Therefore the channel bandwidth of wireless network has been wider and wider. Also, for the hosts moving across a wide area, mobile communication systems support to be built upon heterogeneous wireless overly networks that include traditional Bluetooth, WI-FI, 3G, etc. Figure 1 shows the general architecture of wireless overlay networks for smart communication environments [3].

The wireless overlay network systems must support the seamless communication services, which include vertical handoff to deliver a call from a wireless cell to the other cells among different layers as well as horizontal handoff only in the same layer.

We call controller-based scheme for traditional communication schemes through base station (BS) or mobile switching center (MSC) in this paper [4]. Thus, the bottleneck in the centralized scheme by MSC as well as the decentralized scheme by BS cannot be avoided in smart communication environments with large volume of data. Therefore, conventional cellular architectures have the problems as follows. First, they appear bottleneck effects by high call contention of large data volume, which

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 602–611, 2013.

defer or block call services of mobile hosts in a base station controller. Second, they have low communication channel bandwidth and the small number of channel path. Finally, they cannot avoid a single point of failure since they did not provide channel distribution strategy.

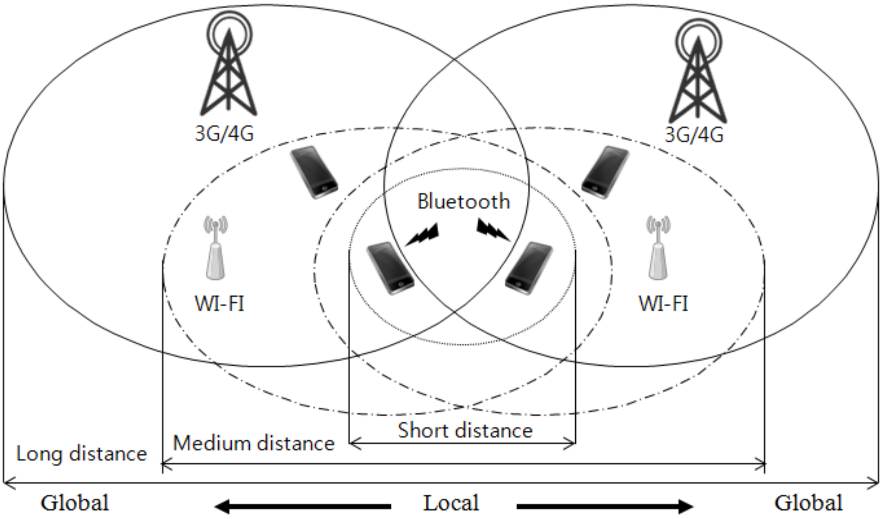


Fig. 1. Smart cellular overlay network architecture

To overcome these problems, we define DNA architecture with *dynamic cells* and different *direct channels*, which can be created or deleted by base stations or smart mobile hosts (MH) [5]. The proposed architecture offers the following advantages. It significantly minimizes call contention in central controller by distributed direct link connection. Then, it has shorter signal propagation path delay than the past one since one more direct communication paths (or channels) are constructed among mobile terminals in the short range. Also, it provides stronger call distribution strategy than conventional architecture. Finally, the proposed architecture using multiple channels of different links saves transfer time compared to channel allocation scheme of the past wireless network. This paper proposes DNA architecture and channel allocation scheme that integrate traditional wireless overlay networks with the dynamic cells.

This paper is described as follows. Section 2 presents cellular system architectures and definitions. Section 3 describes the proposed DNA architecture and algorithm. Section 4 analyzes the performance of our architecture and compares it with past ones. Finally, in Section 5, we conclude this paper with summary and future directions.

2 Cellular System and Definition

2.1 Basic Cellular Architecture

The wireless network architectures, which divide a geographical area into smaller regions called cells, have been constructed as follows. And BS manages call setup and channel assignment to mobile hosts in a cell. The BSs are fully connected by wired networks. The cellular network traditionally has the cellular models such as Figure 2. The “a) Basic model I” shows an infrastructure network based on 3G, 4G, WIFI, etc. The “b) Basic model II” shows an Ad-hoc network based on Bluetooth or WIFI links.

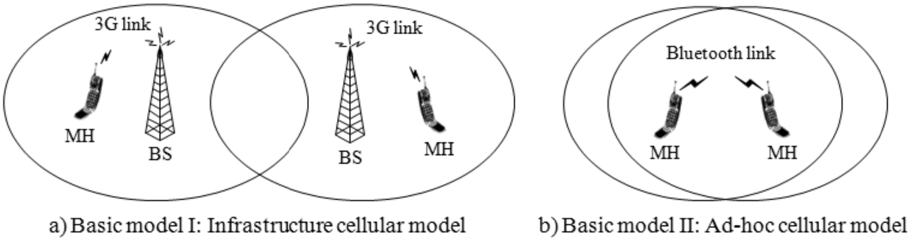

3G link
MH
BS
3G link
BS
MH
Bluetooth link
MH
MH
a) Basic model I: Infrastructure cellular model
b) Basic model II: Ad-hoc cellular model

Fig. 2. Basic cellular architecture models

2.2 Dynamic Cell and the Extensions

Dynamic Cell. Unlike the traditional cellular network that data are transferred only through BS or MSC in a cell, the proposed network can deliver data using direct channels of smart mobile hosts on dynamic cell described in Definition 1 and Figure 3.

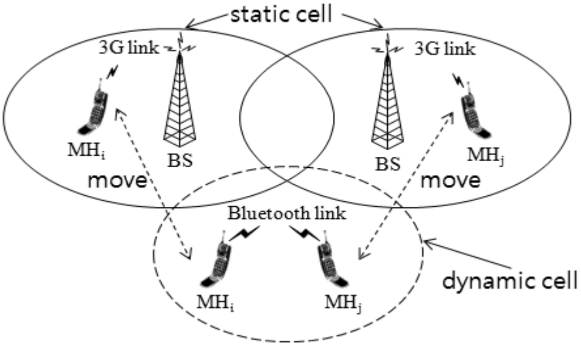


Fig. 3. Static Cell and Dynamic Cell

Definition 1: Dynamic cell is the cell that is generated if the distance (D_{ij}) between a mobile host (MH_i) and correspondent (MH_j) become shorter than distance threshold. The threshold value (T_D) indicates the minimum distance between a mobile host and the correspondent. When the strength of signal that the mobile host receives from the correspondent become larger than the minimum strength of acceptable signal power, direct channel or links are created.

Figure 3 show that each smart host on static cells of 3G networks dynamically moves anywhere in cellular system and they create their dynamic cell and dynamic channel by vertical handoff when two mobile hosts close to each other, and they exchange data through the dynamic channel and vice versa.

Dynamic Cell Extensions. For this organization of dynamic cells, the components of wireless networks should be expanded as follows. Extended dynamic channel models are consisted of the mixture of basic models in Figure 2.

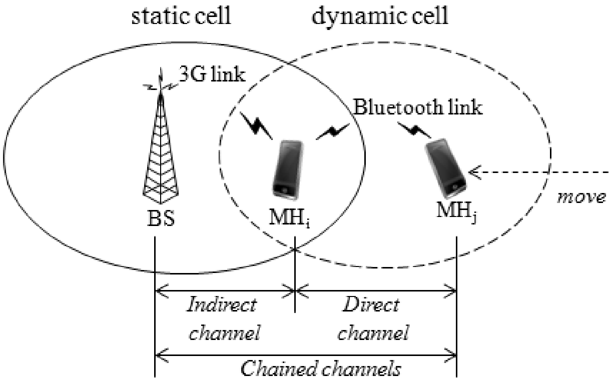


Fig. 4. Chained channel model

First, the chained channel extension consists of the mixture of single indirect 3G channel between MH and BS and one more direct Bluetooth/WIFI channel between MH and MH as shown in Figure 4. It is called of *chained channel*. All MHs except of terminal node participating to the chained channel should operate multiple channels on itself.

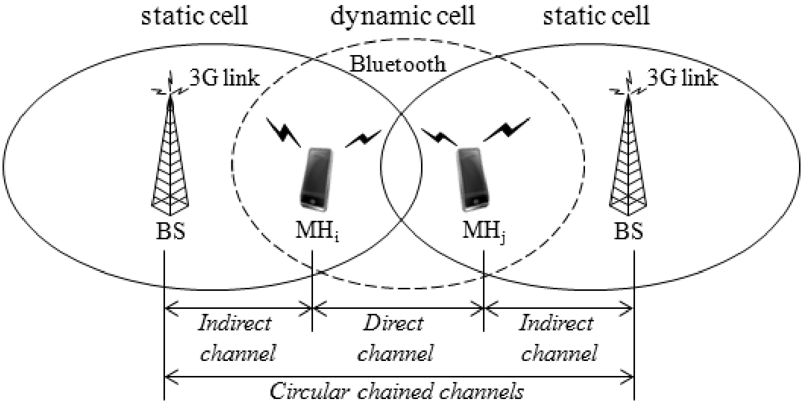


Fig. 5. Circular chained model

Second, the circular chained channel extension consists of the mixture of a indirect 3G channel between MH and BS, and one more direct Bluetooth/WIFI channel between MH and MH, and a indirect 3G channel between MH and BS as shown in Figure 5. Therefore, the path of a wired backbone and wireless access network forms a physical loop through the paths from the BS, via one or more of the MH links, and to the end BS. It is called of *circular channel*. All MHs participating to the circular channel should operate multiple channels on itself.

3 Dynamic Cellular Network Architecture

3.1 DNA System and Cell Operation

The following cell operations can execute the global wireless channel communications in DNA. The mobile hosts and base stations dynamically manage the cells by the following operations [5].

Dynamic Cell Creation. The dynamic creation is operated as controller-initiative and host-initiative as follows. First, for controller-initiative, when the distance between two MHs which exchange data through a BS in the traditional cellular mobile networks is equal to or shorter than T_D, the *BS* asks the creation of dynamic cell and direct channels for the MHs. Then, the MHs perform the vertical handoff process from indirect channels to direct channels. Second, for host-initiative, a MH broadcasts the direct channel request messages to communicate with the correspondents in its dynamic cell. If the correspondents reply acknowledge message within finite time, a dynamic cell (or direct link) between the initiative MH and the correspondents is generated.

Dynamic Cell Destruction. Dynamic cells are destroyed under the following conditions. First, a BS of high-tier cell informs its channel available state to MHs in low-tier dynamic cell overlaid with its cell if it has the number of available channels greater than threshold value. If a MH wants to handoff vertically into the high-tier overlay cell, its dynamic cell is diminished. Second, when a MH finds that the power of its receiving signal go down into the minimum acceptable power of receiving signal, it informs this state to correspondents, and then tries vertical handoff or terminates its communication. Hence, its dynamic cell is destroyed.

Dynamic Vertical Handoff. The vertical handoffs are divided into upward handoff and downward handoff. For upward handoff, the MH in a dynamic cell returns its channel and requires new channel to controller (or BS) in upper network layer. If there is available channel in the upper layer, the controller accepts the MH channel request and handover the MH's services to its upper layer. Otherwise, the services of the MH are terminated. For downward handoff, if the controller delivering messages among MHs found the condition that the MHs in themselves can exchange transmission signal by direct links, the MHs return their channels to each controller and synchronize their dynamic channels and correspondent's dynamic channels.

Dynamic Horizontal Handoff. The dynamic horizontal handoff is the process that the ongoing serve of smart host in a dynamic cell is transferred to different smart hosts in other dynamic cell. This process creates chained channels and circular chained channels to get multiple channels and to increase channel bandwidth.

3.2 Channel Assignment Algorithm

The smart mobile hosts are in either the same traditional cell or different cells. We only describe the host-initiative method in this paper. However, the controller-initiative method is easily updated based on the host-initiative method. There is BS_i or BS_j, which informs dynamic cell setup available to MH_i and the correspondent MH_j.

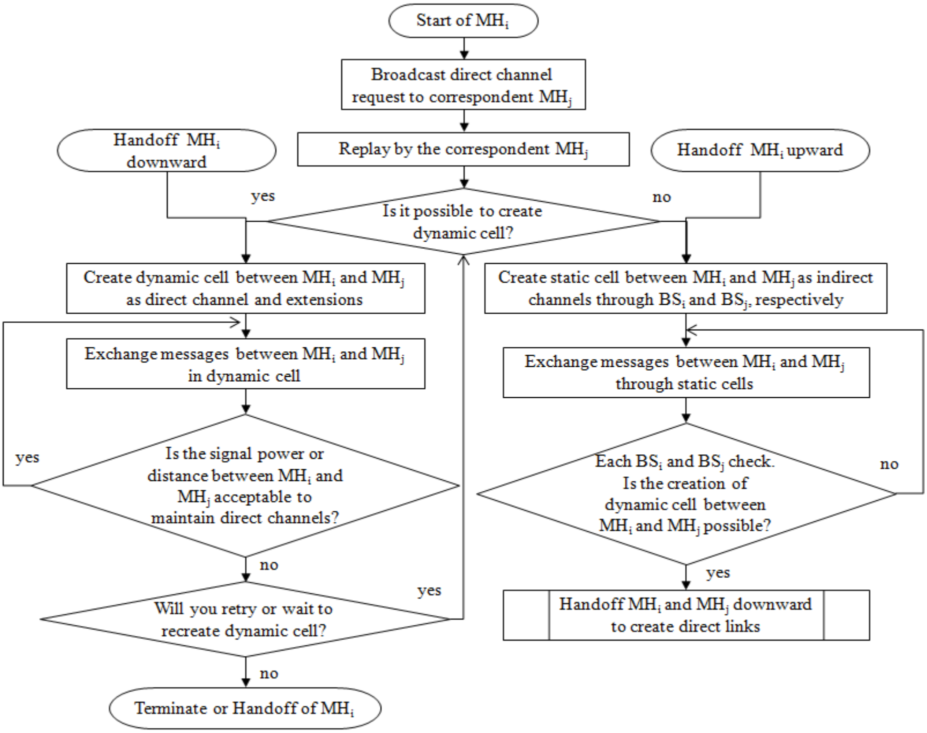

Start of MH$_i$
Broadcast direct channel request to correspondent MH$_j$
Handoff MH$_i$ downward
Replay by the correspondent MH$_j$
Handoff MH$_i$ upward
yes
Is it possible to create dynamic cell?
no
Create dynamic cell between MH$_i$ and MH$_j$ as direct channel and extensions
Create static cell between MH$_i$ and MH$_j$ as indirect channels through BS$_i$ and BS$_j$, respectively
Exchange messages between MH$_i$ and MH$_j$ in dynamic cell
Exchange messages between MH$_i$ and MH$_j$ through static cells
yes
Is the signal power or distance between MH$_i$ and MH$_j$ acceptable to maintain direct channels?
Each BS$_i$ and BS$_j$ check. Is the creation of dynamic cell between MH$_i$ and MH$_j$ possible?
no
no
yes
yes
Will you retry or wait to recreate dynamic cell?
Handoff MH$_i$ and MH$_j$ downward to create direct links
no
Terminate or Handoff of MH$_i$

Fig. 6. Direct Channel Assignment Algorithm

Figure 6 shows that communication channels are dynamically allocated to smart hosts. When a MH_i wants to communicate with its correspondent MH_j ($MH_j \in$ {Neighborhoods of MH_i}), it firstly broadcasts the call request within distance threshold value T_D. Then, the MH_j receives the broadcast messages, the MH_j replies with acknowledge message if it is possible to communicate with the MH_i, and non-acknowledge otherwise. Or MH_i's BS_i receives the broadcast messages, pages the location of MH_j and replies with an acknowledge message if it knows the location of MH_j and if it is possible to create a dynamic cell of MH_i and MH_j, and non-acknowledge message otherwise. Third, if it is possible to generate a dynamic cell between MH_i and the correspondent MH_j, direct channels (including chained or circular channel) are created between MH_i and MH_j, the direct links are continued until the signal power between MH_i and MH_j is worse. Otherwise, if it is impossible to generate a dynamic cell between MH_i and correspondent MH_j, BS$_i$ and BS_j of MH_i and MH_j, respectively, setup indirect communication between MH_i and MH_j as previous channel assignment method. And, the indirect communication is maintained until the direct communication between MH_i and MH_j be possible. Finally, when MHs cannot continue the direct channels any more, they call for vertical handoff setting indirect channels to the controller of upper network layer. Also, when BSs identify that the indirect communication did not need any more, they call for vertical handoff setting direct channels to the MH_i and MH_j for creating of lower dynamic cell.

4 Performance Analysis of DNA

For performance analysis of the proposed DNA, this paper assumes that the network systems have duplex channels as IS-136 and IS-54 separate transmit and receive channels by a constant bandwidth [4]. In the case of full duplex channels, the system provides four independent channels for every MH. It assumes that the calls in cellular architecture are arrived at Poisson process λ and the total number of channels in cellular system is n.

4.1 Channel Capacity Based on Traffic Rates

This section compares efficiency of channel capacity in DNA and that of traditional cellular system. It is assumed that the traffics are generated as Poisson processes.

We consider efficiency of channel capacity of DNA and previous approach in two cases that either one (or half-duplex) channel or two (or full-duplex) channels are supplied for each MH.

First, in the half-duplex system, the channel capacity of previous architecture (C_{hdold}) and our architecture (C_{hdnew}) is expressed as the following equations.

$$C_{hdold} = n - 2\lambda, \quad for \quad n/2 \geq \lambda \geq 0 \tag{1}$$

$$C_{hdnew} = n - \lambda, \quad for \quad n \geq \lambda \geq 0 \tag{2}$$

Second, in the full-duplex system, the channel capacity of previous architecture (C_{fdold}) and our architecture (C_{fdnew}) in the full-duplex system is expressed as the following equations.

$$C_{fdold} = n - 4\lambda, \quad for \quad n/4 \geq \lambda \geq 0 \tag{3}$$

$$C_{fdold} = n - 2\lambda, \quad for \quad n/2 \geq \lambda \geq 0 \tag{4}$$

Then, the following diagrams illustrate these functions of channel capacity.

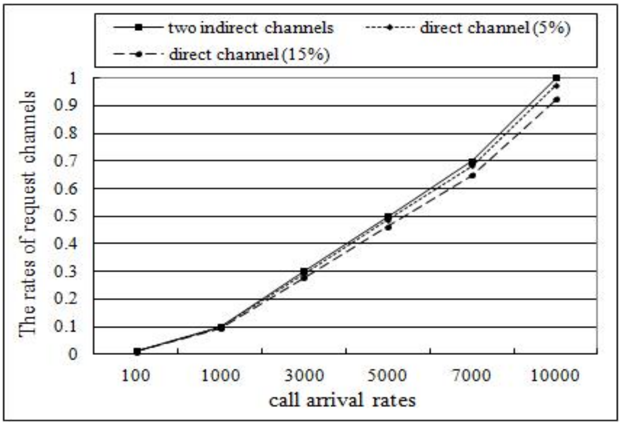


Fig. 7. Comparisons in Half-duplex channels

For showing enhanced performance of our architecture in real world, we will compare the number of request channels of the proposed architecture and that of previous architecture in two cases of half-duplex and full-duplex channels. First, for half-duplex channels, previous cellular architecture using indirect channels between MH and the correspondent needs two indirect channels for a shared channel of each MH. However, DNA needs only one shared direct channel due to direct links. So the efficiency of channel capacity of our DNA overcomes that of previous architecture in all cases that traffic percentages formed by dynamic cells over the total number of network traffics are 5% and 15% as shown in Figure 7.

Then, for full-duplex channels, previous cellular architecture using indirect channels between MH and correspondent need four indirect channels for two separate channels (forward or reverse links) for each MH. However, DNA needs only two shared direct channels due to direct links. So the efficiency of channel capacity of our DNA overcomes that of previous architecture in all cases that traffic percentages formed by dynamic cells over the total number of network traffics are 5% and 15% as shown in Figure 8.

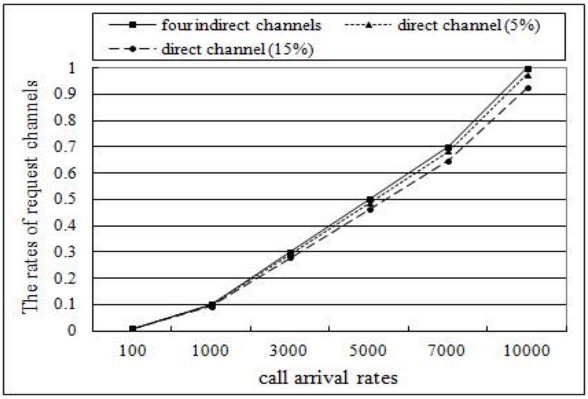


Fig. 8. Comparisons in Full-duplex channels

Because the number of saving channels becomes high as the percentage of DNA become high, DNA is appropriate to the distributed and local network with locality.

4.2 Communication Delay

In this section, we assume that D_d means the signal propagation delay from a *MH* to the correspondent *MH* through direct links of a dynamic cell. And D_i indicates the signal propagation delay from a *MH* to the correspondent *MH* using indirect links relayed by base stations. Thus, as shown in Definition 2, the proposed architecture has delay reduction of the delay difference, D_{di} that subtracts D_d from D_i.

Definition 2: D_{di}, the delay difference of D_d and D_i means the difference of direct communication and indirect communication. The direct communication is constructed by direct links between a MH and the correspondent. But, the indirect communication depends on the controllers that provide indirect links for MHs. The most of D_{di} is the communication delay for assigning communication channels to caller and callee.

If the probability of traffics supported by direct communication in dynamic cell is α, the proposed DNA has the efficiency that reduces communication delay of (α $*D_{di}$).

4.3 Contentions on Central Server

In typical network architectures, all calls including call setup, data signal and call release are served only through a controller (MSC or BS). So the conventional architecture raises high contention problem. Also, it has a single point of failure.

However, DNA architecture with dynamic cells basically can avoid the accesses on a controller for exchanging data. Only when controller-side requires information about the creation of dynamic cells, mobile hosts access on their servers. Therefore, the proposed architecture minimizes the access counts on resource controller. It can avoid a single point of failure. For example, it assumes that the probability of traffics supported by direct communication in dynamic cell is. The ratio of controller-side contentions of asking resource allocation in DNA compared to that in conventional architectures should be minimized as (1-α) ratio. Thus the distribution of traffics is more enhanced by the controller-independent strategy of dynamic cells.

5 Conclusions and Future Works

The proposed architecture shows the following advantages. First, it significantly decreases call contention on a base station as a controller since smart hosts access to the base stations only when they setup their dynamic cell. Second, it minimizes the communication delay because smart hosts in a dynamic cell have the higher bandwidth and the shorter direct path. Third, two mobile hosts using two channels (an uplink and a downlink) in a dynamic cell save wireless channels compared to channel allocation using four channels of past wireless networks. Finally, it served the stronger call distribution strategy since the communications among smart hosts within dynamic cell are possible although the server shutdown.

These current systems including dynamic cells and handoff functions may have the problems that complicate the control logic and deteriorate the signal quality. But, the problems will be solved because future mobile terminals will have the high signal quality and powerful transmission distance through the advanced hardware solutions.

References

1. Poslad, S.: Ubiquitous Computing: Smart Devices, Environments and Interactions. Wiley (2009) ISBN: 978-0-470-03560-3
2. Valhouli, C.A.: The Hammersmith Group, The Internet of things: Networked objects and smart devices, the hammersmith group research report (February 2010)
3. Katz, R.H., Brewer, E.A.: The Case for Wireless Overlay Networks, pp. 621–650. Kluwer Academic Publishers (1996)
4. Harte, L., Prokup, S.,, R.: Cellular and PCS, pp. 118–121. McGrawHill (1997)
5. Lee, T., Hwang, C.-S.: An Enhanced Wireless Network Architecture for Future Communication Environments. In: the 3rd WPMC 2000, pp. 491–496 (November 2000)

Predicting of Abnormal Behavior Using Hierarchical Markov Model Based on User Profile in Ubiquitous Environment

Jaewan Shin, Dongkyoo Shin, and DongIl Shin

Department of Computer Engineering and Science, Sejong University
98 Gunja-Dong, Gwangjin-Gu, Seoul, Korea
shinnom@gce.sejong.ac.kr, {shindk,dshin}@sejong.ac.kr

Abstract. In this paper, we model the multilevel statistical structure as Hierarchical Hidden Markov Models (HHMM) for the problem of predicting the state of human behavior based on user profile in a ubiquitous home network. Algorithms to analyze the behavioral patterns of a user using the information provided by the user in a home network system. We propose the detecting of abnormal behavior algorithm, which builds profile based on the actions taken when the user enters a room. The main contributions of this paper lie in the application of the shared structure HHMM, the estimation of the state of a user's behavior, and the detection of abnormal behavior. The user behavior data from an experiment show that directly modeling shared structures improves the recognition efficiency and prediction accuracy for the state of a human's behavior when compared with a flat HMM.

Keywords: Hierarchy Hidden Markov Model, Viterbi algorithm, Ubiquitous home network, detecting abnormal behavior.

1 Introduction

In the process industries, building intelligent systems in ubiquitous home networks is the goal of much research [1]. As intelligent environments are deployed to build ubiquitous home networks, active operations, rather than passive, can be used for household appliances and tools, and in building a made-to-order model environment reflecting the user's preferences.

For an intelligent home network configuration, the home network provides services tailored to the characteristics of an individual, with minimal intervention in the home's appliances and tools. To accomplish this, information about the user and environment are collected, which require a management and processing system.

Behavior prediction is a much needed requirement to provide convenient and efficient services in a ubiquitous environment [3]. In order to improve the above factors, the intelligent prediction of a user's behavior plays a very important role. The Hidden Markov Model (HMM) recognizes patterns of sequential data and is a proven prediction technique that is used in many fields [4-6].

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 612–622, 2013.

This paper aims to use the HHMM that has multiple levels of states which describe input sequences at different levels of granularity, to tackle two issues: first, modeling and learning sequential behaviors from human indoor spaces and second, detecting and predicting abnormal behaviors with a ubiquitous home network system. The each state of HHMM can contain sequences of nested states or observations. The advantages of HHMM are able to merge specific parts of the entire model. Another advantage of the HHMM over the HMM includes a total number of states needing to be identified. In this paper, our proposed shared structure model recognizes sequential human behaviors and is superior in the recognition and prediction accuracy of the behavior states when compared to the flat HHM. In addition, using the HHMM to model activities allows us to incorporate the structure of the behavioral hierarchy and increases amount of training data leading to better probability. We propose an algorithm for determining that the state of a user's behavior is normal or abnormal using the time data for the length of the user's behavior and the relevance of other activities.

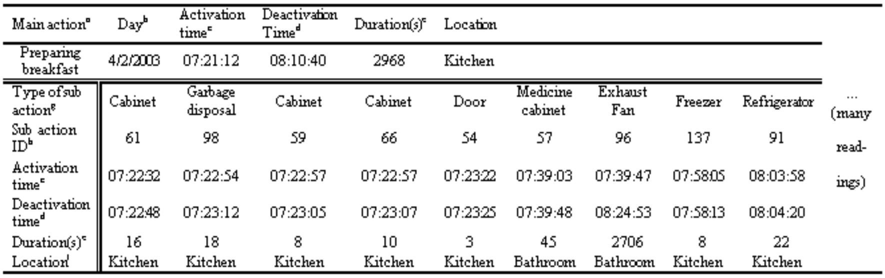
This paper is organized as follows. Section 2 gives a brief overview of the HHMM that is used in this paper to recognize and predict the behavior state. Section 3 describes the data sets [9] for a user's behavior profile, gives details of the architecture of the HHMM with the shared structure model and the algorithm for detecting abnormal behavior, and explains the HHMM implementation on the data sets. Section 4 is devoted to the experimental results, where a comparison is made of the behavior state prediction accuracies of the flat HMM-based method and the HHMM-based method with the shared structure. Finally, the conclusion is given in Section 5.

2 Model Description

2.1 HHMM

The hierarchical hidden Markov models (HHMM) are structured multi-level stochastic processes and generalize the standard hidden Markov model (HMM) [3] by autonomous processes on their own. The HMMs are trained separately and then integrated into the HHMM. After this integration their parameters are kept unchanged with further training impacting only on the higher levels of the HHMM. Since the scheme that we are modeling operates on a number of hierarchical levels, we use a four level HHMM to represent the user's actions and behavior states. Every higher-level state symbol corresponds to a stream of symbols produced by a lower-level sub-HMM. Black edges denote vertical and horizontal transitions. Dashed edges denote returns from the end states of each level to the level's parent state. The HHMM is visualized as a tree structure in which there are three types of states, abnormal and normal behavior states that are divided into two parts, which are hidden states that represent entire stochastic processes. Each behavior state is associated with an observation which is monitored by installed sensors that sensed when everyday household appliances were used. We consider the HMM as the HHMM with just the root node and the hierarchy of production states. The root state is the only level which does not have a final state. Thus, the generation of the observation sequence is completed when control of all the recursive activations is returned to the root state. Then the root state can initiate a new stochastic behavior state model. All states can be reached by a finite number of steps from the root state, so the model is strongly connected.

Table 1. The modified type of data that was acquired by the state change sensors and the experience sampling method (ESM)



Main action[a]	Day[b]	Activation time[c]	Deactivation Time[d]	Duration(s)[e]	Location					
Preparing breakfast	4/2/2003	07:21:12	08:10:40	2968	Kitchen					
Type of sub action[g]	Cabinet	Garbage disposal	Cabinet	Cabinet	Door	Medicine cabinet	Exhaust Fan	Freezer	Refrigerator	… (many read-ings)
Sub action ID[h]	61	98	59	66	54	57	96	137	91	
Activation time[c]	07:22:32	07:22:54	07:22:57	07:22:57	07:23:22	07:39:03	07:39:47	07:58:05	08:03:58	
Deactivation time[d]	07:22:48	07:23:12	07:23:05	07:23:07	07:23:25	07:39:48	08:24:53	07:58:13	08:04:20	
Duration(s)[e]	16	18	8	10	3	45	2706	8	22	
Location[f]	Kitchen	Kitchen	Kitchen	Kitchen	Kitchen	Bathroom	Bathroom	Kitchen	Kitchen	

a	Attribute of the user's activity found by using ESM
b	Day when user's activity was sensed
c	Sensor activation time in seconds starting from 12:00am
d	Deactivation time in seconds starting from 12:00am
e	Time that the sensor was activated
f	Location of the sensor (room) in the house
g	Attribute of the user's activity found by using state change sensor
h	ID number of the sub action

2.2 Structural Model and Parameters

The HHMM consists of the depth of levels D, the number of distinct observation symbols per state (in the discrete case) M, the number of sub-states of an internal state q_i^d (d=1,2,..,D-1, i=i$_{th}$ state) $|q_i^d|$, the state horizontal transition probability distribution A(q d)=(a_{ij}(q d)) and a_{ij} (q^d)=P($q_j^{d+1}|q_i^{d+1}$), the vertical transition probability π(q d)= (π d(q_i^{d+1})) = ($P(q_i^{d+1}|q^d)$), where $P(q_i^{d+1}|q^d)$ is defined to be the probability that state q^d initially activates its child state q_i^{d+1}. The HHMM includes D levels of HMM while each level is independent HMM. Moreover, each HMM only links with its parent and child. The whole parameters set of HHMM is denoted by

$$\lambda=\{\lambda(q^d)\}_{d\in\{1,\dots,D\}}=\{\{A(q^d)\}_{d\in\{1,\dots,D-1\}},\ \{\Pi\ (q^d)\}_{d\in\{1,\dots,D-1\}},\ \{B(q^D)\}\} \qquad (1)$$

A state stays the same until the next time if it does not end. If it ends and the parent state stays the same, it follows a transition to a new child-state of the same parent. The maximum-likelihood parameters for HHMMs are estimated by an EM algorithm, known as forward-backward algorithm since we need to consider stochastic vertical transitions which recursively generate observations.

3 Methodology

3.1 Data Sets

The data for this study were collected from an "Activity Recognition System [10]" that was implemented in the "Massachusetts Institute of Technology (MIT) Placelab [11]".

We modified the original data to create a hierarchical structure and determine the parameters of the HHMM (see Table 1).

A main action is a person's action of using the experience sampling method (ESM) [11,12] to label activities, and the pattern recognition. The advantages of using ESM [11,12] to label the data is that it eases the subject's burden, improves the accuracy of the time-stamps acquired, and reduces the data entry and coding burden of the researcher. The main action can be refined into a sequence of sub actions. A sub-action is a person's action of using a house hold object in which sensors were installed to monitor the user's behavior. The sub-action can continue when the main action is terminated.

3.2 Architecture of the HHMM

The HHMM was first introduced in [8] as a natural generalization to HMM with hierarchical control structure. Every higher-level state symbol corresponds to a stream of symbols produced by a lower-level sub-HMM; a transition at the high level model is invoked only when the lower-level model enters an exit state; observations are only produced by the lowest level states.

In this section, we propose the use of an upper level HMM in order to refine the results from the lower level and produce more accurate decisions by taking into account the relationship. Such four-level HMMs form a Hierarchical HMM and are a generalization of the segment HMM. Each observation of the upper level HMM can be segmented into sub-HMMs in a hierarchical fashion. Figure 2 illustrates the basic idea of the HHMM. A time-series is hierarchically divided into segments, where O_i^1 and O_i^2 represent the observations at the first and second level HMMs, O_i^3 shows the observation at the third level HMM, S_i^3 denotes the state of a user's behavior at the third level HMM, and a block of S_i^3 is the state sequence of the sub-HMMs of O_i^3.

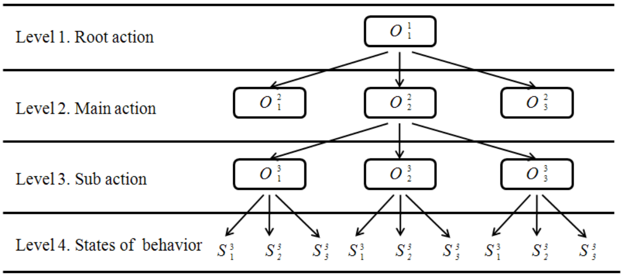


Fig. 1. The architecture of the HHMM

3.3 Algorithm for Detecting Abnormal Behavior

It is difficult to determine abnormal behaviors by using user profile data from sensors used to monitor human behavior. To define the relationship between the sub-actions and main actions, we propose that overlapping time zones where a sub-action and main action are performed indicate that the sub-action is related to other main actions.

We propose the algorithm shown in Table 2 for detecting a user's abnormal behavior based on behavior data collected from state-change sensors installed in household appliances. The algorithm consists of four parts.

Table 2. Algorithm for detecting abnormal behavior

```
If (duration($MAct_i$[a])) > duration($SAct_i$[b]))
  Then $SAct_i \in$ Normal Behavior Category01[c]
If (duration($MAct_i$)) < duration($SAct_i$))
  Then $SAct_i \in$ Exceptional Behavior Category[d]
For (Exceptional Behavior Category($SAct_i$))
  If (time($MAct_i$) $\cap$ time($SAct_i$) $\neq$NULL)
  Then $SAct_i \in$ Normal Behavior Category02[e]
  else
    Then $SAct_i \in$ Abnormal Behavior Category[f]
```

a	main action
b	sub action
c	NBC01, which includes irrelevance to other main action.
d	EBC, which includes normal and abnormal behavior.
e	NBC02, which includes a relevance to other main action.
f	ABC, which includes irrelevance to other main action.

1. *Finding normal behavior pattern01*: Normally a main action lasts longer than a sub-action. We found that 65% of sub-actions are over when the main action is finished. Thus, the behavior patterns represented in Figure 3, -as black rectangular areas and not related to other main actions were classified as *Normal Behavior Category01 (NBC01).*
2. *Finding exceptional behavior pattern*: We found that 35% of the time, sub-actions continued after the main actions were finished. These patterns, which are shown in Figure 3 as dark and light gray rectangular areas, were classified as the *Exceptional Behavior Category (EBC)*. These patterns were divided into *Normal Behavior Category02 (NBC02)* and *Abnormal Behavior Category (ABC)*, based on whether or not they were associated with other main actions.
3. *Finding normal behavior pattern02*: 15% of the patterns in *EBC* were related to other main actions and categorized as *NBC02*. *NBC02* is presented in Figure 3 as light gray areas.
4. *Finding Abnormal behavior pattern*: We found that 20% of the patterns in *EBC* were not related to other main actions.

In this paper, the data for building user profiles are measured using the ESM (Experience Sampling Method) and state-change sensors of the "Activity Recognition System" in the existing article [7].

Table 3. User profile dataset

Activity	Sensor ID	Day	Activation Time	Deactivation time	Duration (sec)	room(opt)	object type(opt)
Preparing breakfast	PDA	12/1/02	08:23:01		10 min		
	23	12/1/02	08:23:03	08:23:07	4	kitchen	drawer
	18	12/1/02	08:23:09	08:23:17	8	kitchen	cabinet
	89	12/1/02	08:23:49	08:23:59	10	kitchen	fridge door

The activity attributes are acquired using experience sampling. The sensor activations are collected by the state-change sensors distributed all around the environment. In Table 3, opt stands for optional attribute. The state-change sensors were installed on doors, windows, cabinets, drawers, microwave ovens, refrigerators, stoves, sinks, toilets, showers, light switches, lamps, some containers (e.g., water, sugar, and cereal), and electric/electronic appliances (e.g., DVDs, stereos, washing machines, dish washers, coffee machines), among other locations. In the measured user behavior pattern data, the sensor ID, Activity Date, time, terminated activity time, duration, activity rooms in place, and sensor location information were recorded [7][8].

To confirm which room users most often visit, the IDs of the rooms were set to Rm_1, Rm_2, Rm_3, $\mathrm{Rm}_4, \ldots \mathrm{Rm}_k$. And, to determine what kind of action a user takes in each room, the ID for the user's behavior was set. The IDs of the user's actions were set to Act_1, Act_2, Act_3, $\mathrm{Act}_4, \ldots$ Act_k. The user set weights to w_1, $\mathrm{w}_2, \ldots, \mathrm{w}_{n,}$representing the user's interests in rooms. These were user-specified and could be revised. The weight is represented by $\mathrm{w_i}(\mathrm{Act_i}, \mathrm{Rm_j})$, $(0 \le \mathrm{w_i}(\mathrm{Act_i}, \mathrm{Rm_j}) \le 1)$, which represented the strength of the relation between $\mathrm{Rm_k}$ and $\mathrm{Act_k}$. These were normalized to a value between 0 (no interest) and 1 (very strong interest) or could contain a NULL-value, if no interest or information was available. This paper suggests Count Profile and Duration Profile to build the user profile.

Count Profile, presenting CntP( ) in this paper, is the profile for the number of actions that a user took in a room. The count( ) function was used to determine the number of actions.

$$\mathrm{CntP(i)} = \frac{\sum_i\left(\mathrm{count}(\mathrm{Act_i}) * \mathrm{w_i}(\mathrm{Act_i}, \mathrm{Rm_j})\right)}{\sum_k \mathrm{count}(\mathrm{Act_k})} \tag{3}$$

To compute CntP(i), a user sets $\mathrm{w_i}(\mathrm{Act_i}, \mathrm{Rm_j})$ in the room where an action occurred, multiplies by the number of actions in the room, and finally sums these values and divides by the total number of $\mathrm{Act_i}$ that the user did in the room.

Duration Profie , representing DurP(i) in this paper, is a profile for the time that the user's action lasted in the room. To determine it, the durn() function is used.

$$\mathrm{DurP(i)} = \frac{\sum_{\mathrm{i}}\left(\mathrm{durn(Act_i)} * \mathrm{w_i(Act_i, Rm_j)}\right)}{\sum_{\mathrm{k}} \mathrm{durn(Act_k)}} \tag{4}$$

$\mathrm{DurP(i)}$ is computed in a similar way. To compute $\mathrm{DurP(i)}$, a user sets $\mathrm{w_i(Act_i, Rm_j)}$ in the room where an action occurred and multiplies the time that the action lasted, which is calculated in minutes. In the last, the user sums these values and divides by the total time of $\mathrm{Act_i}$ that the user did in the room. This article suggests the value $\mathrm{x_i}$ of the BPP algorithm to measure the relevance between a user in a house and a room, which is calculated using the profile data.

$$\begin{aligned} &\mathrm{x_i} = \mathrm{a} * \mathrm{CntP(i)} + \mathrm{b} * \mathrm{DurP(i)}, \\ &\mathrm{where(a+b)} = 1 \end{aligned} \tag{5}$$

Parameters a and b are used to set which profile a user places more importance on in Count Profile and Duration Profile, and can be modified by a user. For example, if a user watching TV places more importance on the time that the action lasted over the number of actions, a larger b than a value is assigned.

3.4 Implementation

In this paper, we apply the HHMM with a shared structure to predict and recognize the behaviors of people in a ubiquitous home network and ubiquitous environment. We consider three main actions – *preparing breakfast, preparing lunch,* and *preparing dinner* – the topologies of which are shown in Figure 4. Note that the topologies of these main actions are specified by using observation data set from MIT Placelab. The main and sub-actions are mapped into a shared-structure HHMM, which has four levels. Level 1 is a root action. Levels 2 and 3 are the main and sub actions, respectively. Level 4 represents the states that represent the relevance for a set of the user's behaviors, by using the time zone in which the user's behaviors were performed.

We define user's behavior states that are normal *(NBC01, NBC02)* and abnormal *(ABC)*. The three states at the fourth level are determined by using the time that the user's behavior lasted and the time zone in which the behavior was performed. The model is described as a sequence of sensing the user's behaviors and at any time as being in one of a set of *N(N=3)* distinct states: S_1, S_2,or S_3. This model undergoes a change of state according to a set of probabilities associated with the observation. In our case, this indicates the relationship between different types of activities and a change in the state on the user's behavior. We denote the time intervals associated with the state changes as t = 1, 2, ..., and the actual state at time t as q_t. This probabilistic description links the current and predecessor states:

$$a_{ij} = P[q_t=S_j|q_{t-1} = S_i],\ 1 <i, j <N,\ \textstyle\sum_j a_{ij} = 1 \tag{6}$$

In this study, we conducted a number of experiments about the behavior of a root action, *preparing a meal* and determined the transition matrix as

$$A = \{a_{ij}\} = \begin{bmatrix} 0.72 & 0.13 & 0.15 \\ 0.35 & 0.42 & 0.23 \\ 0.27 & 0.19 & 0.54 \end{bmatrix} \tag{7}$$

The initial state distribution $\pi_i = P[q_1 = S_i]$, in our case, means the probability distribution of the state of the user's behavior. Another element of the HMM is the observation symbol probability distribution in the state observation symbol probability distribution in state - S_j :$B_j(k)= P[o_k|q_t = S_j]$. b_j shows how likely it is that this model will recognize the observation symbol $O_1, O_2, ..., O_i$. O_i represents the observation made by the 1-3 level HMMs, which corresponds to Figure 4. We use the matrix of each sub-action to represent this B matrix, which can be obtained from the behavior IDs as the identity of the behavior (sub, main action) as shown in Table 1. Note that some sub-actions are shared by multiple main actions; for example, sub_action137 is shared by "preparing breakfast" and "preparing lunch". The parameters of the HHMM are the matrices $\pi^{d,p}$, $A^{d,p}$, $A^{d,p}_{[end]}$, and the observation model B, where d = 1, 2 or 3 and p is a behavior at level d. The Viterbi algorithm [16] is used at the fourth level as shown in Figure 4 to find the single best state sequence $Q = q_1, q_2, ..., q_t$), which represents the most likely underlying behavior state sequence, for the given observation sequence $O = \{O_1, O_2, ..., O_T\}$, which is obtained in the 1-3 level HMMs. Thus, some errors in the first step could be corrected by the upper level HMM.

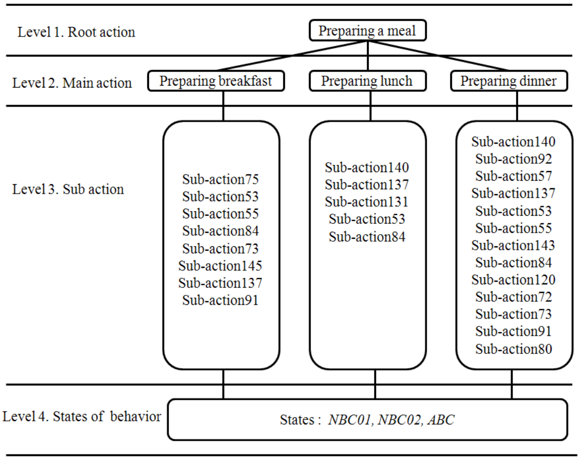


Fig. 2. The behavior and state hierarchy

4 Results and Discussion

The training data for 3, 7, and 14 days consisted of 100, 210, and 430 observation sequences obtained from the root action, "preparing a meal". In each scenario, a person executed three main actions – “preparing breakfast, preparing lunch, and preparing dinner,” and 10-150 sub-actions per main action. In the experiment, we compared the prediction of the behavior state between the HHMM with a shared-structure and the HMM with a flat structure. For the training data sets of 3, 7, and 14 days, the prediction (state sequence of sub-action) was performed for the next 14 consecutive days. The accuracies of the results are given in Table 3, where the accuracy is defined as the total number of correct predictions of the state status per 100 predictions. We can see in Table 4, that as the training set size increased from 3 to 14 days, for both the shared-structure HHMM and flat HMM, the number of correct predictions increased.

Table 4. Accuracy of state prediction

Size of data set		Accuracy	
Training set size	Prediction set size	Shared- structure HHMM	Flat-standard HMM
3days	14days	78.47%	65.32%
7days	14days	85.35%	72.21%
14days	14days	92.14%	85.47%

However, comparing these two models, it is obvious that the performance of the hierarchical HMM with the shared-structure was much better than that of the HMM with the flat structure

Detecting of abnormal behavior algorithm and behavior pattern profiles of residential users over time was used to predict the next action of the user.

Figure 5 shows that the blue points, based on actual user behavior profiles over time in the kitchen, represents a change in the value x$_i$ and the red points show the analysis of the user’s behavior pattern profile data in the kitchen over two weeks and the prediction of the next action of the user in the kitchen over time.

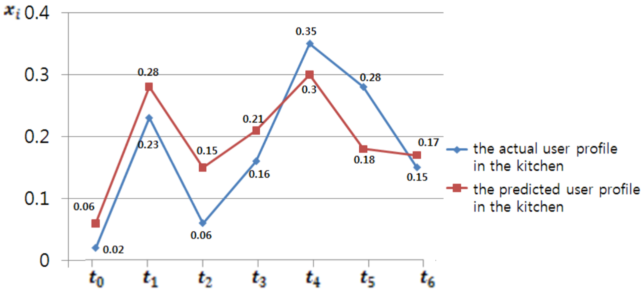


Fig. 3. Experimental result of the user behavior pattern analysis over time

The experimental result between the actual and predicted behavior profiles in the comparison of the difference value x_i represents the minimum value of 0.02 and the maximum value of 0.09.

5 Conclusion

We proposed the use of the shared-structure HHMM to recognize user's behavior states and an algorithm to detect abnormal behavior in a ubiquitous home system. An overlapping time zone determined the relationship between the main and sub-action. Experiments compared the behavior state prediction accuracies of the shared-structure HHMM and flat HMM-based methods. The results showed that by using the HHMM, the accuracy could be improved remarkably. And using information collected from user behavior patterns, can predict the preferred behavior. It was also shown that an increase in the amount of accumulated data improves the accuracy of the predictions.

In the future, will involve a home network system deployment that reflects the user's preference, studies the efficiency of the algorithm for building user profiles, and develops an algorithm that attains better performance for user behavior pattern analysis.

Acknowledgment. This research is supported by Seoul R&BD Program (SS110008).

References

1. Wu, C.L., Fu, L.C.: A human-system interaction framework and algorithm for ubicomp-based smart home. In: Human System Interactions Conference, pp. 257–262 (2008)
2. Galata, A., Johnson, N., Hogg, D.: Learning variable length Markov models of behavior. Computer Vision and Image Understanding Journal 81(3), 398–413 (2001)
3. Oliver, N.M., Rosario, B., Pentland, A.: A Bayesian computer vision system for modeling human interactions. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 831–843 (2000)
4. Oliver, N., Horvitz, E., Garg, A.: Layered representations for human activity recognition. In: Fourth IEEE International Conference on Multimodal Interfaces, pp. 3–8 (October 2002)
5. Pynadath, D.V., Wellman, M.P.: Generalized queries on probabilistic context-free grammars. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(1), 65–77 (1998)
6. Bui, H.H., Venkatesh, S., West, G.: Policy recognition in the abstract hidden Markov model. Journal Journal of Artficial Intelligence Research 17, 451–499 (2002)
7. Bui, H.H., Phung, D.Q., Venkatesh, S.: Hierarchical hidden Markov models with general state hierarchy. In: Proceedings of the Nineteenth National Conference on Artificial Intelligence, San Jose, California, pp. 324–329 (2004)
8. Fine, S., Singer, Y., Tishby, N.: The hierarchical hidden Markov model: Analysis and applications. Machine Learning 32(1), 41–62 (1998)
9. Bui, H.H., Phung, D.Q., Venkatesh, S.: Hierarchical hidden Markov models with general state hierarchy. In: In Proceedings of the Nineteenth National Conference on Artificial Intelligence, San Jose, California (2004)
10. Tapia, E.M., Intille, S.S., Larson, K.: Activity recognition in the home using simple and ubiquitous sensors. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 158–175. Springer, Heidelberg (2004)
11. `http://architecture.mit.edu/house_n/placelab.html`

12. Intille, S.S., Rondoni, J., Kukla, C., Anacona, I., Bao, L.: A context-aware experience sampling tool. In: Proceedings of the Conference on Human Factors and Computing Systems: Extended Abstracts. ACM Press (2003)
13. http://courses.media.mit.edu/2004fall/mas622j/04.projects/home/
14. Csikszentmihalyi, M., Larson, R.: Validity and reliability of the experiencesampling method. The Journal of Nervous and Mental Disease 175(9), 526–536 (1987)
15. Stone, A.A., Shiffman, S.: Ecological momentary assessment (ema) in behavioral medicine. Annals of Behavioral Medicine 16(3), 99–202 (1994)

Advanced Facial Skin Rendering with Actual Fresnel Refractive Index Reflecting Facial Tissue Features

Sunghee Lee and Seongah Chin[*]

Division of Multimedia Engineering, Sungkyul University, Anyang-City, Korea
{sunghi014,solideochin}@gmail.com

Abstract. Realistic shading models can be synthesized in real-time game play environments in which seamless scenes are able to be simulated by reflecting light transport through layered material. In this paper, we have proposed an advanced facial skin rendering as conveying actual Fresnel refractive indices derived from actually measured data on a real human face. The Fresnel index is contingent on the location on a face because facial tissue components are slightly different. To realize physically-based rendering, we have employed a hybrid shading technique that can be merged from both improved Oren-Nayar and layered Phong model. We have shown experimental results to verify the proposed method as well.

1 Introduction

Photo-realistic rendering has to take into consideration the amount of light transport passing through or being reflect as accurately as possible. In addition, light can be more scattered by being reflected, absorbed and refracted specially in layered materials. In recent years, physically-based rendering has increasingly drawn attention in game industry as the performance of GPU becomes powerful. A set of seamless scenes can be rendered through computing the amount of light transport through layered materials. However, material properties such as absorption coefficients, scattering coefficients and Fresnel refractive indices still tend to be approximately acquired. For instance, a fixed Fresnel refractive index for skin is used when synthesizing skin [1][2][3][4]. In real, the quantity of reflectance and refraction are dependent on the location on a face. Facial tissue consists of mainly three layers including epidermis, dermis and subcutis. Each layer contains various components such as melanin, hemoglobin and fibers etc. Hence the amount of the light slightly goes down naturally caused by thickness of a tissue as well as material components when a ray moves.

Ding et al. [5] developed method of reflectance measurement to compute refractive indices of epidermis and dermis. However they investigated Fresnel data mostly sampled from abdomen regions in which only showed optical aspects not considering synthesizing skin. Petrov et al. [6] also presented computational platform to simulate

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 623–630, 2013.

transmittance and reflectance spectra of human skin. The measurements were made from different locations of human hand including fingertip, finger, palm wrist and forearm. The experimental setup might be suitable only for the material that has a certain depth because laser beam pass through the material then light that penetrates the material could be measured.

In this paper, an advanced facial skin rendering technique is proposed. Actual Fresnel refractive indices are derived from actually measured data on a real human face. The Fresnel refractive index is computed in considering the location on a face to deliver accurate amount of light. To simulate the proposed approach, we have used a hybrid shading model that merges improved Oren-Nayar into layered Phong model [7].

2 The Proposed Approach

2.1 Fresnel Measurement

It is critical to acquire accurate Fresnel refractive index when computing the amount of light of materials being composed of several layers. The incoming rays go through layers as be absorbed, refracted or reflected. The property of material has to be taken into consideration to calculate exact the quantity of lights. Reflection and refraction of the light in general have to be considered when light comes through a layer of a refractive index *n1* into a second layer with refractive index *n2*, The Fresnel equations defined as (1) calculate the amount of the light is reflected and is refracted.

$$F_r(\theta) = F0 + (1 - F0)(1 - L \cdot H)^5, \tag{1}$$

where *F0* is the reflectance at the normal incident, *L* is the light direction and *H* is the half vector between *L* and *E* (eye).

By Schlick approximation, equation (1) can be rewritten as (2)

$$F0 = (\tfrac{1-n}{1+n})^2, \tag{2}$$

where *n* is the refractive index.

Spectrophotometry is to measure the amount of the reflection or transmission of a material using a function of wavelength. A spectrophotometer can be used by which intensity is derived from a function of the light source wavelength.

In the proposed research, we have used CM-600d shown in Figure 1 that is a spectrophotometer. This model seems to be more compact and lightweight body while retaining the sophisticated functions of Konica Minolta's than conventional models. The principle of CM-600d is to capture the amount of light using integral sphere.

At first, we capture reflectance values from the various regions of a real human face. There are two modes representing SCI (specular light included) and SCE (specular excluded) respectively. The difference of reflectance between SCI and SCE represents the reflectance at normal direction. The refractive index given *F0* can be solved using equation (2). The red dotted lines indicate reference materials while the blue lines display samples captured in Figure 1. The graphs on the top are from nose regions and the ones on the bottom show the reflectance around zygoma on a face.

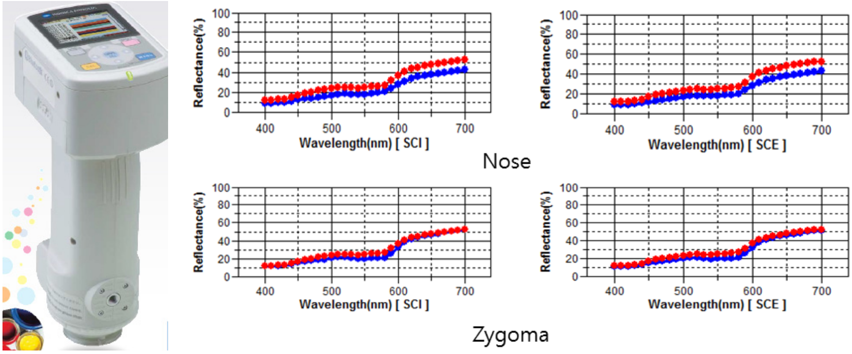


Fig. 1. CM-600d and reflectance graphs

2.2 Improved Oren-Nayar

To simulate skin rendering, we have employed approximated Oren-Nayar model, In particular, slightly rough materials like skin, a diffuse term can be rather critical than the specular light. Oren-Nayar model is designed to simulate a diffuse reflectance model with a Lambertian surface. The method takes into consideration Lambertian diffuse reflection from all microfacets. To realize this, we have used Qualitative Model shown in equation (3) that can be suitable for real-time.

$$L_r = \begin{cases} \text{if}\big(E\cdot L-(N\cdot E)(N\cdot L)\big)\geq 0 \\ \quad \frac{\rho}{\pi}E_0\left[(N\cdot L)\left(1-\frac{1}{\gamma+\alpha s}\right)+\left(\frac{1}{\delta+\beta s}\big(E\cdot L-(N\cdot E)(N\cdot L)\big)\text{Max}\left(1,\frac{N\cdot L}{N\cdot E}\right)\right)\right] \\ \text{otherwise} \\ \quad \frac{\rho}{\pi}E_0\left[(N\cdot L)\left(1-\frac{1}{\gamma+\alpha s}\right)+\left(\frac{1}{\delta+\beta s}\big(E\cdot L-(N\cdot E)(N\cdot L)\big)(N\cdot L)\right)\right] \end{cases}, \tag{3}$$

where $\alpha = 0.65$, $\beta = 0.1$, $\gamma = 2$, $\delta = 2.22$, s is shininess coefficient, ρ is diffuse albedo, E_0 is incoming irradiance, L is light direction, N is surface normal and E is eye direction.

2.3 Layered Phong Model

To render physically based materials, the amount of light through layer of materials has to be computed. When a light transmit into a layer the quantity of light can be reduced because of refraction and absorption. In real materials, it is not hard to find such layered materials. For instance, skin consists of three layers including dermis, epidermis and subcutis. If he or she wants to render these kinds of materials, then results are totally dependents on his or her experience. To overcome the limitations, light transport needs to be traced.

Fundamental Phong model can be approximated with some parameters using equations (4)

$$I_p = k_a I_a + k_d(L\cdot N)i_d + k_s(R\cdot V)^{\alpha} i_s, \tag{4}$$

where i_s is specular light, k_s is secular coefficient, i_d is diffuse light intensity, k_d is diffuse coefficient, i_a is ambient light intensity, k_a is ambient coefficient, and α is shininess.

We realize that skin consists of three layers. However, only Phong model cannot represent light propagation that navigates into three layers of the skin. Subsurface scattering that can be derived physically-based rendering needs to be considered. However computational time for sub-surface scattering is very high. Hence we come to the method that skin rendering can be realized by real measured Fresnel index that takes into account dual layer. We have realized dual-layered Phong model, which simulates far better appearance than a single-layered rendering. To implement this, we introduce Fresnel in the bottom layer as equation (5).

$$F_b(n1,n2) = (1 - F(n1))(F\left(\frac{n2}{n1}\right))\left(1 - F\left(\frac{1}{n1}\right)\right), \tag{5}$$

where $F(n)$ is the Fresnel equation, $n1$ is the refractive index of the top layer and $n2$ is the refractive index of the bottom layer. $F_b(n1, n2)$ is the reflectance of the bottom layer and the Fresnel term. Thus, the specular intensity from the bottom layer can be derived with equation (6)

$$I_{specular} = \mathrm{I_s} \cdot \mathrm{F_b}(n1,n2)max\left(0,1 - \alpha\left(\frac{2}{N \cdot E}\right)\right), \tag{6}$$

where α is modified absorption coefficient associated with a combination of the original coefficient and thickness. I_s is image-based specular light. The diffuse term is defined as equation (7)

$$I_{diffuse} = I_d \cdot (1 - F_b(n1,n2))max\left(0,1 - \alpha\left(2 + \frac{1}{N \cdot E}\right)\right), \tag{7}$$

where I_d is image-based diffuse light.

3 Results and Discussion

We have carried out experiments under Intel Core (TM) i5 CPU 750@2.67GHz, 4.00GB, NVIDIA GeoFore GTS 250 (VGA) with Windows 7 To verify the proposed methods, The proposed methods have been realized using Microsoft MFC and OpenGL API and CG. The first experiment is about calculation of Fresnel refraction index. Know reflection at normal direction, we solve refractive index. Secondly, we have shown rendering results using improved Oren-Nayar and Phong-Layered shading method (IONPL).

3.1 Fresnel Refraction Index

We have obtained reflectance values from the regions of a real human face. As shown in Figure 2, 20 regions on a real face have been captured at first. Color system *L*a*b** is used. *L** indicates reflectance while other two components mean color components. The horizontal axis displays the region index on a face. The vertical axis indicates measured values.

The refractive index known F0 can be solved using equation (2). In Table 1, F_L* is reflectance at normal direction using $L*$ while F_Y is acquired from Y reflectance. Then n_L* refractive index with $L*$ and n_Y refractive index with Y can be solved as well. Interestingly zygoma and forehead show high refractive index values, which implies that lights are found to be rather refracted than other regions. Moreover, around the nose region shown the lowest refractive index since the tissue of nose seems thin meaning that the light pass through tissue layers of nose without heavily refractive light.

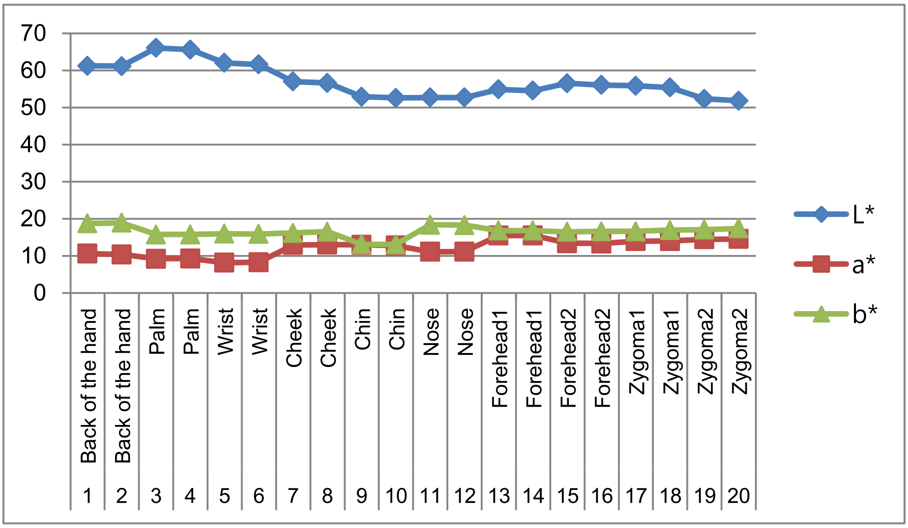


Fig. 2. The graph showing L*a*b*

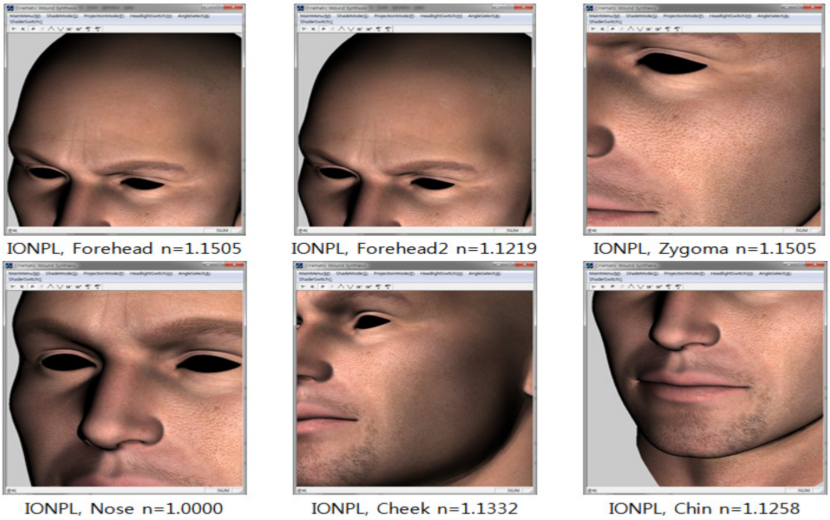
In general, the refractive index of skin is known to be 1.3 [2][7]. However, in our experiments, we have found the refractive index is contingent on the regions on a face. Mostly they are smaller than 1.3.

Table 1. Refractive index

Data	F_L*	F_Y	n_{L*}	n_Y
Hand	0.0007	0.0008	1.0544	1.0582
Palm	0.0048	0.0063	1.1489	1.1724
Wrist	0.0044	0.0052	1.1421	1.1554
Cheek	0.0039	0.0039	1.1332	1.1332
Chin	0.0035	0.0034	1.1258	1.1238
Nose	0.0000	0.0000	1.0000	1.0000
Forehead1	0.0033	0.0032	1.1219	1.1199
Forehead2	0.0049	0.0050	1.1505	1.1522
Zygoma1	0.0049	0.0048	1.1505	1.1489
Zygoma2	0.0049	0.0043	1.1505	1.1404

3.2 Rendering Results

To simulate the proposed method, we have employed improved Oren-Nayar and Phong-Layered shading method (IONPL). In principle, we have merged three methods to reflect light transport with Fresnel coefficients acquired from various regions on a real face shown in Figure 3. Fresnel has been calculated from actually measured reflectance. Most researches seem to simply use approximation of Fresnel refractive n = 1.3000. We also show numerous shading results with various Fresnel coefficients that reflect different amount of light in Figure 4.


IONPL, Forehead n=1.1505

IONPL, Forehead2 n=1.1219

IONPL, Zygoma n=1.1505

IONPL, Nose n=1.0000

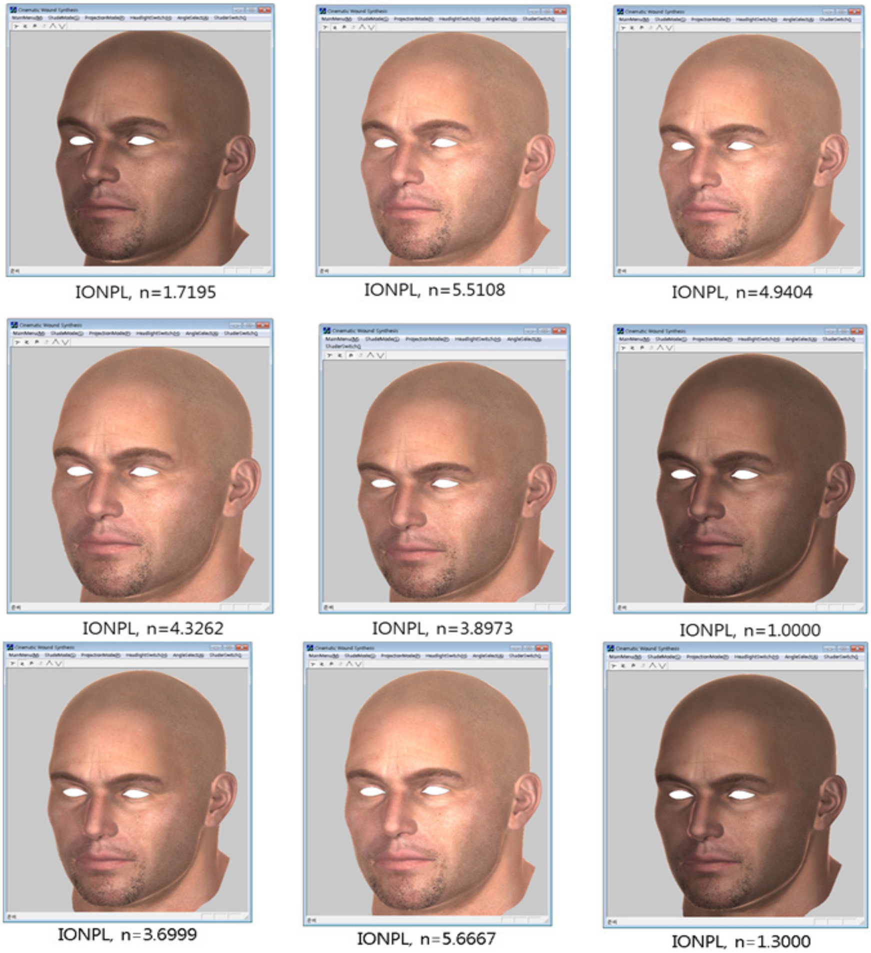
IONPL, Cheek n=1.1332

IONPL, Chin n=1.1258

Fig. 3. Facial skin rendering using IONPL with actual Fresnel coefficients measured from various locations of the real face

4 Concluding Remarks

Seamless scenes both in film and game industry are enough to draw human's interest by computing the amount of light transport through layered materials. However, Fresnel refractive indices of real materials still have limitation since approximately measured or guessed rough ones used. Hence the amount of the light should be computed by conveying thickness of a tissue on a real face as well as material components when a ray moves. Actual Fresnel refractive indices are derived from actually measured data on a real human face. The Fresnel refractive index is gained as taking into consideration the location on a face to deliver accurate amount of light.



IONPL, n=1.7195

IONPL, n=5.5108

IONPL, n=4.9404

IONPL, n=4.3262

IONPL, n=3.8973

IONPL, n=1.0000

IONPL, n=3.6999

IONPL, n=5.6667

IONPL, n=1.3000

Fig. 4. Facial skin renderings simulated using IONPL reflecting various Fresnel coefficients showing different amount of light

Acknowledgement. This research was partially funded by Korea Evaluation Institute of Industrial Technology, KEIT (Physically-based Cinematic Material Rendering Techniques optimized for Gameplay, No-10043453) and by Korea Science and Engineering Foundation (NRF) grant funded by the Korean government (No. 2012-0002889).

References

1. Oren, M., Nayar, S.K.: Generalization of Lambert's Reflectance Model. In: SIGGRAPH, pp. 239–246 (1994)
2. Jensen, H.W., Marschner, S.R., Levoy, M., Hanrahan, P.: A Practical Model for Subsurface Light Transport. In: Proceedings of SIGGRAPH, pp. 511–518 (2001)

3. Hanrahan, P., Krueger, W.: Proceeding SIGGRAPH 1993, Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, pp. 165–174 (1993)
4. Marbach, R., Heise, H.M.: Optical Diffuse Reflectance Accessory for Measurements of Skin Tissue by Near-Infrared Spectroscopy. Applied Optics, 610–621 (1995)
5. Ding, H., Lu, J.Q., Wooden, W.A., Kragel, P.J., Hu, X.: Refractive Indices of Human Skin Tissues at Eight Wavelengths and Estimated Dispersion Relations between 300 and 1600 nm. Physics in Medicine and Biology, 1479–1489 (2006)
6. Petrov, G.I., Doronin, A., Whelan, H.T., Meglinski, I., Yakovlev, V.V.: Human Tissue Color as Viewed in High Dynamic Range Optical Spectral Transmission Measurements. Biomedical Optics Express, 2154–2161 (2012)
7. Gotanda, Y.: Beyond a Simple Physically Based Blinn-Phong Model in Real-Time. SIGGRAPH 2012 Course (2012)

Emotion Recognition Technique Using Complex Biomedical Signal Analysis

Guyoun Hwang, Heejun Cho, Dongkyoo Shin, and DongIl Shin

Department of Computer Engineering Sejong University, Seoul, Korea
{hgy1999,coolheejun}@gce.sejong.ac.kr,
{shindk,dshin}@sejong.ac.kr

Abstract. Recently, there is an increasing interest in and research on human engineering and emotion engineering. As a basic research on biofeedback interface technology, the development of a system for processing and modeling complex biomedical signals is very important, and these technologies will eventually offer a pleasant life environment, so the human-centered system based on biomedical signal analysis is the keyword of the future technology. In this study, a biofeedback interface was designed to analyze biomedical signals (EEG, ECG) to recognize the user concentration and emotion state as well as effectively assessing the user intention. Compared with the existing interface technique using single biomedical signals, the proposed technology can analyze complex biomedical signals to make it easy to assess the user state and intention and enhance the utilization thereof.

Keywords: EEG, ECG, BIOFEEDBACK, INTERFACE.

1 Introduction

Research on human biomedical signals is the main area of not only medicine, biology and physics, but also of other fields, and diverse biomedical signal gauges and processing methods are being studied.

In line with the development of next-generation IT convergence technology, the user emotion state and biomedical signal-based service is increasingly becoming important, so it is an emotion engineering success strategy to determine the user's biomedical signals according to emotion. In South Korea as well, there is an increasing interest in and research on human engineering and emotion engineering [1][2].

In this era of exploding information, it is more important to know how to deliver information than how to create information, and this eventually will lead to an interface problem. In line with the increasing necessity for the development of technology that supports human-centered interfaces and system paradigm shifts, interfaces are an effective means to bridge the gap between technology and users [3].

Biomedical signal-based biofeedback interface technology refers to the technology to use artificial biomedical signals such as ECG and EEG to enable an effective interaction between computers and humans in the virtual reality and real sense field. Biomedical signal-based biofeedback interface technologies are being studied at home and abroad, but effective practical technologies have yet to be unveiled [4][5].

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 631–640, 2013.

It is very important to develop the system designed for processing and modeling the complex biomedical signals as a basic research on biofeedback interface technology, and these technologies will eventually offer a pleasant life environment, and thus the human-centered system based on biomedical signal analysis will be the keyword of the future technology [6].

This study aims to analyze biomedical signals (EEG, ECG) and design the biofeedback interface enabling the recognition of the user's concentration level and emotion state. To that end, Chapter 2 discusses studies on biomedical signals and biofeedback, Chapter 3 explains the proposed biofeedback interface, and Chapter 4 discusses the conclusion and future research.

2 Related Studies

2.1 ELECTROENCEPHALOGRAPHY (EEG)

Human brain cells create particular-shaped regular electric shocks, and these are referred to as brainwaves. In 1929, for the first time in the world, German psychiatrist Hans Berger inserted two platinum electrodes into the skin beneath the skull of a patient with head injuries, and recorded brainwaves, which were referred to as EEG. EEG was initially studied for medical purposes, and efforts are being made to use brainwaves to analyze brain functions and their relevancy to physiological phenomena occurring in the brain, and diagnostic research using this is being made. EEG is being used in a wide range of fields including medicine, military, education, and daily life [7].

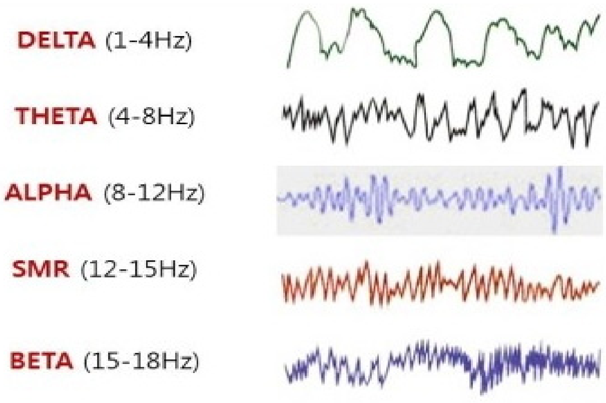


Fig. 1. Brainwave Forms

2.2 ELCTROCARDIOGRATHY (ECG)

ECG refers to the recording of electrical changes occurring in the cardiac muscle during the cardiac cycle. ECG is expressed in terms of atrial systole situation (P) and ventricular systole situation (QRST) graphs, and ECG is widely used in heart diagnoses because cardiac disorders, if any, will cause a change in waveforms. ECG does not necessarily reveal all cardiac disorders, but instead is useful in diagnosing angina, ventricular hypertrophy, myocarditis, ischemic heart disease and arrhythmia [8].

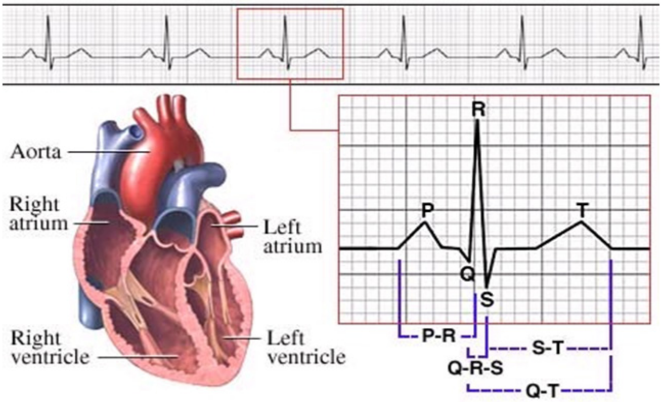


Fig. 2. ECG Waveform

2.3 Biofeedback

Biofeedback uses the biological feedback principle, and its principle and method come from biology. Biological feedback action refers to the action by which when a certain material exists profusely in the body, it will be stimulated, thereby automatically reducing its secretion, while a material, if lacking, will secrete much, thereby maintaining the body's homeostasis.

The main therapy is to attach a sensor to the patient's particular area to visually and auditorily know the body's biomedical signals (muscular tension level, brainwaves, heart rate, skin resistance, temperature, blood pressure) via a computer so as to train himself/herself to adjust to the desired state.

2.4 K-NN

k-NN is the simple and robust classifier. The classifier works by comparing a new sample (testing data) with the baseline data (training data). The classifier finds the k neighborhood in the training data and assign class which appear more frequently in the neighborhood of k. The value of k needs to be varied in order to find the match class between training and testing data. The default value of k is 1. The default neighborhood setting is Euclidean and nearest. The Euclidean distance is used to find the object similarity in the k neighborhood as shown in [9].

$$\mathrm{d}(X_i, X_j) = \sqrt{\textstyle\sum_i (X_i - |X_j)^2} \tag{1}$$

2.5 Support Vector Machine

Support Vector Machine (SVM) is based on statistical learning theory and used for learning classification and regression rules from data. Unlike other predictive models, SVM attempts to minimize the upper bound on the generalization error based on the principle of structural risk minimization (SRM), rather than minimizing the training error. This approach has been found to be superior to the empirical risk minimization (ERM) principle employed in Artificial Neural Networks. In addition, the SRM approach incorporates capacity control that prevents overfitting of the input data. The SVM has a sound orientation towards real-world applications like neuroscience [10].

The SVM technique can be used when there are few samples relative to the number of variables, which is also called the small n large p problem. In decision pattern classification, we have faced this problem because of high dimensionality of input data and features. This is one of the most important reasons why we have selected SVM over other predictive models such as k-Nearest Neighbor or Artificial Neural Network models. Moreover, SVM is computationally efficient in terms of speed and complexity. Furthermore, SVM has good generalization performance meaning that it performs very well with EEG data [11].

3 Design OG Biofeedback Interface

3.1 Biofeedback Signal Context Manager Module

Figure 3 shows a context manager structure designed to acquire, measure and store biomedical signals. The Context Manager receives contexts from the sensor device to analyze their patterns and process contexts in two steps.

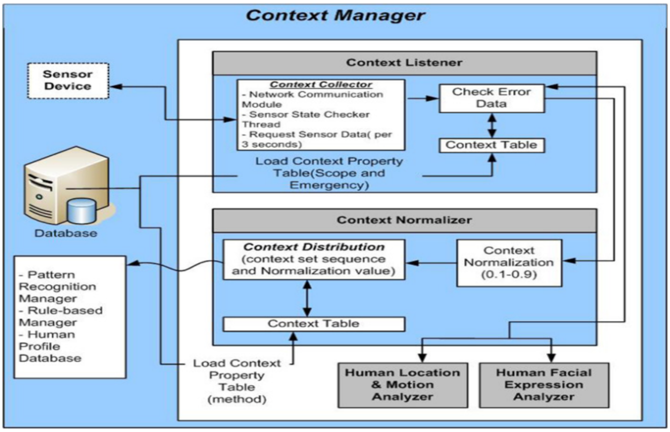

Context Manager
Context Listener
Sensor Device
Context Collector
- Network Communication Module
- Sensor State Checker Thread
- Request Sensor Data(per 3 seconds)
Check Error Data
Context Table
Load Context Property Table(Scope and Emergency)
Database
Context Normalizer
Context Distribution (context set sequence and Normalization value)
Context Normalization (0.1-0.9)
- Pattern Recognition Manager
- Rule-based Manager
- Human Profile Database
Context Table
Load Context Property Table (method)
Human Location & Motion Analyzer
Human Facial Expression Analyzer

Fig. 3. Context Manager Module

First, the Context Manager standardizes contexts ranging from 0.1 to 0.9 from the Collector, and inputs these standardized contents into the prediction module.

Next, the Context Manger stores all contexts in the database to create related rules. The contexts are stored in files to enable an easy application of components, and these created files can be applied to modules that use pattern recognition algorithms.

3.2 EEG Analysis Module

Brain waveforms consist of various sine waves and cosine waves that have different amplitudes and frequencies, and the analysis uses the FFT (Fast Fourier Transform analysis) that can divide waveforms into numerous waves according to the theory of French mathematician Fourier from the 19th century that can divide waveforms into each sine wave or consine wave. Mathematical processing course is shown in Expression 1.

$$H(fn) = \sum_{k=0}^{N-1} h_k e^{-\frac{j2\pi kn}{N}} = H_n \tag{2}$$

$$h_k = \frac{1}{N}\sum_{n=0}^{N-1} H_n e^{-2\pi kn/N} \tag{3}$$

In Expression (2), if an absolute value is taken by both sides of the Expression and is squared, and if is taken and added up, then Expression 3 is determined. The sum of the squaring of circle signals plus the sum of the squaring of signals undergoing FFT become total power. The power spectrum of total signal power value is defined as Expression (4) using the Parseval theorem that makes it all equal in the time and space or in the frequency space.

$$Total\ Power = \sum_{k=0}^{N-1} |h_k|^2 = \frac{1}{N}\sum_{n=0}^{N-1} |H_n|^2 \tag{4}$$

Specifically, the power spectrum, determined by the FFT analysis, transforms the time series signals into the frequency area, and again sorts them out according to the brainwave wavelength range and transforms them into numbers.

$$\begin{aligned}
&\mathrm{P}(f_0) = P(0) = \frac{1}{N^2}|H_0|^2 \\
&\mathrm{P}(f_n) = \frac{1}{N^2}[|H_n|^2 + |H_{N-n}|^2]n = 1, 2, \Lambda, \left(\frac{N}{2} - 1\right) \\
&\mathrm{P}\left(f_{\frac{n}{2}}\right) = p(f_c) = \frac{1}{N^2}\left|H_{N/2}\right|^2
\end{aligned} \tag{5}$$

Transform time series signals - that change according to time - into the frequency area, and then judge the signal patterns according to the changing frequency. Using this analysis, data should be classified by frequency feature, and the classified frequency features' density and distribution are determined.

As in the following Expression, the concentration indicator is calculated in terms of the ratio of SMR and M-Beta power to Theta power.

Concentration indicator =
Power Ratio of (SMR + M-Beta) / Theta

3.3 ECG Analysis Module (Feature Point Extraction)

ECG creates P, QRS and T waves once whenever the heart circulates one round, and takes similar, continual waveforms periodically. Thus, using this cycle, each waveform is distinguished and analyzed.

R WAVE

Normal ECG waveforms generally have an R-R interval of 0.6 ~ 1 second. In the case of normal ECG, the R wave summit has the highest voltage value in an ECG waveform. At the cycle of 0.8 second, the maximum value should be evaluated, and the average value of each sample should be set as the baseline.

If the heart cycle is irregular, points other than the R wave summit may be expressed as the R wave summit from time to time, and to prevent this from happening, those points not exceeding the baseline should be discarded, and the remaining points should be selected as the R wave summit.

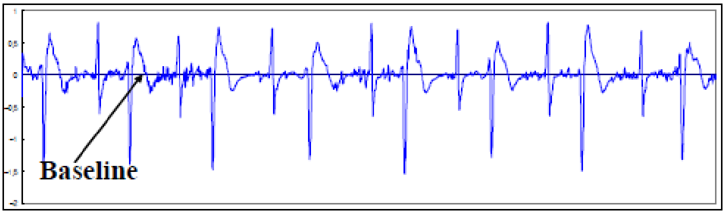


Fig. 4. Establishment of Initial Baseline

Q Wave and S Wave

Q wave and S wave are located on the left and right sides of R wave, respectively, and the summits of Q wave and S wave take the downward curve. Thus, based on the determined R wave summit, Q wave summit and S wave summit can be determined by finding out the lowest voltage points on the right and left sides of the R wave summit. The starting point is the point that meets the baseline on the left of Q wave summit, and S wave end point is the point that meets the baseline on the right side of the determined S wave summit. There can be no Q wave and S wave depending on signals, and in this case, it is assumed that there is only R wave, and thus the two points where R wave meets the baseline are determined as the QRS group's starting point and end point, respectively. If Q wave and S wave are determined, the distance between the Q wave starting point and the S wave end point can be evaluated, and this distance will be the QRS interval. Generally, if the QRS interval is over 0.12 second, it is regarded as abnormal, and is referred to as Right Bundle Branch Block (RBBB) or Left Bundle Branch Block (LBBB).

P Wave and T Wave

P wave and T wave are located on the left and right sides of the QRS group, respectively. P wave is a curve with its summit heading upwards, and the point with the largest voltage value on the left side of the QRS group starting point is the P wave summit. P wave starting point and end point are the points that meet the baseline on the left and right sides of P wave summit, respectively, because P wave summit is S-T junction's end point, which stays on the baseline. T wave is a curve with the normal waveform summit heading upwards, while LBBB takes a curve with the summit going downwards. Thus, T wave summit is determined by evaluating the maximum value and minimum value on the right side of the QRS group, and selecting the value with greater difference from the baseball of them. P wave starting point and T wave end point can be determined by finding the two points that meet the baseline on the left and right sides of the determined summit, respectively. If P wave and T wave are determined, the P-R interval can be evaluated by the distance between P wave starting point and QRS group starting point, and the Q-T interval, by the distance between QRS group starting point and T wave end point.

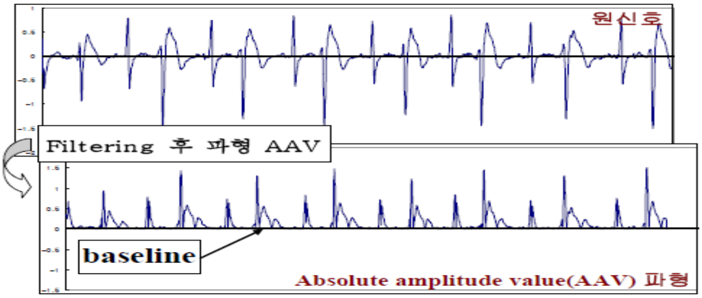


Fig. 5. Waveform AAV (Absolute Amplitude Value)

Specifically, the power spectrum, determined by the FFT analysis, transforms the time series signals into the frequency area, and again sorts them out according to the brainwave wavelength range and transforms them into numbers.

3.4 Combined EEG/ECG Biomedical Signal Analysis Module

Two types of biomedical signals should be simultaneously stored and analyzed to design the system to enable the user to repeat input feedback via the feedback module.

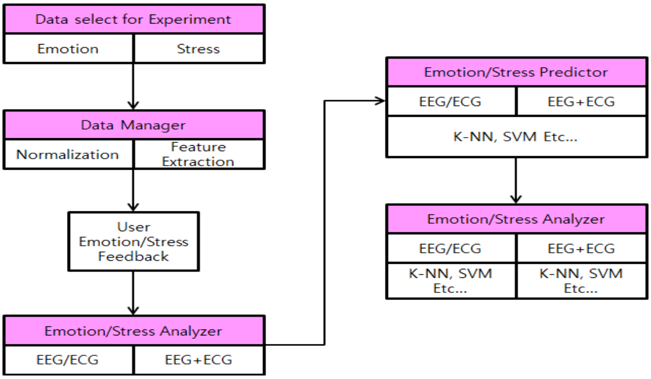


Fig. 6. Stress/Emotion Recognition System

- Data Selector is a preparation stage for emotion recognition experiment, and is used to experiment with biomedical signals of stress, nervous and relaxed situations. Data Manager normalizes biomedical signal data, evaluates particular points by normalized data, and stores evaluated particular point values in the database. The user is allowed to select his/her actual feeling and then go to the next learning stage along with the resultant feedback data.

- - Emotion Learner is responsible for learning, and learning consists of two stages. First, of particular point values stored in the database, emotion obtained from ECG data and user's feedback should be learned, and then EEG/ECG data and the user's feedback emotion should be learned together. The algorithm for use in learning is determined by selecting the representative algorithm of Supervised and Unsupervised learning algorithms.

- Emotion Predictor retrieves data except data used in the learning stage, undergoes data normalization by Data Manager as with the learning stage, determines particular point values, and stores them in the database. Of the particular point values stored in the database, only ECG data should be applied to mechanical learning algorithm, and EEG and ECG data together should be applied to SVM and K-NN.

3.5 Biofeedback Interface Class Tree Map

The main class EmotionRecongnitionFrame is located at the top, and below it are located Dialog and ControlPanel classes. Placed under the EmotionRecognitionControlPanel classed are the LoginPanel class for login, LearningPanel class for learning pages, DBManager class for managing database, Prediction Panel class for predicting emotion and stress, ImagePanel class for screens and UI, and AnalysisPanel class for analysis.

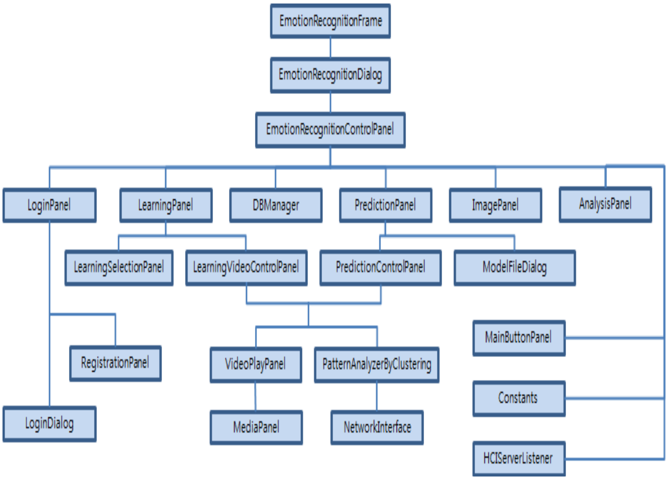


Fig. 7. Class Tree Map

3.6 Map Design of Content

Arbitrary contents were designed to use the interface. Each object requires a physical engine and a rendering unit, and the physical engine calculates physical attributes and the rendering unit implements 2-dimensional and 3-dimensional screens. There is an exclusive thread for handling networks, which exchanges data with the computer.

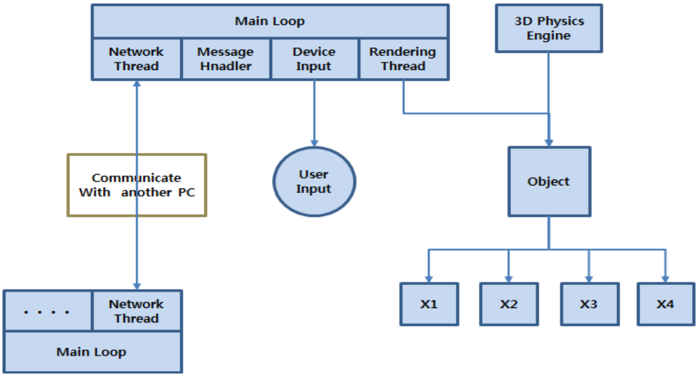


Fig. 8. Content Design Structure

4 Conclusion

This study designed a biofeedback interface system to analyze EEG and ECG, assess the user's concentration and emotion state, use the result, and effectively reflect the user's intention.

Compared with the interface technique using single biomedical signals, the proposed interface system can analyze complex biomedical signals to better assess the user's state and intention and to improve the utilization of the system.

Biofeedback interface technology can be linked with the sophisticated HCI technology to contribute to the development of the SW industry, to scientifically analyze humans' five senses and motions and structure the data, enabling a scientifical analysis of human life, quantifying the results to create the foundations for the development of diverse human-centered technologies.

To evaluate the proposed system performance, the future research should compare and analyze the implementation of other biofeedback interface techniques.

Acknowledgment. This research is supported by Seoul R&BD Program(SS110008).

References

1. Ferreira, A., Celeste, W.C., Cheein, F.A., Bastos-Filho, T.F., Sarcinelli-Filho, M., Carelli, R.: Human-machine interfaces based on EMG and EEG applied to robotic systems. Journal of NeuroEngineering and Rehabilitation (2008)
2. Thielscher, A., Pessoa, L.: Neural Correlates of Perceptual Choice and Decision Making during Fear–Disgust Discrimination. The Journal of Neuroscience, 2908–2917 (2007)
3. Adeli, H., Zhou, Z., Dadmehr, N.: Analysis of EEG records in an epileptic patient using wavelet transform. J. Neurosci. Meth. 123(1), 69–87 (2003)
4. Acir, N., Guzelis, C.: Automatic spike detection in EEG by a two stage procedure based on support vector machines. Comput. Biol. Med. 34, 561–575 (2004)
5. Khosrowabadi, R., et al.: EEG-based Emotion Recognition Using Self-Organizing Map for Boundary Detection. In: 20th International Conference on Pattern Recognition (ICPR), pp. 4242–4245 (2010)
6. Acharya, R.: Classification of cardiac abnormalities using heart rate signals. Medical & Biological Engineering & Computing 42, 288–293 (2004)
7. Ting, W., Guo-Zheng, Y., Bang-Hua, Y., Hong, S.: EEG Feature Extraction Based on Wavelet Packet Decomposition for Brain Computer Interface. Transactions of the Institute of Measurement & Control 41, 618–625 (2008)
8. Islam, M., Fraz, M.R., Zahid, Z., Arif, M.: Optimizing Common Spatial Pattern and feature extraction algorithm for Brain Computer Interface. In: Emerging Technologies (ICET), pp. 1–6 (2011)
9. Thielscher, A., Pessoa, L.: Neural Correlates of Perceptual Choice and Decision Making during Fear–Disgust Discrimination. The Journal of Neuroscience, 2908–2917 (2007)
10. Zhao, Y., Hong, W., Xu, Y., Zhang, T.: Multichannel Epileptic EEG Classification Using Quaternions and Neural Network. In: Pervasive Computing Signal Processing and Applications (PCSPA), pp. 568–571 (2010)
11. Ito, S.-I., Mitsukura, Y., Cao, J., Fukumi, M.: A design of the EEG feature detection and condition classification. In: Annual Conference, pp. 2798–2803 (2007)

RWA : Reduced Whole Ack Mechanism for Underwater Acoustic Sensor Network

Soo Young Shin and Soo Hyun Park

Graduate School of BIT, Kookmin University, Jeongneung-ro 77, Seongbuk-gu, Seoul 136-702
Korea
{sy-shin,shpark21}@kookmin.ac.kr

Abstract. In this paper, we proposed Reduced Whole Acknowledgement (RWA) method which was based on Multiple Acknowledgement (MA) [1] method and proposed by Smart Block Medium Access Control (SBMAC) [2]. The proposed method can reply the information of transmission error states of all Senders within smaller storage space. Especially, the number of transmission and the frame length was minimized to reduce transmission error. The performance of the proposed method and conventional methods such as Normal Multiple Acknowledgement (NMA) and Selective Multiple Acknowledgement (SMA) method [3] was calculated and compared with each other. The calculation results showed the best performance in case of the proposed method .

Keywords: SBMAC, RWA, SMA, NMA, MA, Underwater MAC.

1 Introduction

Recently, reliable communications in an extreme condition, such as underwater environment, have been studied consistently[4,5]. Research on modem as transmission device has been successed after several years and underwater Medium Access Control (MAC), efficient error recovering and re-transmission techniques have been drawing much interests[5,6]. Especially, the authors have been trying to reduce transmission numbers and frame sizes by enhancing conventional method such as Automatic Repeat-reQuest (ARQ)[7~11] and Block Acknowledgement (BA) [10,12]. Our research works is an another results of best performance and efficiency on the line of those trial.

A few knowledge is required to understand RWA and SMA.

1) We have to understand that it is possible to use various transmission and error recovering methods in SBMAC system, which is developed for underwater network. 2) The theme of this paper is about error detection and re-transmission mechanism. We used MA concept, which is able to reduce the number of transmissions significantly by broadcasting via kinds of feedback way. The transmitted information contains Acknowledgement (Ack) information about transmitted sensor nodes. Especially, this paper proposed RWA method by enhancing conventional NMA and SMA method. 3) The core contents of the paper is the changed frame structure, which has significantly reduced RWA frame size and improved transmission efficiency[3,4].

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 641–648, 2013.

In Chapter 2, MA method is explained. In Chapter 3, the proposed method is compared with conventional SMA method to explain mathematical model and performance evaluation. Conclusion is in Chapter 4.

2 MA

MA method is to transmit Ack to many objects simultaneously [1]. In unit cluster, Cluster Head(master, coordinator) conducts Broadcasting of Ack information of data transmitted from many receivers from Super-frame(round) within Beacon signal, which is a control frame transmitted at periodically transmitting Beacon interval. Figure 1 shows an example of Normal ARQ and Multiple Ack method in case of Multiple Access. S1~S3 is senders and sensor nodes. CH is a cluster header and a receiver. In case of MA, frame including control frames is transmitted 4 times (Beacon 1 + data 3) within super-frame. On the other hand, ARQ transmitted 7 times (Beacon 1 + data 3 + Ack 3). In Figure 1, MA and ARQ is compared with each other. Ack transmitting time and Guard time for Acks transmission were reduced. Red dotted line of Figure 1 shows the possible energy reduction sections. With MA method, the total number of frame transmission, transmission time and Guard time is reduced. The reduced Duty cycle, which was resulted from the increased possible energy reduction section, contributes to the increased Network lifetime. In addition, It removes the network complexity so it is a efficient method in case of poor condition such as underwater environment [1,3].

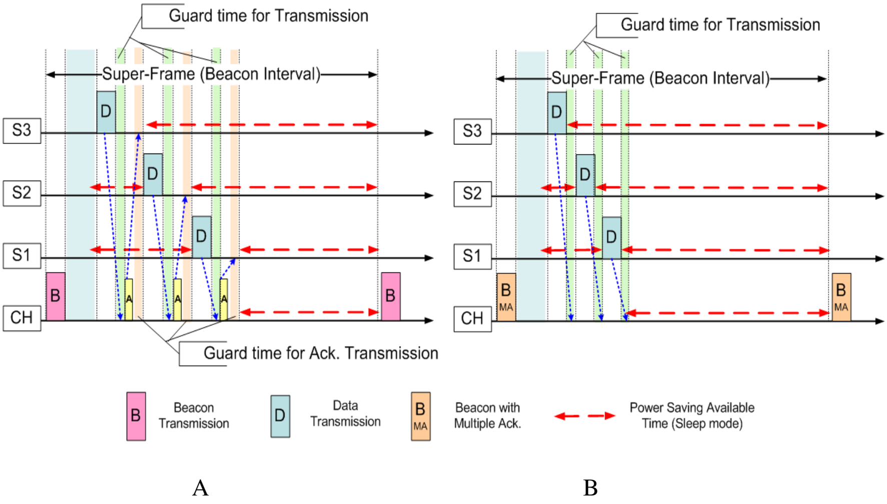

Guard time for Transmission
Super-Frame (Beacon Interval)
S3
S2
S1
CH
D
D
D
B
A
A
A
B
Guard time for Ack. Transmission
Guard time for Transmission
Super-Frame (Beacon Interval)
S3
S2
S1
CH
D
D
D
B MA
B MA
B Beacon Transmission
D Data Transmission
B MA Beacon with Multiple Ack.
Power Saving Available Time (Sleep mode)
A
B

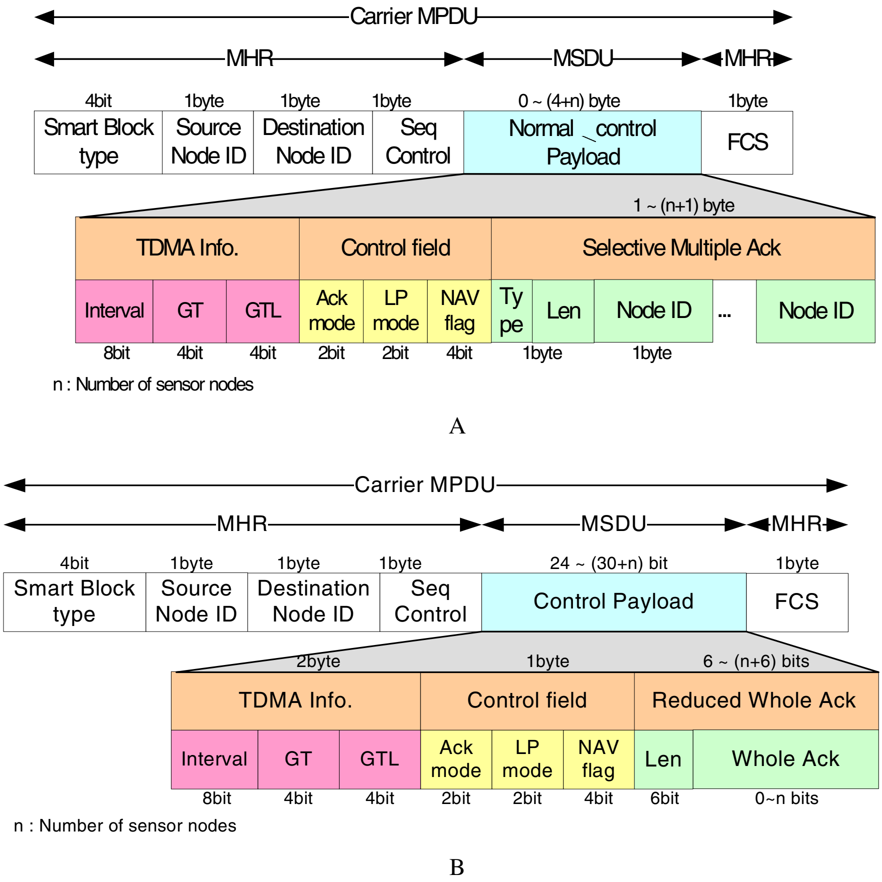
Fig. 1. An example of Super Frame; A, ARQ; B, MA

3 Proposed RWA Mechanism

3.1 SMA and RWA

The difference between RWA and SMA is whether it contains partial information or total information. SMA selectively reply Ack or Nack information, and RWA conducts node numbering and transmit 0 in case of Ack and 1 in case of Nack in a sequential way in 1 bit. Figure 2 shows frame format and their differences.

Fig. 2. Beacon frame format with Acks; A, SMA; B, RWA.



A

B

Fig. 2. Frame Format of SMA and RWA

For example, #2 and #8 node is failed to be transmitted, the following picture's information is stored respectively. If Ack/Nack is not very small such as 0~2, RWA can more reduce the frame size. With RWA, Ack type, which is needed in SMA, is not necessary. Since it has all the information of Ack/Nack, it is not necessary to classify types. Length is stored in 6 bit information and it is used for the efficient transmission. Figure 2-B shows Beacon frame format containing RWA. Green colored part is different part with SMA and Ack/Nack information is stored as 1 bit per node.

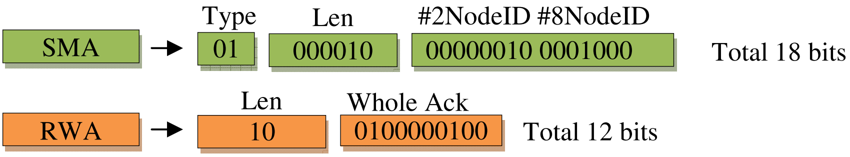


Fig. 3. Example of Ack fields

3.2 Mathematical Model

3.2.1. Analytical Formula of SMA/RWA Scheme

Variables for deduction of the formulas are listed in Table .

Table 1. Variables

Notation	Definition
C	Network Bandwidth
R	Data Rate
N	Number of Nodes
data	Data Frame with control information
int()	Function of integer
ACK	ACK Frame
L_{total}	Length of total frame
$L_{payload}$	Length of MSDU(Payload)
$L_{control}$	Length of control
L_{ack}	Length of *ACK*
$\Sigma\ L_{ack}$	Total length of *ACK* on link
Len()	Function of frame length
L_{data}	Length of *data*
BEACON	Periodic Broadcasting Frame

Channel usability can be expressed as *R/C* - Frame transmission rate over the total bandwidth. The efficiency of the channel being used means the rate of data length over the total transmitted frame. This can be expressed by $\frac{L_{payload}}{L_{total}} = \frac{L_{total} - L_{control}}{L_{total}}$. The length of the total transmitted frame is the payload length plus the control information length. The equations defined in this paper are derived from reference[3].

NMA, SMA, RWA Ack part frame format, which are defined above, can be expressed as follows.

$$ACK_{NMA} = Normal\ Ack(Type + Len + (Corresponding\ ID *\ N) \tag{3.1}$$

$$ACK_{SMA} = Selective\ Ack(Type + Len + Corresponding\ ID\ *\ int(\frac{N}{2}) \tag{3.2}$$

$$ACK_{RWA} = Reduced\ Whole\ Ack(Len + 1\ bit\ *\ N) \tag{3.3}$$

The data frame lengths of NMA, SMA and RWA are the same. This means that channel efficiency is derived from the difference of the Ack methods and control frame length. The number of transmissions of the Ack Frame and Control Frame and the total length of messages are explained. The number of data transmissions is 100. All MA does not need to transmit additional control frames. This efficiency improvement is the consequence of the minimization of information inside Ack and Data. There is no transmission number for Ack in the cases of NMA, SMA and RWA since all *ACK* information is transmitted with *BEACON*. Additionally, the sum of the *ACK* length is less than that in the other three methods. Equations 3.5-3.7 are for NMA, SMA and RWA, respectively:

$$N_{ack.xMA} = 0 \tag{3.4}$$

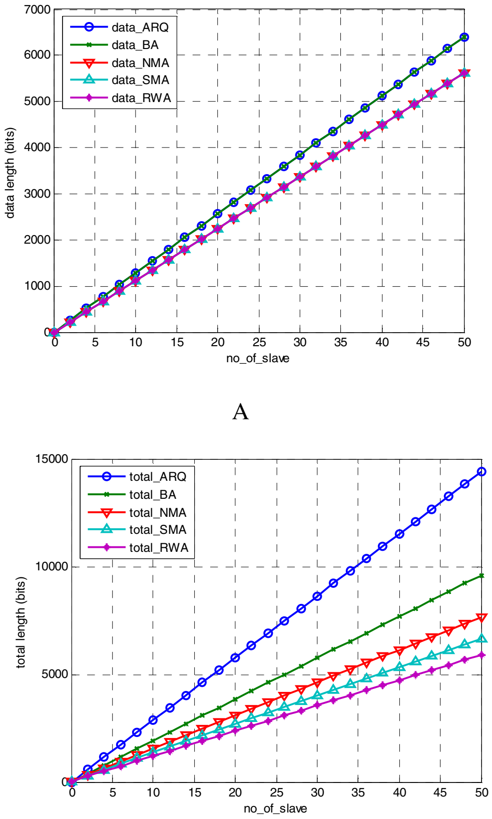
$$\sum L_{ack.NMA} = Len(BEACON(Normal.Ack.Field)) \tag{3.5}$$

$$\sum L_{ack.SMA} = Len(BEACON(Selective.Ack.Field)) \tag{3.6}$$

$$\sum L_{ack.RWA} = Len(BEACON(Reduced\_Whole.Ack.Field)) \tag{3.7}$$

3.2.2 Numerical Result

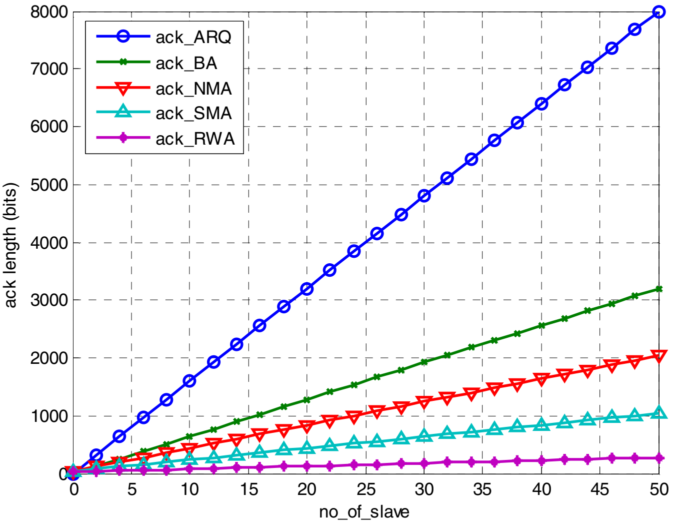
Numerical simulation results using Matlab is shown in Figure 4.



A

B

Fig. 4. Frame Length, A, data frame length; B, total frame length; C, Ack frame(Beacon) length



C

Fig. 4 (*Continued*)

Based on the comparative analysis of ARQ, BA, NMA, SMA and RWA, it is verified that the Ack transmission length is increased as the number of nodes increase. In this case, the efficiency of RWA increased significantly and the overall performance also increased.

4 Conclusion

In this paper, Reduced Whole Acknowledgement method was proposed for underwater communications. With the proposed method, the information of whether the transmission was succeeded or not of all sending Senders can be replied within minimum space. In addition, the method minimize frame length as well as the number of transmission which is necessary for minimization of transmission failure in an extreme condition such as underwater environment. Performance of the proposed RWA method and conventional methods, such as NMA, SMA, was compared. The comparison results showed that the proposed method can transmit Ack information minimized especially in case that the number of nodes increased.

References

1. Shin, S.Y., Lee, S.J., Park, S.H.: MA: Multiple Acknowledgement Mechanism for UWSN - Underwater Sensor Network. Journal of Korea Multimedia Society 12(12), 1769–1777 (2009)

2. Shin, S.Y., Park, S.H.: SBMAC: Smart Blocking MAC Mechanism for Variable UW-ASN (Underwater Acoustic Sensor Network) Environment. Sensors 10(1), 501–525 (2010)
3. Shin, S.Y., Park, S.H.: A Cost Effective Block Framing Scheme for Underwater Communication. Sensors, 11717–11735 (November 2011), doi:10.3390/s111211717
4. Etter, P.C.: Underwater Acoustic Modeling and Simulation. Spon Press (2003)
5. Alan, F.A., Dario, P., Tommaso, M.: State-of-the Art in protocol Research for Underwater Acoustic Sensor Networks. In: WUWNet 2006, pp. 7–16 (September 2006)
6. Leroy, C.C., Parthiot, F.: Depth-pressure relationships in the oceans and seas. J. Acoust. Soc. Amer. 103, 1346–1352 (1998)
7. Yao, Y.-D.: Performance of ARQ and NAK-Based ARQ on a Correlated Fading Channel. In: VTC 1999, pp. 2706–2710 (1999)
8. Lu, D.-L.: Analysis of ARQ Protocols via Signal Flow Graphs. IEEE Transactions on Communications 37, 245–251 (1989)
9. Peng, X.: Performance of hybrid ARQ techniques based on turbo codes for high-speed packet transmission. In: IEEE 7th Int. Symp. on Spread-Spectrum Tech., pp. 682–686 (2002)
10. Tinnirello, I.: Efficiency Analysis of Burst Transmissions with Block ACK in Contention-Based 802.11e WLANs. In: 2005 IEEE International Conference on Communications, ICC 2005, vol. 5, pp. 3455–3460 (2005)
11. IEEE Standard 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition, Reaff 2003 (2005)

Data Hiding Based on Palette Images Using Weak Bases of Z_2-Modules

Phan Trung Huy[1], Cheonshik Kim[2,*], Nguyen Tuan Anh[1],
Le Quang Hoa[1], and Ching-Nung Yang[3]

[1] Hanoi University of Science and Technology, Vietnam
huyfr2002@yahoo.com, huypt-fami@mail.hut.edu.vn
[2] Dept. of Digital Media Engineering, Anyang University
22, Samdeok-ro 37beon-gil, Manan-gu, Anyang-si, Gyeonggi-do, 430-714, Korea
mipsan@paran.com
[3] Department of Computer Science and Information Engineering,
National Dong Hwa University, #1, Sec. 2, Da Hsueh Rd., Hualien, Taiwan
cnyang@mail.ndhu.edu.tw

Abstract. Many steganography schemes were invented for the purpose of safe communication. Such previous schemes often show good and reasonable performance, however, few have been based on paletted images; it is not easy to invent good steganographic schemes with little evidence of data hiding. In this paper, we propose new hiding schemes (*r*, *N*, *k*) based on binary images. The proposed hiding scheme (*r*, *N*, *k*) applied to paletted images using the Optimal Parity Assignment (OPA) approach. Experimental results show that the proposed palette-scheme (3,18,9) exhibits good performance compared to that of previous schemes.

Keywords: data hiding, paletted image, Z_2-modules, r-weak base, optimal parity assignment.

1 Introduction

For secret data transmission over the internet, digital images are used for carriers to conceal a message. The most challenging problem is how to hide secret data into images with a high ratio of secret data and a low distortion of stego-images. In block-wise approaches for data hiding scheme, a given image should be divided into disjoint blocks with the same size to hide secret data in each block. A hiding scheme is good if a large amount of data can be embedded in each block while only a small number of pixels are changed. Bierbrauer and Fridrich [1] proposed a steganography scheme using the block-wise direct sum of code factorizations. A more general problem is to use the known families of good Z4-linear codes for the construction of covering codes and covering functions. Tseng and Pan [8] proposed a data hiding scheme based on binary images using a block-wise technique. This scheme's basic ideas are: (i) to use a different binary operator XOR to protect the secret key from being compromised, and

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 649–658, 2013.

(ii) to use a weight matrix to increase the data hiding rate while maintaining high quality in the host image.

In EZ Stego [2], the palette is first sorted by luminance. In the reordered palette, neighboring palette entries are typically near to each other in the color space, as well. EZ Stego embeds the message in a binary form into the LSB of randomly chosen pointers to the palette colors. For the cases of paletted images, Fridrich [3] has proposed to hide message bits into the parity bits of closest colors in the palette. Huy and Nguyen[4] proposed a data hiding scheme using a block-wise technique. The principal of the scheme is similar to [8]. Since the number of colors in paletted images is generally small, to prevent steganalysis, as shown in [6,7] if the ratio of pixels changed to total pixels in each block in a given image is not larger than about 1/5 (the smaller the better), it is possible to obtain a good hiding scheme for protecting against revelations of the existence of hidden data by steganalysis. Zhang et al.[9] proposes a novel multibit assignment steganography for palette images, in which each gregarious color that possesses close neighboring colors in the palette is exploited to represent several secret bits. For any pixel with a gregarious color, a data-hider can always find a suitable neighbor to the original color that corresponds to a prefix of secret bit-sequence and then replaces the original color with the neighboring color.

In this paper, we define a notion of *r*-weak bases of modules over the field Z_2 and propose data hiding schemes to hide secret data in palette images by using *r*-weak bases for *r* small, $r = 2, 3$. For our proposed schemes (r, N, k), one can change at most *r* pixels to hide *k* secret bits in each block of *N* pixels of paletted images.

The remainder of this paper is organized as follows. Section 2 introduces *r*-hiding schemes based on *r*-weak bases of modules over the field $\mathbf{Z}_2$ for data hiding which are extended from [4]. In section 3, we propose a way to obtain new high-quality hiding schemes for paletted images by combining the previous schemes for binary images with an Optimal Parity Assignment (OPA) approach. The experimental results are presented in section 4. Section 5 summarizes our findings and suggests future research directions.

2 Application of $\mathbf{Z}_2$-Module in Data Hiding

Recall that each (right) module *M* over the field $\mathbf{Z}_2$ is an additive abelian group *M* with zero 0 together with a scalar multiplication "." to assign each couple (d, k) in $M\times\mathbf{Z}_2$ to an element $d.k$ in *M*. Let $\mathbf{Z}_2=\{0,1\}$. The following notion will be used for the data hiding scheme:

P1) $d.\mathbf{0} = 0$; $d.\mathbf{1}=d$

P2) $c+d = c+d$ for all c,d in M {commutative property for + on M}

P3) $d.(\boldsymbol{k}+\boldsymbol{l}) = d.\boldsymbol{k} + d.\boldsymbol{l}$ for all d in M, $\boldsymbol{k},\boldsymbol{l}$ in $\mathbf{Z}_2$ {distributive law}

P4) The addition + on $\mathbf{Z}_2$ and on M are commutative and associative law

For binary images, the addition "+" in $\mathbf{Z}_2$ can be seen as the operation ⊕ (exclusive – OR) on bits, and $V_k=\mathbf{Z}_2 \times \mathbf{Z}_2\times..\times \mathbf{Z}_2$ is the *k*-fold Cartesian product of $\mathbf{Z}_2$ which can be seen as a (right) $\mathbf{Z}_2$-module, each element $x=(x_1,x_2,..,x_k)$ in V_k can be presented as a *k*-bit stream $x=x_1x_2..x_k$, with operations defined by:

D1) For any $x=x_1x_2..x_k$, $y=y_1..y_k$ in M, $\boldsymbol{k}$ in $\mathbf{Z}_2$, $x+y = z_1z_2..z_k$ where $z_i=x_i +y_i= x_i \oplus y_i$, $i=1,..,n$ (compute by bitwise XOR).

D2) $x.\boldsymbol{k}= z_1z_2..z_n$ where $z_i=\mathrm{x}_i.\boldsymbol{k}$ (= x_i AND $\boldsymbol{k}$).

Definition 2.1. Let U be a subset of a $\mathbf{Z}_2$-module M, $0 \notin U$. We call U

i) An *r-base* of M if any element in M-{0} can be presented as a linear combination (or, equivalently, a sum by XOR operation) of at most r elements in U.

ii) An *r-weak base* of M if almost elements in M-{0} can be presented as a linear combination (or a sum by XOR operation) of at most r elements in U. The remaining elements in M-{0} can be presented as a linear combination of r+1 elements in U and the number of these elements is not larger than $|U|$.

Remark 2.1 On average, with an r-weak base U, we can consider that any element in M-{0} can be presented as a linear combination of at most r elements in U.

Set C_G= Z_2 ={0,1} is the set of two colors in each binary image. We define the function Next: $\mathbf{Z}_2 \rightarrow \mathbf{Z}_2$ by

(2.1) Next(c)= 1-c, (also $c \oplus 1$), for all c in $\mathbf{Z}_2$

and changing a color c means that c is replaced by c'=Next(c).

Given a set $F=\{f_1,f_2,..,f_N\}$ of N pixels in a binary image G, an *r-weak base* U of the $\mathbf{Z}_2$-module $M = V_k = \mathbf{Z}_2 \times \mathbf{Z}_2 \times .. \times \mathbf{Z}_2$ the k-fold Cartesian product of $\mathbf{Z}_2$ as above, $N \geq |U|$, we define a *weight* function as a surjective mapping:

(2.2) h: $\{1,2,..,N\} \rightarrow U$ for which, each f_i in F, w=h(i) is called the weight of f_i.

Given a binary image G which is split into separated blocks of the same size N (pixels), with an *r-weak base* U of at most N elements of V_k the k-fold Cartesian product of $\mathbf{Z}_2$ as above, a secret set $K=\{k_i \in \mathbf{Z}_2\text{: } 1 \leq i \leq N\}$ of N key bits k_i , and a weight function h: $\{1,2,..,N\} \rightarrow U$, we can hide a k secret bit stream $b= b_1b_2..b_k$ as an element of V_k in any block F of G by changing color of at most r pixels.

2.1 Algorithm: Hiding the Secret Item *b* into *F*

The inputs to our scheme and notation are:

- F: a host bitmap, which is to be modified to embded data. (We will partition F into blocks of size m×n. For simplicity, we assume that the size of F is multiple of m×n.)
- K: a secret key shared by the sender and the receiver. It is a randomly selected bitmap of size m×n (to be stated later).
- W: a secret weight matrix shared by the sender and the receiver. It is an integer matrix of size m×n whose content satisfies some requirements (to be stated later).
- r: the number of bits to be embedded in each m×n block of F. The value of r satisfies 2^r-1≤ mn.
- b: 5-bit binary number in scheme (2,8,5), and 9-bit binary number in (3,18,9).
- $h(\bullet)$: h is a function and return a weighted value of F, i.e., h(i) = $[W]_i$.
- t: elements of $[(F \oplus K)]_i$.

- f: elements of $[F]_i$.
- k: elements of $[K]_i$.

Step 1: Compute $T = F \oplus K=\{t_1,..,t_N\}$ where $t_i=f_i\oplus k_i$, t_i,f_i,k_i as elements in $\mathbf{Z}_2$, i=1,..,N.
Step 2: Compute $s = \sum_{l\leq i \leq N} h(i).t_i$ in the $\mathbf{Z}_2$ –module M.
Step 3: Case $s = b$: keep F intact;
Case $s \neq b$:
3.1) take $x = b\oplus s$, i.e., $(x \neq 0)$,
3.2) Find l indexes from W with the smallest number, i.e., i= $([W]_i = x)$, where $l \leq r$+1 and i is index (since U is an r-weak base of V_k). The x can be presented as a linear combination (or a XOR sum) of l elements in U: $x= h(i_1)\oplus h(i_2) \oplus..\oplus h(i_l)$.
3.3) Change the colors f_i into $f_i^{'}$ = Next(f_i)=$f_i\oplus$1 for all $i = i_1,i_2,\ldots,i_l$.
Step 4: Exit

2.2 Algorithm: Extracting the Secret Item *b* from *F*

Step 1: Compute $T = F \oplus K=\{t_1,..,t_N\}$ where $t_i=f_i\oplus k_i$, t_i,f_i,k_i as elements in $\mathbf{Z}_2$, i=1,..,N.
Step 2: Compute $s = \sum_{l\leq i \leq N} h(i).t_i$ in the $\mathbf{Z}_2$ –module M.
Step 3: Return(s); {s : secret values }.

Correctness of the Method. By using commutative and associative laws of the addition + on V_k and on $\mathbf{Z}_2$, by the distributive law (P3) we deduce the following fact: P5) In the $\mathbf{Z}_2$ –module M, the sum $s = \sum_{1\leq i \leq N} h(i).t_i$ after changing color of l pixels f_i, $i=i_1,i_2,..,i_l$ is changed into the new sum $s'=s + \sum_{1\leq j \leq l} h(i_j)$.

Indeed, after changing the color of pixels f_i, $i=i_1,i_2,..,i_l$ we have l new colors $f_i'=f_i\oplus 1=f_i+1$, $i=i_1,i_2,..,i_l$ and $f'_i=f_i$ for all $i\neq i_1,i_2,..,i_l$. Hence, we have the new sum on new block F after changing colors of pixels:

$s'=\sum_{l\leq i \leq N} h(i).t_i^{'}$ where $t_i'=k_i\oplus(f_i\oplus 1)= (k_i\oplus f_i)\oplus 1=t_i \oplus 1$ for all $i = i_1,i_2,\ldots,i_l$ and $t_i'=t_i$ for all $i\neq i_1,i_2,..,i_l$. Therefore $s'=\sum_{1\leq i \leq N} h(i).t_i + \sum_{1\leq j \leq l} h(i_j)=s+\sum_{1\leq j \leq l} h(i_j)$.

Since h is surjective and U is an r-weak base, in step 3.3) in the hiding phase we always find successful l elements as claimed. Using the above fact P5, in step 3.2) we deduce $b\oplus s = x = \sum_{1\leq j \leq l} h(i_j)$ and therefore

$$b = s \oplus x = s+ \sum_{1\leq j \leq l} h(i_j).$$

Hence after step 3.3) we can obtain the new sum $s'=\sum_{1\leq i \leq N} h(i).t_i^{'} = s + \sum_{1\leq j \leq l} h(i_j)$ when taking sum on F'. This implies the following proposition:

Proposition 2.1. The secret item b embedded in the stego-block F after changing colors of pixels in F by the algorithm 2.1 can be extracted exactly by using the algorithm 2.2. With r-weak bases of N elements of modules $V_k=\mathbf{Z}_2^k$ over the field $\mathbf{Z}_2$, the schemes (r, N, k) permit us to hide in each palette image G of mN pixels by changing r pixels in each block of the stego-images.

In our proposed schemes (r, N, k), it can be change at most (on average) r pixels to hide k secret bits in each block of N pixels of paletted images. Let us remark that the scheme (3,18,9) for binary images is fitted with secret data in bytes – hence it allows the high quality of stego-images to be maintained. The scheme (3,18,9) also achieves a high standard of security when the ratio of changed pixels is not higher than 20%.

Example 1: For the following scheme (2,8,5), in V_5 whose elements are 5-bit strings, e.g.,00001,00010,..,11111 as binary number, we use the 2-weak base U = {1, 2, 3, 4, 5, 8, 16, 31}. Denote 1 = w_1, 2 = w_2, 3 = w_3,.., 31= w_8. There are only three elements 14,22,25 in V_5 that cannot be presented by a sum of 2 elements in U, but they can be presented by a sum of 3 elements in V_5 as follows:

$$25 = w_1 \oplus w_6 \oplus w_7 = w_2 \oplus w_4 \oplus w_8 = w_3 \oplus w_5 \oplus w_8;$$
$$22 = w_1 \oplus w_6 \oplus w_8 = w_2 \oplus w_4 \oplus w_7 = w_3 \oplus w_5 \oplus w_7;$$
$$14 = w_1 \oplus w_7 \oplus w_8 = w_2 \oplus w_4 \oplus w_6 = w_3 \oplus w_5 \oplus w_6;$$

Since the number of these elements satisfies 3=|{f in M | f is not expressed as combination of 2 elements in U}| < |U|=8, the probability of appearance x=b⊕s in the set {14,22,25} is smaller than the other case, i.e., the value by a sum of two elements in U. That means that the number of cases where we need to change 3 pixels in all blocks F of the image G is smaller (on average) than the number of cases where we need to change 1 pixel in all blocks F of the image G. When algorithms 2.1 and 2.2 are applied into the scheme (2,8,5), not more than two pixels are changed when concealing 5 bits in each block F of at least 8 pixels.

Example 2: The elements of V_9 is 9-bit strings which have values 0,1,..,511, i.e., V_9= {0, 1, 2, .. ,511}. A 3-weak base U (= W) has the set of weights in V_9, W = {w_1, w_2, ..,w_{18}}, as shown in the following table:

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}	w_{11}	w_{12}	w_{13}	w_{14}	w_{15}	w_{16}	w_{17}	w_{18}
11	51	119	39	30	29	1	2	4	8	16	32	128	192	256	320	384	448

There are 7 elements in V_9 which cannot be presented by 3 elements in U. They are: 67, 180, 244, 308, 372, 436, 500. Each of them can be expressed as combinations of 4 elements in U:

067 is a sum (XOR) of w_1,w_{10},w_{13},w_{14} and 10 other combinations.
180 is a sum (XOR) of w_1, w_3,w_{10},w_{14} and 4 other combinations.
244 is a sum (XOR) of w_1, w_3,w_{10},w_{13} and 4 other combinations.
308 is a sum (XOR) of w_1, w_3,w_{10},w_{16} and 4 other combinations.
372 is a sum (XOR) of w_1, w_3,w_{10},w_{15} and 4 other combinations.
436 is a sum (XOR) of w_1, w_3,w_{10},w_{18} and 4 other combinations.
500 is a sum (XOR) of w_1, w_3,w_{10},w_{17} and 4 other combinations.

Since 7=|{f in V_9 | f is not expressed as a sum of 3 elements in U }| < |U|=18, we can use the scheme (3,18,9), in each block of 18 pixels of a binary image, we can hide 9 bits by changing at most 3 pixels.

$K=$

1	0	1	1	1	0
0	0	0	1	0	1
1	1	0	1	0	1

For details, suppose we have the key binary set K given as a matrix of size 3×6 as follows: and W is a matrix of weights selected from U with the same size 3×6 as follows:

$W=$

39	30	119	2	4	448
128	11	8	51	320	29
16	32	192	384	1	256

In this case, we can rewrite the mapping h:$\{(i,j)$: $1\leq i \leq 3$, $1\leq j\leq 6\} \rightarrow W$ by taking $h(i,j)=w_{ij}$ for all (i,j), and all of the related properties remain true. Given a block F of pixels in a binary image as a binary matrix $F=(f_{ij})_{3\times 6}$ of the same size 3×6, such as

$F=$

0	0	1	0	1	1
0	1	0	1	0	0
1	0	1	0	1	1

For any 9-bit string, such as b = 155, with binary presentation b = 010011011, we can hide b in F by taking:

$T=F\oplus K=$

1	0	0	1	0	1
0	1	0	0	0	1
0	1	1	1	1	0

And then compute

$$s = \sum_{1\leq i \leq 3,\ 1\leq j\leq 6,} h(i,j).t_{i,j}= \sum_{1\leq i \leq 3,\ 1\leq j\leq 6,} w_{ij}.t_{i,j}$$

$$= 39 \oplus 2 \oplus 448 \oplus 11 \oplus 29 \oplus 32 \oplus 192 \oplus 384 \oplus 1 = 146.$$

Since $s\neq b$, we have $x=b\oplus s= 9$ as the binary string 000001001.

We find w_{23}=8 and w_{35} satisfy $x= w_{23} \oplus w_{35}$. Hence, we need to change f_{23} = 0 to 1 and f_{35} = 1 to 0, and the new stego-block F' after hiding is given by

$F'=$

0	0	1	0	1	1
0	1	(1)	1	0	0
1	0	1	0	(0)	1

In the extracting phase with F' we take

$T'=F'\oplus K=$

1	0	0	1	0	1
0	1	1	0	0	1
0	1	1	1	0	0

Then we obtain the sum

$s'=39 \oplus 2 \oplus 448 \oplus 11 \oplus 29 \oplus 32 \oplus 192 \oplus 384 \oplus 8 = 155 = b$ as claimed.

3 A Hiding Scheme for Maintaining High Quality of Palette Images

In this section, we propose a data hiding scheme based on palette images using the Optimal Parity Assignment (OPA) with algorithms 2.1, 2.2 and Example 2. Start with a given palette image G with the palette $F=\{f_1,..,f_t\}$. Denote by Val(f) the parity value of any pixel of G, where the Val function Val: $F \to \mathbf{Z}_2$ can be defined as a function in [5] together with a Next function Next: $F \to F$. This satisfies *the OPA condition* for the palette F:

(3.1) Next $(f) = f'$ is the nearest color of f in the palette F such that
Val(f') = Val(f)⊕1.

We can use this couple (*Val*, *Next*) to establish a good hiding scheme for the palette image G: In each block of 18 pixels of G we can hide 9 bits by changing, on average, at most 3 pixels.

As in the previous scheme in Example 2,

1. We take a key binary set K as a matrix of size 3×6 randomly, such as (as in Example 2):

K=

1	0	1	1	1	0
0	0	0	1	0	1
1	1	0	1	0	1

2. We choose a matrix of weights W with the same size 3×6 whose entries are filled with all elements of U, for example, W can be chosen as in Example 2:

W=

39	30	119	2	4	448
128	11	8	51	320	29
16	32	192	384	1	256

In this case, we can rewrite the mapping h:$\{(i,j)$: $1\leq i \leq 3$, $1\leq j\leq 6\} \to W$ by taking $h(i,j)=w_{ij}$ for all (i,j). All of the related properties will also remain true.

3.1 Algorithm for Hiding 9 Bits

Given a block F of G as a matrix $F=(f_{ij})_{3\times 6}$ of the same size 3×6,
For any 9-bit string, $b = b_9b_8\ldots b_1$ we can hide b in F by taking:

Step 1: Take the binary matrix F^*=Val(F)=(Val$(f_{i,j})_{3\times 6}$
(The matrix F in Example 2).

F^*=

0	0	1	0	1	1
0	1	0	1	0	0
1	0	1	0	1	1

Step 2: Compute $T=F^*\oplus K$;

Step 3: Compute $s = \sum_{1\leq i \leq 3,\ 1\leq j\leq 6,} w_{ij}.t_{i,j}$, then compare s with b:

3.1) case $s = b$: F is kept intact. (F is considered a successful stego-image with b hidden in).

3.2) case $s \neq b$:

- take $x = b \oplus s$;
- find as small as possible a number r ($1 \leq r \leq 3$) elements in W such that x is the sum of r elements w_{ij} in W,
- change r related elements f_{ij} of F into Next(f_{ij})=$1 \oplus f_{ij}$.

Step 4: Exit.

3.2 Algorithm for Extracting 9 Hidden Bits in *F*

Step 1: Take the binary matrix F^*=Val(F)=(Val($f_{i,j}$))$_{3x6}$

F^*=

0	0	1	0	1	1
0	1	0	1	0	0
1	0	1	0	1	1

Step 2: Take T=$F^* \oplus K$;

Step 3: Compute $s = \sum_{1 \leq i \leq 3,\ 1 \leq j \leq 6,} w_{ij}.t_{i,j}$

Step 4: Return(s);{s is exactly the embedded 9- bit string b}

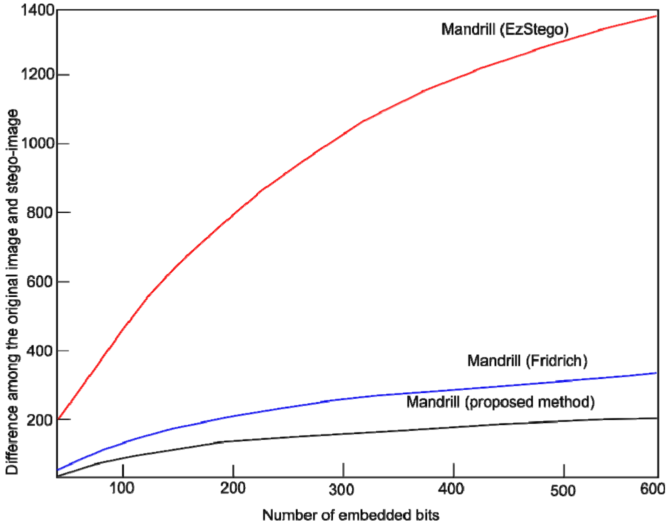


Fig. 1. Comparison between the distortion introduced by EzStego, Fridrich, and that of the proposed method

4 Experimental Results

We have experimented with the proposed data hiding scheme using the pallete-scheme (3,18,9) for palette images. The testing images are 256-color paletted images, whose sizes are 512 × 512 and 316 × 530 (original image of Fig.2). Fig.1 presents the differences between the stego-images and the original image measured as the Euclidean distance between two vectors (matrices). The number of flipping bits for embedding messages is plotted along the Y axis, and the number of embedded bits is plotted along the X axis. A method where the number of flipping pixels is lower than in any other scheme must be considered as a good scheme. The distance between the original and stego-images is more than five times smaller for the proposed scheme. If a scheme produces a small number of distortions in the stego-image, such a method stands a better chance of passing statistical tests than a method that introduces larger distortions.

Table 1. Results for Lena, Mandrill, Peppers, Murphy by the hiding scheme (3,18,9)

Image	Size		Scheme (3, 18, 9)		
	Width	Height	Hidden bytes	Rate	PSNR
Lena	512	512	16383	49.997%	40.42
Mandrill	512	512	16383	49.997%	42.12
Peppers	512	512	16383	49.997%	38.07
Murphy	316	530	10466	49.94%	43.67

In Table 1, we present some resulting stego-images and their capacity for embedded data in bytes. As a result of experiment, our proposed scheme show high embedding rate and good image quality.

Mandrill (512×512) Lena (512×512)

Pepper (512×512) Murphy (316×530)

Fig. 2. Stego-images using the scheme (3,18,9) for palette images

5 Conclusion

In order to make a good steganography scheme, we need to control the quality of the stego-image. Our proposed scheme can control visual quality of an image using a block-wise scheme. In this paper, we present methods for hiding a large number of bits in an image by block. We introduce a notion of *r*-weak bases of modules over the field Z_2 and propose data hiding schemes to hide secret data in palette images by using *r*-weak bases for *r* small, *r* =2, 3. With *r*-weak bases of *N* elements of modules $V_k=\mathbf{Z}_2^k$ over the field $\mathbf{Z}_2$, the schemes (*r*, *N*, *k*) permit us to hide in each palette image *G* of *mN* pixels, a total of *m.k* secret bits by changing, on average, at most *r* pixels in each block of the stego-images. In this scheme we use a key matrix, e.g., 180 bit key matrix, and split this to 10 sub-matrices which are used for ten consecutive blocks of 18 pixels as well. The quality can be controlled as needed using a large set of binary keys to protect from even exhaustive attacks. As our experimental images show, our proposed schemes are appropriate for a steganographic scheme. In the future, we will improve on our proposed schemes in order to apply them to a video format.

Acknowledgement. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), by the Ministry of Education, Science and Technology (20120192).

References

1. Bierbrauer, J., Fridrich, J.: Constructing Good Covering Codes for Applications in Steganography. In: Shi, Y.Q. (ed.) Transactions on Data Hiding and Multimedia Security III. LNCS, vol. 4920, pp. 1–22. Springer, Heidelberg (2008)
2. EZStego[EB/OL]: `http://www.informatik.htw-dresden.de/~fritzsch/VWA/Source/` (since December 10, 2012)
3. Fridrich, J., Du, R.: Secure Steganographic Methods for Palette Images. In: Pfitzmann, A. (ed.) IH 1999. LNCS, vol. 1768, pp. 47–60. Springer, Heidelberg (2000)
4. Phan, T.H., Nguyen, H.T.: On the Maximality of Secret Data Ratio in CPTE Schemes. In: Nguyen, N.T., Kim, C.-G., Janiak, A. (eds.) ACIIDS 2011, Part I. LNCS, vol. 6591, pp. 88–99. Springer, Heidelberg (2011)
5. Huy, P.T., Thanh, N.H., Thang, T.M., Dat, N.T.: On Fastest Optimal Parity Assignments in Palette Images. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ACIIDS 2012, Part II. LNCS, vol. 7197, pp. 234–244. Springer, Heidelberg (2012)
6. Zhang, X., Wang, S.: Vulnerability of Pixel-Value Differencing Steganography to Histogram Analysis and Modification for Enhanced Security. Pattern Recognition Letters 25, 331–339 (2004)
7. Zhang, X., Wang, S.: Analysis of Parity Assignment Steganography in Palette Images. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005, Part III. LNCS (LNAI), vol. 3683, pp. 1025–1031. Springer, Heidelberg (2005)
8. Tseng, Y.-C., Pan, H.-K.: Secure and Invisible Data Hiding in 2-Color Images. In: Proceedings of INFOCOM 2001, pp. 887–896 (2001)
9. Zhang, X., Wang, S., Zhou, Z.: Multibit Assignment Steganography in Palette Images. IEEE Signal Processing Letters 15, 553–556 (2008)

Adaptive Smart Vehicle Middleware Platform for Aspect Oriented Software Engineering

Jin-Hong Kim and Seung-Cheon Kim*

Department of Information and Communication Engineering, HanSung University,
116, Samseongyoro-16gil, Seongbuk-gu, Seoul, Korea
{jinhkm,kimsc}@hansung.ac.kr

Abstract. The most of very large system by growing the variety of applications, the relationships between the requirements and the program components are more complex. A single requirement may be implemented by a number of components and each component may include elements of several requirements. Moreover, these requirements become critical when considering conceptual model by smart applications and smart platform, which are capable of optimizing their behavior or context of execution depending on themselves. Accordingly, we propose to aspect oriented software engineering in our adaptive smart vehicle middleware platform to influence the development and the concern OSGi oriented requirement in this paper.

Keywords: Smart Platform, Smart Application, Aspect Oriented Software Engineering (AOSE), Smart Vehicle Middleware Platform, OSGi.

1 Introduction

Today, as software system become larger and more complex, the limitations of the object oriented paradigm become more obvious. Large scale systems that involve many different subsystems can be complicated to envision as objects as well as, in most this system, the relationship between the requirements and the program components are complex. A single requirement may be implemented by a number of components and each component may include elements of several requirements. In addition, changes of the heterogeneous context including stakeholder, smart application, platform, and environment are aspect oriented software engineering [1-3]. However, these environments demand plenty of computation resources for functional requests and performance requirements. To realize the idea of above computation resources and performance requirements, so called model, a various of information and communication technologies should be developed and be integrated into our environment from processors, sensors, and actuators connected via wireless high-speed networks, such as Long Term Evolution and WiFi. Accordingly, to deal with these complex dynamic environments, several solutions based on aspect-oriented software engineering have been proposed [4]. Nevertheless, it is difficult apply to the different computing environments and variant executable conceptual model. This paper

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 659–664, 2013.

presents the Adaptive Smart Vehicle Middleware Platform for aspect-oriented software engineering using OSGi. The rest of the paper is organized as follows. Section 2 is Aspect-Oriented Software Engineering based Programming (AOP), and Section 3 presents our support for System Infrastructure for SVMP. Section 4 and 5 shows Design of SVMP for AOSE and implementation for our research prototype. Finally, Section 6 is conclusion.

2 AOSE Based Programming

Aspect-Oriented Software Engineering (AOSE) is based on the Aspect Oriented Programming (AOP) such as CBD and OOP so on. AOP aims at extending languages to improve the software modularity [5-7]. This supports the construction of reconfigurable systems by enforcing separation of concerns. An aspect is defined as a set of pieces of code to execute in particular points of an application. Programming language abstractions, such as procedures and classes, also are the mechanism that this normally uses to organize and structure the core concerns of a system. However, the implementations of the core concerns in conventional programming language usually additional code to implement the cross-cutting, functional, quality of service and policy concerns[8][9]. More importantly, existing approaches fail to specify the impact of aspect weaving on the QoS of an existing application. Accordingly, our research proposes that Smart Vehicle Middleware Platform (SVMP) can improve to existing AOP approaches by providing a run-time support for the dynamic selection of aspects.

3 System Infrastructure for SVMP

We present the aspect-oriented software engineering based smart vehicle middleware platform for encapsulate knowledge about the organization of system architecture, and is a system whose correct operation depends on both the results produced by the system as following in Figure.1 and the time at which these results are produced. This can be thought of as reactive systems; that is they must react to events in their environment at the speed of that environment.

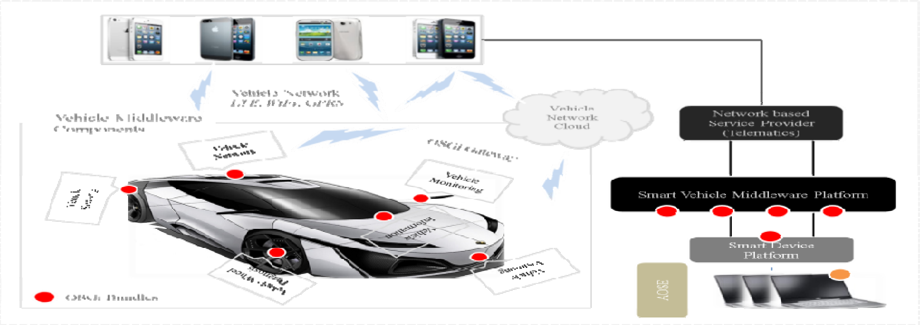


Fig. 1. Smart Vehicle System Architecture

The conceptual ideas of the proposed SVMP are as followings [10-13]: 1) The OSGi gateway is deployed as central control point for interconnecting various internal/external networks and bundles in which is Controller Area Network (CAN). CAN bus using CAN is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other within a vehicle without an outer (host) computer, and Local Interconnect Network (LIN) is a serial network protocol used for communication between components in vehicles. 2) Their service gateway can provide local area connection for the smart platform/devices in the vehicle system through MOST and wireless (WiFi, LTE, 3G/4G, and so on); provide wide area connection to the smart vehicle network through GRPS, and provide hotspot communication through IEEE 802.11x, Bluetooth, and personal wireless network service. It's also for bridging different service delivery/discovery protocols such as UPnP, Jini, SOAP and SIP. 3) The component (bundle)-based infrastructure also provides complexity of mechanism for remote configuration with services. 4) The OSGi service framework is designed to facilitate the acquisition, aggregation, interpretation and information.

Although, their components of the SVMP infrastructure like the OSGi base Aspect-oriented Software architecture in the point of view research work consists of two parts: First, Middleware system manage to smart network service, functions and gateway. Second, we are shown in Figure 2, it includes the following conceptual components; 1) ***Gateway Controller*** where the OSGi framework is embedded. 2) ***Gateway Viewer (Monitoring)*** in which is managed a set of service gateway. 3) ***Gateway Middleware server (Module)*** that provides a flexible modules with a symbolic mapping of names to components of a larger software distribution and to collections of directories and files.

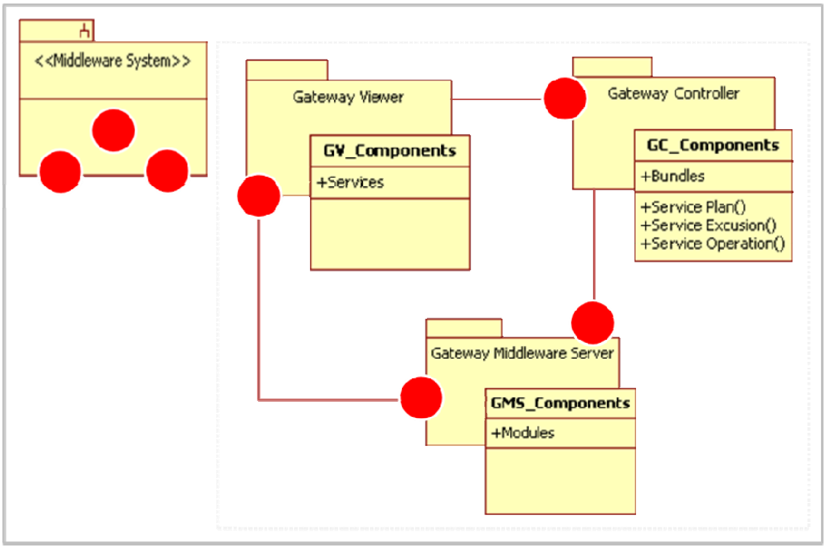
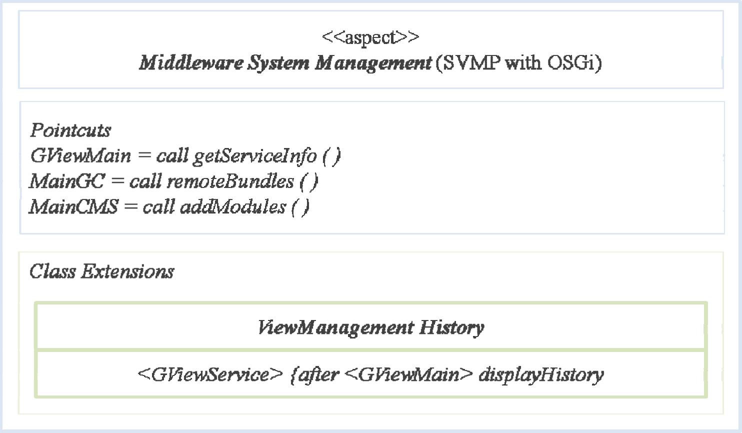


Fig. 2. Middleware System for SVMP with OSGi

4 Design of SVMP as AOSE

In this paper, we illustrate the design of SVMP as an aspect and describe how this point of view is dynamically adapted according to the approach. As we known that section 2, 3 is explained already, we have identified the core functionality and the extensions to that functionality to be denoted as conceptual aspect as following in Figure 3. The focus of the programming process should then be to write code implementing the core and extension functionality and, critically, to specify the pointcuts in the aspects actually.



<<aspect>>
Middleware System Management (SVMP with OSGi)
Pointcuts
GViewMain = call getServiceInfo ( )
MainGC = call remoteBundles ( )
MainCMS = call addModules ( )
Class Extensions
ViewManagement History
<GViewService> {after <GViewMain> displayHistory

Fig. 3. Part of a SVMP Model of an Aspect

5 Implementing SVMP as AOSE

The development of aspect is based on existing AOP technology, tangling occurs when a module includes code that implements different subsystem. So code associated with the synchronization concern is shown as red color code.

```
 Synchronized void put (SensorRecord rec) {
 If ( numberOfBundles == GViewOfService )
 Wait ( ) ; // add record to inventory remotely at buffer
Store [back] = new SensorRecord (rec.sensorId, rec.sensorVal) ;
……………
numberOfBundles = numberOfBundls + 1 ;
notify ( );
}
```

For example, say a vehicle emergency record management system, such as the vehicle accident, has a number of components concerned with managing bundle information, sensory data, vision/images, diagnosis, and treatments. These implement the core concern of the system: maintaining records from Vehicle Cloud, Network based Service Provider or vehicle middleware system respectively. The system can be configured for different types of operating system by selecting the components as several of bundles that provide the functionality needed for this situation. We assume that all components in the system use a consistent naming strategy and that all database/inventory updates are implemented by methods starting with ‘update’. There are therefore methods in the system such as:

```
updateVehicleInformation (BundleOfVehicleId, inforupdate)
updateVehicleService (BundleOfServiceId, VehicleServiceupdate)
```

The vehicle is identified by *BundelOfVehicleId* and the changes to be made are encoded in the second parameter; the details of encoding are not important for this example. Updates are made by Network based Service Provider, which are notified into the system. Alternatively, the system could be modified so that each time an update method is called, method calls are added before call to do the authentication, and then after to notify the changes made.

6 Conclusion

This research paper has introduced the Smart Vehicle Middleware Platform for the support of Aspect-Oriented Software Engineering with OSGi principles. Our adaptive smart vehicle middleware platform is good approaches form AOSE by controlling the Gateway Vehicle Component. However, this limitation is resolved by simple modeling and prototype the aspect mechanism. Nevertheless, our smart vehicle middleware platform is getting to more advantages, supporting the self-adaptive vehicle system of the aspect mechanism and the associating policies depending on semantic information variation especially.

In our advanced future research, how aspects should be specified so that tests for these aspect may be derived, as well as aspects interface occurs when more aspects use the same pointcuts specification from adaptive smart vehicle middleware platform. Furthermore, we are also a significant to the adoption of aspect-oriented software engineering in real world and interested in investing the identification and the detection of compatible aspects using the architectural smart vehicle platform supported by our modeling approach.

Acknowledgements. This research was financially supported by Hansung University.

References

1. Greenwood, P., Blair, L.: Using dynamic aspect-oriented programming to implement an autonomic system. In: Proceedings of the 2003 Dynamic Aspect Workshop (DAW04 2003), RIACS (2003)
2. Griswold, W.G., Sullivan, K., Song, Y., Shonle, M., Tewari, N., Cai, Y., Rajan, H.: Modular software design with crosscutting interfaces. IEEE Software 23(1), 51–60 (2006)

3. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C.V., Loingtier, J.M., Irwin, J.: Aspect-oriented programming. In: Akşit, M., Matsuoka, S. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997)
4. Maes, P.: Concepts and experiments in computational reflection. SIGPLAN Not. 22(12), 147–155 (1987)
5. Navarro, L.D.B., Sdholt, M., Douence, R., Menaud, J.-M.: Invasive patterns for distributed programs. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 772–789. Springer, Heidelberg (2007)
6. Grace, P., Lagaisse, B., Truyen, E., Joosen, W.: A Reflective Framework for Fine-Grained Adaptation of Aspect-Oriented Compositions. In: Pautasso, C., Tanter, É. (eds.) SC 2008. LNCS, vol. 4954, pp. 215–230. Springer, Heidelberg (2008)
7. Khan, M.U., Reichle, R., Geihs, K.: Applying Architectural Constraints in the Modelling of Self-adaptive Component-based Applications. In: ECOOP Workshop on Model Driven Software Adaptation (M-ADAPT), Berlin, Germany (July/August 2007)
8. Kiczales, G., Mezini, M.: Aspect-Oriented Programming and Modular Reasoning. In: 27th Int. Conference on Software Engineering (ICSE), St. Louis, MO, USA, pp. 49–58. ACM (May 2005)
9. Lundesgaard, S.A., Solberg, A., Oldevik, J., France, R.B., Aagedal, J.Ø., Eliassen, F.: Construction and Execution of Adaptable Applications Using an Aspect-Oriented and Model Driven Approach. In: Indulska, J., Raymond, K. (eds.) DAIS 2007. LNCS, vol. 4531, pp. 76–89. Springer, Heidelberg (2007)
10. Pessemier, N., Seinturier, L., Coupaye, T., Duchien, L.: A Model for Developing Component-Based and Aspect-Oriented Systems. In: Löwe, W., Südholt, M. (eds.) SC 2006. LNCS, vol. 4089, pp. 259–274. Springer, Heidelberg (2006)
11. Rouvoy, R., Eliassen, F., Floch, J., Hallsteinsen, S., Stav, E.: Composing Components and Services Using a Planning-Based Adaptation Middleware. In: Pautasso, C., Tanter, É. (eds.) SC 2008. LNCS, vol. 4954, pp. 52–67. Springer, Heidelberg (2008)
12. Sharma, P.K., Loyall, J.P., Heineman, G.T., Schantz, R., Shapiro, R., Duzan, G.: Component-Based Dynamic QoS Adaptations in Distributed Real-Time and Embedded Systems. In: Meersman, R. (ed.) OTM 2004, Part II. LNCS, vol. 3291, pp. 1208–1224. Springer, Heidelberg (2004)
13. Kim, J.-H., Kim, S.-C.: Design of Architectural Smart Vehicle Middleware. INFORMATION: An International Interdisciplinary Journal (III), ISSN 1343-4500. 2013

Parallel Generation of Optimal Mortgage Refinancing Threshold Rates

Nan Zhang[1,4], Dejun Xie[2,4], Eng Gee Lim[3,4], Kaiyu Wan[1,4], and Ka Lok Man[1,4]

[1] Department of Computer Science and Software Engineering
[2] Department of Mathematical Sciences
[3] Department of Electrical and Electronic Engineering
[4] Xi'an Jiaotong-Liverpool University, China
{nan.zhang,dejun.xie,enggee.lim,kaiyu.wan,ka.man}@xjtlu.edu.cn

Abstract. We present our study on the optimal mortgage refinancing problem under a stochastic interest rate environment. Through Monte Carlo simulations we try to identify the optimal time for refinancing such that the overall cost is minimised. Experimental results reveal that such a time is more likely to appear at the early stage of a mortgage contract. Through simulations we also generate time-dependent threshold rates for optimal refinancing. At a particular time, if market interest rate falls below such a threshold refinancing is most likely to be optimal. To accelerate the generation of the threshold rates we developed a multi-threaded program, which demonstrated more than three-time speedups against an efficiently-written sequential program on a quad-core Intel Corei7 2600 in all the test cases.

Keywords: Parallel computing, Monte Carlo simulation, Mortgage refinancing, Stochastic interest rate model.

1 Introduction

Residential mortgage contract is one of the most-widely used financial instruments in both the primary and secondary markets. Such a contract grants the borrower the right to pay back the loan in monthly instalments over a certain period of time. Two repayment schemes are commonly in use to calculate the monthly instalments: matching the principal repayment method and matching the repayment of principal and interest method. With the former scheme, each month the borrower pays back an equal fraction of the principal. Counting in the interest payment the amount of total monthly payment decreases as time moves towards the end of the contract. However, with the latter scheme, the borrower pays back an equal amount each month during the whole contract period. Within this fixed amount the portion of interest payment decreases as time moves towards the end of the contract period.

A mortgage contract typically grants the borrower the right to pay back all of the outstanding debt at a time before the end of the contract although the lender's preference may be to keep on receiving the monthly payments. The borrower may choose to pay back early because of a lower market interest rate. In such a case the borrower can borrow from another bank an amount of money equal to the outstanding balance owed

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 665–675, 2013.

under the first mortgage contract, and enter into a second mortgage contract with the second bank. By refinancing the outstanding debt from another mortgage contract with a lower interest rate the amount of payment on interest will be reduced.

In this work we study the optimal refinancing problem under three assumptions.

1. Only one opportunity is allowed for a borrower to pay back all the outstanding debt during the lifetime of a mortgage contract.
2. The first lender charges no refinancing fee from the borrower if he/she chooses to pay back the outstanding debt.
3. The dynamics of interest rate follows a mean-reverting stochastic model.

Under such assumptions the optimal refinancing problem is investigated from two different aspects. First, when is the optimal time for the refinancing such that the overall cost under the two contracts is minimal? Second, at a particular time, what is the threshold interest rate below which if refinancing takes place the overall cost is most likely to be minimal? Preliminary work on this topic is found in [1,9].

We investigate these questions through Monte Carlo simulations. We use Vasicek's short rate model [6,3] to describe the stochastic process of interest rate. This is a model incorporating the mean reversion property. For its applications in computing bond prices and other more complicated financial products, see [6,2,5,8] for instance. Through simulations we found that the optimal refinancing time is more likely to appear at the early stage of a contract than at the later stage. Through simulations we also generate curves of the time-dependent threshold rates for the optimal refinancing.

To speedup the generation of the threshold rates a parallel program was developed. The parallelisation was achieved through POSIX multi-threading, working on shared-memory x86 multi-core processors. In all the tests we ran on a quad-core Intel Corei7 2600 the parallel program demonstrated more than three-time speedups against an efficiently-written sequential program in generating the rates.

2 The Vasicek Short Rate Model

In Vasicek's model, the risk-neutral process for the instantaneous short rate r is

$$dr_t = k(\theta - r_t) + \sigma dW_t \tag{1}$$

where reversion rate k, long-term mean level θ and volatility σ are positive constants. This model incorporates mean reversion in that the short rate r is pulled to the long-term mean level θ at the speed k. The second part σdW in the equation is a normally distributed stochastic term superimposed upon the mean reversion. When the equation is used with Monte Carlo simulations the short rate r_t at time t is calculated from the rate r_{t-1} at time $t-1$ as

$$r_t = r_{t-1} + k(\theta - r_{t-1})\Delta t + \sigma\epsilon_{t-1}\sqrt{\Delta t} \tag{2}$$

where Δt is the time interval between times $t-1$ and t, and $\epsilon_{t-1} \sim \mathcal{N}(0,1)$ is a standardised normally distributed random number. Fig. 1 shows two paths of interest rates simulated using Equation 2. The two paths demonstrate the mean reversion property of the model. Each of the two paths starts with an initial rate significantly different from the mean level, and then is pulled back to the long-term mean. After that, the paths just oscillate around the mean level.

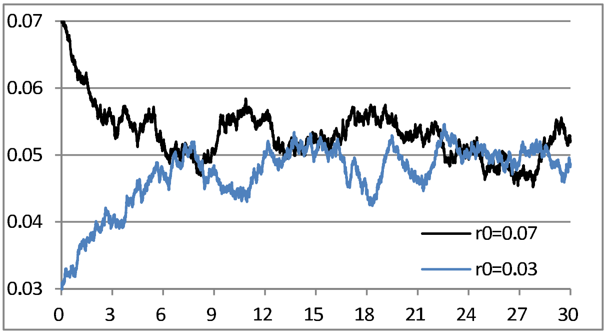


Fig. 1. The mean reversion property of Vasicek's model. Simulations over a period of 30 years with $k = 0.5$, $\theta = 0.05$ per year, $\Delta t = 1/365$ years, $\sigma = 0.003$ and $r_0 = 0.07$ per year and 0.03 per year, respectively.

3 Mortgage Repayment Settings

In this work we consider the matching the repayment of principal and interest method, in which a fixed amount of payment is made in each month during the whole period of the mortgage contract. The typical settings in such a scheme is that a principal P_0 is borrowed at time 0 with monthly interest rate r_0, and the principal is to be paid back over a period of N months. The first payment is made at month 1, and the last at month N. In each month a fixed amount of payment m is made and this monthly payment m is calculated by

$$m = \frac{P_0 r_0}{1 - (1 + r_0)^{-N}} \tag{3}$$

At the kth, $k \in \{1, 2, \ldots, N\}$, month in this scheme, after the monthly payment m has been made, the outstanding balancing P_k owed to the lender is

$$P_k = \frac{m}{r_0}(1 - (1 + r_0)^{k-N}) \tag{4}$$

where m is the monthly payment in the period between month 1 and month k, and r_0 is the applied monthly interest rate.

4 Optimal Mortgage Refinancing Time

At the kth, $k \in \{1, 2, \ldots, N\}$, month, after the kth monthly payment has been made, suppose because of a lower monthly interest rate r_k the debtor would borrow P_k from a second bank to pay back in one go the remaining P_k owed to the first lender, and enter into another mortgage contract with the second bank. The new contract is parameterised with principal P_k and repayment month $k + 1$, $k + 2$, ..., N. In this case the total payment M made by the borrower under the two contract is

$$M = km_1 + (N - k)m_2 \tag{5}$$

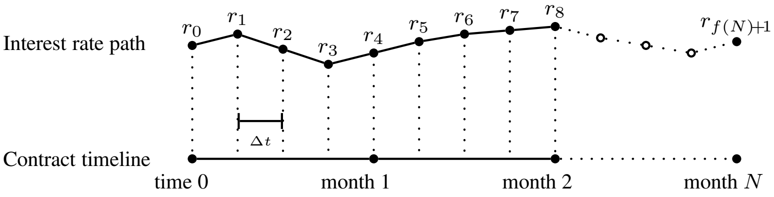
where

$$\begin{cases} m_1 = \frac{P_0 r_0}{1-(1+r_0)^{-N}} \\ m_2 = \frac{P_k r_k}{1-(1+r_k)^{k-N}} \\ P_k = \frac{m_1}{r_0}(1-(1+r_0)^{k-N}) \end{cases} \tag{6}$$

But Equation 5 does not consider the time value of money. To take the time value of money into consideration the monthly payments must be properly discounted. Suppose a path of annualised interest rates has been generated using Equation 2. We are interested to know the time 0 value of an amount m paid out at month k if we use the rates in the generated path to discount. In the generated path an annualised rate r_i applies to the period between times $i\varDelta t$ and $(i+1)\varDelta t$. Between time 0 and month k the number of time intervals is $k/12/\varDelta t$, because $\varDelta t$ is measured in years. We assume $k/12$ is an integral multiple of $\varDelta t$. Let function $f(k) = k/12/\varDelta t - 1$, and so function $f(k)$ is the index of the rate in the generated path that applies to the time interval whose end is month k. So at month k the time 0 discounting factor $F(0,k)$ is

$$F(0,k) = (1 + r_0\varDelta t)(1 + r_1\varDelta t)(1 + r_2\varDelta t)\cdots(1 + r_{f(k)}\varDelta t) \tag{7}$$

where $r_0, r_1, r_2, \ldots, r_{f(k)}$ are the annualised interest rates applying to time intervals from 0 to 1, 1 to 2, 2 to 3,. . ., and $f(k)$ to $f(k)+1$. The length of the time intervals is $\varDelta t$. So, the time 0 value of an amount m paid out at month k is $m/F(0,k)$. The calculation for the discounting factors is illustrated in Fig. 2.



$\Delta t = 1/4$ month

$f(1) = 3,\ r_{f(1)} = r_3,\ f(2) = 7,\ r_{f(2)} = r_7$

$F(0,1) = r_0 \cdot r_1 \cdot r_2 \cdot r_3,\ F(0,2) = F(0,1) \cdot r_4 \cdot r_5 \cdot r_6 \cdot r_7$

Fig. 2. Calculation for the discounting factors $F(0,k)$

If we use $F(0)^{-1}M_k$ to denote the aggregated time 0 values of the monthly payments made under the two contracts when the refinancing takes place at month k, $F(0)^{-1}M_k$ is calculated as

$$F(0)^{-1}M_k = \frac{m_1}{F(0,1)} + \cdots + \frac{m_1}{F(0,k)} + \frac{m_2}{F(0,k{+}1)} + \cdots + \frac{m_2}{F(0,N)} \tag{8}$$

where m_1 and m_2 are defined in Equation 6.

The purpose of the refinancing is to minimise $F(0)^{-1}M_k$, the aggregated discounted monthly repayments. If only one refinancing opportunity is allowed for the borrower during the period of the N months, we are interested to know which $k \in \{1, 2, \ldots, N\}$

makes the $F(0)^{-1}M_k$ minimal in the set $\{F(0)^{-1}M_1, F(0)^{-1}M_2, \ldots, F(0)^{-1}M_N\}$ in a stochastic interest rate environment.

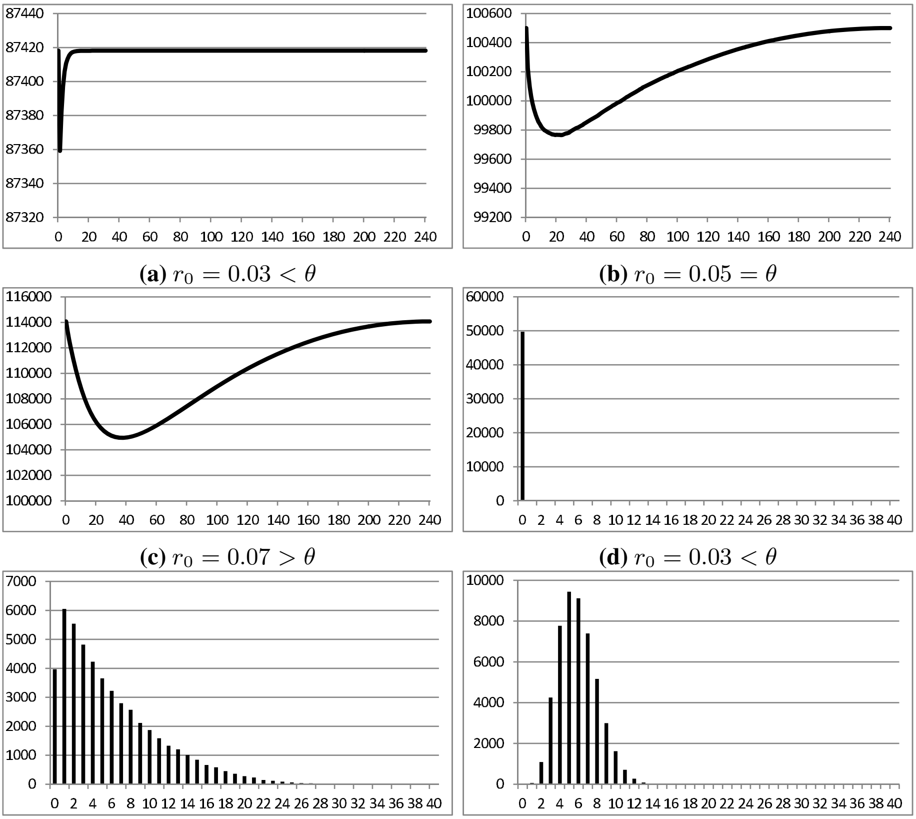
We did a series of simulation tests to locate the optimal refinancing month. The tests were made under the conditions where $r_0 = \theta$, $r_0 < \theta$ and $r_0 > \theta$. In each of the simulations we generated 50,000 paths using Equation 2. We set $\Delta t = 1/365$ and $N = 240$. So the simulation covers a period of 20 years. In each simulated path we generate $N/12/\Delta t$ annualised rates. At each monthly point along a path we assume the borrower refinances his/her outstanding debt using the new rate at that month point if the new rate is lower than the initial rate r_0, and calculate the aggregated discounted payments using Equation 8. However, if at a monthly point, the simulated rate is equal to or more than the initial rate r_0 we assume refinance will not happen. Such aggregated discounted payments are averaged over all the 50,000 generated paths. For a monthly point on a simulated interest rate where there is no refinancing we use the aggregated discounted payments without refinancing as the value in calculating the average. The plots of the averaged values are shown in Fig. 3(a)-(c). The other parameters in the simulations are set as $k = 0.5$, $\theta = 0.05$ and $\sigma = 0.003$ according to the previous study [7].

From the plots in Fig. 3(a)-(c) it can be seen that the optimal refinancing month is likely to appear in the early stage of the mortgage contract period. (Note that the aggregated discounted payments without refinancing is the value pointed out at month 0 and 240.) To confirm this observation we divide the period of 240 months into 40 sub-periods with 6 months in each, and count the number of times that the optimal refinancing month falling within each of the sub-periods. The count is collected over the 50,000 simulations. The histograms are plotted in Fig. 3(d)-(f). The pattern also shows that the optimal refinancing month is more likely to appear in the early stage of the 240-month period. Moreover, as the initial rate r_0 increases the optimal refinancing month appears relatively later.

5 Parallel Generation of the Optimal Refinancing Threshold Rates

Besides locating the optimal refinancing month we are also interested to know that for a given month k, $k \in \{1, 2, \ldots, N\}$ what is the threshold rate for optimal refinancing. At month k the threshold rate for optimal refinancing is an interest rate value that if the borrower refinances his/her outstanding debt with any rate below it the aggregated discounted payments $F(0)^{-1}M_k$ calculated by Equation 8 will most likely be minimal among all such aggregated discounted payments in the set $\{F(0)^{-1}M_k, F(0)^{-1}M_{k+1}, \ldots, F(0)^{-1}M_N\}$. We have developed a parallel program working on shared-memory x86 multi-core processors that generates such optimal threshold rates for all the months from 1 to N which start a year.

Given a month k, $k \in \{1, 2, \ldots, N\}$, and a rate r_k we use $P(k, r_k)$ to denote the probability of $F(0)^{-1}M_k$ being minimal in the set $\{F(0)^{-1}M_k, F(0)^{-1}M_{k+1}, \ldots, F(0)^{-1}M_N\}$. This probability is calculated by simulations. Suppose n simulation paths are generated, and the kth month is found to be the optimal month for refinancing with rate r_k on totally n_k paths, $P(k, r_k)$ is set to be n_k/n.



(a) $r_0 = 0.03 < \theta$

(b) $r_0 = 0.05 = \theta$

(c) $r_0 = 0.07 > \theta$

(d) $r_0 = 0.03 < \theta$

(e) $r_0 = 0.05 = \theta$

(f) $r_0 = 0.07 > \theta$

Fig. 3. Plots of averaged aggregated discounted payments and histogram of number of optimal refinancing months

The definition for optimality in this section is slightly different from what we have discussed in Section 4. In Section 4 we consider time 0 discounted values of the payments, but, here, to compute $P(k, r_k)$ we only discount to month k. We have assumed that only one refinancing opportunity is allowed within the period of the N months. So if we compare the aggregated discounted payments when refinancing happens at month k with that value when refinancing happens at a future month j, with $k, j \in \{1, 2, \ldots, N\}$ and $j > k$, the payments made in the months before k do not need to be considered. No matter what happens after month k the payments made before month k do not change. So to compare the two aggregated payments when refinancing happen at months k and j we only need to make the discounting back to month k. As in Equation 7 we define month k discounting factor $F(k, j)$ for a future month j with $j \geq k$ to be

$$F(k,j) = \begin{cases} 1 & j = k \\ (1 + r_{f(k)+1}\Delta t)(1 + r_{f(k)+2}\Delta t)\cdots(1 + r_{f(j)}\Delta t) & j > k \end{cases} \tag{9}$$

where function $f(j) = j/12/\Delta t - 1$, and $r_{f(k)+1}$ denotes the rate applying to the time interval starting from month k and ending at time $k + \Delta t$. If refinance happens at month j with $j \geq k$ the month k aggregated discounted payments $F(k)^{-1}M_j$ under the two contracts is calculated by

$$F(k)^{-1}M_j = \frac{m_1}{F(k,k)} + \cdots + \frac{m_1}{F(k,j)} + \frac{m_2}{F(0,j+1)} + \cdots + \frac{m_2}{F(0,N)} \tag{10}$$

where monthly payments m_1 and m_2 are defined as that in Equation 6.

Because the monthly payments before k can be ignored, when computing $P(k, r_k)$ interest rate paths are simulated from month k for the remaining period until month N with initial rate r_k. On any path if the aggregated discounted payments $F(k)^{-1}M_k$ is minimal among $\{F(k)^{-1}M_k, F(k)^{-1}M_{k+1}, \ldots, F(k)^{-1}M_N\}$ then the refinancing at month k with rate r_k is counted as optimal.

In our program, given a month k and a rate r_k the probability $P(k, r_k)$ is computed in parallel by multiple processors. To generate a path of interest rates from month k with time interval Δt, the number of random numbers needed is $(N - k)/12/\Delta t$, assuming this is an integral value denoted by R_k. When totally n simulation paths are launched the number of random numbers needed is nR_k. When the simulation is carried out by p processors in parallel, we make each processor generates nR_k/p random numbers and computes on n/p paths. To generate random numbers in parallel, each processor must skip a certain number from a global random number stream. For the ith processor in the system, assuming processor index starting from zero, it starts from the (inR_k/p)-th position in the stream and generates a segment consisting of nR_k/p random numbers. For a given month k and a rate r_k, each processor counts the number of simulated paths on which refinancing at month k with rate r_k is optimal, and returns this number to the calling routine. After all processors finish the counting the calling routine will sum up all the returned counts and divided the sum by the number of totally generated paths. The quotient is the value for $P(k, r_k)$.

To search for the optimal threshold rate for month k, as in [9], we set the optimal probability range to be $[90.2\%, 90.4\%]$. For a month k, if with a rate r_k, $P(k, r_k)$ falls within this range we count it as an optimal threshold rate for month k. We use an iterative bisection algorithm to determine the optimal threshold rate for each month. For month 1 the intitial searching range is set to $[0, r_0]$, and then for each successive month the lower bound for the initial range is set to the optimal threshold rate just found for the proceeding month. For a given month k, Algorithm 1 summarises the bisection searching we use in determining its optimal threshold rate.

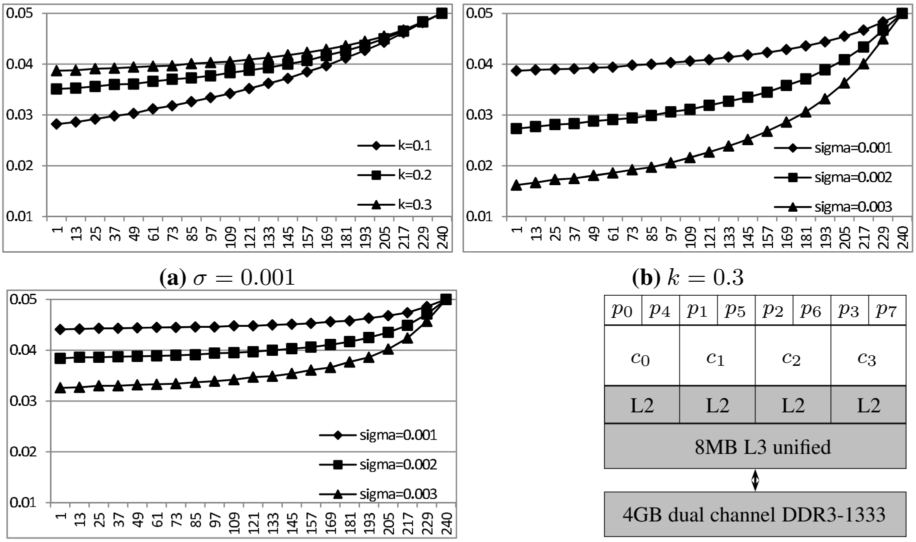
We did three groups of simulation tests under different parameter settings. In each group of the tests we output three curves. We did not compute the optimal threshold rates for all the months from 1 to N, but only for those months starting a year, that is, months 1, 13, 25, ..., 229. For month N, the optimal threshold rate r_N^O equals to r_0. This does not need to be computed. The common parameters in the tests are set as $P_0 = 100,000$, $r_0 = 0.05$, $\theta = 0.05$, $N = 240$, $n = 50,000$ and $\Delta t = 1/365$. The values for parameters k and σ are varied. The generated curves are plotted in Fig. 4(a)-(c) and the rates are reported in Table 1.

From the output results several interesting patterns can be observed. First, the optimal refinancing threshold rate is an increasing function in time. Second, with all else

Algorithm 1. Finding optimal refinancing threshold rate r_k^O for month k

```
Input: Model parameters ($P_0$, $r_0$, $\theta$, $k$ (reversion rate), $\sigma$, $N$, $\Delta t$), month $k$, number $n$ of
       paths, number $p$ of processors, initial lower bound $r_{k-1}^O$.
Output: Optimal refinancing threshold rate $r_k^O$ for month $k$
begin
    $r_k^L \leftarrow r_{k-1}^O$ // $r_k^L$ is the lower bound of the searching range.
    $r_k^H \leftarrow r_0$ // $r_k^H$ is the upper bound of the searching range.
    $r_k^O \leftarrow 0$
    $P(k, r_k^O) \leftarrow 0$
    while ($P(k, r_k^O) < 90.2\%$ or $P(k, r_k^O) > 90.4\%$) and ($r_k^H - r_k^L > 0.00001$) do
        $r_k^O \leftarrow (r_k^L + r_k^H)/2$
        with $p$ processors in parallel do
            Compute $P(k, r_k^O)$ by simulations
        if $P(k, r_k^O) < 90.2\%$ then
            $r_k^H \leftarrow r_k^O$
        else if $P(k, r_k^O) > 90.4\%$ then
            $r_k^L \leftarrow r_k^O$

    return $r_k^O$
end
```



(a) $\sigma = 0.001$

(b) $k = 0.3$

(c) $k = 0.7$

(d) quad-core Intel Core i7 2600

Fig. 4. Optimal refinancing threshold rate curves and the processor that generates them

being the same, a higher volatility σ will result in lower optimal threshold rates at the early stage of a mortgage contract (see Fig. 4(b) and Fig. 4(c)). This is understandable

Table 1. Optimal refinancing threshold rates under different values of k and σ

Month	$\sigma = 0.001$			$k = 0.3$			$k = 0.7$		
	$k = 0.1$	$k = 0.2$	$k = 0.3$	σ=0.001	σ=0.002	σ=0.003	σ=0.001	σ=0.002	σ=0.003
1	0.0282	0.0351	0.0387	0.0387	0.0273	0.0162	0.0441	0.0384	0.0326
13	0.0286	0.0353	0.0388	0.0389	0.0277	0.0167	0.0442	0.0386	0.0327
25	0.0292	0.0356	0.0391	0.039	0.0281	0.0173	0.0443	0.0386	0.033
37	0.0298	0.036	0.0392	0.0391	0.0283	0.0175	0.0443	0.0387	0.033
49	0.0303	0.0361	0.0394	0.0393	0.0288	0.0181	0.0444	0.0388	0.0332
61	0.0312	0.0366	0.0396	0.0394	0.0291	0.0186	0.0445	0.0389	0.0333
73	0.0318	0.037	0.0397	0.0398	0.0294	0.0192	0.0445	0.039	0.0334
85	0.0326	0.0373	0.0401	0.04	0.0299	0.0197	0.0446	0.0391	0.0337
97	0.0334	0.0377	0.0403	0.0403	0.0306	0.0206	0.0446	0.0394	0.0339
109	0.0342	0.0383	0.0405	0.0406	0.0311	0.0216	0.0448	0.0395	0.0342
121	0.0352	0.0388	0.0409	0.0409	0.0319	0.0227	0.0448	0.0397	0.0347
133	0.0362	0.0393	0.0413	0.0414	0.0327	0.0239	0.045	0.04	0.0349
145	0.0372	0.04	0.0418	0.0418	0.0335	0.0252	0.0451	0.0403	0.0354
157	0.0385	0.0407	0.0423	0.0423	0.0345	0.0268	0.0453	0.0406	0.0361
169	0.0397	0.0417	0.0429	0.0429	0.0358	0.0286	0.0456	0.0411	0.0366
181	0.0412	0.0426	0.0436	0.0436	0.0371	0.0306	0.0458	0.0417	0.0377
193	0.0427	0.0437	0.0445	0.0444	0.0389	0.0332	0.0463	0.0425	0.0386
205	0.0443	0.045	0.0455	0.0455	0.0409	0.0363	0.0468	0.0435	0.0402
217	0.0462	0.0465	0.0467	0.0467	0.0434	0.0401	0.0474	0.0449	0.0424
229	0.0482	0.0483	0.0483	0.0483	0.0467	0.045	0.0486	0.0471	0.045
240	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

because in a more volatile market environment the debtor needs lower refinancing rates to guarantee optimality. Third, with all else being the same the higher the reversion rate k the more flat the curve becomes (see Fig. 4(a) and compare Fig. 4(b) and Fig. 4(c)).

6 Parallel Performance

We wrote the parallel program in C/C++. The NPTL (native POSIX thread library) 2.12.1 was used for the threading. To test the performance of the parallel program we created an efficient sequential program that generates the optimal refinancing threshold rates. The sequential program uses the same method to compute the threshold rates as that has been discussed except that when $P(k, r_k)$ is computed it just uses one processor, rather than several. We ran the sequential program under the same setting and recorded the runtimes. The tests were made on a 3.4GHz (turbo boost to 3.8GHz) quad-core (8 threads with hyperthreading) Intel Core i7 2600 processor (see Fig. 4(d)) running Ubuntu Linux 10.10 (64 bit). The binary executables were compiled by the Intel compiler icpc 12.0 for Linux with -O3 and -ipo optimisations. For the random number generation we used the functions provided in Intel's math kernel library (MKL) [4].

The runtimes of the parallel (T_P) and sequential (T_S) programs in generating the curves and the calculated parallel speedups (S) are reported in Table 2. The runtimes are reported in seconds. The parallel speedup S is calculated as $S = T_S/T_P$.

Table 2. Runtimes and speedups in parallel performance tests

	$\sigma = 0.001$			$k = 0.3$			$k = 0.7$		
	$k = 0.1$	$k = 0.2$	$k = 0.3$	σ=0.001	σ=0.002	σ=0.003	σ=0.001	σ=0.002	σ=0.003
T_S	557.3	569.6	629.8	580.9	608.6	612.3	654.4	672.2	631.4
T_P	167.2	186.9	185.2	213.0	189.0	186.3	187.5	206.7	179.6
S	3.3	3.0	3.4	2.7	3.2	3.3	3.5	3.3	3.5

7 Conclusions

In this paper we have examined the optimal mortgage refinancing problem. Under the assumption that only one refinancing opportunity is allowed during the lifetime of a contract period, the optimal refinancing problem can be asked from two different but related aspects. First, if the debtor chooses to refinance his/her total outstanding debt due to a lower interest rate, when is likely to be the optimal time for the refinancing such that the overall discounted costs under the two contracts is minimal? Second, at a particular time within the lifetime of a mortgage contract what is the threshold rate below which refinancing is most likely to be optimal?

In the settings we set up for the investigation we assume the dynamics of interest rate follows Vasicek's short rate model, and the loan is paid back using the matching the repayment of principal and interest method. Through Monte Carlo simulations we found that the optimal refinancing time is more likely to appear at the early stage in the contract's lifetime than at the later stage. Through simulations we generated curves of the time-dependent optimal refinancing threshold rates. The threshold rate increases as the refinancing time moves towards the end of the contract period. The threshold rates become lower as the volatility of the interest rate model increases. As the reversion rate of the model increases the shape of the curves becomes increasingly flattened.

To accelerate the generation of the threshold rates we developed a parallel, multi-threaded program. On a quad-core Intel Corei7 2600 the parallel program demonstrated more than 3 times speedups against an efficiently-written sequential program in generating all the curves.

Acknowledgement. This work is jointly funded by the Natural Sciences Foundation BK2011350 of Jiangsu Provinceand the RDF grant 10-03-08 of Xi'an Jiaotong-Liverpool University.

References

1. Gan, S., Zheng, J., Feng, X., Xie, D.: When to Refinance Mortgage Loans in a Stochastic Interest Rate Environment. In: Proceedings of the International MultiConference of Engineers and Computer Scientists 2012, Hong Kong, vol. 2 (March 2012)
2. Hürlimann, W.: Valuation of Fixed and Variable Rate Mortgages: Binomial Tree versus Analytical Approximations. Decisions in Economics and Finance 35(2), 171–202 (2012)
3. Hull, J.C.: Options, Futures, and Other Derivatives, 8th edn., ch. 30.2. Prentice Hall (2012)
4. Intel Corporation: Intel Math Kernel Library for Linux OS: User's Guide, Document Number: 314774-018US (2011), `http://software.intel.com/en-us/articles/intel-math-kernel-library-documentation/`

5. Lo, C., Lau, C., Hui, C.: Valuation of Fixed Rate Mortgages by Moving Boundary Approach. In: Proceedings of the World Congress on Engineering 2009, London, UK, vol. 2 (July 2009)
6. Vasicek, O.A.: An Equilibrium Characterization of the Term Structure. Journal of Financial Economics 5(2), 177–188 (1977)
7. Xie, D.: Parametric Estimation for Treasury Bills. International Research Journal of Finance and Economics 17 (2008)
8. Xie, D., Chen, X., Chadam, J.: Optimal Payment of Mortgages. European Journal of Applied Mathematics 18(3), 363–388 (2007)
9. Zheng, J., Gan, S., Feng, X., Xie, D.: Optimal Mortgage Refinancing Based on Monte Carlo Simulation. IAENG International Journal of Applied Mathematics 42(2) (May 2012)

Pricing American Options on Dividend-Paying Stocks and Estimating the Greek Letters Using Leisen-Reimer Binomial Trees

Nan Zhang[1,3], Kaiyu Wan[1,3], Eng Gee Lim[2,3], and Ka Lok Man[1,3]

[1] Department of Computer Science and Software Engineering
[2] Department of Electrical and Electronic Engineering
[3] Xi'an Jiaotong-Liverpool University, China
{nan.zhang,kaiyu.wan,enggee.lim,ka.man}@xjtlu.edu.cn

Abstract. We present out work on computing the prices of American call and put options and the values of their Greek letters. The underlying stocks of the options are assumed to pay out cash dividends. For calculating option prices and their Greek letters we use the Leisen-Reimer binomial method. Through experiments we demonstrate that it converges both faster and more smoothly than the Cox-Ross-Rubinstein binomial method. We also present plots for the Greek letters calculated from American call and put options on non-dividend paying stocks. The calculation of the Greek letters with the Leisen-Reimer binomial method is explained.

Keywords: American options, Option pricing, Dividend-paying stocks, Greek letters estimation, Leisen-Reimer binomial tree.

1 Introduction

An American call (put) option is a financial contract that gives the option holder the right but not the obligation to buy (sell) a unit of the underlying stock at a fixed strike price anytime before or at a future expiration time. In order to get this right of buying (selling) at the fixed strike price a certain amount of money must be paid, which is the option's price.

The fair price of an option contract is jointly determined by several factors, such as price of the underlying stock, the volatility of the stock price, risk-free interest rate, the fixed strike price, time to expiration and dividends paid out within the lifetime of the option. To measure the change rates of the option's price with respect to some of these factors, people often calculate five Greek letters: delta, gamma, theta, vega and rho. Because of American options' early exercise feature their prices and the corresponding Greek letters cannot be calculated by the Black-Scholes-Merton (BSM) formulae [1,5]. Their values can only be evaluated by numerical methods.

The binomial tree method is an often-used numerical procedure for evaluating the prices of American options. In this work we use the Leisen-Reimer (LR) binomial tree [4] to compute the prices of American options on dividend-paying stocks and the values of the Greek letters. Compared to the commonly-used Cox-Ross-Rubinstein (CRR) binomial tree [2], the LR binomial tree converges much faster and more smoothly. In what

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 676–685, 2013.

follows, we will discuss the option pricing on dividend-paying stocks, the estimation of the Greek letters and their properties.

2 The LR Binomial Tree v.s. the CRR Binomial Tree

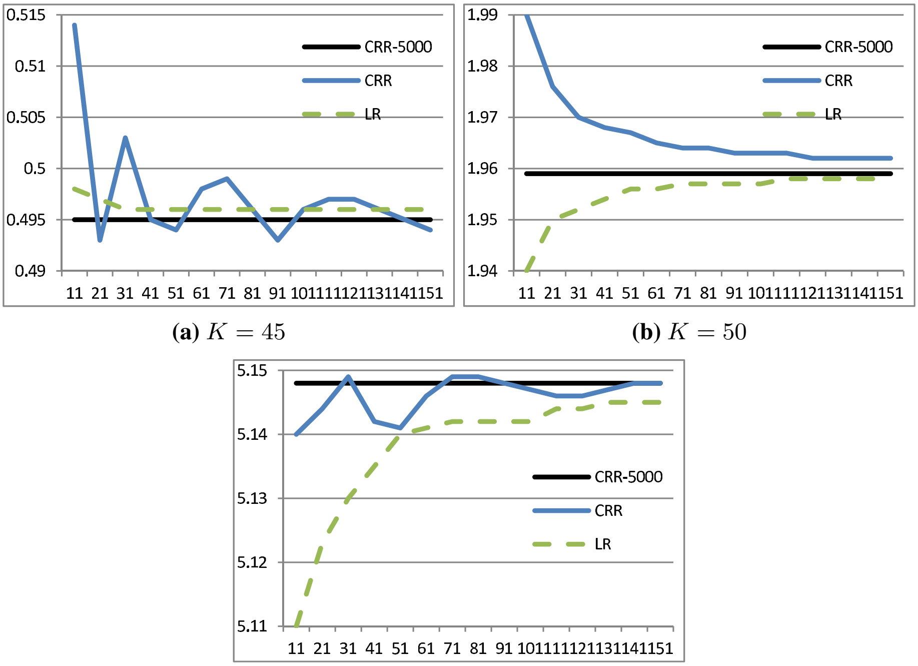
We use S_0 to denote the underlying stock's current price, K the fixed strike price, σ the annualised volatility of the stock price, r the annual continuously compound interest rate, T the option's expiration time and N the number of time steps modelled in the binomial tree. In the CRR binomial tree [2], the risk-neutral up-move probability p, the up-move factor u and the down-move factor d are set as:

$$\begin{aligned}
u &= \exp(\sigma\sqrt{T/N}) \\
d &= 1/u = \exp(-\sigma\sqrt{T/N}) \\
p &= \frac{\exp(rT/N) - d}{u - d}
\end{aligned}$$

But the parameters p, u and d in the LR binomial tree [4] are set through the following equations. The number N of time steps used in a LR binomial tree must be an odd number.

$$\begin{aligned}
d_1 &= \frac{\ln(S_0/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}} \\
d_2 &= \frac{\ln(S_0/K) + (r - \sigma^2/2)T}{\sigma\sqrt{T}} \\
h^{-1}(z) &= \frac{1}{2} + \frac{\text{sgn}(z)}{2}\sqrt{1 - \exp\left[-\left(\frac{z}{N+\frac{1}{3}}\right)^2\left(N + \frac{1}{6}\right)\right]} \\
\text{sgn}(z) &= \begin{cases} 1 & z > 0 \\ 0 & z = 0 \\ -1 & z < 0 \end{cases} \\
p\prime &= h^{-1}(d_1) \\
p &= h^{-1}(d_2) \\
u &= \exp(rT/N)\frac{p\prime}{p} \\
d &= \frac{\exp(rT/N) - pu}{(1-p)}
\end{aligned}$$

The LR binomial tree converges much faster and more smoothly than the CRR binomial tree. The plots in Fig. 1 demonstrate a comparison between the two methods on pricing in-the-money, at-the-money and out-of-the-money American put options. The plots show the change of the computed option prices as the increment of the time step. The



(a) $K = 45$

(b) $K = 50$

(c) $K = 55$

Fig. 1. Convergence comparison between the LR binomial method and the CRR binomial method on American put options. The parameters are set as: $S_0 = 50$, $K = 50$, $r = 0.1$, $\sigma = 0.2$ and $T = 0.5$. No dividend is paid out by the underlying stock.

reference value is the option's price calculated using the CRR binomial method with 5,000 time steps. From the plots it can be seen that the behaviour of the LR binomial tree is much more consistent than that of the CRR binomial tree.

3 Pricing American Options on Dividend-Paying Stocks

To price American options on dividend-paying stocks we follow the method discussed by John Hull [3]. We present the method using the example shown in Fig. 2. The example illustrates the pricing of an American put option using the LR binomial tree method. The parameters are set as: $S_0 = 50$, $K = 50$, $r = 0.1$, $\sigma = 0.2$, $T = 0.5$ and $N = 5$. During the lifetime of the option the underlying stock pays out a 15-dollar dividend at three month's time ($t = 0.25$). Although in our discussion we assume there is only one dividend to be paid out by the underlying stock, the method can be easily generalised to handle multiple cash dividends.

The method separates S_0 into a certain component and an uncertain component. The certain component is the aggregated time 0 value of the cash dividends that are going to be paid out during the lifetime of the option. The uncertain component is S_0 less the certain component. It then uses this uncertain component as root and builds a binomial tree with the up-move and down-move factors u and d. In our example, the value of

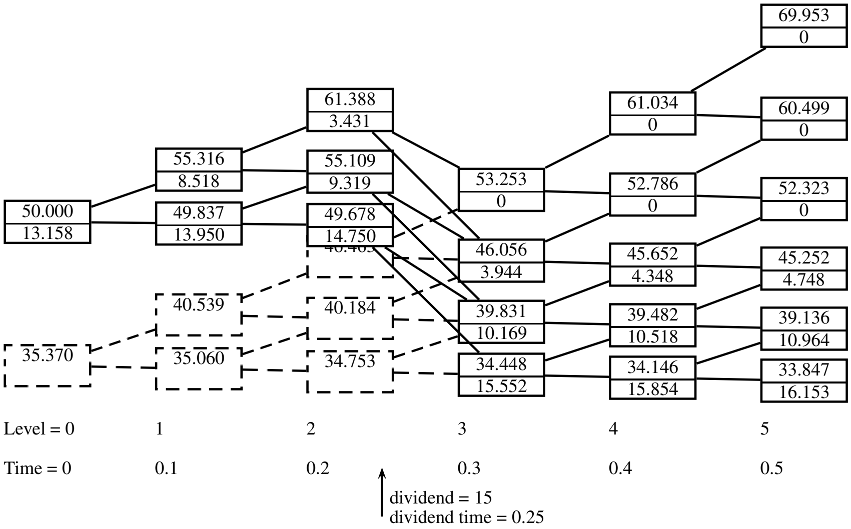


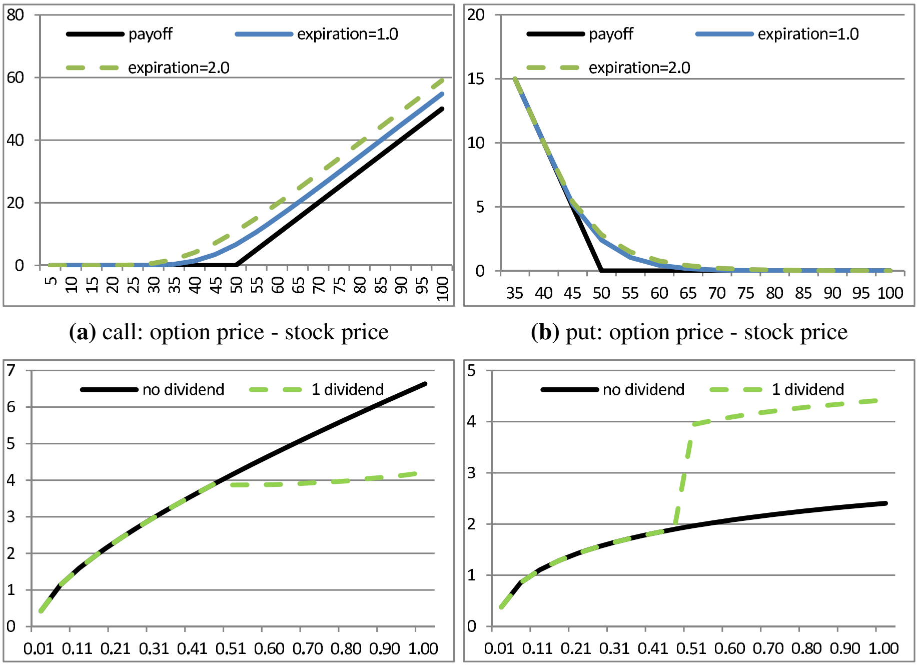
Fig. 2. Pricing an American put option on a dividend-paying stock using the LR binomial tree method. The parameters are set as: $S_0 = 50$, $K = 50$, $r = 0.1$, $\sigma = 0.2$, $T = 0.5$ and $N = 5$. A dividend of 15 dollars is paid out at time $t = 0.25$. Stock prices are shown in the upper parts of the boxes, and option prices in the lower parts.

this uncertain component is 35.370. To incorporate the cash dividends, it then adds the discounted value of each dividend onto the stock prices whose nodes proceed the time of the dividend. In the example, the dividend is paid out at time $t = 0.25$, and so the discounted values of the dividend are added upon the stock prices represented by the nodes at levels 2, 1 and 0. After the prices are updated, the backward inductive binomial pricing is performed in a usual way. Note that the numbers in Fig. 2 are calculated using the LR binomial method. If the CRR binomial method were used the numbers would be different, but the option price and the stock price at the root node would not differ much.

Fig. 3(a) and Fig. 3(b) show the change of option's price as that of the underlying stock's price, where the underlying stock pays no dividend. The option's prices are compared with their payoff functions. This payoff function is $P = \max(I(S - K), 0)$, where S is stock price, K is the strike price, $I = 1$ for a call option and $I = -1$ for a put option. From the plots it can be seen that options with longer expirations have higher values. Fig. 3(c) and Fig. 3(d) show the change of option's price as the change of time to expiration. In the dividend-paying cases, a dividend of 5 dollars is paid out at time $t = 0.5$. This has effect on options that expire in more than half year's time. For this reason, in the plots at $t = 0.5$ there are remarkable jumps in the prices of the options on the dividend-paying stocks.

4 Computing Greek Letter Delta

The delta (Δ) of an option is defined as the rate of change of the option price with respect to the price of underlying stock. For an American option its delta must be



(a) call: option price - stock price

(b) put: option price - stock price

(c) call: option price - time to expiration

(d) put: option price - time to expiration

Fig. 3. Patterns for Prices of American call and put options. The common parameters are set as: $K = 50$, $r = 0.1$, $\sigma = 0.2$, and $N = 5$. In figures (c) and (d), $S_0 = 50$, dividend = 5 dollars and dividend time $t = 0.5$.

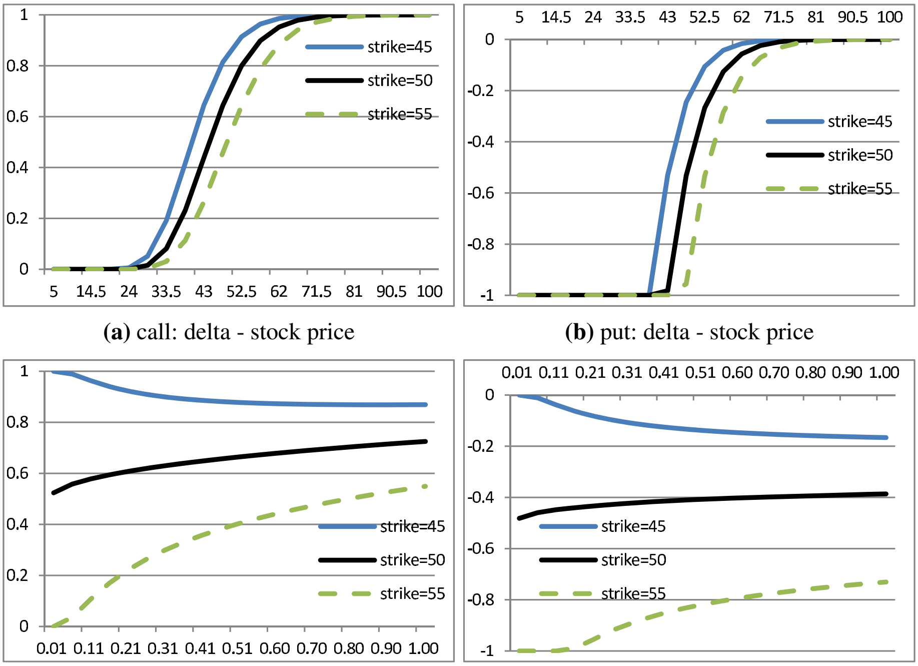
computed using numerical methods. We explain how delta is calculated in the binomial method using the example in Fig. 2. As the figure shows, in level 1 there are two nodes. We use $S(1,0)$ to denote the stock price represented by the upper node, which is 55.316, and $S(1,1)$ to denote that represented by the lower node. We also use $P(1,0)$ to denote the option's price represented by the upper node and $P(1,1)$ to denote that represented by the lower node. Delta is calculated as:

$$\varDelta = \frac{P(1,0) - P(1,1)}{S(1,0) - S(1,1)}$$

Fig. 4 plots deltas for American call and put options on non-dividend paying stocks. The parameters are set as: $S_0 = 50$, $r = 0.1$, $\sigma = 0.2$, and $N = 121$. Expiration for the options in Fig. 4(a) and Fig. 4(b) is set to $T = 1$.

5 Computing Greek Letter Gamma

The gamma ($\varGamma$) of an option is the rate of change of the option's delta with respect to the price of the underlying stock. It is the second partial derivative of the option with respect to stock price. With the help of Fig. 2 we explain how gamma is calculated in



(a) call: delta - stock price

(b) put: delta - stock price

(c) call: delta - time to expiration

(d) put: delta - time to expiration

Fig. 4. Delta plots for American call and put options on non-dividend paying stocks

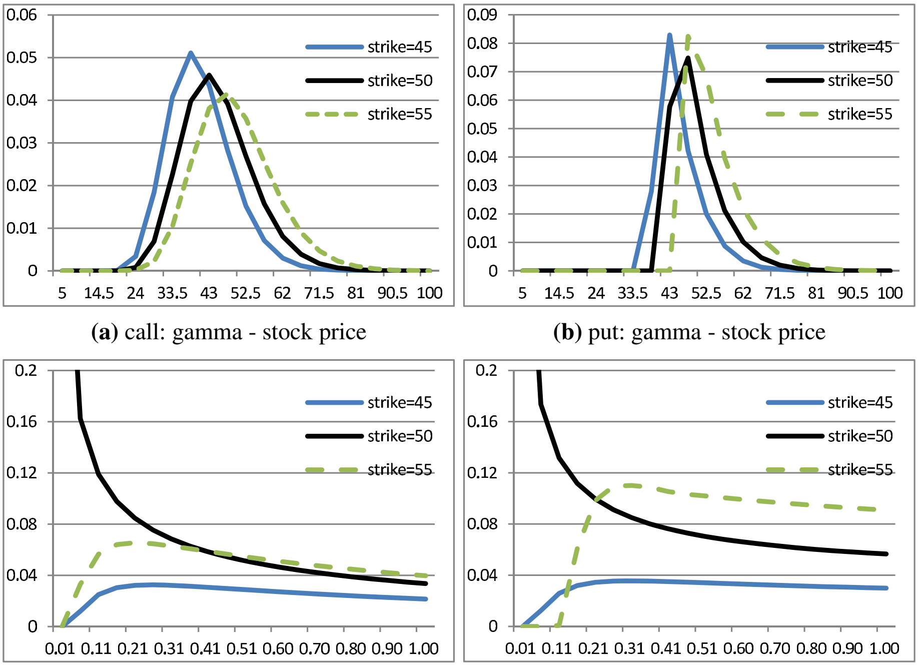
the binomial method. Like the notations we used in explaining the computation of delta, we use $P(2,0)$, $P(2,1)$, $P(2,2)$ to denote the option prices represented by the nodes in level 2 from top to bottom. Also, we use $S(2,0)$, $S(2,1)$, $S(2,2)$ to denote the stock prices at level 2 from top to bottom. Then gamma is calculated as:

$$\varGamma = \frac{\frac{(P(2,0)-P(2,1))}{(S(2,0)-S(2,1))} - \frac{(P(2,1)-P(2,2))}{(S(2,1)-S(2,2))}}{(S(2,0)-S(2,2))/2}$$

Some plots of gamma for American call and put options on non-dividend paying stocks are shown in Fig. 5. The parameters are set to the same values as in the case of delta.

6 Computing Greek Letter Theta

Theta ($\varTheta$) is the rate of change of the option's price with respect to the passage of time will all else remaining the same. On a CRR binomial tree theta can be calculated similarly to that of delta and gamma, because the stock price represented by the middle node at level 2 equals S_0. But this is not true on a LR binomial tree. See Fig. 2 for an example. To compute theta on a LR binomial tree we make a small change in T, construct a new tree and obtain a new option price. We denote this new option price by $P(S_0, K, r, \sigma, T - \varDelta t, N)$, and we denote the original option price by



(a) call: gamma - stock price

(b) put: gamma - stock price

(c) call: gamma - time to expiration

(d) put: gamma - time to expiration

Fig. 5. Gamma plots for American call and put options on non-dividend paying stocks

$P(S_0, K, r, \sigma, T, N)$. Note that these are option prices on non-dividend paying stocks. With these notations the option's theta measured as per calendar day is:

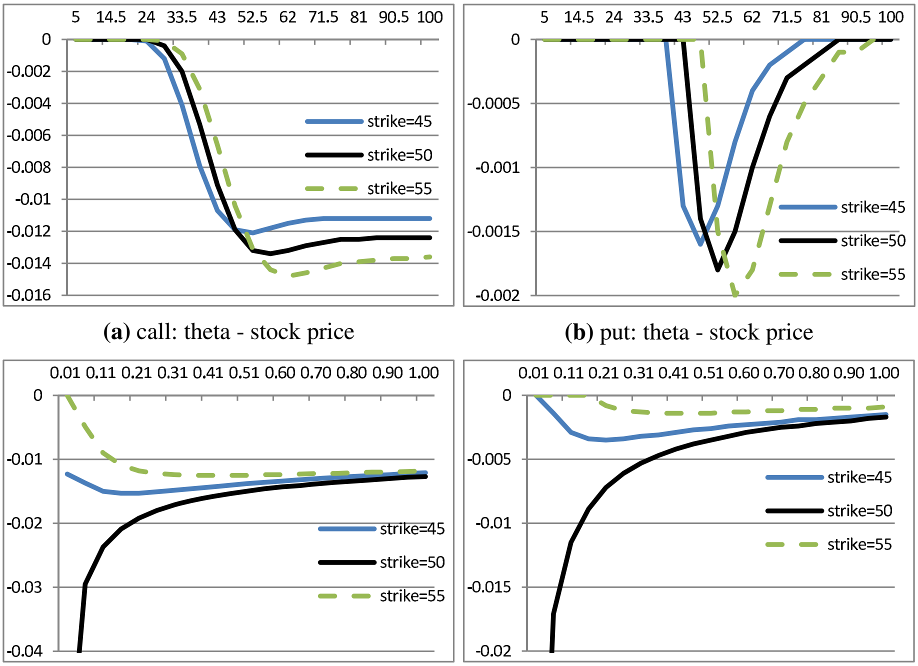
$$\varTheta = \frac{P(S_0, K, r, \sigma, T - \varDelta t, N) - P(S_0, K, r, \sigma, T, N)}{365\varDelta t}$$

Some theta plots calculated from American call and put options on non-dividend paying stocks are shown in Fig. 6. The parameters are set to the same values as before.

7 Computing Greek Letter Vega

Vega (ν) is the rate of change of option's price with respect to the volatility of the underlying stock. To compute vega, again, we make a small change in the volatility, construct a new tree and obtain a new price for the option with all else remaining the same. We denote this new price by $P(S_0, K, r, \sigma + \varDelta\sigma, T, N)$, and the old price by $P(S_0, K, r, \sigma, T, N)$. Vega of the option measured in per 1% change in the volatility of the underlying stock is:

$$\nu = \frac{P(S_0, K, r, \sigma + \varDelta\sigma, T - P(S_0, K, r, \sigma, T, N)}{100\varDelta\sigma}$$



(a) call: theta - stock price

(b) put: theta - stock price

(c) call: theta - time to expiration

(d) put: theta - time to expiration

Fig. 6. Theta plots for American call and put options on non-dividend paying stocks

Some vega plots calculated from American call and put options on non-dividend paying stocks are shown in Fig. 7. The parameters are set to the same values as before.

8 Computing Greek Letter Rho

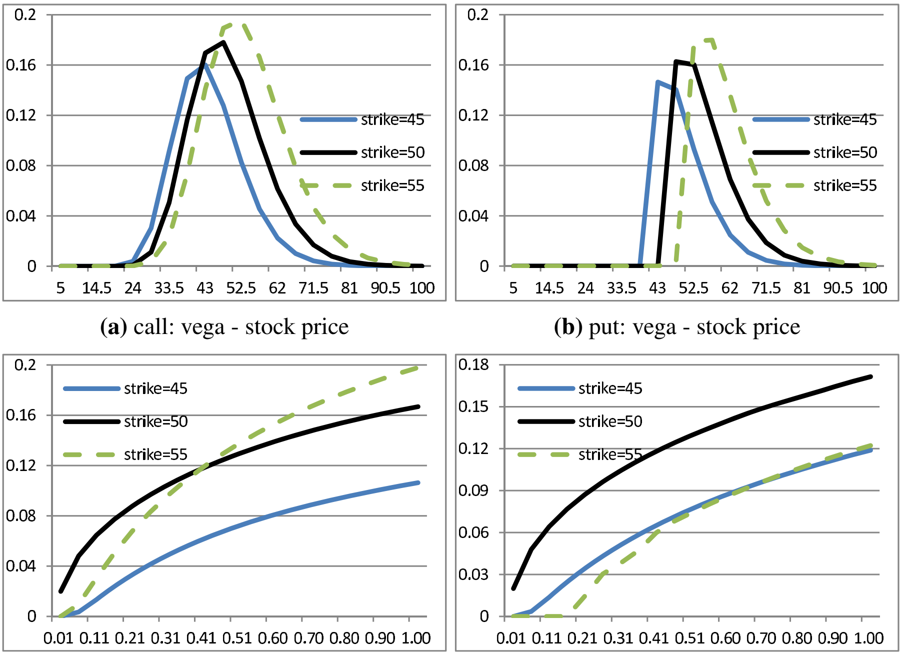
Rho (ρ) is the rate of change of the option's price with respect to the interest rate. To compute rho, again, we make a small change in the interest rate and obtain a new option price $P(S_0, K, r+\Delta r, \sigma, T, N)$, and we use $P(S_0, K, r, \sigma, T, N)$ to denote the old price before the change. Rho of the option measured in per 1% change in the interest rate is:

$$\rho = \frac{P(S_0, K, r + \Delta r, \sigma, T - P(S_0, K, r, \sigma, T, N)}{100\Delta r}$$

Some rho plots calculated from American call and put options on non-dividend paying stocks are shown in Fig. 8. The parameters are set to the same values as before.

9 Conclusions

We have presented our work on pricing American call and put options using the LR binomial trees. The underlying stocks of the options are assumed to pay out dollar cash



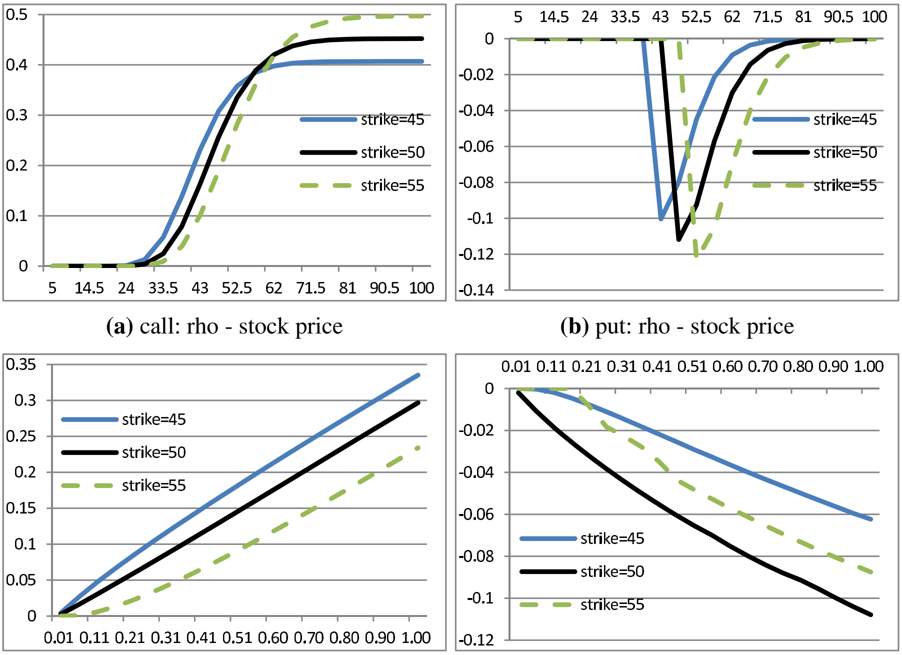
(a) call: vega - stock price

(b) put: vega - stock price

(c) call: vega - time to expiration

(d) put: vega - time to expiration

Fig. 7. Vega plots for American call and put options on non-dividend paying stocks



(a) call: rho - stock price

(b) put: rho - stock price

(c) call: rho - time to expiration

(d) put: rho - time to expiration

Fig. 8. Rho plots for American call and put options on non-dividend paying stocks

dividends within the lifetime of the options. We have compared the performances of the LR binomial and the CRR binomial methods. We confirm the previous report that the LR binomial method converges both faster and more smoothly than the CRR binomial method. We have also presented methods for estimating the Greek letters delta, gamma, theta, vega and rho using the LR binomial method, as well as some plots for the Greek letters.

Acknowledgement. This work is supported by the Research Development Fund 10-03-08 of Xi'an Jiaotong-Liverpool University.

References

1. Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. The Journal of Political Economy 81(3), 637–659 (1973)
2. Cox, J.C., Ross, S.A., Rubinstein, M.: Option Pricing: A Simplified Approach. Journal of Financial Economics 7(3), 229–263 (1979)
3. Hull, J.C.: Options, Futures, and Other Derivatives, 8th edn., ch. 20.3. Prentice Hall (2012)
4. Leisen, D., Reimer, M.: Binomial Models for Option Valuation - Examining and Improving Convergence. Applied Mathematical Finance 3, 319–346 (1996)
5. Merton, R.: Theory of Rational Option Pricing. Bell Journal of Economics and Management Science 4, 141–183 (1973)

A Resource-Centric Architecture for Service-Oriented Cyber Physical System

Kaiyu Wan[1] and Vangalur Alagar[2]

[1] Xi'an Jiaotong-Liverpool University, Suzhou, PRC
Kaiyu.Wan@xjtlu.edu.cn
[2] Concordia University, Montreal, Canada
alagar@cse.concordia.ca

Abstract. The strategic application domains of Cyber Physical Systems (CPS) [7,6] include health care, transportation, managing large-scale physical infrastructures, and defense systems (avionics). In all these applications there is a need to acquire reliable resources in order to provide trustworthy services at every service request context. Hence we view CPS as a large distributed highway for services and supply chain management. In traditional service-oriented systems service, but not resource, is a first class entity in the architecture model and resources are assumed to be available at run time to provide services. However resource quality and availability are determining factors for timeliness and trustworthiness of CPS services, especially during emergencies. So in the service-oriented view of CPS discussed in this paper we place services around resources, because resource constrain service quality. We investigate a *resource-centric*, and *context-dependent* model for *service-oriented* CPS and discuss *3-tiered* architecture for *service-oriented* CPS in this paper.

Keywords: Resource, Service-oriented Architecture, Service Model, Cyber Physical System.

1 Introduction

The NSF program description [7] states that CPS initiative [2] is "to transform our world with systems that respond more quickly, are more precise, work in dangerous and inaccessible environments, and provide *large scale distributed services*." This paper is a contribution to specify resources and resource-centric services. The term *resource* is used in a generic sense to denote an entity that is relevant in either producing or consuming a service. In CPS, physical devices are resources, which are hence first class entities. Services may be either generated or consumed by physical devices, which might in turn be consumed by cyber computational resources, such as communication protocols. Software services may be generated by the computational resources that reside either in a static or dynamic host computer in CPS network and may be consumed by other physical devices (actuators) to make changes in the environment. In general, a CPS resource might offer many services, a CPS service might require several resources, a CPS resource might *use* other resources, and a CPS (complex) service may be produced by combining several services and resources. Thus the service-oriented view of

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 686–693, 2013.

CPS is more complex than the service-oriented view required for traditional business applications, as discussed in SOC literature [1].

In this paper we regard the three conceptual layers of CPS resources as *physical*, *logical*, and *process*. Selic [9] uses the term resource to denote *any runtime entity for which the services can be quantified by one or more quality of service characteristics*. Thus the UML resource model of Selic [9] is restricted to the run-time (process) environment of a specific application in a centralized real-time system. The Resource-Explicit Service Model (RESM) proposed by Huang uses Entity Relationship diagrams to model resources and services as equal citizens [4]. It is possible to refine a physical layer model to a process layer model by adding more details. In doing so the complexity of the diagram will increase. Expressing logical dependencies between resources in this modeling notation is hard. RDF [10] is meant to describe web resources, which according to our classification are *virtual resources*. Resource models at physical and logical layers can use RDF. The Resource Space Model (RSM) describes the resource space and logical relationship between resources, for a specific application domain. This model will have multiple descriptions of one resource when that resource is used in different applications involving different resources. So, this approach does not support modeling the physical layer and to model resources at process level will be quite complex. In all these models context information, and QoS properties (such as reliability and availability) are absent.

CPS applications in areas such as health care, flood monitoring, and emergency evacuation require timely services, which in turn depend upon *availability* and *reliability* of resources. Both quantity and quality must be negotiated as often as necessary, and with as many resource providers as possible within the time limit set to complete the requested service. The absence of availability of reliable resources, and the emergence of severe competition for resources among services might cause the deadline not be met. For strict real-time applications, such as emergency evacuation, such situations are unsafe. Even when reliable resources are available in sufficient quantity, their distribution and cyber communication to service requesters may fail causing the deadline to fault. Consequently quality properties, attributes, context of use and availability constraints for resources must be published by resource producers in advance in order to enable service providers repeatedly discover resources required for providing services. Such discovery of resources maximizes the creation of services in advance and minimizes non-availability of resources at run-time. This is the motivation why we investigate resource-centric service model for Cyber Physical Systems in this paper.

Throughout the paper we suggest the underlying formalism without being formal. In Section 2, the resource-centric abstract service model is specified. In Section 3 we introduce the basics of context formalism necessary to understand how satisfaction relation is to be evaluated in a context. In Section 4 we use three-tiered approach to specify resource-centric service-oriented architecture for Cyber Physical Systems. We conclude the paper in Section 5 with a brief summary of its significance and our ongoing work.

2 Abstract Service Model

Abstractly, the three major stakeholders in CSP are *Resource Producer* (RP), *Service Provider* (SP), and *Service Requester* (SR). A SP may interact with one or more RPs

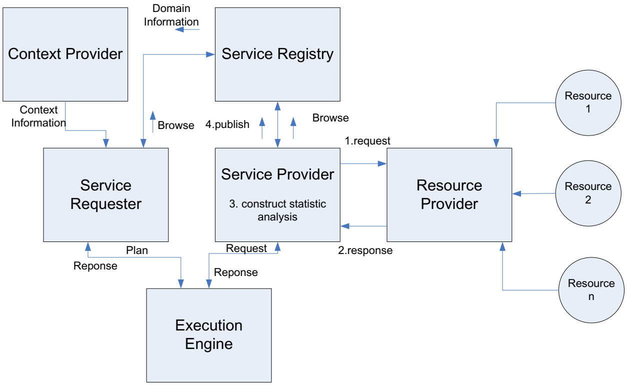


Fig. 1. Resource-Centric Abstract Service Model

and one or more SRs. A RP may not be *directly visible* to any SR in the system. So, a SR gets to know about resources used for service composition and delivery only from the service descriptions posted by the SPs. In this abstract CPS model shown Figure 1, every RP creates a resource model for each resource in its ownership and publishes it to all SPs who subscribe to its services. Thus, it is a comprehensive description of the physical, logical, and process layer needs. This specification will enable the SPs conduct a static analysis of published resource descriptions and request their distribution across CPS nodes in a demand-driven fashion. That is, they may acquire the resources and create their services well before service execution times. We regard *reliability* and *availability* as fundamental attributes for resource acquisition. A reliable resource is one that adds *economic value* for the client who uses it, by *satisfying the QoS characteristics of the client*. That is, the QoS characteristics *provided by* the resource satisfies the QoS characteristics *required by* the client. So, reliability is part of the QoS *contract* between the client and the resource used by it. In order that the client may use the resource to its full advantage, the resource must be *available* in sufficient quantity and when required by the client. So, availability has both a quantitative and temporal dimension. Consequently, both reliability and availability are made part of resource model. Once the resource model is published by a RP, the SPs who are clients of RPs will have an opportunity to independently verify the claims made in service descriptions before selecting it for use in the services created by them.

A SP creates service descriptions for services provided by it. A service description includes the functionality of the service, its non-functional properties, a list of resources used in creating and delivering the service, and a service contract. A SP publishes service descriptions and make them available to SRs who subscribe to its services. The SP guarantees the quality of service through a list of *claims*, which should be validated by the SP when challenged by the SRs. A SR creates a demand model of service. This model is very much dependent upon the application. It may be as simple as the ‘quality of result specification’. Examples include (1) ‘the cost should not exceed $50’, and (2) ‘the service should be delivered within 2 hours from the time the contract is signed’. Once the SR presents its model, after choosing a service type, the SP is expected to deliver a service whose quality attributes satisfy the quality attributes in the model presented by the SR.

Satisfaction Criteria
Therefore, in order to have matched CPS services the two essential conditions are

- *Provided-by*(RP_q) *SAT Required-by*(SP_q)
- *Provided-by*(SP_q) *SAT Required-by*(SR_q)

where *Provided-by*(X_q) means the 'quality attributes provided by the entity X', *Required-by*(Y_q) means the 'quality attributes required by the entity Y', and *SAT* is the 'satisfaction relation'. So we posit that the resource model should include *Provided-by*(RP_q), and the service model should include *Required-by*(SP_q), *Provided-by*(SP_q), and *Required-by*(SR_q). We assume that a SP, by whichever *Required-by*(SP_q) model it has, will select the resources in order to satisfy the relation *Provided-by*(RP_q) *SAT Required-by*(SP_q). We assume that a SR, by whichever *Required-by*(SR_q) model it has, will select the services in order to satisfy the relation *Provided-by*(SP_q) *SAT Required-by*(SR_q). Thus, the resource description should enable a formal execution of the *SAT* relation. Typical *SAT* relations are *implies* ($\rightarrow$), and *includes* (subset relation $\subset$)). These are resolved using Logic and Set Theory provers. We discuss in Section 3 a method to resolve situation constraints in different contexts.

3 Context-Dependence

In service-oriented systems and in particular for CPS, service contracts are usually context-dependent. Resource availability must be assessed from a combination of several factors ranging from rarity of the resource to legal implications in delivering it. An ubiquitous resource, such as water, may not be sold by a RP to a SP who is located in a zone Z either because the RP is not permitted to supply water in Zone Z or the water quality does not meet the standards of zone Z. In many countries strict environmental laws might forbid or restrict the use of certain types of energy resources. These examples are to motivate the necessity to include context information as part of resource and service descriptions. In order that such descriptions be formalized we need a formal representation of context. We use the formal notation of context and context toolkit developed by Wan [11,12] in order to formalize context information. A *context space* is defined for an application in a domain and contexts are constructed within that space. A context space includes a finite set of *dimensions* and a *type* associated with each dimension. The typed values are called *tags* along each dimension. A RP may define a context space with (1) *who* needs the resources? (2) *what* resource types are available? (3) *where* a resource can be delivered? (4) *when* the resource will be available? and (5) *why* the resource might be required? Contexts are constructed from the knowledge collected in the five dimensions *who*, *what*, *where*, *when*, and *why*. A RP can construct contexts that include all or only a subset or a superset of these dimensions. In a similar way a SP can construct contexts related to service provision. In the notation of Wan [11] a context is represented as $c = [WHERE : \text{Chicago}, WHEN : 04/07/2012, WHO : XYZ, WHAT : \text{EPR2}]$.

The interpretation is that c defines the setting in which the RP XYZ at $Chicago$ has the resource $EPR2$ at time $04/07/2012$.

A context in itself is not useful, unless it is associated with *events* or *situations* that are of interest in the context. The context formalism [11,12] allows evaluating situations at a context. A situation is encoded as a logical formula p on the dimension names and other variables. In order to check that a situation p is true in a context c, the dimension names in p are bound to the tag values in the definition of context c and p is evaluated. An example situation is the predicate $can\_deliver == (\mid x - WHERE \mid < 100) \ \wedge \ (d_2 < 10 + WHEN)$, where $\mid \ldots \mid$ denotes the distance expression and $(10 + WHEN)$, meaning within 10 days of specified time. When evaluated at c we will get the expression $(\mid x - Chicago \mid < 100) \ \wedge \ (d_2 < 14/07/2012)$. Once the values for the location variable x and date variable d_2 are known this expression can be evaluated to either true or false. This approach is used to resolve the *SAT* relation involving context situation constraints.

4 A Three-Tiered Architecture for Service-Oriented Cyber Physical System

In this section we put forth a *resource-centric*, and *context-dependent* model for *service-oriented* CPS. A *3-tiered* approach is shown in Figure 2. Tier-1 is the physical layer in which the attributes and properties of a resource are specified together with legal and contextual constraints. Tier-2 is the logical layer which imports specifications from Tier-1, introduces dependencies and constraints and lists possible ways to utilize the imported resource in services. Tier-3 imports resource class specifications from Tier-2 and specifies configured services by adding QoS properties of created service.

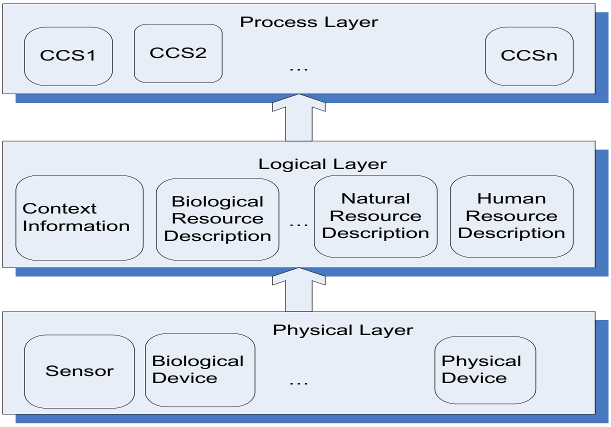


Fig. 2. Three-tiered architecture for CPS

4.1 Physical Description Layer

In this section we discuss the attributes for modeling resources in the physical layer. The model that we create is called *Resource Description Template* (RDT). We may assume that CPS resources are categorized so that all resources in a category are of the same *type*. One such classification is *human resources*, *biological resources*, *natural resources*, *man made resources*, and *virtual resources*. Human resources are well understood and many human resources management systems are available today. Resources required by a living being for survival and growth are biological resources. Examples include water and food. Natural resources are derived from the environment. Examples include trees, minerals, metals, gas, oil, and some fertilizers. Biological resource type is a subtype of natural resource type. Man made resources include physical entities such as bricks or mortar, books and journals for learning, and machineries. Any virtual component of limited availability in a computer is a virtual resource. Examples of virtual resource are *virtual memory*, *CPU time*, and the whole collection of Java resources [8].

A RP and its experts determine the essential features and properties to be specified in a resource model. The main attributes of resources, especially when it comes to their *adaptation* for providing services, are *utility*, *availability*, *cost*, *sustainability*, *renewability*, *reuse*. The utility factor for a resource defines its relevance, and often expressed either as a numerical value u, $0 < u < 1$, or as an enumerated set of values $\{$*critical*, *essential*, *recommended* $\}$. In the former case, a value closer to 1 is regarded as critical. In the later case the values are listed in decreasing order of relevance. A *Resource Producer* (RP) may choose the representation $\{\langle a_1, u_1\rangle, \langle a_2, u_2\rangle, \ldots, \langle a_k, u_k\rangle\}$ showing the utility factor u_i for the resource in application area a_i for each resource produced by it. The utility factors published by a RP are to be regarded as recommendations based on some scientific study and engineering analysis of the resources conducted by the experts at the RP sites. Cost might depend upon duration of supply (as in power supply) or extent of use (as in gas supply), or in required measure (as in the supply of minerals). Dependency between resources can often be expressed as situations, in which predicate names are resources.

4.2 Logical Layer Description

For the resource-centric CPS model we need to follow the resource-centric service approach, which is somewhat similar to the order-centric approach [13]. The activities in the service are ordered, and the list of activities per single resource are handled taking into account resource dependencies. This calls for a specification for each resource in which the dependencies on other resources and the tasks that can be done with that resource are listed. This is the logical view and we call this specification a *Resource Class Specification* (RCS). To realize the resource-centric model of CPS it is necessary that every CPS site publishes the RDTs of resources owned (or produced) by it as well as the RDTs acquired from other RPs, develop a mechanism for allocating resources in different service request contexts, and create a RCS.

4.3 Process Layer Model

The process layer for resource-centric CPS should model how services are configured, discovered, composed, and optimized. Among these process layer activities only service configuration activity requires a language description, the other activities require algorithms. So we restrict to service configuration description below.

In a service-oriented model the center piece is service and resources are not fully addressed within service model. On the other hand, in a resource-centric model, such as in [3], the center piece is resource class specifications and service model is ignored. In our resource-centric service model, resource class specifications are included in configuring and composing service specifications. The first step for SP is browsing the sites of those RPs, examining the RDTs published by them, and then selecting the RCSs published by them. The second step is that the SP selects the RPs from whom the RCSs can be bought. The final step for SP is to create services that can be provided by putting together the atomic tasks in the RCSs. We introduce the *CyberConfiguredService* (CCS) notation for this purpose. In CCS the service with its contract, quality assurances, and other legal rules for transacting business are included. Such configured services are published in the site of the SP.

Abstractly viewed, a service is a function. In business, a service not only has functionality but also has non-functional properties, legal issues for providing the service, and context information for service delivery. These are bundled together by the SP in a configured service. We define a *CyberConfiguredService* (CCS) is a service package that includes all the information necessary that a service requester in CPS needs to know in order to use that service. It will include (1) service functionality, (2) a list of resources used to create the service, together with resource specifications, (3) nonfunctional attributes of service, (4) quality attributes of the service, and (5) contract details. Legal rules, context information on service availability and service delivery, and privacy guarantees are part of contract details. The service and contract parts are integrated in CCS, and consequently no service exists in our model without a contract. The contract part in CCS includes QoS contract *Provided-by*(SP_q) as well as the QoS contract *Provided-by*(RP_q). These contracts must be resolved at service discovery and service execution times using methods explained in Section 3.

5 Conclusion

In this paper we have put forth a *resource-centric*, and *context-dependent* model for *service-oriented* CPS. Our contribution is a *3-tiered* approach. Tier-1 is the physical layer in which the attributes and properties of a resource are specified together with legal and contextual constraints. The attributes are typed, properties and legal rules can be formulated in logic, and context has a relational semantics [12]. As such Tier-1 specification has a semantic basis. Tier-2 is the logical layer which imports specifications from Tier-1, introduces dependencies and constraints and lists possible ways to utilize the imported resource in services. Tier-3 imports resource class specifications from Tier-2 and specifies configured services by adding QoS properties of created service. A specification from a lower tier can be included in more than one specification in the next higher tier. Modifications to a higher tier specification do not affect their constituent lower tier specifications. This 3-tier approach has the advantages of separation of concerns and modularity, the essential software engineering principles for developing large systems. In the near future we will continue our work on resource modeling, investigate formal notation for describing resource-centric services, and illustrate our ideas through proof-of-concept case studies.

Acknowledgement. This research is supported by Research Grants from National Natural Science Foundation of China (Project Number 61103029), Natural Science Foundation of Jiangsu Province, China (Project Number BK2011351) , and Natural Sciences and Engineering Research Council, Canada.

References

1. Georgakopolous, D., Papazoglou, M.P.: Service-oriented Computing. The MIT Press (2008)
2. C.S. Group. Cyber-physical systems: Executive summary. Report (2008), `http://varma.ece.cmu.edu/summit/CPS-Executive-Summary.pdf`
3. Zhuge, Y.H., Shi, P.: Resource space model, owl and database: Mapping and integration. ACM Transactions on Internet Technology 8(4) (2008)
4. Jian Huang, I.-L.Y., Bastani, F., Jeng, J.-J.: Toward a smart cyber-physical space: A context-sensitive resource-explicit service model. In: 33rd Annual IEEE International Computer Software and Applications Conference. IEEE Press (2009)
5. John, J.J.H., Guttag, V., Wing, J.M.: The larch family of specification languages. IEEE Transactions on Software Engineering
6. Networking, I.T. Research, and D. Program. High-confidence medical devices: Cyber-physical systems for 21st century health care. Technical report, NITRD (2009)
7. NSF. Usa nsf program solicitation, nsf-08-611. Report, NSF (2008)
8. Oracle. Java resources. Web report, Oracle (2003)
9. Selic, B.: A generic framework for modeling resources with uml. IEEE Computer 33(6), 64–69 (2000)
10. W3C. W3c recommendation. Technical report
11. Wan, K.: Lucx: Lucid Enriched with Context. Phd thesis, Concordia University, Montreal, Canada (2006)
12. Wan, K.: A brief history of context. International Journal of Computer Science Issues 6(2) (2009)
13. zur Muehlen, M.: Resource modeling in workflow applications. In: Proceedings of Workflow Management Conference

Implied Volatilities of S&P 100 Index with Applications to Financial Market

Jin Zheng[1], Nan Zhang[2], and Dejun Xie[3]

[1] Department of Mathematical Sciences, University of Liverpool, UK
[2] Department of Computer Sciences, Xian Jiaotong Liverpool University, China
[3] Department of Mathematical Sciences, Xian Jiaotong Liverpool University, China

Abstract. This paper studies the implied volatilities of the S &P 100 from the prices of the American put options written on the same index. The computations are based on a recursive Binomial algorithm with prescribed error tolerance. The results show that the volatility smile exists, thus the classic Black-Scholes's approach of using a constant volatility for pricing options with different trading conditions is not plausible. The method discussed in this work contrasts the likelihood ratio method contained in [6]. Further studies with expanded data set are recommended for comparing the effectiveness of these two methods in forecasting stock market shocks.

Keywords: Binomial Methods, Implied Volatility, Option Pricing.

1 Introduction

The Binomial tree method is developed as a stable tool for pricing Black-Schole options (see [1]), where the volatilities of the same underlying asset with the same expiration date are assumed as constant, even when the strike prices are different. However, abundant literatures have questioned the validity of this assumption. For example, Derman et al. (see [2]) argue that volatility is a function of time to maturity and strikes. Ederington and Guan (see [3]) show the non-flat property of implied volatilities based on a study of the S&P 500 futures options. They find that the implied volatility is higher for deep in- or out-of-the-money options than that of the at-the-money options. In this study, the implied volatilities are obtained by a recursive Binomial algorithm whose goal is to minimize the error between the theoretical price computed from the Binomial tree and the price recorded market observation.

The rest of the paper is organized as follows. Section 2 describes the model of our study, including data description and model derivation. Section 3 demonstrates the empirical results of our study. Section 4 discusses the applications of our modeling and estimation procedure for financial crisis prediction. In particular, a comparison is drawn with our pervious approach introdued in [6]. Concluding remarks and and possible future directions are provided in Section 5.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 694–699, 2013.

2 Methodology

2.1 Model Description

Binomial methods for valuing options arises from discrete random walks models, which assumes that the future stock price is not fully predictable, but progresses only from the value of the current time. The basic model expressed in terms of a geometric Brownian motion is as follows:

$$dS = rSdt + \sigma SdW \tag{1}$$

where r is the risk-free interest rate, σ is the volatility of stock price and W is the standard Brownian motion. The binomial tree method for valuing options assumes that both r and σ are constant. We define S_0 is the stock price at time 0. At time 1, the stock price S_1 can be either uS_0 with probability p or dS_0 with probability $1 - p$, where

$$u = A + \sqrt{A^2 + 1}; d = A - \sqrt{A^2 + 1}; A = \frac{1}{2}(e^{-r\delta t} + e^{(r+\sigma^2)\delta t}) \tag{2}$$

and

$$p = \frac{e^{r\delta t} - d}{u - d} \tag{3}$$

Let V_n^m denotes the value of American put option at time step $m\delta t$, the payoff function of the American put can be described by (see [5])

$$V_n^m = max(max(E - S_n^m, 0), e^{-r\delta t}(pV_{n+1}^{m+1} + (1-p)V_n^{m+1})) \tag{4}$$

where K is the strike price and S_n^m is the stock price estimated by Binomial method at time m. Therefore, in each step, we can identify whether to exercise the American put. The traditional Binomial tree method can only solve the problem about option pricing by assuming constant implied volatility. In our research, we apply the Binomial method by calibrating the value of σ to obtain the option price. The implied volatility, σ will be adjusted until the lowest error between the result by Binomial method and the option price on option exchange market is be achieved. A suedo-Matlab program is as follows.

1. Define the initial stock price S_0, time to maturity T, and strike price K. The parameters of random walk r and σ should also be determined.
2. Compute u , d and p. For each time step, calculate the stock price by using u and d and construct the tree.
3. Compute the put option values at time $t = T$, where $V = max(K - S_n^m, 0)$.
4. Obtain the value of American put at each node by applying equation (4). Record the option at the initial point, which is denoted as theoretical option value.
5. Observe the option value at option exchange market, and compare the error between the theoretical option value and the observed market value.
6. Update σ and repeat the above procedure until the minimum relative error is achieved.

2.2 Data Selection

The data used are daily S&P 100 index options which were observed from Chicago Board of Exchange (CBOE). S&P 100 index (OEX) options are options whose underlying assets are the values of the the S&P 100 index. As the barometer for the American economy, S&P 100 index option is an actively trading option and data set is abundant and representative for the experiment in this model. All the options on the S&P 100 index board will contain the expiration date and maturity days, closing market price of the options, daily index returns, S&P 100 closing stock prices and the interest rate at the maturity date. We choose six sample sets of American put from the available data with different maturity dates. The observations are carried out on December 18, 2012. Observations are made for all three types of options: in-the-money, at-the-money and out-of-the-money options.

3 Empirical Results

The following table and figures provides some of the numerical results. The sample period chosen contains short-term, medium-term and long-term options. Specifically, the maturity days are 31 days, 59 days, 87 days, 185 days, 367 days and 731 days. The graphs show a U-shape smile or half of the smile. For out-of-the-money put options ($S < K$), the implied volatility is descending, while for in-the-money put options($S > K$), the trend of the implied volatilities is ascending. In general, the turning point occurs when $S = K$, which is called at-the-money option. The short-term options show a more uniform smile than the long-term options. This implied volatility smile is not flat, which indicates that the Black-Scholes model's presumption of the same implied volatility for the options with the same maturity and observed on the same date is not appropriate. The results are reasonable, and consistent compared to the previous literatures.

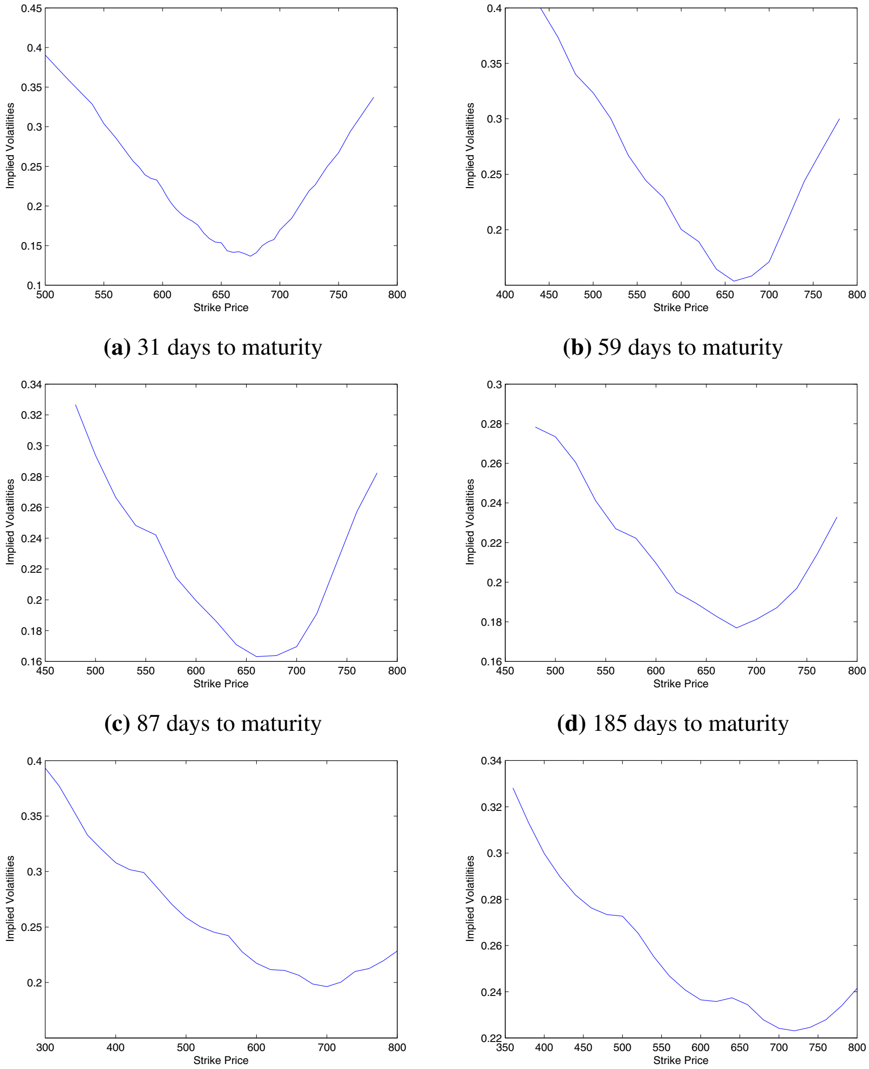
4 Applications of Stock Market Crisis

The study of implied volatility has many applications in the financial market. As one of the examples, the implied volatilities can be used as an efficient forecast of possible stock crisis. In our previous work (see [6]),the exponential average method is used to generate the daily implied volatilities. With the daily implied volatilities generated, we have addressed the problem by a structural modeling approach where the CIR mean-reverting process is adopted to describe the movement of implied volatilities over time. According to our previous research, the values of mean reversion speed, are higher immediately after the stock crisis than before the crisis, the conclusion of which is supported by the concept of mean-reversion disillusion (see [4]). In contrast to [6], one can apply the recursive Binomial method to obtain the implied volatilities. Since it has been shown that the implied volatilities can be captured by CIR model, the same procedures can be adopted to estimate the parameters of CIR model:

$$dv(t) = k(\theta - v(t))dt + \sigma\sqrt{v(t)}dZ(t) \tag{5}$$

Table 1. The implied volatilities for 6 samples of each strike price

K	$S1$	K	$S2$	K	$S3$	K	$S4$	K	$S5$	K	$S6$
500	0.3916	440	0.4	480	0.3267	480	0.2783	300	0.3932	360	0.3282
510	0.3757	460	0.3736	500	0.2938	500	0.2734	320	0.3769	380	0.3131
520	0.3602	480	0.34	520	0.2665	520	0.2605	340	0.3552	400	0.2998
530	0.3449	500	0.3235	540	0.2482	540	0.2411	360	0.3329	420	0.2898
540	0.3299	520	0.3003	560	0.2421	560	0.227	380	0.3201	440	0.2818
550	0.3056	540	0.267	580	0.2145	580	0.2222	400	0.308	460	0.2763
560	0.2878	560	0.2442	600	0.1994	600	0.2096	420	0.3017	480	0.2734
570	0.2678	580	0.2289	620	0.186	620	0.195	440	0.2991	500	0.2727
575	0.258	600	0.2003	640	0.1709	640	0.1893	460	0.2848	520	0.2653
580	0.2507	620	0.1893	660	0.1631	660	0.1828	480	0.2703	540	0.2552
584	0.2411	640	0.1645	680	0.1638	680	0.1769	500	0.2584	560	0.2468
590	0.2365	660	0.1537	700	0.1696	700	0.1813	520	0.2504	580	0.2408
595	0.2344	680	0.1583	720	0.191	720	0.1871	540	0.2453	600	0.2365
600	0.2239	700	0.1711	740	0.2242	740	0.1969	560	0.2423	620	0.2358
605	0.2112	720	0.2071	760	0.2574	760	0.214	580	0.2275	640	0.2374
610	0.2008	740	0.2438	780	0.2823	780	0.2329	600	0.2174	660	0.2344
615	0.1936	760	0.2723					620	0.2116	680	0.2279
620	0.1875	780	0.3001					640	0.2109	700	0.2242
625	0.1835							660	0.2066	720	0.2231
630	0.1793							680	0.1986	740	0.2246
635	0.1692							700	0.1963	760	0.2279
640	0.162							720	0.2003	780	0.2339
645	0.1576							740	0.2099	800	0.2415
650	0.1587							760	0.2126		
655	0.1488							780	0.2195		
660	0.1468							800	0.2284		
665	0.1475										
670	0.1503										
675	0.147										
680	0.1518										
685	0.1604										
690	0.1709										
695	0.1831										
700	0.1972										
705	0.2044										
710	0.2117										
715	0.2233										
720	0.2349										
725	0.2464										
730	0.2534										
740	0.276										
750	0.2939										
760	0.3204										
780	0.3636										



(a) 31 days to maturity

(b) 59 days to maturity

(c) 87 days to maturity

(d) 185 days to maturity

(e) 367 days to maturity

(f) 731 days to maturity

Fig. 1. The implied volatility smiles for S&P 100 index put options with different maturities

where where $v(t)$ is the variance of the stock, and $Z(t)$ is a Brownian Motion, and k, θ and σ are positive constants. We apply the maximum likelihood method to estimate the three parameters and the results are presented in Table 2.

The significant change of k between sample 1 and sample 2 indicates that there should be a fluctuation in the stock market between January and Feburary, 2013. The result is not surprising due to the current economic conditions in U.S. The U.S. fiscal cliff may cause the economic turndown if there is no effective measurement to prevent it.

Table 2. The estimated parameters for CIR model

Sample	k	θ	σ
Sample 1	13.697	0.137	0.035
Sample 2	66.669	0.045	0.039
Sample 3	74.368	0.061	0.023
Sample 4	70.771	0.052	0.015
Sample 5	51.539	0.043	0.034
Sample 6	59.769	0.053	0.016

5 Concluding Remarks

This paper focuses on the Binomial approach for obtaining the implied volatilities in a stochastic financial environment. In this work, discrete random walk model is assumed and applied to scale the stock prices at different time nodes. We iterate the Binomial method recursively to obtain the implied volatilities by achieving the lowest error between the theoretical option value and the observed market value. Results from these empirical experiments demonstrate that the implied volatilities display a U-shape or half of the smile-shape, which is consistent with the existing literatures. Further studies can be carried out with larger data set for calculating the likelihood ratios to predict the stock market crisis, and draw comparisons.

Acknowledgement. The work is partially supported by Natural Science Foundation of Jiangsu Province of China (BK2011350).

References

1. Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. Journal of Political Economy 81, 637–654 (1973)
2. Derman, E., Iraj, K.: The Volatility Smile and Its Implied Tree, Quantitative Strategies Research Notes, Goldman Sachs (1994)
3. Ederington, L., Guan, W.: Why Those Options Smiling. The Journal of Derivatives 10, 9–34 (2002)
4. Hillebrand, E.: A Mean-Reversion Theory of Stock-Market Crashes, Center for Complex Systems and Visualization, University at Bremen, Department of Mathematics, Stanford University (2003)
5. Wilmott, P., Howison, S., Dewynne, J.: The Mathematics of Financial Derivatives, pp. 187–189. Cambridge University Press
6. Zheng, J., Xie, D.: Stochastic Modeling and Estimation of Market Volatilities with Applications in Financial Forecasting. International Journal of Statistics and Probability 1, 2–19 (2012)

RF Characteristics of Wireless Capsule Endoscopy in Human Body

Meng Zhang[2], Eng Gee Lim [1,*], Zhao Wang[1], Tammam Tillo[1],
Ka Lok Man[1], and Jing Chen Wang[2]

[1] Xi'an Jiaotong-Liverpool University, Suzhou, China
{enggee.lim,zhao.wang,tammam.tillo,ka.man}@xjtlu.edu.cn
[2] Xi'an Jiaotong University, Xian, China
{zhangmeng6176,Elain16}@xjtu.edu.cn

Abstract. Wireless capsule endoscopy (WCE) is an ingestible electronic diagnostic device capable of working wirelessly, without all the limitations of traditional wired diagnosing tools, such as cable discomfort and the inability to examine highly convoluted sections of the small intestine. However, this technique is still encountering a lot of practical challenges and requires further improvements. This paper is to propose the methodology of investigating the performance of a WCE system by studying its electromagnetic (EM) wave propagation through the human body. Based on this investigation, the capsule's positioning information can be obtained. The WCE transmission channel model is constructed to evaluate signal attenuations and to determine capsule position. The detail of this proposed research methodology is presented in this paper.

Keywords: Wireless capsule endoscopy (WCE), Electromagnetic wave propagation, positioning, transmission channel.

1 Introduction

The use of endoscopes to examine the body's internal organs dates back to the 19th century [1], where a Mainz scientist developed the 'Lichtleiter' to examine human bladder and bowel with candle light. Later, various types of endoscopes were developed to examine the body's internal organs in greater detail. Timely detection and diagnosis are extremely important since the majority of gastrointestinal (GI) cancers are curable if caught early. Traditional surgical treatments through the use of endoscopies were developed into two branches: gastroscopy for examining the stomach and colonoscopy for the intestines. Each branch developed rapidly in the last two decades, eventually culminating in the birth of capsule endoscopy. Compared to earlier techniques, capsule endoscopy as shown in figure 1 is non-invasive and hence more comfortable to patients. It can examine deeper GI tracts in the human body inaccessible with existing wired endoscopes.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 700–706, 2013.

The Wireless Capsule Endoscope (WCE), a small capsule-shaped device containing a video camera, LED lights, a power source and a wireless transmitter, is used to detect various diseases within the digestive system (e.g. in duodenum, jejunum, ileum, etc.). There are many different types of wireless capsule endoscopes and they are mostly developed and manufactured by Olympus [2], Intromedic [3] and Given Imaging [4].

There are drawbacks limiting the application of WCE. First of all, the collected physiological data, such as the GI tract images, are insufficient for clinical diagnosis without the presence of capsule positioning data. Secondly, most capsules are powered by an internal battery cell that in turn restricts capsule miniaturization. Lastly, current systems do not have continuous communication due to random orientations of the Capsule [5].

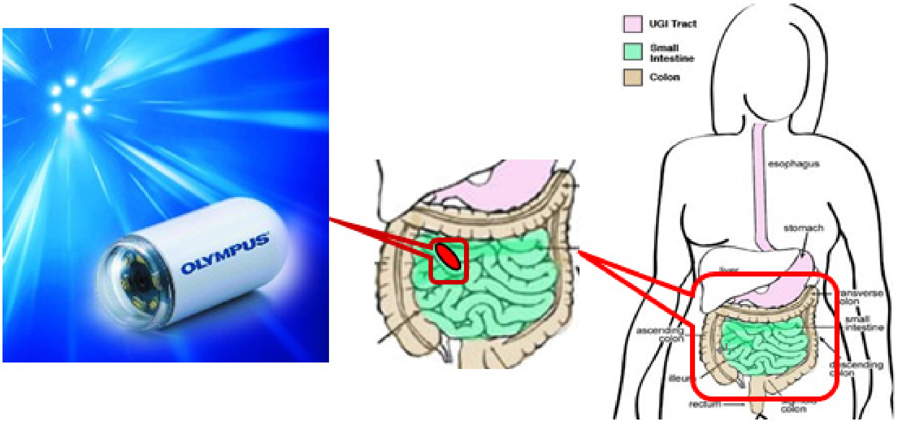


Fig. 1. Human GI tract and wireless endoscopy capsule [2]

This paper is proposing a methodology to investigate the performance of a WCE communication system by studying its EM wave propagation. This investigation serves to determine signal attenuation and capsule position. There are two main reasons for this proposed research. The first one is that some EM energy would be absorbed by the organs when waves are transmitted through the human body, which could lead to large signal distortions. In addition, the human body is a frequency dispersive system with frequency dependent parameters (permittivity and conductivity) [6] that influence the electrical and magnetic properties of the signal transmission channel. The parameters change when wide-band signals are applied to the system, which require human body models to simulate the signal transmission with frequency dependent permittivity and conductivity. Furthermore, positioning of the capsule in the human body can be achieved by studying the EM wave propagation of the system, enabling tracking of position and orientation of the capsule without adding additional sensors. This allows more capsule space to be allocated for other components.

For the above reasons, this proposed project works through three aspects. Firstly, the level of signal distortion in the transmission system is investigated. Secondly, determine the EM wave propagation properties of WCE with different transmission distances when the transmitting and receiving antennas are in the same work plane (z=0 plane) by simulating the communication process through the human body model. Lastly, the second step is repeated but with varying work planes to simulate cases where the transmitting and receiving antennas are not in the same plane (z≠0 planes).

2 Overview and Proposed Methodology

EM wave propagation of the WCE transmission channel is studied by examining signal distortions, extracting the WCE's location information and determining the capsule position. Therefore, a WCE communication system is built with a transmitter, a receiver and a communication channel. In order to better understand the system, each component will be studied separately.

2.1 EM Wave Propagation Environment

The abdominal environment is highly complex and the small intestines, which lie in close proximity, greatly influence the results. Therefore, the inhomogeneous human body module is simplified to a homogeneous body model which uses muscle material whose relative permittivity equals 56 and conductivity equals 0.83 S/m [7]. The shape of the body model can be cylindrical or ellipsoidal based on the needs of the study. Additional tissue layers may be added for further investigations.

2.2 Transmitting and Receiving Antennas

To implement the communication system, a suitable transmitting and receiving antenna will be selected to operate in the human body environment. The WCE antenna should be less sensitive to human tissue influences as the EM wave transmits in the body. Lossy dielectric material absorbs a number of waves and thus attenuates the receiving signal, causing strong negative effects on the EM wave propagation. A much wider bandwidth is required to enable transmission of high resolution images and large amounts of data. The detection of transmitted signal is preferred to be independent of the transmitter's position and hence, the transmitting antenna should have an omni-directional radiation pattern.

On the basis of the Friis formula [8], the total loss between the transmitter and receiver is calculated by the distance between the transmitting and receiving antennas, which is 15 cm. It was calculated that the minimum total loss is achieved when the operating frequency is between 400-600 MHz [9-10]. Therefore the antenna in [9] which operates at 410 MHz (as shown in Figure 2) is chosen for the modelling of the communication channel. To match the size and phase (of S21) of the transmitter and receiver pairs, a similar antenna design is applied to both the transmitter and receiver.

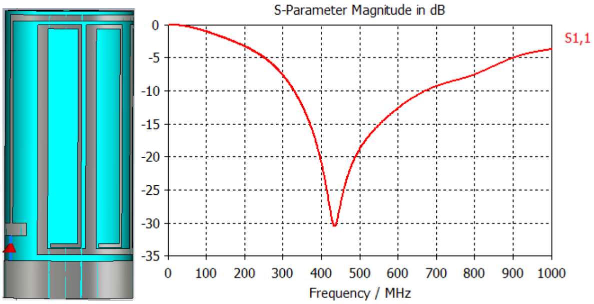
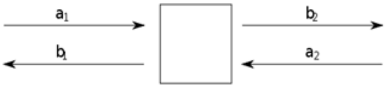


Fig. 2. Physical layout of conformal antenna and the return loss

The communication system including the transmitter, intermediate material and receiver can be considered as a two port network. Therefore, scattering parameters can be used to analyze this system as shown in Figure 3. S11 is the return loss used to determine the channel bandwidth and S21 the ratio of voltage reflected at port 2 over the voltage sent from port 1, called the forward voltage gain, is used in the following sections.



$$S11 = \frac{b1}{a1} = \frac{V_{reflect\,at\,port1}}{V_{towards\,port1}}\bigg|_{a2=0} \qquad S12 = \frac{b1}{a2} = \frac{V_{reflect\,at\,port1}}{V_{towards\,port2}}\bigg|_{a1=0}$$

$$S21 = \frac{b2}{a1} = \frac{V_{reflect\,at\,port2}}{V_{towards\,port1}}\bigg|_{a2=0} \qquad S22 = \frac{b2}{a2} = \frac{V_{reflect\,at\,port2}}{V_{towards\,port2}}\bigg|_{a1=0}$$

Fig. 3. Scattering parameters

3 Methodology and Feasibility Analysis

The following methodologies are proposed in order to study the electromagnetic wave propagation of Wireless Capsule Endoscopy in human body.

3.1 Relative Angle Position between TX and RX

Antenna is not symmetrical structure in general, therefore the influence of the antenna radiation pattern should also be taken into consideration. To test the system, one direction of the transmitter with the appropriate radiation pattern is chosen. Excitation signals are supplied into the transmitting antenna while signals at the receiving antenna are compared to the input signal to check for attenuations. The receiver is placed at different angles, surrounding the transmitter and separated by 45 degrees each (illustrated in Figure 4).

The unstable forward voltage gain influences the accuracy of localization results. If the capsule is rotated around the center of the outer shell, errors are introduced to the localization calculations. This influence needs to be taken into consideration while performing the localization estimation.

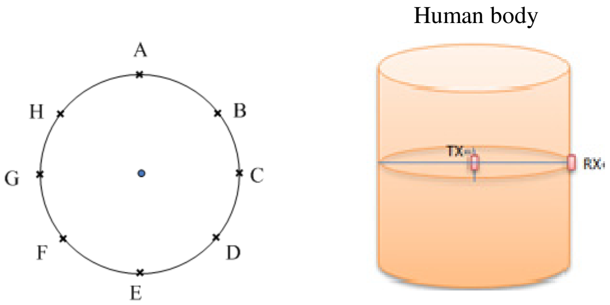


Fig. 4. Evaluate the RP effects and the in body communication system

3.2 Transfer Function and S21

The simulated S21 results are examined to see if it could be considered the transfer function. Since the Finite Integration Technique makes CST's simulation results in the time domain more accurate than the frequency domain, Discrete Fourier Transform (DFT) will be used for both input and output port signals to calculate the transfer function, and it is compared with S21 results achieved from CST. Limited by the signal time gap between two discrete values, the number of points within 1 GHz is small, but the transfer function trend can be obtained..

3.3 Relative Distance between TX and RX(z=0 plane)

At this stage, different offsets between transmitter and receiver are applied to this system to collect the simulation results of S21, the forward voltage gain. To perform capsule localization, signal transmission distances are swept from 0 to 160 mm with 20 mm steps. With the position of RX fixed, TX is moved in the human body model to obtain the EM wave propagation properties of WCE with different offsets as shown in Figure 5.

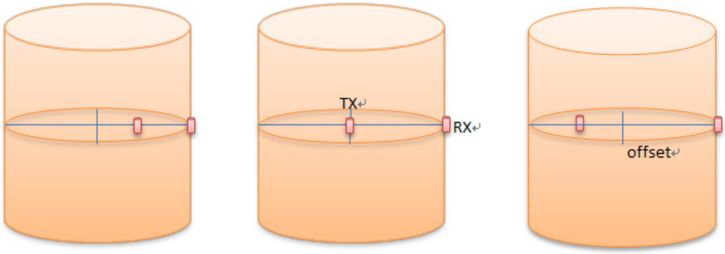


Fig. 5. Layouts of TX and RX (offsets: 50, 100, and 150 mm)

3.4 Relative Position between TX and RX(z≠0 plane)

Since the position of the capsule endoscope in the gastrointestinal tract is changing, knowing the transmission characteristics of the wireless signal in the same plane is not sufficient. The step in Section 3.3 will be repeated but with the transmitter and receiver in different planes as shown in Figure 6 so that the location and quantity of the receiver and transmitter can be determined. Three-dimensional positioning and tracking of the capsule endoscope is therefore possible. As receiver moves along the z-axis, the radio propagation properties of WCE with different signal transmission distances when the transmitting and receiving antennas in the different plane (z≠0 plane) can be obtained.

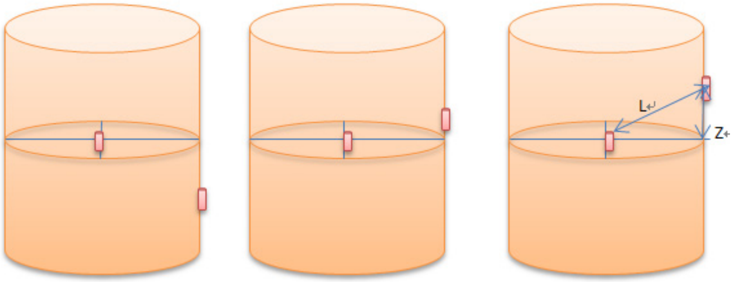


Fig. 6. TX and RX are in different work planes

4 Conclusion and Future Work

In this paper, the proposed methodology was to investigate the performance of a WCE system. Based on this investigation, the capsule's positioning information can be obtained. The WCE transmission channel model was constructed in order to examine signal attenuations and determine capsule position. The outcome of this investigation will be useful for researchers to carry out further research in locating the WCE position within the human body.

Acknowledgement. This work is partially supported by the Natural Science Foundation of Jiangsu province (No. BK2010251 and BK2011352), Suzhou Science and Technology Bureau (No. SYG201011 and SYG201211), and XJTLU Research Development Fund (No. 10-03-16.).

References

1. Endoscopy, From Wikipedia, the free encyclopedia (2011), http://en.wikipedia.org/wiki/Endoscopy
2. Web page of Olympus, http://www.olympus-europa.com/endoscopy/2001_5491.htm

3. Web page of Intromedic, http://www.intromedic.com/
4. Web page of Given Imaging, http://www.givenimaging.com/
5. Fireman, Z., Kopelman, Y.: New frontiers in capsule endoscopy. J. Gastroenterol. Hepato., 1174–1177 (2007)
6. The Finite Integration Technique, CST Computer Simulation Technology AG. (2011), http://www.cst.com/Content/Products/MWS/FIT.aspx
7. Kwak, S.I., Chang, K., Yoon, Y.J.: The helical antenna for the capsule endoscope system. In: Proc. IEEE Antennas Propag. Symp., vol. 2B, pp. 804–807 (July 2005)
8. Lee, J., Nam, S.: Q evaluation of small insulated antennas in a lossy medium and practical radiation efficiency estimation. In: Proc. Korea–Jpn. Microw. Conf., pp. 65–68 (November 2007)
9. Kim, K., Yun, S., Lee, S., Nam, S., Yoon, Y.J., Cheon, C.: A Design of a High-Speed and High-Efficiency Capsule Endoscopy Syste. IEEE Transactions on Biomedical Engineering 59(4) (April 2012)
10. Wang, J.C., Lim, E.G., Wang, Z., Huang, Y., Tillo, T., Zhang, M., Alrawashdeh, R.: UWB Planar Antennas for Wireless Capsule Endoscopy. In: International Workshop on Antenna Technology (March 2013)

Building a Laboratory Surveillance System via a Wireless Sensor Network

Chi-Un Lei[1,⋆], J.K. Seon[2], Zhun Shen[3], Ka Lok Man[3,4],
Danny Hughes[5], and Youngmin Kim[6]

[1] Department of Electrical and Electronic Engineering
The University of Hong Kong, Pokfulam Road, Hong Kong
culei@eee.hku.hk
[2] LS Industrial Systems, South Korea
jkseon@lsis.biz
[3] Department of Computer Science and Software Engineering
Xi'an Jiaotong-Liverpool University, Suzhou, China P.R.C.
Zhun.Shen08@student.xjtlu.edu.cn, ka.man@xjtlu.edu.cn
[4] Myongji University, South Korea; Baltic Institute of Advanced Technology, Lithuania
[5] Department of Computer Science, KU Leuven, Flanders, Belgium
danny.hughes@cs.kuleuven.be
[6] UNIST Academy-Industry Research Corporation, South Korea
youngmin@unist.ac.kr

Abstract. Contemporary technical experimentations become complicated. Therefore, a smart laboratory environment is needed for effective laboratory activities. In particular, a monitoring/surveillance system is needed to detect and regulate extreme ambient conditions in the laboratory. In this paper, we describe how a thermal comfort laboratory surveillance system is constructed via the deployment of a wireless sensor network (WSN). In order to prolong system lifetime as well as improve system reliability, a habit-based adaptive sensing mechanism has been proposed. Evaluations of on-site deployment results indicate the functionality and feasibility of the proposed WSN.

Keywords: laboratory surveillance, wireless sensor network.

1 Introduction

Wireless sensor networks (WSNs) are wireless network systems that contain numerous distributed, linked, and autonomously operated sensor nodes [1–3]. In each sensor node, sensors are used to examine environmental conditions in distributed locations, for assisting human to make better decisions or allowing machines to make decisions automatically. Results show that WSNs have been successfully deployed for clinical monitoring [4], environmental surveillance [5] and other surveillance applications [1]. However, WSNs were seldom used in monitoring environments in high-end industries.

[⋆] This research is partially funded by the Xi'an Jiaotong-Liverpool University Research Development Fund, under grants 2010-26 and 10-02-03; the Xi'an Jiaotong-Liverpool University Educational Development Foundation, under grant 201211; and Transcend Epoch Intl. Co., Ltd, Hong Kong.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 707–714, 2013.

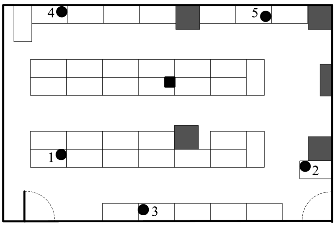


Fig. 1. The deployment of the WSN in the laboratory. The black square, black circles, grey squares, and white rectangles denote the base station, sensor nodes, wall structures, and laboratory benches, respectively.

On the other hand, contemporary technical experimentations, instrumentations, and productions become complicated. For examples, researchers have to work with volatile chemicals/biological substances and sophisticated equipment. Therefore, a smart laboratory environment is needed for effective laboratory activities. In particular, the smart laboratory environment should be able to i) detect hazards in laboratories, such as existence of pest, accidental release of chemical and bacteria, and extreme ambient conditions, ii) monitor the health of users, conditions of equipment as well as the laboratory environment, iii) track the existence of equipment, chemicals and other substances for maintaining safety and security, and iv) regulate the environment of the laboratory to reduce power/resource consumption without deteriorating efficiency of activities in the laboratory. Laboratory management systems [6] and remote laboratory systems [7] have been proposed for data logging and utilization of laboratory equipment, respectively. However, no monitoring/surveillance systems have been developed to gather adequate information, transfer data to information and knowledge, and eventually providing useful and prompt services for hand-on laboratory activities.

In this paper, we describe how a laboratory surveillance system is constructed via the deployment of a WSN. In particular, the WSN is used to gather and act on relevant information about the thermal comfort of physical environments. In order to prolong the operation time of sensor nodes, a habit-based adaptive sensing mechanism has been proposed. In this paper, functionalities and requirements for the surveillance in laboratories as well as hardware and habit-based adaptive sensing mechanism are discussed in Section 2. The performance evaluation of the developed WSN is shown in Section 3. Section 4 concludes the paper.

2 Laboratory Surveillance System

Our proposed WSN contains numerous distributed sensor nodes and a base station. When the WSN operates, sensor nodes autonomously examine relevant environmental conditions in the laboratory. Measured physical quantities are then sent back to the base station for analysis and post processing.

2.1 Deployed Environment and System Requirements

The proposed WSN has been deployed in a project laboratory in an university. The laboratory is located in the basement of the building. We observed that the air ventilation of the laboratory may not be adequate for large-class activities with about 160 students and after office hours. Furthermore, laboratory activities are sometimes affected by ground vibrations and are not always supervised by technicians. Therefore, the major task of the system is to measure vibration, temperature and humidity. The floor-plan of the laboratory and the deployment of five sensor nodes are shown in Fig. 1. In order to ensure a complete coverage of surveillance, the base station is placed in the middle of the laboratory. Meanwhile, sensor nodes are distributed throughout the laboratory. In addition, there are a few requirements for the deployment of WSNs in laboratories:

- There are no restrictions in the placement of nodes. However, they should be placed seamlessly on work benches, in order to measure realistic environments as well as to prevent destructions by impact of equipment and human activities.
- Most power plugs have been occupied by equipment, therefore only the base station can be powered by a power plug. In other words, sensor nodes are powered by battery packs. Furthermore, low-volume/-power wireless communication should be used in the WSN, and computation-intensive data analysis should be done by the base station. Furthermore, habit-oriented adaptive sensing mechanisms should be introduced for power saving.
- Wireless communication may be blocked occasionally because of physical and electromagnetic obstacles in the laboratory (e.g., instruments and computers). Therefore, reliable transmission mechanisms and measures are needed.
- Sensor nodes are located in predetermined locations. Therefore, localization and beaconing are not required.

2.2 Relevant Physical Quantities for Measurements

The core function of the proposed WSN is to measure physical quantities for laboratory surveillance. These quantities can be used to keep people and instruments away from hazards, accidents, and usurping, as well as to provide a suitable, comfort, and productive environment. In summary, laboratory productivity can be related to many physical quantities. Examples of relevant quantities are shown in Table 1.

In laboratory surveillances, the two major relevant physical quantities are temperature, and relative humidity (RH). For example, people can be less concentrated and productive if the thermal comfort is not provided [8]. Reliability of equipment can also be affected by the temperature. In addition, people become not comfortable when the RH of the environment is lower than 25% or higher than 60%. Meanwhile, products, materials, equipment, and performance of reactions can also be deteriorated if the environment has an abnormal humidity. For example, low RH may cause problems with static electricity, which may cause damages to static-sensitive equipment and materials as well as may cause fires and explosions when working with flammable liquids and gases. Meanwhile, when RH is high (e.g. $>70\%$), there may be condensations on surfaces of instruments, which leads to corrosions and moisture-related deteriorations. Therefore, these quantities should be examined and moderated in the laboratory.

Table 1. Relevant Physical Quantities for Laboratory Surveillance

Monitored condition	Relevant physical quantities
Human comfortability	temperature, humidity, air movement, light intensity
Human safety/healthy	toxic gas, particulates
Instrumentation effectiveness	temperature, moisture, particulates, vibration, air pressure
Equipment safety and security	rotation, acceleration, collusion, vibration, existence
Power effectiveness	current, power, temperature

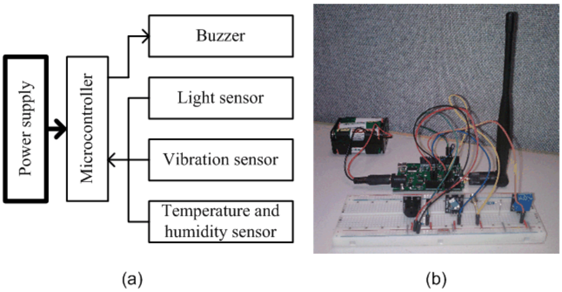


Fig. 2. Implemented sensor node for laboratory surveillance: (a) block diagram, and (b) the implemented prototype

2.3 Hardware of Sensor Nodes

The implemented sensor node of the proposed WSN is shown in Fig. 2. A sensor node detects physical quantities from sensors on the microcontroller board. Measured quantities are then processed on the board, and transmitted to a relay/base station via a wireless transceiver. In the proposed WSN, Arduino-compatible and transceiver-embedded microcontroller platform Zigduino is used as the microcontroller board of sensor nodes. The board contains a Atmega128RFA1 processor and a 2.4 GHz antenna. Furthermore, there are 14 digital input/output ports and six analog input ports on the microcontroller board that can be connected to sensors. The board also has 128 KB of flash memory, 16 KB of SRAM and 4 KB of EEPROM. These adequate peripherals allow the system to provide a versatile operation. In addition, the node draws 15 mA, 6 mA, and 250 μA in the mode for transmitting, sensing, and sleeping, respectively.

Besides microcontroller boards, sensors are also critical in WSNs. In WSNs, sensors have to provide an accurate, reliable and low-power/-volume measurement. The implemented prototype has a resistive humidity sensor component and a negative temperature coefficient (NTC) temperature detection component. The measurable temperature range is from $0°C$ to $50°C$, and RH range is from 20% to 90%. The measurement mechanism is power-aware and reliable because i) the sensor is switched on only when an interrupt is received from the microcontroller, and ii) the sensor sends the measured quantity to controller board with a checksum for verifications. Besides sensing of temperature and RH, the prototype also has a light sensor and vibration sensor to examine existences of research activities and vibrations.

Table 2. Meaning of the node message {AXXYYS} in the WSN

Condition	XX	YY
If $0 \leq XX \leq 50$	Temperature	Relative humidity
If $XX = YY$ and $XX > 50$	Abnormal condition	55: Too high temperature 60: Vibration 65: Sensor failure

A: Node number
S: Checksum (Last digit of summation of digits of A, XX and YY)

e.g. {125356} : “Node 1: Temperature = 25°C; Relative humidity = 35%
e.g. {165653} : “Node 1: Sensor failure
e.g. {165654} and {165350}: Invalid messages

2.4 Communications between Nodes

When the microcontroller in the sensor node indicates the validity of the sensor reading, a low-volume message is sent to the base station through the transmitter. Examples of transmitted/received messages are shown in Table 2. Normally, each message consists of its sensor node number, a temperature reading, a RH reading, and a checksum. Furthermore, since the temperature range of the sensor is from $0°C$ to $50°C$, redundant temperature range has been used to indicate abnormal conditions (“hazards”), such as vibrations and sensor failures.

The proposed WSN involves multiple sensor nodes interacting over the same communication channel. If the sensor node or the base station suffer from communication collisions in the network, the surveillance of a certain region or even the whole region in the laboratory can be terminated. Therefore, communication mechanisms should be developed to minimize effects on system dynamics due to node failures. Therefore, for reliable communications, a quasi-802.15.4 MAC mechanism with acknowledgement and a disconnection alarm have been used in the implemented WSN, such that the WSN can notify technicians for re-configurations. In particular, if the sensor node cannot successfully transmit messages to the base station, the sensor node is classified as “disconnected”. In addition, if the quality of transmission is continuously low (e.g., Link Quality Indicator (LQI) < 200), the sensor node is classified as “poorly connected”.

2.5 Adaptive Sensing for Low-Power Operations

A core task in WSNs is to minimize the energy used for node communications. In our deployed environment, the base station can be powered by a power plug. However, sensor nodes are mainly powered by 5V 3Ah battery packs. If there is no duty cycling, the node may only be operated for one day. Therefore, an adaptive sensing mechanism is proposed for a long-term surveillance. In other words, based on the computed duty cycle, the sensor node turns on the sensor (“sensing mode”) and radio transceiver (“transmitting mode”) for sensing and message transmissions, respectively. Finally, the sensor node is switched into “sleeping mode” when the message is acknowledged.

The adaptive sensing mechanism adjusts the duty cycle of sensor nodes, based on i) the existences of laboratory activities, ii) the existences of abnormal conditions, and

Table 3. The configuration of duty cycle adjustment

Condition	Laboratory activities	Initial duration (sec.)	Increment of duration in each step (sec.)	Maximum duration (sec.)
Normal	Exist	15	5	30
Normal	Not exist	30	10	300
Abnormal	Exist	5	0	5
Abnormal	Not exist	5	5	60

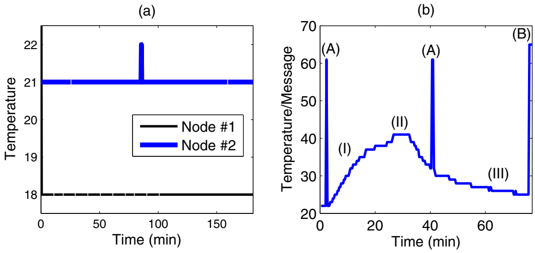


Fig. 3. Obtained readings of temperature and relative humidity. (a) From Sensor Nodes #1 and #2 in normal conditions; and (b) From a sensor node in abnormal conditions.

iii) the status of current operations. For example, if the condition does not change, the duty cycle can be prolonged gradually until the duty cycle reaches the pre-determined maximum duty cycle. Meanwhile, the maximum duty cycle can be longer if there are no activities or abnormal conditions in the laboratory. Apart from this, we observe that researchers sometimes work overnight in the laboratory, therefore light intensity instead of time (e.g. 8AM-5PM) is used as the indication of existences of laboratory activities. Furthermore, time synchronization is not required. In summary, the configuration of duty cycle adjustment is shown in Table 3.

3 Evaluation of the System Performance

3.1 Detecting Normal and Abnormal Environment Conditions in the Laboratory

Collected sensor readings have been analyzed to detect the onset of deterioration of laboratory environment. Results in Fig. 3(a) show the temperature difference between Node #1 and Node #2 in a normal situation. As shown from the figure, temperature should be moderated through ventilation. Readings in abnormal conditions are shown in Fig. 3(b). From received messages, the base station noticed that the ambient became hot from Time (I) to Time (III). In particular, the base station noticed that i) the temperature have been changed rapidly at Time (I), and ii) the temperature became extremely high at Time (II). Therefore, a message has been sent to switch on the ventilation system and remove the hot source. Meanwhile, the base station also noticed that i) there were a vibration and a sensor failure at Time (A) and (B), respectively. In aforementioned abnormal conditions, notifications had been sent to technicians. These examples show that the WSN is capable of detecting deteriorations of conditions in the laboratory.

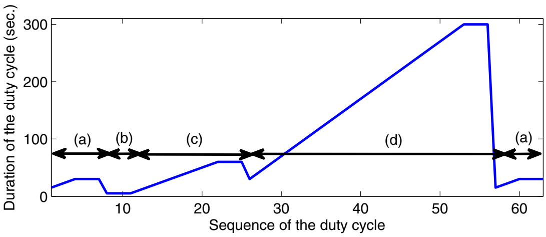


Fig. 4. The duty cycle of a sensor node with the adaptive sensing mechanism. (a) Normal conditions with laboratory activities; (b) Abnormal conditions with laboratory activities; (c) Abnormal conditions with no laboratory activities; and (d) Normal conditions with no laboratory activities.

Table 4. System reliability and network reliability in the WSN.

Sensor Node	System reliability	Network reliability (LSI; Max: 255)	Power (dBm)
#1	96.87%	253.99	-90.06
#2	98.53%	227.09	-89.50
#3	98.71%	255.00	-81.03
#4	96.69%	254.63	-89.73
#5	97.24%	254.20	-84.89

3.2 Service Reliability and Power Consumption of the Adaptive Sensing Mechanism

The adaptive sensing mechanism of an 1.8-hour surveillance example has been evaluated. In the example, the sensor node switched on and off based on the calculated duty cycle. An example of the duty cycle of a sensor node is shown in Fig. 4. In the example, there were i) laboratory activities from 0 seconds to 200 seconds and the last 180 seconds (Period (a) and (b)), and ii) abnormal conditions in the laboratory from 170 seconds to 770 seconds (Period (c) and (d)). The sensor node with the adaptive sensing configuration switched on 63 times and the microcontroller drew 2549 mA current for operations. Meanwhile, in the fixed-schedule dense-sampling configuration (i.e., sensing for every five seconds), the sensor node switched on 1296 times and drew 50544 mA current. The excessive power consumption is caused by the unnecessary measurements in low-risk situations (e.g. no laboratory activities). The excessive power consumption will significantly shorten the lifetime of the portable system. If a fixed-schedule sparse-sampling configuration (i.e. sensing for every 102.9 seconds) is used, the sensor node also switched on 63 times and drew 2549 mA current. However, there was a 35.8 seconds detection delay. The delay can be fatal, because a high temperature can cause fire or even explosion in the laboratory. In conclusion, the adaptive sensing mechanism allows the WSN maintains an effective measurement with low power consumption.

3.3 System Reliability

In order to evaluate the system performance, five sensor nodes have been deployed simultaneously for collecting readings of temperature and RH during the three-hour trial.

In the trial, 2715 (i.e., 543 messages per sensor node) messages have been received and analyzed. Among received messages, 2640 messages had valid temperature and RH readings, i.e., the mean system reliability is about 97.24%. Furthermore, the system reliability of each sensor nodes is summarized in Table 4. On the other hand, received signal strength indicators in Table 4 shows that Node #2 had the worse network reliability (i.e., lowest LSI). The problem may be due to the obstruction (e.g. computers, equipment, and wall structures). But in overall, all sensor nodes have been appropriately placed to provide a complete coverage as well as a open channel for communications.

4 Conclusion

This paper presents a design of a WSN for laboratory surveillance. On-site measurements show the functionality of the adaptive sensing mechanism. The measurements also verify the feasibility of applying WSNs for real-time laboratory surveillance and construction of smart laboratories. In the future, we hope that the proposed WSN can be verified and validated in a large-scale deployment [9].

References

1. Bulusu, N., Jha, S.: Wireless sensor networks. Artech House, Boston (2005)
2. Krishnamurthy, L.: Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea. In: Proc. ACM International Conf. on Embedded Networked Sensor Systems, pp. 64–75 (November 2005)
3. Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Computer Networks 52(12), 2292–2330 (2008)
4. Chipara, O., Lu, C., Bailey, T.C., Roman, G.-C.: Reliable clinical monitoring using wireless sensor networks: experiences in a step-down hospital unit. In: Proc. the ACM Conf. on Embedded Networked Sensor Systems, pp. 155–168 (November 2010)
5. Dyo, V.: Evolution and sustainability of a wildlife monitoring sensor network. In: Proc. ACM Conf. on Embedded Networked Sensor Systems, pp. 127–140 (November 2010)
6. Piho, G., Tepandi, J., Parman, M.: Towards LIMS software in global context. In: Proc. IEEE Intl. Convention on Information and Communication Technology, Electronics and Microelectronics, pp. 721–726 (December 2012)
7. Cooper, M., Ferreira, J.: Remote laboratories extending access to science and engineering curricular. IEEE Transactions on Learning Technologies 2(4), 342–353 (2009)
8. Yatim, S., Zain, M., Darus, F., Ismail, Z.: Thermal comfort in air-conditioned learning environment. In: Proc. IEEE Intl. Symp. and Exhibition in Sustainable Energy and Environment, pp. 194–197 (June 2011)
9. Shen, Z., Man, K.L., Lei, C.U., Lim, E., Choi, J.: Assuring system reliability in wireless sensor networks via verification and validation. In: Proc. IEEE Intl. SoC Design Conference, pp. 285–288 (November 2012)

S-Theory: A Unified Theory of Multi-paradigm Software Development

Danny Hughes[1], Nelly Bencomo[2], Brice Morin[3], Christophe Huygens[1],
Zhun Shen[3], and Ka Lok Man[4]

[1] IBBT-DistriNet, KU Leuven, Leuven, Belgium
{firstname.lastname}@cs.kuleuven.be
[2] INRIA Paris-Rocquencourt, Rocquencourt, France
nelly@acm.org
[3] SINTEF ICT, Oslo, Norway
brice.morin@sintef.no
[4] Xi'an-Jiaotong Liverpool University, Suzhou, China.
ka.man@xjtlu.edu.cn

Abstract. Many problems facing software engineers demand 'optimal' performance in multiple dimensions, such as computational overhead and development overhead. For these complex problems, designing an optimal solution based upon a single programming paradigm is not feasible. A more appropriate solution is to create a solution framework that embraces multiple programming paradigms, each of which is optimal for a well-defined region of the problem space. This paper proposes a theory for creating multi-paradigm software solutions that is inspired by two contributions from theoretical physics: model dependent realism and M-Theory. The proposed theoretical framework, which we call 'S-Theory', promotes the creation of actor-optimal solution frameworks, encourages technology reuse and identifies promising research directions. We use the field of sensor networks as a running example.

Keywords: M-Theory, S-Theory, multi-paradigm programming.

1 Introduction

In this paper, we propose 'S-Theory', a theoretical approach to combining multiple programming paradigms to realize optimal software solutions. S-Theory provides a methodology for analyzing problems and designing software solutions using multiple programming paradigms. A complete validation of S-Theory is beyond the scope of this paper. Instead, we provide a high-level outline of the theoretical framework for feedback from the research community.

Contemporary application scenarios require solutions that provide 'optimal' performance for multiple actors. Performance is a multi-dimensional concept with different implications at different stages of the software lifecycle. Performance at build-time entails considerations such as the man-hours required to build software, while performance at run-time focuses on issues such as CPU, memory and network requirements. In this paper, we use the problem of building Wireless Sensor Network

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 715–722, 2013.

(WSN) applications as a motivating example. While WSN problems showcase the need for multi-dimensional optimization, in our view, similar problems are encountered in many different application areas.

Building efficient node-local software for WSN platforms requires low-level abstractions that are capable of providing fine-grained control over every available byte of memory, CPU cycle and joule of battery power. Conversely, effective administration and tailoring of a large sensor network application requires higher-level abstractions that allow for reasoning at the granularity of groups of nodes or an entire WSN. As such, a range of software models operating at different scales of abstraction is required throughout the software lifecycle to serve the needs of all actors. While the Model Driven Engineering (MDE) community recognizes this problem, the cost of using multiple models has received comparatively little attention. On resource constrained WSN platforms, the use of each additional model consumes extra computational and memory resources and, in all cases, introduces communication barriers between actors who use different paradigms. The question for software engineers therefore becomes: *How to design a multi-paradigm solution that supports the needs of all actors while minimizing costs?*

Physicists have grappled with the problem of reconciling multiple theoretical models of the universe into a 'grand unified theory' for decades. Classical Mechanics is a sound model for predicting interactions between macroscopic entities travelling at low speeds. However, the predictive power of this theory breaks down at scales smaller than 10^{-9} meters. At these scales Quantum Mechanics has better predictive power. Conversely, Quantum Mechanics cannot be used to accurately predict the behavior of macroscopic entities. Model dependent realism [1] is a philosophical approach that posits that reality is best understood using a number of co-existing and complementary models, each appropriate for a different region of the problem space. M-Theory [1] builds upon model-dependent realism to unify disparate scientific theories such as Classical and Quantum Mechanics. In M-Theory, a model is valid if it has better predictive power than existing models for a specific region of the problem space, or if it has equal predictive power, but covers a greater region of the problem space. Thus, M-Theory allows for the reconciliation of multiple theoretical models in a unified theoretical framework. We propose S-Theory to provide a comparable theoretical framework for analyzing software problems. Furthermore, while physicists are limited to observing the universe, software engineers are free to redefine their 'software universe' and thus S-Theory can also be used as a tool to create optimal solutions.

The remainder of this paper is structured as follows: Section 2 provides an overview of S-Theory, the implications of which are discussed in Section 3. S-Theory is positioned against related work in Section 4. Finally, we conclude in Section 5 and outline directions for future work.

2 S-Theory

S-Theory provides a methodology to analyze problems, identify requirements, design and optimize a software solution. These elements of the S-Theory methodology are described in Sections 2.1 to 2.4 respectively.

2.1 Mapping the Problem Space

All software problems may be plotted against a common multi-dimensional *problem space*, each axis of which describes a problem characteristic such as: the phase of the software lifecycle at which concerns are introduced, network scale, requirement-dynamism, network-dynamism and timeliness. The scale of each axis may be quantitative, or qualitative depending upon the characteristics of the dimension being described. Where a dimension is considered irrelevant, for example network-dynamism in node-local problems, it may be omitted from the plot. The various dimensions of the problem space are analogous to the physical dimensions of M-Theory, such as length, breadth, depth and time. Figure 1 plots a simplified example of a WSN problem on the common problem space. For comparison, a cloud-computing problem is also plotted.

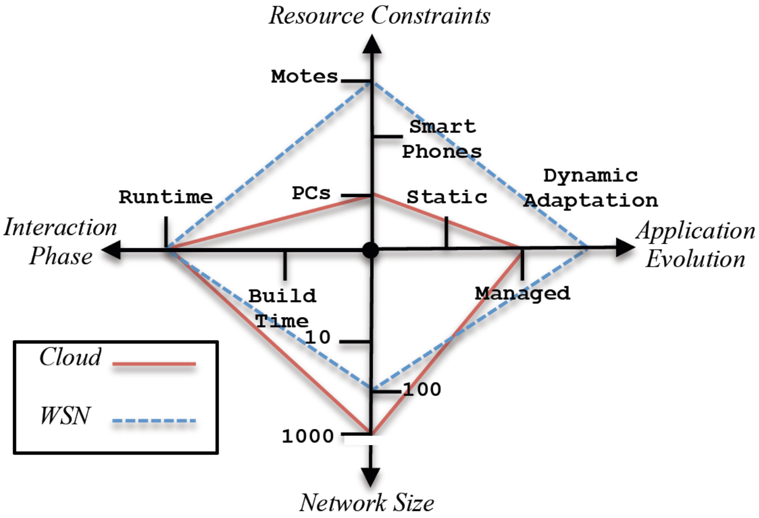


Fig. 1. Simplified Plot of WSN and Cloud Computing against a common problem space

As can be seen from Figure 1, the scale of cloud computing problems is larger than WSN, however, cloud computing has less extreme resource constraints and a simpler model of software evolution. Based upon this problem space it is possible to identify actor regions, as described in Section 2.2.

2.2 Identifying Actor Regions

For large and complex problems, each actor that is involved in developing a solution typically operates within a limited area of the total problem space, which we refer to as the *'actor region'*. In the WSN domain, Picco et al. [2] identify three actors: (i.) the 'WSN Geek', who develops embedded software and therefore requires specialized programming abstractions that focus on resource efficiency, (ii.) the 'WSN Technician' who manages and administers the network and thus requires mechanisms to manage a large-scale sensor network and (iii.) the ‘Domain Expert’ who requires simple mechanisms to integrate data from the WSN into their back-end applications. Huygens et al. [3] provide a similar set of actor roles for WSN, validating this work.

It can be seen that each actor works within a specific region of the problem space, depending on his objectives and the phase of the software lifecycle in which the actor works. For example, the WSN Geek requires abstractions that expose the low-level features of embedded sensor nodes, while the WSN technician reasons in terms of groups of nodes. Figure 2 plots actor areas for the WSN Geek and WSN Technician. For clarity and simplicity, only these two roles are plotted. Drawing from model-dependent realism, S-Theory allows each actor to create and reason about the system through the most appropriate software model.

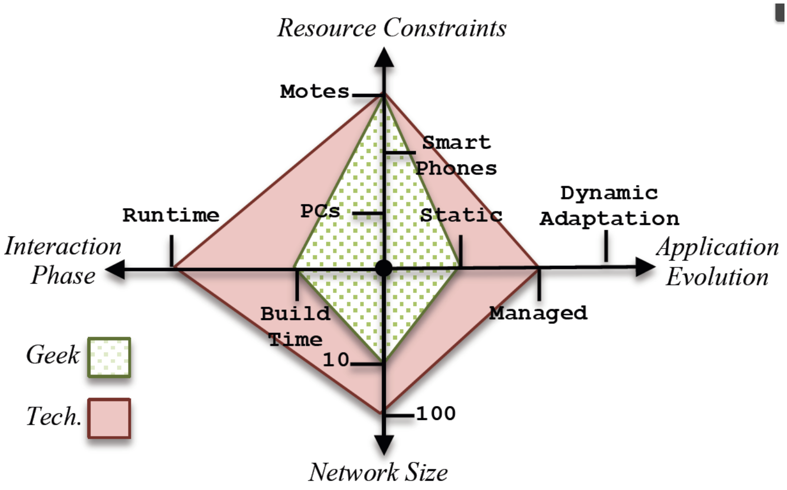


Fig. 2. WSN Geek and WSN Technician Actor Regions in the WSN Problem Space

As can be seen from Figure 2, the ‘WSN Geek’ works primarily at build-time before the system is deployed, while the ‘WSN Technician’ works at run-time to maintain and tailor system behavior. The ‘WSN Geek’ reasons at the level of an individual mote, while the ‘WSN Technician’ reasons at the level of many of motes. The ‘WSN Geek’ builds static system software, while the ‘WSN Technician’ manages software evolution. Finally, both actors must work within the resource constraints of contemporary mote platforms. Considering the sketch of the problem space provided in Figure 2, it is clear that no one programming paradigm can meet the different needs of both actors.

2.3 Building a Solution Space

A *complete solution space* for any problem should provide a set of software models that together give complete coverage of all actor regions, i.e. the collection of models should allow all actors to work at all points within their ‘actor regions’. For most problems, multiple valid solution spaces are possible, each of which has an associated *cost*, which is calculated based upon the sum costs of the constituent software models.

The costs of any software model are also multi-dimensional, including factors such as developer effort, message-passing overhead and CPU load and the prioritization of costs is dependent upon the application domain. Cost is also a rationale for including software life-cycle phase as a dimension in the problem space, as the cost of a software model is dependent on the point in the problem space at which it is applied. For example, at development time, human related costs such as development time are likely to dominate, while at runtime, resource related costs such as CPU and memory overhead are more relevant. For each software model, a cost function should be provided which transforms a position in the n-dimensional problem space to a set of software costs for each cost dimension. It should be noted that the metrics used to measure costs are defined along with the problem space as described in Section 2.1.

2.4 Optimization of the Solution Space

A software solution-space is considered *optimal* where the following four conditions are met:

1. Actors are provided with software models that cover their actor-region and that allow them to accomplish all necessary tasks at all times.
2. The solution contains no models that serve only a region external to the problem space, as this would constitute unnecessary redundancy and overhead.
3. There is no known software model that could cover an actor region at lower cost than currently used models.
4. There is no known software model that could cover two actor regions at the same cost as the current models and thereby reduce overhead and eliminate communication barriers between actors.

The problem space and solution space are expected to be dynamic and to change over time. For the problem space, dynamism is driven by the emergence of new actors and the expansion of problem space dimensions to scales that were unforeseen during the initial modeling of the problem. For the solution space, dynamism is driven by the emergence of new software models with different capabilities and costs. The solution space should therefore be continually re-optimized by applying the four rules listed above.

3 Applying S-Theory

S-Theory provides a scientific methodology for building and optimizing software solutions to complex multi-actor problems. While individual researchers may apply S-Theory as a tool to analyze and optimize their own work, this entails significant overhead in terms of generating cost-functions for 3rd party software models. In our view, the full potential of S-Theory will most effectively be realized, if it is adopted and used in a collaborative fashion by multiple research groups.

We envisage that for each application area of computer science, researchers could first use S-Theory to map their problem space. Once the key dimensions of the problem space have been mapped and accepted by the research community within an application area, actor roles and regions should be identified, discussed and refined along with the dimensions of software solution costs. This process may be viewed as a formalization of the existing pattern where new research fields are initiated by timely and influential position papers. In comparison, mapping the problem space using S-Theory is more formal and specific and provides a sound foundation for future work within a sub-domain.

Once a sub-discipline has agreed upon the dimensions of the problem space and the actor regions, appropriate software models from related application areas should be identified and reused. Where no appropriate model exists for an actor region, S-Theory identifies a clear research agenda: to develop a new model that meets the requirements of the un-served actor region. As new software models emerge, they should be evaluated against existing models based upon their cost functions and their coverage of the problem space using the rules described in Section II.D. It is likely that as a sub-discipline matures, the boundaries of the problem space will change, necessitating optimization of solution spaces.

4 Related Work

Focusing on our example application area of WSN, the necessity of creating multi-paradigm WSN solutions is supported by Picco et al. who argue for finding “the right abstractions for the right application” and “for the right developer” [2]. This requirement is also reflected in contemporary WSN software stacks. UC Berkeley provides a multi-paradigm software solution composed of: the NesC [4] component model for efficient node-local programming, the Mate VM [5] for run-time tailoring and TinyDB [6] to integrate the WSN with back-end applications. In contrast, the University of Lancaster provides OpenCOM [7] for efficient node-local programming, Open Overlays [8] for distributed (re)-configuration and Genie [9] for goal-based modeling of application behavior. KU Leuven provide yet another WSN software stack, with LooCI [10] for local and distributed programming, PMA [11] for policy-based administration and QARI [12] for goal-based enforcement of quality objectives. Each software stack provides a programming paradigm targeted at the WSN Geek, the WSN Technician and the Domain Expert roles. However, using current methods it is difficult to assess which software stacks are most optimal for a given application. S-Theory promises to rationalize divergent multi-paradigm solutions and is therefore complementary to existing multi-paradigm work in our application area.

Moving from our example application domain of WSN to consider the general case, the difficulty of building software solutions for complex applications was noted by Zadeh [15], who states that: “the complexity of a system and the precision with which it can be analyzed bear a roughly inverse relation to one another”. S-Theory provides a promising approach to mitigating Zadeh’s dilemma. By decomposing a problem space into smaller actor regions and selecting appropriate abstractions for each region, complexity is minimized and performance is maximized for each actor.

Model-Driven Engineering (MDE) is related to S-Theory in that it advocates the use of multiple domain-specific abstractions as an important complement to general purpose modeling languages to solve specific problems. Modern general purpose programming languages like Scala [14] propose advanced features to define and seamlessly integrate DSLs into the main language, making it possible to use more elegant, concise and comprehensive abstractions for certain parts of a program. Research [16] has shown that families of more focused modeling languages are more successful and practical. In contrast to MDE, S-Theory specifically considers the cost of additional abstractions. However, we view the flexible abstractions offered by MDE [17.18] as a key enabler to support S-Theory.

In comparison to model-dependent realism [1], from which we draw inspiration, S-Theory also allows for multiple valid models of a complex system, however, while model-dependent-realism only allows for multiple valid models for understanding phenomena, S-Theory also allows for multiple valid models for enacting change. While theories that deal with observable reality from fields such as physics or psychology can be considered as a subset of M-Theory [1], in our view S-Theory is not a subset of M-Theory, as S-Theory also concerns the creation of software 'realities'.

5 Conclusions and Future Work

In this paper we have presented of a theoretical framework for developing optimal software solutions using multiple programming paradigms: S-Theory. We then sketched a vision of how S-Theory could be applied by the research community. The work presented in this paper is at an early stage, and represents the first steps in realizing our vision.

Our future work will first focus on a thorough and complete validation of S-Theory in the field of WSN. Specifically, we will map the WSN problem space, develop cost models and benchmark available WSN programming paradigms using these models. Based upon this analysis, we will assemble an optimal WSN software stack from available paradigms and identify areas that are not well served by current approaches. In the longer term, we intend to evaluate the applicability of S-Theory to application domains other than WSN. To support this, we actively invite collaborations from software engineers working in any area.

References

1. Hawking, S., Mlodinow, L.: The Grand Design, pp. 85–120. Bantam Books, New York (2010)
2. Picco, G.P., Mottola, L.: Middleware for Wireless Sensor Networks: An Outlook. Journal of Internet Services and Applications 3(1), 31–39 (2011)
3. Huygens, C., Hughes, D., Lagaisse, B., Joosen, W.: Streamlining development for Networked Embedded Systems using multiple paradigms. IEEE Software 27(5), 45–52 (2010)
4. Gay, D., Levis, P., Von Behren, R., Welsh, M., Brewer, E., Culler, D.: The NesC Language: A Holistic Approach to Networked Embedded Systems. In: Proc. of the ACM confernece on Programming Language Design and Implementation, SIGPLAN PLDI 2003, San Diego, CA, USA, June 9-11, pp. 1–11 (2003)

5. Levis, P., Culler, D.: Maté: A Tiny Virtual Machine for Sensor Networks. In: Proc. of Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2002), San Jose, CA, US, October 5-9, pp. 85–95 (2002)
6. Madden, S.R., Franklin, M.J., Hellerstein, J.M., Hong, W.: TinyDB: an acquisitional query processing system for sensor networks. ACM Transactions on Database Systems 30(1), 122–173 (2005)
7. Coulson, G., Blair, G., Grace, P., Taiani, F., Joolia, A., Lee, K., Ueyama, J., Sivaharan, T.: A generic component model for building systems software. ACM Transactions on Computer Systems 26(1), 1–42 (2008)
8. Grace, P., Hughes, D., Porter, B., Blair, G., Coulson, G., Taiani, F.: Experiences with Open Overlays: A Middleware Approach to Network Heterogeneity. In: Proc. of European Conference on Computer Systems (EuroSys 2008), Glasgow, UK, March 31- April, pp. 123–136 (2008)
9. Goldsby, H.J., Sawyer, P., Bencomo, N., Hughes, D., Cheng, B.H.C.: Goal-Based Modeling of Dynamically Adaptive System Requirements. In: Proc. of International Conference on Engineering of Computer-Based Systems (ECBS 2008), Belfast, Northern Ireland, March 31-April 1, pp. 36–45 (2008)
10. Hughes, D., Thoelen, K., Maerien, J., Matthys, N., Del Cid, J., Horré, W., Huygens, C., Michiels, S., Joosen, W.: LooCI: the Loosely-coupled Component Infrastructure. To appear in Proc. of 11th International Symposium on Network Computing and Applications (NCA 2012), Cambridge, MA, US, August 23-25 (2012)
11. Horré, W., Hughes, D., Michiels, S., Joosen, W.: Advanced sensor network software deployment using application-level quality goals. Journal of Software 6(4), 528–535 (2011)
12. Matthys, N., Huygens, C., Hughes, D., Ueyama, J., Michiels, S., Joosen, W.: Policy-driven tailoring of sensor networks. In: Par, G., Morrow, P. (eds.) S-CUBE 2010. LNICST, vol. 57, pp. 20–35. Springer, Heidelberg (2011)
13. Cheng, B., Atlee, J.: Research Direction in requirements Engineering. In: Proc. of Future of Software Engineering (FOSE 2007), Minneapolis, MN, USA, May 20-26, pp. 285–303 (2007)
14. Odersky, M., Spoon, L., Venners, B.: Programming in Scala: A Comprehensive Step-by-step Guide. Artima Inc. (2008)
15. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning. Information Sciences, Part I: 8, 199–249; Part II: 8, 301–357; Part III: 9, pp. 43–80
16. Hutchinson, J., Whittle, J., Rouncefield, M., Kristoffersen, S.: Empirical assessment of MDE in industry. In: Proc. of 33rd International Conference on Software Engineering (ICSE 2011), Waikiki, Hawaii, US, May 21-28, pp. 471–480 (2011)

Design of J-VTS Middleware Based on IVEF Protocol*

Taekyeong Kang[1,2] and Namje Park[1,2,**]

[1] Science Technology in Society Research Center (STSRC), Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
[2] Department of Computer Education, Teachers College, Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
{ktg,namjepark}@jejunu.ac.kr

Abstract. The IVEF service is the draft standard designed for exchange of information on sea traffic between the vessel traffic systems and between the vessels. Standardization of this service is under way as a part of the next-generation navigation system, called e-Navigation. The International Association of Lighthouse Authorities (IALA) suggests, on its recommendation V-145, the IVEF service model and the protocol for provisioning of this service. But the detailed configuration of this service must be designed by the users. This paper suggests, based on the basic service model and protocol provided in the recommendation V-145, the design of the J-VTS middleware which will facilitate exchange of information on sea traffic.

Keywords: VTS Security, e-Navigation, IVEF, VTS, Middleware Platform, J-VTS, Inter-VTS Data Exchange Format.

1 Introduction

The IVEF service is the draft standard designed for exchange of information on sea traffic between the vessel traffic systems and between the vessels. Standardization of this service is under way as a part of the next-generation navigation system, called e-Navigation. The International Association of Lighthouse Authorities (IALA) suggests, on its recommendation V-145, the IVEF service model and the protocol for provisioning of this service. But the detailed configuration of this service must be designed by the users. IVEF service is aimed at establishing a common framework to ensure exchange of ship information between VTS centers gained via automatic ship identification device, CCTV, radar system and other devices. For this, a number of models such as data model, interaction model and security model are being proposed and XML-type basic protocol to deliver them has been distributed. Robust exchange of marine traffic information between VTS centers through IVEF service will enable related authorities to identify location of domestic and international ships in the coastal waters real-time and predict expected sea route, thus ensure effective sea route control and pre-emptive response to potential disasters or accidents. Furthermore, an

* This research was supported by the 2013 scientific promotion program funded by Jeju National University.

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 723–729, 2013.

international collaboration system between countries that use standard protocol assists effective response to threats from pirates.

This paper suggests, based on the basic service model and protocol provided in the recommendation V-145, the design of the J-VTS middleware which will facilitate exchange of information on sea traffic. The J-VTS middleware consists of various components for providing the IVEF service and for processing the IVEF message protocols. The vessel traffic systems and the vessels corresponding to upper-layer applications may use the IVEF service with the functions provided by the J-VTS middleware, and the services are designed to be accessed according to the security level of users.

2 IVEF Protocol in VTS Service

IVEF service is a server/client model serving as a protocol to exchange traffic information between VTS systems. Its development based on open source is underway by IALA and its protocol and sample program can be checked by downloading SDK in OpenIVEF website [2]. Basic actions to provide service between server/client take three steps as follows. In the first step, a client requests server certification and receives log-in reply if he/she is a legitimate user. In the second step, the server provides a certain service for the specific user only if it has such service. If it does not offer such service, it provides a basic service defined in the standard called BIS (Basic IVEF Services). In this step, the client can designate area of interest, data renewal period or data form based on his/her preference. In the third step, the client sends log-out message to the server in order to end use of IVEF service. Since the server does not give a separate reply on the log-out message, all the client has to do is just cancel access to server when he/she sends the message [4].

IALA, which is the basic protocol to provide IVEF service between VTS centers, defines nine messages as shown in Table 1. Definition of these messages is composed of XML-type schema and all messages are composed of sub-elements of MSG_IVEF, which is the most significant element. Message of each sub-element also has its own sub-elements based on message characteristics. IVEF messages are broadly divided into control information message and real-time information message. The former consists of user certification and termination, service request to the server and its reply message and others to provide information on server status. The latter controls ship's current location, expected route, destination port and other physical information in an object data. Main purpose of object data is to exchange the following information.

- Real-time Tracking positions
- Static Vessel Information
- Voyage related Information

Table 1. IVEF Interface Message Define

Message	From	To
Control Information Message		
Login	Client	Server
Login Response	Server	Client
Logout	Client	Server
Ping	Both	Both
Pong	Both	Both
Service Request	Client	Server
Service Request Response	Server	Client
Service Status	Server	Client
Real-time Information Message		
Object Data	Server	User

3 Proposed J-VTS Middleware

This paper suggests the J-VTS (Jeju National University - Vessel Traffic System) middleware structure to implement the IVEF service model. J-VTS consists of components and functions which abstract the IVEF protocol and the IVEF service, enabling the vessel traffic system and vessels to easily use the IVEF service. The J-VTS middleware is designed in consideration of all models recommended in the IALA recommendation V-145: the data model, the security model, the interface model, the interaction model, the test model and the admin model. The J-VTS middleware has additional components; one analyzes and creates the IVEF messages in consideration of the IVEF message protocol, and the other provides specific actions in accordance with the result of the analysis. Figure 2 illustrates the overall architecture of the J-VTS middleware. J-VTS is divided into three layers. The first layer is the application layer, which is corresponding to the vessel traffic system or vessels in service. The second layer is the J-VTS standard function layer. It consists of functions which, on the application layer, communicate with major components of the J-VTS middleware and control the related components. The third layer consists of 12 components which provide the IVEF service, send/receive messages under the IVEF protocol base and collect/analyze various sensor data.

3.1 Description on Client Implementation Codes

A client program is written with Android-based Java. As for network, it is written based on the Java TCP/IP protocol stack, and the UI components provided by the Android platform for user interface. The main codes for implementation of the client are as follows:

1) Developed on the Android Gingerbread platform.
2) MainActivity shows the protocol list.

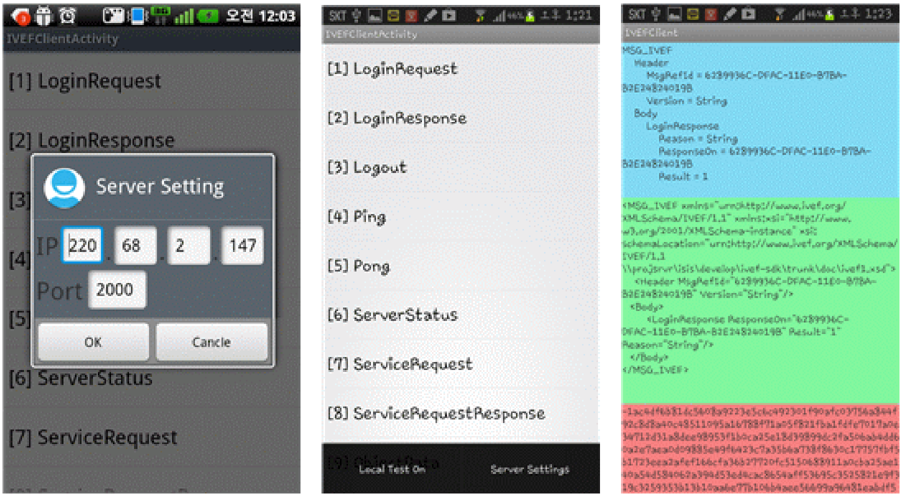
3) The thread that requests messages from the server through the socket communication, and extracts useful information from the received xml file.

4) ViewerActivity showing the xml file and useful information on the display.

5) Actual V-145 recommended library

6) The XmlMsgCreator code creating the xml file based on the recommended library.

7) The handler processing the library-parsed codes



IP 220 . 68 . 2 . 147

Port 2000

```
MSG_IVEF
  Header
    MsgRefId = 6289936C-DFAC-11E0-B7BA-
B2E248240198
    Version = String
  Body
    LoginResponse
      Reason = String
      ResponseOn = 6289936C-DFAC-11E0-B7BA-
B2E248240198
      Result = 1
```

```
<MSG_IVEF xmlns="urn:http://www.ivef.org/
XMLSchema/IVEF/1.1" xmlns:xsi="http://www.
w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="urn:http://www.ivef.org/XMLSchema/
IVEF/1.1
\\projsrvr\isis\develop\ivef-sdk\trunk\doc\ivef1.xsd">
  <Header MsgRefId="6289936C-DFAC-11E0-B7BA-
B2E248240198" Version="String"/>
  <Body>
    <LoginResponse ResponseOn="6289936C-
DFAC-11E0-B7BA-B2E248240198" Result="1"
Reason="String"/>
  </Body>
</MSG_IVEF>
```

```
-1ac4df6b81dc5b0f8a9223e5cbc492301f90afc037f56a8444f
92c8d8a40c4951109 5a16f88f71a05f821fba1fdfe7017a0e
34f712d31a8dee98953f1b4ca25e18d398 99dc3fa506ab4dd6
0a2e7aea0d09885e49fb423c7a35b6a7384f8630c17757fb45
b1723eea2afef166cfa3663 7720fc51506 88911a0cba25ae1
40a54d5840 62a394d53ed4cac8654aff53695c3525821e9f3
19c3259353613b10aa6e77b10bb4aee56499a96481eabdf5
```

Fig. 1. IVEF's Client Menu

3.2 Description on the Server Implementation Codes

The server program is written with the Java language. The daemon runs with the TCP/IP protocol stack provided by Java, and the user interface is implemented with Java Swing. If the server program is started initially and a user starts the server, a thread is created, waiting for access of a client. This thread uses the protocol developed in this project as the library, and performs the functions in accordance with this protocol. The main components for implementation of server and the implementation details are as follows:

1) Using the Java-based TCP/IP protocol stack

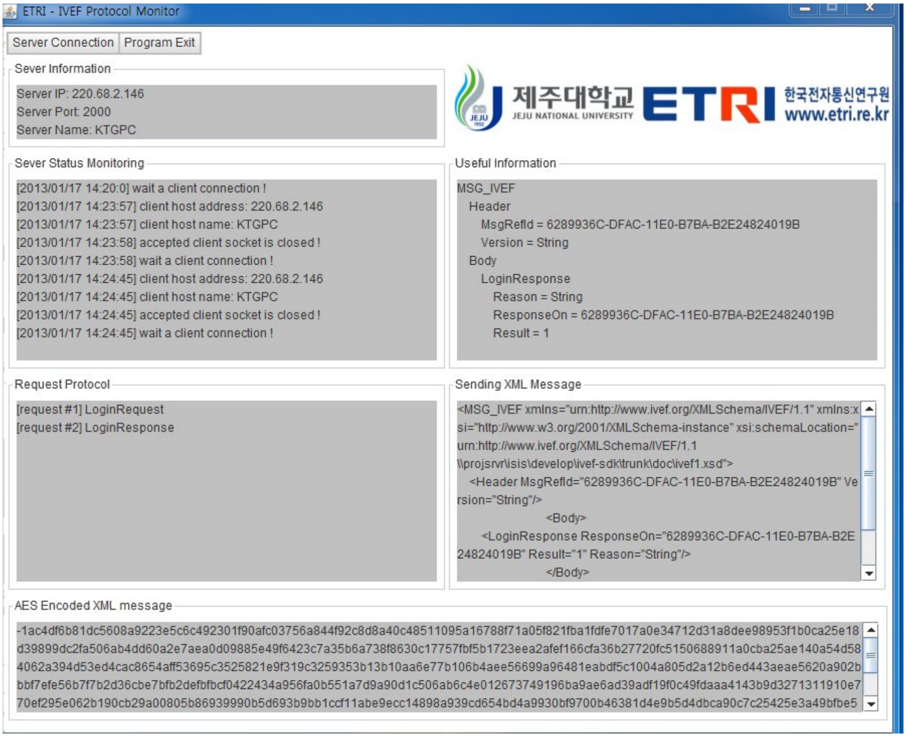
2) Using the Java Swing-based UI framework

3) Using the v-145 recommended protocol library developed in the project

4) The thread creating and transmitting messages requested by the clients through the socket communication.

5) The handler code processing the final XML-parsed result

6) The message creator code encoding messages into the recommended XML files



```
Server IP: 220.68.2.146
Server Port: 2000
Server Name: KTGPC
```

```
[2013/01/17 14:20:0] wait a client connection !
[2013/01/17 14:23:57] client host address: 220.68.2.146
[2013/01/17 14:23:57] client host name: KTGPC
[2013/01/17 14:23:58] accepted client socket is closed !
[2013/01/17 14:23:58] wait a client connection !
[2013/01/17 14:24:45] client host address: 220.68.2.146
[2013/01/17 14:24:45] client host name: KTGPC
[2013/01/17 14:24:45] accepted client socket is closed !
[2013/01/17 14:24:45] wait a client connection !
```

```
MSG_IVEF
  Header
    MsgRefId = 6289936C-DFAC-11E0-B7BA-B2E24824019B
    Version = String
  Body
    LoginResponse
      Reason = String
      ResponseOn = 6289936C-DFAC-11E0-B7BA-B2E24824019B
      Result = 1
```

```
[request #1] LoginRequest
[request #2] LoginResponse
```

```
<MSG_IVEF xmlns="urn:http://www.ivef.org/XMLSchema/IVEF/1.1" xmlns:x
si="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="
urn:http://www.ivef.org/XMLSchema/IVEF/1.1
\\projsrvr\isis\develop\ivef-sdk\trunk\doc\ivef1.xsd">
   <Header MsgRefId="6289936C-DFAC-11E0-B7BA-B2E24824019B" Ve
rsion="String"/>
                    <Body>
      <LoginResponse ResponseOn="6289936C-DFAC-11E0-B7BA-B2E
24824019B" Result="1" Reason="String"/>
                    </Body>
```

-1ac4df6b81dc5608a9223e5c6c492301f90afc03756a844f92c8d8a40c48511095a16788f71a05f821fba1fdfe7017a0e34712d31a8dee98953f1b0ca25e18
d39899dc2fa506ab4dd60a2e7aea0d09885e49f6423c7a35b6a738f8630c17757fbf5b1723eea2afef166cfa36b27720fc5150688911a0cba25ae140a54d58
4062a394d53ed4cac8654aff53695c3525821e9f319c3259353b13b10aa6e77b106b4aee56699a96481eabdf5c1004a805d2a12b6ed443aeae5620a902b
bbf7efe56b7f7b2d36cbe7bfb2defbfbcf0422434a956fa0b551a7d9a90d1c506ab6c4e012673749196ba9ae6ad39adf19f0c49fdaaa4143b9d3271311910e7
70ef295e062b190cb29a00805b86939990b5d693b9bb1ccf11abe9ecc14898a939cd654bd4a9930bf9700b46381d4e9b5d4dbca90c7c25425e3a49bfbe5

Fig. 2. IVEF's Server operation

4 Conclusion

This paper suggests J-VTS as the middleware that provides the IVEF service. J-VTS is the middleware that provides the IVEF service between the vessel traffic systems and between vessels. It is designed to provide, using various functions provided by J-VTS, functions, such as IVEF service provisioning, data security, protocol test, incorporation of data and incorporation of sea traffic image. The overall structure design of the middleware is finished, but detailed design of each component and the actual implementation work have not been done. Therefore, further study is required to finish the detailed design of the J-VTS middleware suggested in this study, to verify applicability, and to examine the improvement.

Acknowledgments. This research was supported by the 2013 scientific promotion program funded by Jeju National University. The corresponding author is prof. namje Park.

References

1. IALA Recommendation V-145 on the Inter-VTS Exchange Format(IVEF) Service (2011)
2. `http://en.wikipedia.org/wiki/E-Navigation`
3. OpenIVEF, `http://www.openivef.org`

4. International Association of Lighthouses and Aids-to-Navigation Authorities (IALA). Interface Control Document for IVEF, release 0.1.7
5. A Security Architecture of the inter-VTS System for shore side collaboration of e-Navigation (2012)
6. Park, N., Kwak, J., Kim, S., Won, D., Kim, H.: WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 741–748. Springer, Heidelberg (2006)
7. Park, N.: Security scheme for managing a large quantity of individual information in RFID environment. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) ICICA 2010. CCIS, vol. 106, pp. 72–79. Springer, Heidelberg (2010)
8. Park, N.: Secure UHF/HF Dual-Band RFID: Strategic Framework Approaches and Application Solutions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 488–496. Springer, Heidelberg (2011)
9. Park, N.: Implementation of Terminal Middleware Platform for Mobile RFID computing. International Journal of Ad Hoc and Ubiquitous Computing 8(4), 205–219 (2011)
10. Park, N., Kim, Y.: Harmful Adult Multimedia Contents Filtering Method in Mobile RFID Service Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 193–202. Springer, Heidelberg (2010)
11. Park, N., Song, Y.: AONT Encryption Based Application Data Management in Mobile RFID Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 142–152. Springer, Heidelberg (2010)
12. Park, N.: Customized Healthcare Infrastructure Using Privacy Weight Level Based on Smart Device. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. CCIS, vol. 206, pp. 467–474. Springer, Heidelberg (2011)
13. Park, N.: Secure Data Access Control Scheme Using Type-Based Re-encryption in Cloud Environment. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 319–327. Springer, Heidelberg (2011)
14. Park, N., Song, Y.: Secure RFID Application Data Management Using All-Or-Nothing Transform Encryption. In: Pandurangan, G., Anil Kumar, V.S., Ming, G., Liu, Y., Li, Y. (eds.) WASA 2010. LNCS, vol. 6221, pp. 245–252. Springer, Heidelberg (2010)
15. Park, N.: The Implementation of Open Embedded S/W Platform for Secure Mobile RFID Reader. The Journal of Korea Information and Communications Society 35(5), 785–793 (2010)
16. Kim, Y., Park, N.: Development and Application of STEAM Teaching Model Based on the Rube Goldberg's Invention. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) CSA 2012. LNEE, vol. 203, pp. 693–698. Springer, Heidelberg (2012)
17. Park, N., Cho, S., Kim, B., Lee, B., Won, D.: Security Enhancement of User Authentication Scheme Using IVEF in Vessel Traffic Service System. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) CSA 2012. LNEE, vol. 203, pp. 699–705. Springer, Heidelberg (2012)
18. Kim, K., Kim, B.-D., Lee, B., Park, N.: Design and Implementation of IVEF Protocol Using Wireless Communication on Android Mobile Platform. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES3 2012. CCIS, vol. 339, pp. 94–100. Springer, Heidelberg (2012)

19. Ko, Y., An, J., Park, N.: Development of Computer, Math, Art Convergence Education Lesson Plans Based on Smart Grid Technology. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES3 2012. CCIS, vol. 339, pp. 109–114. Springer, Heidelberg (2012)
20. Kim, Y., Park, N.: The Effect of STEAM Education on Elementary School Student's Creativity Improvement. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES3 2012. CCIS, vol. 339, pp. 115–121. Springer, Heidelberg (2012)
21. Lee, J.W., Park, N.: Individual Information Protection in Smart Grid. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES3 2012. CCIS, vol. 339, pp. 153–159. Springer, Heidelberg (2012)

On the Use of a Hash Function in a 3-Party Password-Based Authenticated Key Exchange Protocol*

Youngsook Lee[1] and Dongho Won[2,**]

[1] Department of Cyber Investigation Police, Howon University, Korea
ysooklee@howon.ac.kr
[2] Department of Computer Engineering, Sungkyunkwan University, Korea
dhwon@security.re.kr

Abstract. This paper is concerned with the security of a three-party password-authenticated key exchange protocol presented by Abdalla and Pointcheval in FC'05. Abdalla and Pointcheval's protocol makes use of a hash function F whose outputs are elements of a cyclic group G of prime order. Such a hash function F can be constructed from a typical hash function in various ways. In this paper, we consider the case that $F(\cdot) = g^{\mathrm{h}(\cdot)}$, where g is an arbitrary generator of G and h is a hash function such as SHA-1 and MD5. Our result is that such a construction of F immediately leads to the vulnerability of the Abdalla-Pointcheval protocol to an off-line dictionary attack. We also show how to address this weakness of the protocol.

Keywords: Key exchange protocol, hash function, password, dictionary attack.

1 Introduction

Password-authenticated key exchange (PAKE) protocols are fundamental primitives for securing distributed systems where communications are taking place through public networks. Such password-based protocols allow two communicating parties to generate a cryptographic *session key* using their easy-to-remember *passwords*, and thereby to establish a secure communication channel over a public insecure network. Despite all the work conducted over the last two decades, the design of secure PAKE protocols is still non-trivial [7, 1, 9, 10]. In particular, the notorious *dictionary attacks* have always been a major security concern in designing PAKE protocols. Dictionary attacks are often classified into two types: on-line dictionary attacks and off-line dictionary attacks. In an on-line dictionary attack, each password guess is checked in a new run of the protocol, whereas in an off-line dictionary attack [7, 1, 9], password guesses are checked off-line by an automated computer program. Therefore, on-line dictionary attacks are not quite practical whereas off-line dictionary attacks are practical enough to be exploited by adversaries and must be prevented.

* This work was supported by Howon University in 2013.

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 730–736, 2013.

In this work, we show that the three-party PAKE protocol presented by Abdalla and Pointcheval [3] in FC 2005 may be vulnerable to an off-line dictionary on how the hash function used in the protocol is instantiated. In the three-party setting, each party, commonly called a client, registers their individual password with a trusted server and must keep the password private from all other third parties including other registered clients. This means that when two clients run a three-party PAKE protocol, the password of one client must be protected against the other client who can legitimately obtain the session key [2, 11, 12]. This situation makes it more difficult to design a three-party PAKE protocol secure against off-line dictionary attacks. Abdalla and Pointcheval's three-party PAKE protocol, which we denote by AP-3PAKE, features many merits; it is very simple and efficient, and is authenticated using passwords only. The AP-3PAKE protocol makes use of a hash function F whose outputs are elements of a cyclic group G of prime order. Such a hash function F can be constructed from a typical hash function in various ways. In this paper, we investigate the security of AP-3PAKE in the case that $F(\ \cdot\ ) = g^{\mathrm{h}(\cdot)}$, where g is an arbitrary generator of G and h is a hash function such as SHA-1 and MD5. Our result is that such a construction of F immediately leads to the vulnerability of AP-3PAKE to an off-line dictionary attack. We also show how to address this weakness of AP-3PAKE.

2 Abdalla and Pointcheval' 3-Party PAKE Protocol

The AP-3PAKE protocol [3] is based on the password-based key exchange protocols of [4, 8, 6], which in turn are based on the encrypted key exchange of Bellovin and Merritt [5]. The protocol runs among the three participants: the authentication server S and two clients A and B. The server S assists the clients A and B in establishing a session key by providing them with a central authentication service. Let pw_A and pw_B be the passwords of A and B, respectively. Each client holds their individual password shared securely with the authentication server S. The public system parameters of the protocol are:

- A large cyclic group G with prime order q and an arbitrary fixed generator g of the group G.
- A hash function H which outputs l-bit strings. Here, l is a security parameter representing the length of session keys. H is modeled as a random oracle.
- Two hash functions F and G which outputs the elements of the cyclic group G. F and G are both modeled as random oracles.

The AP-3PAKE protocol works as follows:

1. Client A chooses a random $x \in \mathrm{Z}_q$and computes $X = g^x$, $pw_{A,1} = F\ (A,\ B,\ pw_A)$ and $X^* = X \cdot pw_{A,1}$. Then A sends X^* to the server S.
2. Similarly, client B chooses a random $y \in \mathrm{Z}_q$ and computes $Y = g^y$, $pw_{B,1} = F\ (A,\ B,\ pw_B)$ and $Y^* = Y \cdot pw_{B,1}$. Then B sends Y^* to S.

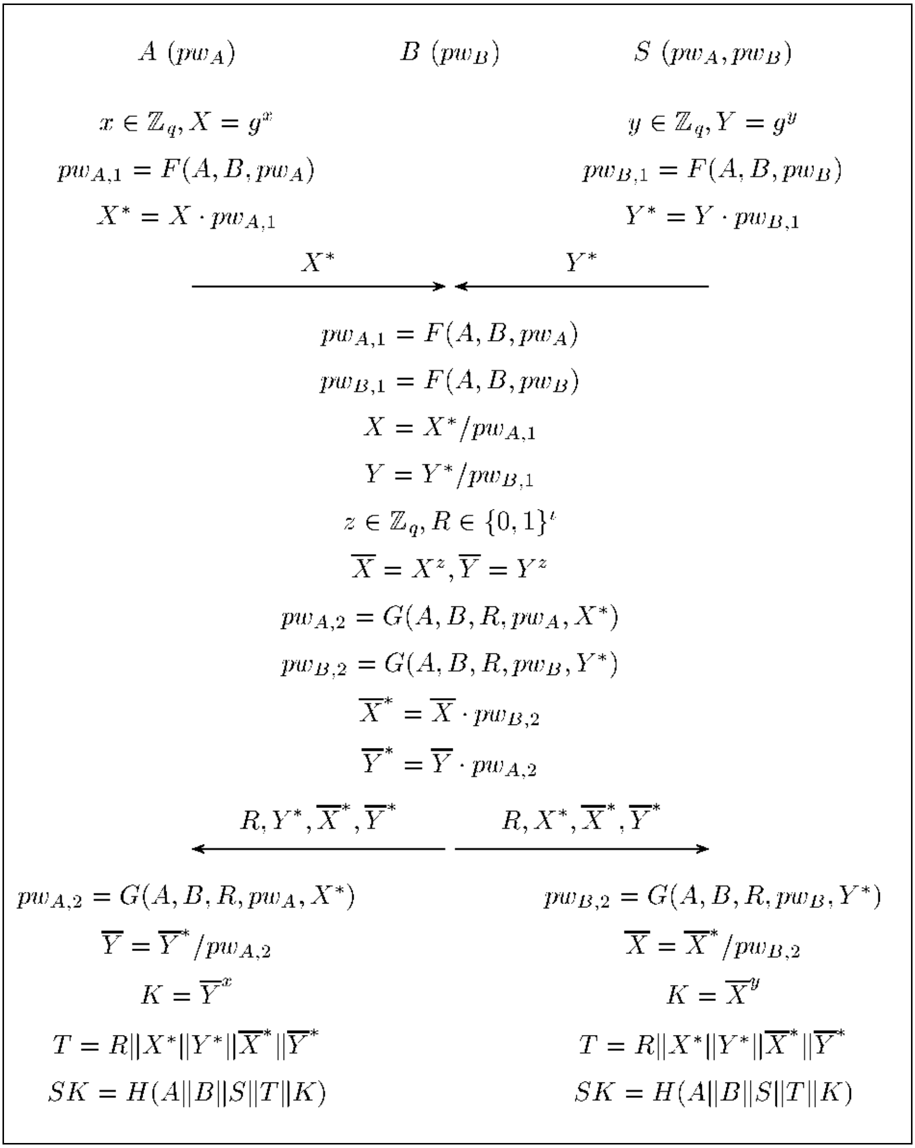

$A\ (pw_A)$
$B\ (pw_B)$
$S\ (pw_A, pw_B)$
$x \in \mathbb{Z}_q, X = g^x$
$pw_{A,1} = F(A, B, pw_A)$
$X^* = X \cdot pw_{A,1}$
$y \in \mathbb{Z}_q, Y = g^y$
$pw_{B,1} = F(A, B, pw_B)$
$Y^* = Y \cdot pw_{B,1}$
X^*
Y^*
$pw_{A,1} = F(A, B, pw_A)$
$pw_{B,1} = F(A, B, pw_B)$
$X = X^*/pw_{A,1}$
$Y = Y^*/pw_{B,1}$
$z \in \mathbb{Z}_q, R \in \{0,1\}^l$
$\overline{X} = X^z, \overline{Y} = Y^z$
$pw_{A,2} = G(A, B, R, pw_A, X^*)$
$pw_{B,2} = G(A, B, R, pw_B, Y^*)$
$\overline{X}^* = \overline{X} \cdot pw_{B,2}$
$\overline{Y}^* = \overline{Y} \cdot pw_{A,2}$
$R, Y^*, \overline{X}^*, \overline{Y}^*$
$R, X^*, \overline{X}^*, \overline{Y}^*$
$pw_{A,2} = G(A, B, R, pw_A, X^*)$
$\overline{Y} = \overline{Y}^*/pw_{A,2}$
$K = \overline{Y}^x$
$T = R\|X^*\|Y^*\|\overline{X}^*\|\overline{Y}^*$
$SK = H(A\|B\|S\|T\|K)$
$pw_{B,2} = G(A, B, R, pw_B, Y^*)$
$\overline{X} = \overline{X}^*/pw_{B,2}$
$K = \overline{X}^y$
$T = R\|X^*\|Y^*\|\overline{X}^*\|\overline{Y}^*$
$SK = H(A\|B\|S\|T\|K)$

Fig. 1. Abdalla and Pointcheval's three-party PAKE protocol

3. After receiving X^* and Y^*, S first recovers X and Y by computing $X = X^* /F$ (A, B, pw_A) and $Y = Y^*/F\ (A, B, pw_B)$. Next, S selects a random element $z \in$ Zq and a random string $R \in \{0,1\}^l$, where l is a security parameter which determines the bit-length of R. S then computes.

$$\begin{aligned}\bar{X} &= X^z,\\ \bar{Y} &= Y^z,\\ pw_{A,2} &= G\,(A,\,B,\,R,\,pw_A, X^*),\\ pw_{B,2} &= G\,(A,\,B,\,R,\,pw_B, Y^*),\\ \bar{X}^* &= \mathrm{X} \cdot pw_{B,2},\\ \bar{Y}^* &= \mathrm{Y} \cdot pw_{A,2},\end{aligned}$$

and sends <R, $Y^*, \bar{X}^*$, $\bar{Y}^*$> and <R, $X^*, \bar{X}^*$, $\bar{Y}^*$> to A and B, respectively.

4. Upon receiving <R, $Y^*, \bar{X}^*$, $\bar{Y}^*$> from S, A computes

$$\begin{aligned}pw_{A,2} &= \mathrm{G}\ (\mathrm{A, B, R}, pw_A, X^*\ ),\\ \bar{Y} &= \left(\frac{\bar{Y}^*}{pw_{A,2}}\right),\\ \mathrm{K} &= \bar{Y}^x.\end{aligned}$$

A then defines the transcript $T = R||X^*||Y^*||\bar{X}^*||\bar{Y}^*$and computes the session key $SK = \mathrm{H}(A||B||S||T||K)$.

5. Upon receiving <R, $X^*, \bar{X}^*$, $\bar{Y}^*$> from S, B computes

$$\begin{aligned}pw_{B,2} &= \mathrm{G}\ (\mathrm{A, B, R}, pw_B, Y^*),\\ \bar{X} &= \left(\frac{\bar{X}^*}{pw_{B,2}}\right),\\ \mathrm{K} &= \bar{X}^y.\end{aligned}$$

B then defines the transcript $T = R\big||X^*|\big|Y^*\big||\bar{X}^*|\big|\bar{Y}^*$ and computes the session key $SK = \mathrm{H}(A||B||S||T||K)$.

The correctness of AP-3PAKE can be verified from the equations

$$\begin{aligned}\mathrm{K} &= \left(\frac{\bar{X}^*}{pw_{B,2}}\right)^y\\ &= \left(\frac{\bar{\mathrm{X}} \cdot pw_{B,2}}{pw_{B,2}}\right)^y\\ &= g^{xyz}\end{aligned}$$

and

$$\begin{aligned}\mathrm{K} &= \left(\frac{\bar{Y}^*}{pw_{A,2}}\right)^x\\ &= \left(\frac{\bar{\mathrm{Y}} \cdot pw_{A,2}}{pw_{A,2}}\right)^x\\ &= g^{xyz}\end{aligned}$$

As can be easily seen from Fig. 1, the AP-3PAKE protocol takes two rounds of communications.

3 A Bad Instantiation of the Hash Function

Let's consider the case that the hash function *F*, whose outputs are elements of G, is defined as $F(\ \cdot\ ) = g^{h(\cdot)}$, where *h* is a typical cryptographic hash function such as SHA-1 and MD5. As shown below, such an instantiation of *F* leads to the vulnerability of AP-3PAKE to an off-line dictionary attack.

Assume that $F(\ \cdot\ ) = g^{h(\cdot)}$ and thus, $pw_{A,1}= g^{h(A,B,pw_A)}$ and $pw_{B,1} = g^{h(A,B,pw_B)}$. Assume also that B is a malicious client, and wants to find out the password of client A. Then B can mount an off-line dictionary attack against A's password as follows:

Phase 1. In this first phase, the attacker B runs the protocol with the server S while playing dual roles of B itself and the victim A.

1. B selects two random numbers x, $y \in Z_q$ and computes X^* and Y^* as

$$\begin{aligned} X^* &= g^x, \\ Y^* &= g^y \cdot pw_{B,1} \\ &= g^y \cdot g^{h(A,B,pw_B)} \end{aligned}$$

Then, B sends X^* to S as if it is from A while sending Y^* (to S) as its own message.

2. S will send <R, Y^*, $\bar{X}^*$, $\bar{Y}^*$> and <R, X^*, $\bar{X}^*$, $\bar{Y}^*$> respectively to A and B in response to X^* and Y^*. B intercepts the message <R, Y^*, $\bar{X}^*$, $\bar{Y}^*$>.

Notice here that $\bar{X}^*$ is set equal to $g^{(x-h(A,B,pw_A))z} \cdot pw_{B,2}$ because S computes it as

$$\begin{aligned} \bar{X}^* &= \bar{X} \cdot pw_{B,2} \\ &= X^z \cdot pw_{B,2} \\ &= \left(\frac{X^*}{pw_{A,1}}\right)^z \cdot pw_{B,2} \\ &= \left(\frac{g^x}{g^{h(A,B,pw_A)}}\right)^z \cdot pw_{B,2} \\ &= g^{(x-h(A,B,pw_A))z} \cdot pw_{B,2} \end{aligned}$$

But, $\bar{Y}^*$ is set equal to $g^{yz} \cdot pw_{A,2}$ as in an honest execution of the protocol.

Phase 2. Using $\bar{X}^*$ and $\bar{Y}^*$ obtained in Phase 1, B now guesses possible passwords and checks them for correctness.

1. First, B computes $pw_{B,2} = G(A, B, R, pw_B, Y^*)$ and

$$\begin{aligned} K &= \left(\frac{\bar{X}^*}{pw_{B,2}}\right)^y \\ &= \left(\frac{g^{(x-h(A,B,pw_A))z} \cdot pw_{B,2}}{pw_{B,2}}\right)^y \\ &= \mathrm{g}^{(x-h(A,B,pw_A))yz}. \end{aligned}$$

2. Next, B makes a guess pw'_A for the password pw_A and computes $pw'_{A,2}= G(A, B, R, pw'_A, X^*)$ and

$$\begin{aligned} K' &= \left(\frac{\bar{Y}^*}{pw'_{A,2}}\right)^{(x-h(A,B,pw'_A))} \\ &= \left(\frac{g^{yz} \cdot pw_{A,2}}{pw'_{A,2}}\right)^{(x-h(A,B,pw'_A))}. \end{aligned}$$

3. B verifies the correctness of pw'_A by checking that K is equal to K'. Note that if pw'_A and pw_A are equal, then the equation $K = K'$ ought to be satisfied.

4. B repeats steps 2 and 3 of this phase until a correct password is found.

The off-line dictionary attack described above can be mounted by any client against any other clients and does not even require the participation of the victim. Notice in the attack that the steps for verifying password guesses can be performed in an off-line manner by an automated program. The existence of the attack means that the security of AP-3PAKE depends on the instantiation of the hash function *F*. In particular, the hash function *F* must not be instantiated as $F(\ \cdot\ ) = g^{h(\cdot)}$, where h is a wildly used hash function like SHA-1 and MD5. Otherwise, the protocol cannot guarantee the security of clients' passwords.

However, we stress that our attack does not work if the hash function *F* is instantiated with the construction suggested for the PAK suite [8].

4 An Improved Protocol

In this section we improve the AP-3PAKE protocol to make it immune from off-line dictionary attacks. Our improved protocol makes use of a block cipher *E* mapping the elements of G to G. Let $\{0,1\}^n$ be the space of keys for the cipher. Then each key $k \in \{0,1\}^n$ determines a permutation $E_k = E\ (k,\ \ \cdot)$ on the cyclic group G. Let $D_k$denote the inverse permutation of E_k. We redefine the hash function F: $\{0,1\}^* \rightarrow \{0,1\}^n$ whose outputs will be used as keys for the cipher.
Our improved protocol works as follows:

1. Client *A* chooses a random $x \in \mathbb{Z}_q$ and computes $X = g^x$, $k_{A,1} = F\ (A,\ B, pw_A)$ and $X^* = E_{k_{A,1}}\ (X)$. Then *A* sends X^* to the server *S*.
2. Client *B* chooses a random $y \in \mathbb{Z}_q$and computes $Y = g^y$, $k_{B,1} = F\ (A\ B, pw_B)$ and $Y^* = E_{k_{B,1}}\ (Y)$. Then *B* sends Y^* to *S*.
3. Upon receiving $X^*$and Y^*, *S* first recovers *X* and *Y* by computing $X = D_{k_{A,1}}\ (X^*)$ and $Y = D_{k_{B,1}}\ (Y^*)$ where $k_{A,1}$ and $k_{B,1}$ are as computed above. Next, *S* selects a random element $z \in \mathbb{Z}_q$ and computes

$$\bar{X} = X^z,$$
$$\bar{Y} = Y^z$$
$$k_{A,2} = F\ (A,\ B,\ pw_{A,}\ X^*),$$
$$k_{B,2} = F\ (A,\ B,\ pw_{B,}\ Y^*),$$
$$\bar{X}^* = E_{k_{B,2}}\ (\bar{X}),$$
$$\bar{Y}^* = E_{k_{A,2}}\ (\bar{Y}).$$

Then *S* sends <Y^*, $\bar{X}^*$, $\bar{Y}^*$> and < X^*, $\bar{X}^*$, $\bar{Y}^*$> to *A* and *B*, respectively.
4. After receiving <Y^*, $\bar{X}^*$, $\bar{Y}^*$> from *S*, *A* computes

$$k_{A,2} = F\ (A,\ B, pw_{A,}\ X^*),$$
$$\bar{Y} = D_{k_{A,2}}\ (\bar{Y}^*),$$
$$K = \bar{Y}^x.$$

A then defines the transcript $T = X^* || Y^* || \bar{X}^* || \bar{Y}^*$ and computes the session key $SK = H\ (A||B||S||T||K)$.

5. After receiving $<X^*, \bar{X}^*, \bar{Y}^*>$ from S, B computes

$$k_{B,2} = F(A, B, pw_{B,} Y^*),$$
$$\bar{X} = D_{k_{B,2}} (\bar{X}^*),$$
$$K = \bar{X}^y.$$

B then defines the transcript $T = X^* \| Y^* \| \bar{X}^* \| \bar{Y}^*$ and computes the session key $SK = H(A \| B \| S \| T \| K)$.

This improved protocol effectively prevents the off-line dictionary attack because the symmetric encryptions of X and Y destroy the algebraic property required for the attack. Although not explicitly stated above, S should abort the protocol immediately if any of X and Y is found to be equal to 1. Other-wise, the protocol is still vulnerable to a variant of the off-line dictionary attack described in the previous section. We finally note that our improved protocol is as efficient as the AP-3PAKE protocol.

Acknowledgments. This work was supported by Howon University in 2013. The primary(first) author is Youngsook Lee (ysooklee@howon.ac.kr).

References

1. Abdalla, M., Bresson, E., Chevassut, O., Pointcheval, D.: Password-based group key exchange in a constant number of rounds. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T. (eds.) PKC 2006. LNCS, vol. 3958, pp. 427–442. Springer, Heidelberg (2006)
2. Abdalla, M., Fouque, P., Pointcheval, D.: Password-based authenticated key exchange in the three-party setting. In: Vaudenay, S. (ed.) PKC 2005. LNCS, vol. 3386, pp. 65–84. Springer, Heidelberg (2005)
3. Abdalla, M., Pointcheval, D.: Interactive Diffie-Hellman assumptions with applications to password-based authentication. In: S. Patrick, A., Yung, M. (eds.) FC 2005. LNCS, vol. 3570, pp. 341–356. Springer, Heidelberg (2005)
4. Bellare, M., Rogaway, P.: The AuthA protocol for password-based authenticated key exchange. Contributions to IEEE P1363 (2000)
5. Bellovin, S., Merritt, M.: Encrypted key exchange: Password-based protocols secure against dictionary attacks. In: 1992 IEEE Symposium on Research in Security and Privacy, pp. 72–84 (1992)
6. Bresson, E., Chevassut, O., Pointcheval, D.: New security results on encrypted key exchange. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 145–158. Springer, Heidelberg (2004)
7. Lin, C., Sun, H., Hwang, T.: Three-party encrypted key exchange: Attacks and a solution. ACM SIGOPS Operating Systems Review 34(4), 12–20 (2000)
8. MacKenzie, P.: The PAK suite: Protocols for password-authenticated key exchange. Contributions to IEEE P1363.2 (2002)
9. Nam, J., Paik, J., Kang, H., Kim, U., Won, D.: An off-line dictionary attack on a simple three-party key exchange protocol. IEEE Communications Letters 13(3), 205–207 (2009)
10. Nam, J., Paik, J., Won, D.: A security weakness in Abdalla et al.'s generic construction of a group key exchange protocol. Information Sciences 181(1), 234–238 (2011)
11. Yoneyama, K.: Efficient and strongly secure password-based server aided key exchange (Extended abstract). In: Chowdhury, D.R., Rijmen, V., Das, A. (eds.) INDOCRYPT 2008. LNCS, vol. 5365, pp. 172–184. Springer, Heidelberg (2008)
12. Zhao, J., Gu, D.: Provably secure three-party password-based authenticated key exchange protocol. Information Sciences 184(1), 310–323 (2012)

A Sequence Classification Model Based on Pattern Coverage Rate

I-Hui Li[1], Jyun-Yao Huang[2], I-En Liao[2], and Jin-Han Lin[2]

[1] Department of Information Networking and System
Administration, Ling Tung University, Taichung, Taiwan
[2] Department of Computer Science and Engineering,
National Chung Hsing University, Taichung, Taiwan
{Sanityih,allen501pc}@gmail.com,
ieliao@nchu.edu.tw, semag33@hotmail.com

Abstract. The technique of classification can sort data into various categories for data mining studies. The demand for sequence data classification has increased with the development of information technology. Several applications involve decision prediction based on sequence data, but the traditional classification methods are unsuitable for sequence data. Thus, this paper proposes a Pattern Coverage Rate-based Sequence Classification Model (PCRSCM) to integrate sequential pattern mining and classification techniques. PCRSCM mines sequential patterns to find characteristics of each class, and then calculates pattern coverage rates and class scores to predict the class of a sequence. The experimental results show that PCRSCM exhibits excellent prediction performance on synthetic and real sequence data.

1 Introduction

With the development of data mining technologies, classification has become an effective technique to predict the category information of interest data. However, in real life, numerous data are ordered according to their timestamps, and are known as sequence data, such as consumption records of customers. The traditional classification methods are unsuitable for obtaining the desired results, such as using the financial situations of customers to predict a debt assessment error [1]. Thus, it is crucial to develop sequence classification techniques to analyze sequence data.

Exarchos et al. [2] proposed a sequence classification model with two phases. This model stores the score of each sequence and compares it to each sequential pattern of a class using a score matrix. Finally, the total score of each class with each sequence is calculated, and each sequence is predicted to a class with the highest score. In the second phase, this model uses optimization software [3] to assign the appropriate weights of each sequential pattern and each class to achieve higher accuracy. Li et al. [4] proposed a novel method based on the model proposed by Exarchos et al. [2]; however, this model did not include optimization. The experimental results indicated that this method achieves excellent accuracy without the optimization phase.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 737–745, 2013.

This study established a classification model for sequence data, that is, Pattern Coverage Rate-based Sequence Classification Model (PCRSCM). The proposed model can achieve excellent classification accuracy. The main contributions of this study are as follows:

(1) A sequence classification model that integrates sequential pattern mining and the classification architecture.

(2) An estimation method based on pattern coverage rate to determine the sequence classification. The proposed method can speed up the prediction. Although the data are skewed, the proposed PCRSCM can avoid error judgment through score calculation.

(3) The proposed PCRSCM uses a sophisticated evaluation formula that combines four types of information: the similarity comparison, the characteristic length, the characteristic support rate, and sequential pattern rate of each class. Such a score formula can classify sequence categories more accurately.

The remainder of this paper is organized as follows: Section 2 introduces the system architecture and detailed methods of the proposed PCRSCM; Section 3 provides simulations and comparisons with other schemes; and finally, Section 4 offers a summary and conclusion.

2 Pattern Coverage Rate-Based Sequence Classification Model

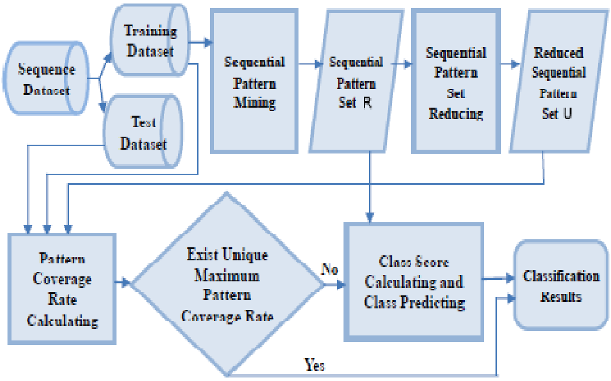

Sequence Dataset
Training Dataset
Test Dataset
Sequential Pattern Mining
Sequential Pattern Set R
Sequential Pattern Set Reducing
Reduced Sequential Pattern Set U
Pattern Coverage Rate Calculating
Exist Unique Maximum Pattern Coverage Rate
No
Yes
Class Score Calculating and Class Predicting
Classification Results

Fig. 1. Pattern Coverage Rate-based Sequence Classification Model

2.1 System Architecture

The system architecture of PCRSCM is shown in Fig. 1. First, PCRSCM divides a sequence dataset into a training dataset and test dataset. PCRSCM includes two phases. The first phase finds the sequential patterns of each class (*R* in Fig. 1) using PrefixSpan from training dataset. PCRSCM subsequently deletes repeated sequential patterns between all classes from the sequential pattern set *R*. The reduced sequential pattern set is called *U*. The second phase determines whether a sequence can be directly predicted by its class using the pattern coverage rate. First, the training dataset is inputted to calculate the pattern coverage rate of each sequence data. From the reduced sequential pattern set *U*, PCRSCM determines the class of each sequence with a unique category; i.e., direct prediction. If the predicted class of a sequence is not unique, PCRSCM further distinguishes them using class scores of the sequential pattern set *R*.

2.2 Pattern Coverage Rate Based Sequence Classification Method

The second phase in PCRSCM uses a pattern coverage rate-based sequence classification method for efficient prediction. PCRSCM uses the reduced sequential pattern set *U* and patterns coverage rate to distinguish direct predictable sequences. If the predicted class of a sequence is not unique, PCRSCM further distinguishes them using class scores of the sequential pattern set *R*.

2.2.1 Pattern Coverage Rate Calculation and Direct Prediction

Assume that *g* sequence data must be predicted. PCRSCM compares each sequence data S_j with *U* to identify the pattern coverage rate, where $1 \leqq j \leqq g;$ that is, the proportion of the number of each sequential pattern in U_i which is the sub-sequence to every sequence S_j. For example, suppose sequential pattern P_1=<(c)(e)>, P_2=<(cd)(e)>, sequence S_1=<(a)(cfd)(e)>, S_2=<(a)(c)(d)(e)>; because P_1 and P_2 are sub-sequences of S_1, the sub-sequence of S_2 is only P_1, the pattern coverage rate of S_1 is equal to 100%, and the pattern coverage rate of S_2 is equal to 50%.

After calculating the pattern coverage rate between S_j and U_i, assume that the coverage rates of each sequence S_{j} in every class are $T_{\mathrm{j},1}$, $T_{\mathrm{j},2}$, …, $T_{\mathrm{j,n}}$. Subsequently, PCRSCM finds $T_{j,h}=\max\{T_{\mathrm{j},1},T_{\mathrm{j},2},\ldots,T_{\mathrm{j,n}}\}$, where $1 \leqq h \leqq n$. Finally, the predicted class of S_{j} is *h*.

While the characteristics of each class are obvious, the pattern coverage rate can quickly predict the classes of sequences. This method does not waste time in calculating class score [2],[3],[4], and can prevent error prediction from skewed data.

2.2.2 Class Score Calculation and Class Predicting

If no unique maximal pattern coverage rate of a sequence is available; that is, $T_{j,h}=T_{j,s}=T_{j,t}$, $h, s, t \in \{1,\ldots,n\}$, PCRSCM uses class score calculation by comparing *R* and S_{j} to further determine the class of a sequence. The class score calculation includes three procedures, as follows:

(1) Similarity Calculation

PCRSCM refers to a sequential pattern similarity calculation method called Similarity Measure for the Sequential Patterns (S2MP), which was proposed by Laurent et al. [5].

(2) Pattern Score Matrices Generation

PCRSCM generates the pattern score matrix for each class. The rows of the pattern score matrix represent each sequential pattern of a class in the sequential pattern set *R*, whereas the columns represent all sequence data. The values in the matrix are the scores of each sequence S_j corresponding to every sequential pattern. When calculating these pattern scores, PCRSCM considers the length, support, support proportion, the proportion of the sequential patterns, and the degree of similarity between the sequential patterns and sequences. The pattern score $P\_S_{i,k,j}$ (the pattern score of the j^{th} sequence corresponding to the k^{th} sequential pattern in the i^{th} class) is as follows:

$$P\_S_{i,k,j} = \text{Len}(P_{i,k}) \times \text{SupWg}_{i,k} \times \text{SimDeg}_{i,k,j} \times \text{ClaWg}_i \quad (1)$$

where, $k=1,2,...,m_i$

$P_{i,k}$: The k^{th} sequential pattern in i^{th} class.

Len($P_{i,k}$): $P_{i,k}$'s length.

$\text{SupWg}_{i,k}$: $P_{i,k}$'s support/$\sum_{k=1}^{m_i} P_{i,k}$'s support

$\text{SimDeg}_{i,k,j}$: The similarity degree between $P_{i,k}$ and S_j.

ClaWg_i: The sequential patterns count in i^{th} class/All sequential patterns count.

(3) Class Score Calculation

When the pattern score matrix of each class is calculated, PCRSCM produces a class score matrix; the rows of the class score matrix represent classes, whereas the columns of the class score matrix represent sequences data. The values in the matrix are the scores of each sequence S_j corresponding to every class. The class score $C\_S_{i,j}$ (the class score of the j^{th} sequence corresponding to the i^{th} class) is shown in Eq. (2):

$$C\_S_{i,j} = \sum_{k=1}^{m_i} P\_S_{i,k,j} \quad (2)$$

(4) Class Prediction

According to the values in the class score matrix, PCRSCM predicts the class of each sequence as shown in Eq. (3):

$$\begin{aligned} &\text{prdeict\_class}_j = i_{\text{pred}} \\ &\text{prdeict\_class}_j\text{: The predict calss of Sequence } S_j \\ &i_{\text{pred}}\text{: The class of } \max\{C\_S_{i,j}\} \end{aligned} \quad (3)$$

If the number of i_{pred} is greater than one, PCRSCM randomly selects a class from as the prediction result.

3 Experiments and Results

PCRSCM is assessed by the prediction accuracy of classification results, as shown in Eq. (4):

$$\text{accuracy} = \frac{\text{Number of correctly predicted sequences}}{\text{Number of sequences}} \quad (4)$$

Three types of datasets were used in experiments: (1) The synthetic dataset from [2]; (2) The synthetic datasets generated using the IBM Quest Synthetic Data Generator [6], the notation for which is shown in Table 1. (3) The real dataset used in SCM. This study designed 11 experiments, the detailed experimental designs and results are as follows.

3.1 Synthetic Dataset

Experiment 1: This experiment used the synthetic dataset in [2], in which the minimum support was 0.5 (50%), the sequence data consisted of three classes, the training dataset had 18 sequences, and the test dataset had 6 sequences. The content of this synthetic dataset is shown in Table 2.

Table 1. The notation of IBM Quest Synthetic Data Generator

C	how many thousands of customers (sequences) exist within the dataset
T	the average transaction (itemset) length
S	the average sequence length (i.e., average number of itemset)
N	the number of distinct items

Table 2. Synthetic Dataset [2]

Training Dataset			Test Dataset		
No.	Sequence	Class	No.	Sequence	Class
1	bbcc	1	1	bbbc	1
2	baca	1	2	accc	1
3	abac	1	3	bbba	2
4	bbca	1	4	caab	2
5	cccb	1	5	caaa	3
6	bbcb	1	6	aacc	3
7	abab	2			
8	cbca	2			
9	cbbb	2			
10	abaa	2			
11	cbbc	2			
12	baba	2			
13	ccbb	3			
14	bbbb	3			
15	bcca	3			
16	acab	3			
17	acca	3			
18	acaa	3			

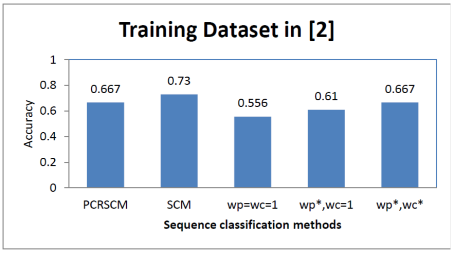


Fig. 2. Prediction performance of different methods

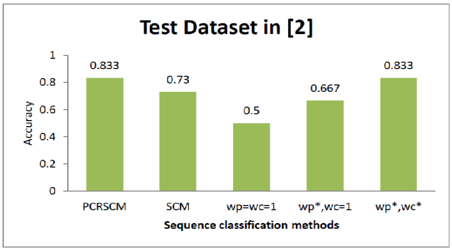


Fig. 3. Prediction performance of different methods

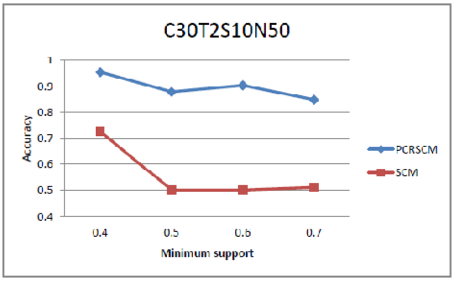


Fig. 4. Prediction performance of variation of the minimum support

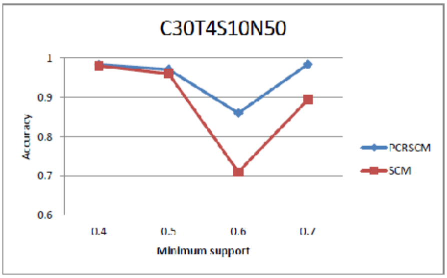


Fig. 5. Prediction performance of variation of the minimum support

Figure 2 shows the accuracy of the training dataset. Figure 3 shows the prediction accuracy of the test dataset. As shown in the figure, the next three results are from classification methods proposed in [2]; (1) wp (pattern weight)=wc (class weight)=1 represents the un-optimized classification results; (2) wc=1 is used to find the optimal solution of wp (wp*); and (3) the optimal pattern weight obtained by (2) is used to find the optimal class weight (wc*). As shown in Figs. 2 and 3, PCRSCM has accuracy that is superior to that of the method proposed in [2] and SCM.

Experiment 2: This experiment produced a synthetic dataset C30T2S10N50 using the IBM Quest Synthetic Data Generator, and varied the minimum support over 0.4, 0.5, 0.6, 0.7. The sequence consisted of two classes. The results are shown in Fig. 4. PCRSCM had accuracy that is superior to that of SCM in each minimum support value.

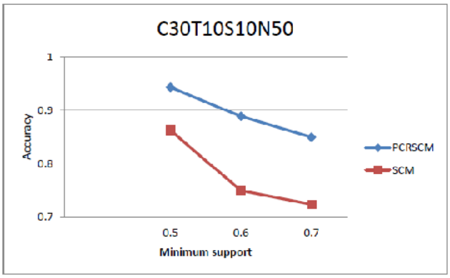


Fig. 6. Prediction performance of variation of the minimum support

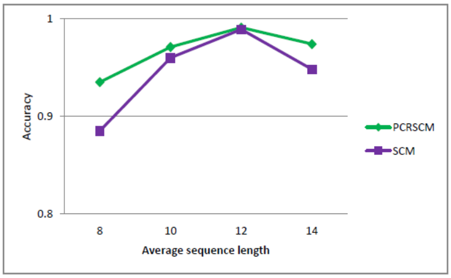


Fig. 7. Prediction performance of variation of the average sequence length (S)

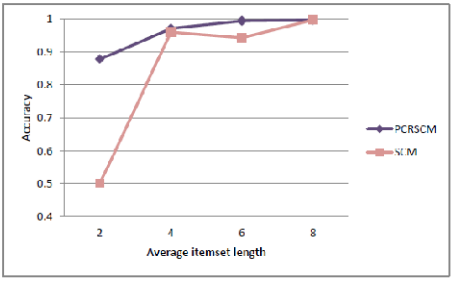


Fig. 8. Prediction performance of variation of the average itemset length (T)

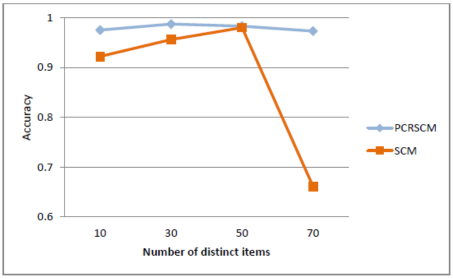


Fig. 9. Prediction performance of variation of the distinct number of items (N)

Experiment 3: To observe the prediction ability of various minimum support values with increased average itemset length (T), this experiment was performed using C30T4S10N50 by varying the minimum support over 0.4, 0.5, 0.6, and 0.7. The sequence consisted of two classes. As shown in Fig. 5, the average accuracy of PCRSCM reached 95%, which is superior to that of SCM (87%) in each minimum support value.

Experiment 4: This experiment used the larger value of T to observe the prediction ability of various minimum support values. C30T10S10N50 was simulated by varying the minimum support over 0.5, 0.6, and 0.7. The sequence consisted of two classes. As shown in Fig. 6, when the minimum support was 0.7, the prediction accuracy of PCRSCM reached 85%.

Experiment 5: This experiment was performed using the C30T10S8N50 dataset by varying the average sequence length (S) over 8, 10, 12, and 14. The minimum support was 0.5, and the sequence consisted of two classes. Fig. 7 indicate that the two methods of classification perform optimally when S is 12. The average prediction accuracy of PCRSCM was 97%, which is superior to that of SCM (95%).

Experiment 6: This experiment was performed using the C30T2S10N50 dataset by varying the average itemset length (T) over 2, 4, 6, and 8. The minimum support was 0.5, and the sequence consisted of two classes. With T set to 2, the prediction was more difficult because the sequential patterns were limited. However, the accuracy was 88% in PCRSCM. The average accuracy of PCRSCM was 96%, and that of SCM was 85%. The results are shown in Fig. 8.

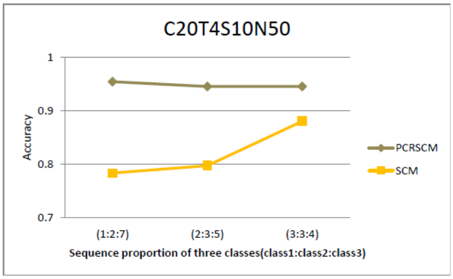


Fig. 10. Prediction performance of variation of the sequence proportion of three classes

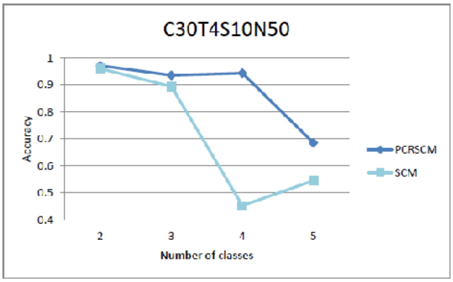


Fig. 11. Prediction performance of variation of the number of classes

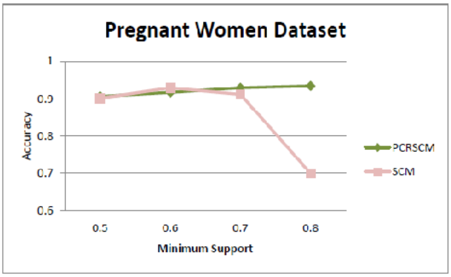


Fig. 12. Prediction performance of variation of the minimum support

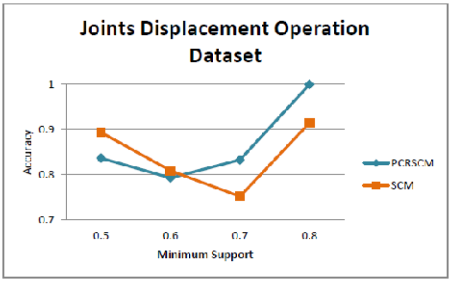


Fig. 13. Prediction performance of variation of the minimum support

Experiment 7: This experiment was performed using the C30T4S10N10 dataset by varying the distinct number of items (N) over 10, 30, 50, and 70. The minimum support was 0.5, and the sequence consisted of two classes. As shown in Fig. 9, the accuracy of PCRSCM is higher than 97%. The average accuracy of PCRSCM was 98%, and that of SCM was 88%.

Experiment 8: This experiment was performed using the C30T4S10N50 dataset. The minimum support was 0.5 and the sequence consisted of three classes. The sequence proportion of these three classes was varied over 1:2:7, 2:3:5, and 3:3:4. As shown in Fig. 10, the average accuracy of PCRSCM is 95%, and that of SCM is 82%.

Experiment 9: This experiment was performed using the C30T4S10N50 dataset. The minimum support was 0.5, and the number of classes was varied over 2, 3, 4, and 5. The sequence proportion of each class was equal. As shown in Fig. 11, the accuracy of PCRSCM decreases to 70%. However, the average prediction accuracy of PCRSCM was 88%, which is superior to that of SCM (71%).

3.2 Real Datasets

The real datasets were collected from a hospital in Taichung, Taiwan, and included data of pregnant women in hospital, and data of hospital patients who have underwent joint displacement operations during 2009. The experiment used three attributes from the original data to generate a sequence dataset: the patient ID, the physical examination item, and the date; that is, the experiment ordered the physical examination items by the date of each patient ID.

Experiment 10: This experiment used the pregnant women dataset, and converted them into 395 sequences of data. Two classes were used in this dataset: natural childbirth and Caesarean birth. This experiment varied the minimum support over 0.5, 0.6, 0.7, and 0.8. The average accuracy of PCRSCM was 92%, and that of SCM was 86%. The experimental results are shown in Fig. 12.

Experiment 11: This experiment used a dataset from patients who underwent joint displacement operations, and converted them into 246 sequences of data. Two classes were used in this dataset: total hip joint displacement operations and total knee joint displacement operations. This experiment varied the minimal support over 0.5, 0.6, 0.7, and 0.8. The average accuracy of PCRSCM was 87%, and that of SCM was 84%, the experimental results are shown in Fig. 13.

4 Conclusions

The PCRSCM proposed in this study can effectively analyze sequences to establish a classification model. It obtains excellent classification results and is not susceptible to various parameters. Moreover, when classifying, PCRSCM can provide more comprehensive information on sequence data. The experimental results indicated the average accuracy of PCRSCM is 92%, and the lowest accuracy of PCRSCM is 70%. Therefore, PCRSCM can achieve certain accuracy without any optimization processes. Future studies will enhance the concept of ensemble classifiers to improve the accuracy of the classification model.

References

1. Zhao, Y., Zhang, H., Wu, S., Pei, J., Cao, L., Zhang, C., Bohlscheid, H.: Debt Detection in Social Security by Sequence Classification Using Both Positive and Negative Patterns. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009, Part II. LNCS, vol. 5782, pp. 648–663. Springer, Heidelberg (2009)
2. Exarchos, T.P., Tsipouras, M.G., Papaloukas, C., Fotiadis, D.I.: A two-stage methodology for sequence classification based on sequential pattern mining and optimization. Data & Knowledge Engineering 66, 467–487 (2008)
3. Papageorgiou, D.G., Demetropoulos, I.N., Lagaris, I.E.: MERLIN-3.1.1. A new version of the Merlin optimization environment. Computer Physics Communications 159, 70–71 (2004)
4. Li, I.H., Lin, M.C., Liao, I.E.: A Sequential Pattern Length Based Sequence Classifier Model. In: International Conference on Information Management, p. 95 (2011)
5. Saneifar, H., Bringay, S., Laurent, A., Teisseire, M.: S2mp: Similarity Measure for Sequential Patterns. In: Proc. of 7th Australasian Data Mining Conference, pp. 95–104 (2008)
6. IBM Quest Market-Basket Synthetic Data Generator, `http://www.cs.rpi.edu/~zaki/software/IBM-datagen.tar.gz`

Development of STEAM Program and Teaching Method for Using LEGO Line Tracer Robot in Elementary School[*]

Yeonghae Ko and Namje Park[**]

Department of Computer Education, Teachers College, Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
{smakor,namjepark}@jejunu.ac.kr

Abstract. The 7th National Curriculum includes the courses on electronic goods (Practical Course) and roles of robot in the future (Social Studies), which are not sufficient for students to learn about characteristics and functions of robots. As a complementary measure, in order to enhance logical thinking of students through programming of movement of robots, robot education programs have been suggested and are under development. This study suggests the programmable line tracer robot course which provides the elementary school students with the experience of producing and controlling robots. This study also suggests the convergence robot education program to be developed link with the regular curriculum, and analyzes the method of teaching based on the teaching plans.

Keywords: Line Tracer robot, LEGO, STEAM, Elementary School, Teaching Method.

1 Introduction

The 21st century is a knowledge-based era. The government suggests the image of Korea in 2040 as ‘the world with nature’ and ‘abundant, healthy and convenient world’, and expects that the robot industry will become the major industry in 2040 to make the ‘convenient world’. The 7th National Curriculum includes the courses on electronic goods (Practical Course) and roles of robot in the future (Social Studies). However, the courses are not sufficient for students to know about characteristics and functions of robots because they are not consistent with the level of students or compliant with the subjects. As a complementary measure, in order to enhance logical thinking of students through programming of movement of robots, robot education programs have been suggested and are under development.

The robot manufacturing is roughly divided into modeling and movement control. Programming line tracer is the suitable model to be applied to the elementary school

[*] This work was supported by the Korea Foundation for the Advancement of Science and Creativity(KOFAC) grant funded by the Korea government(MEST).

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 746–751, 2013.

course. This paper suggests the programmable line tracer robot course which provides the elementary school students with the experience of producing and controlling robots. This paper also suggests the convergence robot education program to be developed link with the regular curriculum, and analyzes the method of teaching based on the teaching plans.

2 Requirement of STEAM Education

To realize the STEAM education, the factors on how to interrelate and integrate science, technology, engineering, art, and mathematics as well as the factors that are needed in realizing the STEAM education in creativity in addition to the considered factors in contents need to be decided, which in reality, makes the creation of STEAM materials into a system science or system engineering. In other words, the many factors need to harmonize in a creative and appropriate way along with the theoretical foundation and applications in a systematic way.

Many questions on STEAM education include whether only S and T or T and E could be realized, how it is different from the field trips to research centers or science centers, and how different it is from the existing STS education.

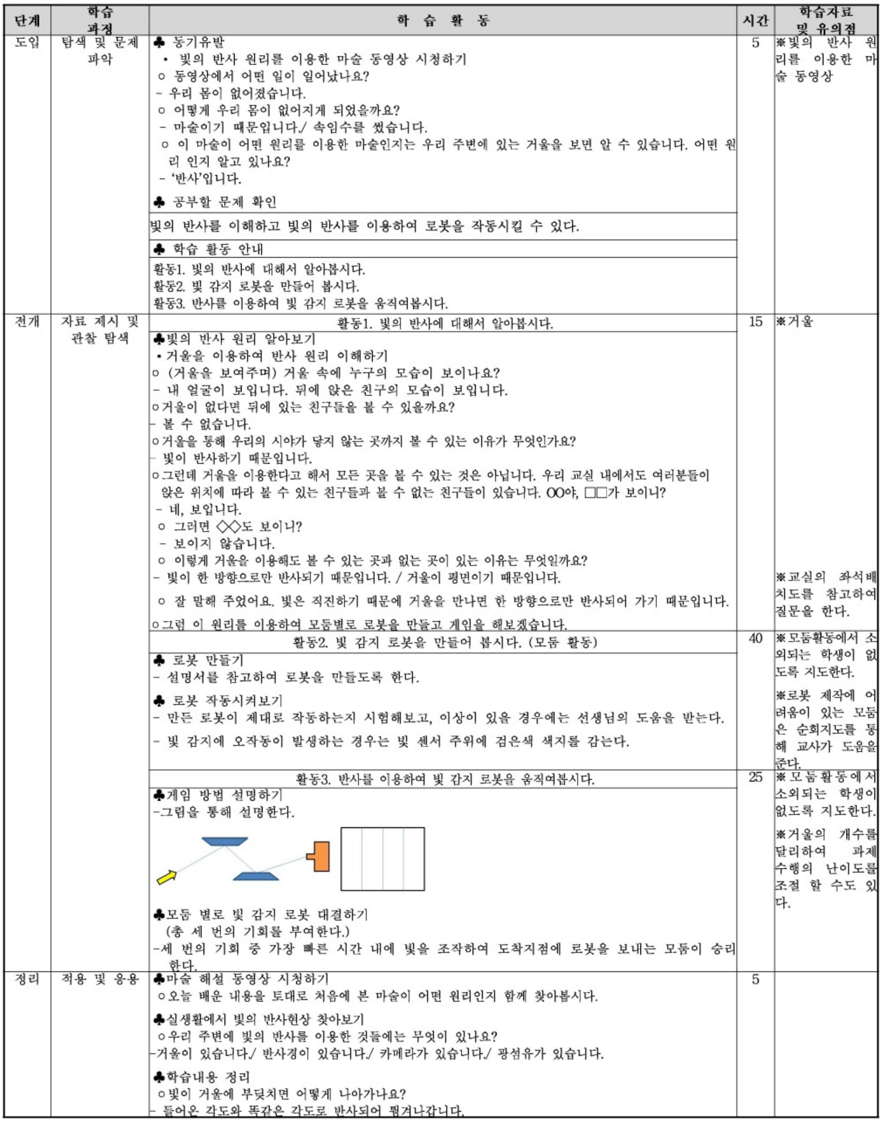
First, the materials of STEAM education could hold an important meaning. Therefore, whether to start from current textbook science theories and systematically increase into the engineering and technology or whether to create a new structure of reverse engineering on a topic to allow students find the theories as they dissemble a product could be an issue but to smoothly transition to STEAM project with minimal friction, the former would be more preferable than the latter. This is because if the reverse engineering is utilized for the concept understanding in science education, the students may not fully understand the science theories and may require additional curricular activities. In reality, the reflecting factors of S, T, E, A, and M in STEAM education are naturally included in the systematic connection based on a key factor of storytelling and the process to describe the variety of science technology engineering.

3 Proposed STEAM Program

To apply on site without creating conflicts with current curriculum, a systematic connection into the basic science technology engineering is required. In addition, integrative thought or fused thought activities could be organized separately or together with each area of STEAM.

1) Introduction: Exploration and problem analysis
   - Motivation: Watching videos on magic using reflection of light
2) Development: Presentation of data and observation
   - Activity 1. Let's find out about reflection of light.
   - Activity 2. Let's make a light-sensing robot.
   - Activity 3. Let's move the light-sensing robot by using reflection of light.

3) Finish: Practice and application
- Viewing videos on explanation of magic
- Finding reflection of light in a real life
- Summary



단계	학습 과정	학 습 활 동	시간	학습자료 및 유의점
도입	탐색 및 문제 파악	♣ 동기유발 ▪ 빛의 반사 원리를 이용한 마술 동영상 시청하기 ○ 동영상에서 어떤 일이 일어났나요? - 우리 몸이 없어졌습니다. ○ 어떻게 우리 몸이 없어지게 되었을까요? - 마술이기 때문입니다./ 속임수를 썼습니다. ○ 이 마술이 어떤 원리를 이용한 마술인지는 우리 주변에 있는 거울을 보면 알 수 있습니다. 어떤 원리 인지 알고 있나요? - '반사'입니다. ♣ 공부할 문제 확인 빛의 반사를 이해하고 빛의 반사를 이용하여 로봇을 작동시킬 수 있다. ♣ 학습 활동 안내 활동1. 빛의 반사에 대해서 알아봅시다. 활동2. 빛 감지 로봇을 만들어 봅시다. 활동3. 반사를 이용하여 빛 감지 로봇을 움직여봅시다.	5	※빛의 반사 원리를 이용한 마술 동영상
전개	자료 제시 및 관찰 탐색	활동1. 빛의 반사에 대해서 알아봅시다. ♣빛의 반사 원리 알아보기 ▪거울을 이용하여 반사 원리 이해하기 ○ (거울을 보여주며) 거울 속에 누구의 모습이 보이나요? - 내 얼굴이 보입니다. 뒤에 앉은 친구의 모습이 보입니다. ○거울이 없다면 뒤에 있는 친구들을 볼 수 있을까요? - 볼 수 없습니다. ○거울을 통해 우리의 시야가 닿지 않는 곳까지 볼 수 있는 이유가 무엇인가요? - 빛이 반사하기 때문입니다. ○그런데 거울을 이용한다고 해서 모든 곳을 볼 수 있는 것은 아닙니다. 우리 교실 내에서도 여러분들이 앉은 위치에 따라 볼 수 있는 친구들과 볼 수 없는 친구들이 있습니다. OO야, □□가 보이니? - 네, 보입니다. ○ 그러면 ◇◇도 보이니? - 보이지 않습니다. ○ 이렇게 거울을 이용해도 볼 수 있는 곳과 없는 곳이 있는 이유는 무엇일까요? - 빛이 한 방향으로만 반사되기 때문입니다. / 거울이 평면이기 때문입니다. ○ 잘 말해 주었어요. 빛은 직진하기 때문에 거울을 만나면 한 방향으로만 반사되어 가기 때문입니다. ○그럼 이 원리를 이용하여 모둠별로 로봇을 만들고 게임을 해보겠습니다.	15	※거울 ※교실의 좌석배치도를 참고하여 질문을 한다.
		활동2. 빛 감지 로봇을 만들어 봅시다. (모둠 활동) ♣ 로봇 만들기 - 설명서를 참고하여 로봇을 만들도록 한다. ♣ 로봇 작동시켜보기 - 만든 로봇이 제대로 작동하는지 시험해보고, 이상이 있을 경우에는 선생님의 도움을 받는다. - 빛 감지에 오작동이 발생하는 경우는 빛 센서 주위에 검은색 색지를 감는다.	40	※모둠활동에서 소외되는 학생이 없도록 지도한다. ※로봇 제작에 어려움이 있는 모둠은 순회지도를 통해 교사가 도움을 준다.
		활동3. 반사를 이용하여 빛 감지 로봇을 움직여봅시다. ♣게임 방법 설명하기 -그림을 통해 설명한다. ♣모둠 별로 빛 감지 로봇 대결하기 (총 세 번의 기회를 부여한다.) -세 번의 기회 중 가장 빠른 시간 내에 빛을 조작하여 도착지점에 로봇을 보내는 모둠이 승리한다.	25	※모둠활동에서 소외되는 학생이 없도록 지도한다. ※거울의 개수를 달리하여 과제 수행의 난이도를 조절 할 수도 있다.
정리	적용 및 응용	♣마술 해설 동영상 시청하기 ○오늘 배운 내용을 토대로 처음에 본 마술이 어떤 원리인지 함께 찾아봅시다. ♣실생활에서 빛의 반사현상 찾아보기 ○우리 주변에 빛의 반사를 이용한 것들에는 무엇이 있나요? -거울이 있습니다./ 반사경이 있습니다./ 카메라가 있습니다./ 광섬유가 있습니다. ♣학습내용 정리 ○빛이 거울에 부딪치면 어떻게 나아가나요? - 들어온 각도와 똑같은 각도로 반사되어 튕겨나갑니다.	5	

Fig. 1. Example of teaching guidance methods (Korea's document format)

Utilization in the class and guidelines are as follows.

- Carry out the activities that may attract attention of the students, and evaluate achievement of the students through the activities. Set the starting point and the arrival point of robots, and divide the distance into sections. Measure time taken by light-sensing robots to reach the arrival point.

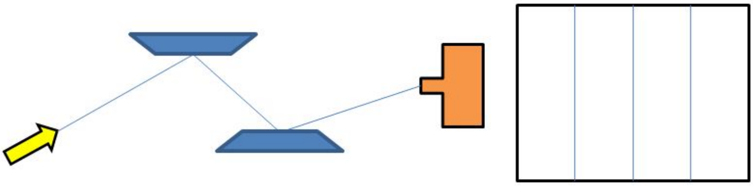


Fig. 2. Start of a robot utilizing activity

Figure 2 shows the start of a robot utilizing activity. The theory of reflection of light, that is, the theory of incidence angle and reflection angle is used to have a light from a laser point reached to the sensor of the robot. In this course of activity, no quantitative approach (incidence angle = reflection angle) shall be used.

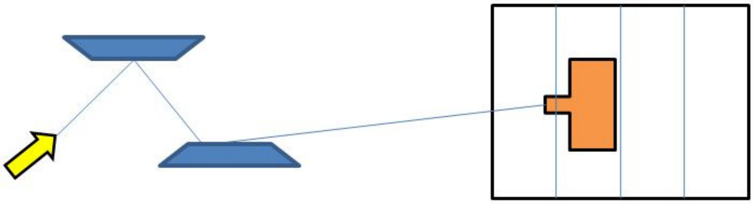


Fig. 3. Adjusting incidence angle at a point

A robot starts to move as it senses light, and stops at the point 1. In order to have this robot to move again, students need to have light to reach the robot. Because the position of the robot is changed, students need to control the angle of a mirror or the incident angle of light.

4 Conclusion

This paper suggests the course in which students improve creativity and problem-solving capability by designing program and using the convergence programming-based line tracer robots. Students are generally interested in the course on programming language and robot, but this kind of course is provided mainly for talented students. Further studies are required to assess and analyze the effect of the program suggested in this study.

References

1. An, J., Kim, E., Byeon, H., Ko, Y., An, J., Park, N.: A Study on STEAM Program Development and Teaching Guidance Methods using Line Tracer Robot. In: Proceedings of JCCIS Conference, The Third Conference on Computer and Information Science, vol. 6(2), pp. 17–19 (2012)
2. Lee, J.W., Park, N.: Individual Information Protection in Smart Grid. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 153–159. Springer, Heidelberg (2012)
3. Kang, S.: Korean Morphological Analysis and Information Retrieval. Hongreung Science Press (2002)
4. Support Vector Machine, Wikipedia. the free Encyclopedia, `http://en.wikipedia.org/wiki/SVM`
5. Thesaurus, Wikipedia, The Free Encyclopedia (2008), `http://en.wikipedia.org/wiki/Thesaurus`
6. Frakes, W., Baeza-Yates, R.: Information Retrieval: Data Structures and Algorithms. Prentice Hall (1992)
7. Kim, Y., Park, N., Hong, D., Won, D.: Context-based classification for harmful web documents and comparison of feature selecting algorithms. Journal of Korea Multimedia Society 12(6) (2009)
8. Yang, Y., Pederson, J.: A Comparative Study on Feature Selection in text Categorization. In: Proceedings of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
9. Park, N., Kwak, J., Kim, S., Won, D., Kim, H.: WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 741–748. Springer, Heidelberg (2006)
10. Park, N.: Security scheme for managing a large quantity of individual information in RFID environment. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) ICICA 2010. CCIS, vol. 106, pp. 72–79. Springer, Heidelberg (2010)
11. Park, N.: Secure UHF/HF Dual-Band RFID: Strategic Framework Approaches and Application Solutions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 488–496. Springer, Heidelberg (2011)
12. Park, N.: Implementation of Terminal Middleware Platform for Mobile RFID computing. International Journal of Ad Hoc and Ubiquitous Computing 8(4), 205–219 (2011)
13. Park, N., Kim, Y.: Harmful Adult Multimedia Contents Filtering Method in Mobile RFID Service Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 193–202. Springer, Heidelberg (2010)
14. Park, N., Song, Y.: AONT Encryption Based Application Data Management in Mobile RFID Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS, vol. 6422, pp. 142–152. Springer, Heidelberg (2010)
15. Park, N.: Customized Healthcare Infrastructure Using Privacy Weight Level Based on Smart Device. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. CCIS, vol. 206, pp. 467–474. Springer, Heidelberg (2011)
16. Park, N.: Secure Data Access Control Scheme Using Type-Based Re-encryption in Cloud Environment. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 319–327. Springer, Heidelberg (2011)

17. Park, N., Song, Y.: Secure RFID Application Data Management Using All-Or-Nothing Transform Encryption. In: Pandurangan, G., Anil Kumar, V.S., Ming, G., Liu, Y., Li, Y. (eds.) WASA 2010. LNCS, vol. 6221, pp. 245–252. Springer, Heidelberg (2010)
18. Park, N.: The Implementation of Open Embedded S/W Platform for Secure Mobile RFID Reader. The Journal of Korea Information and Communications Society 35(5), 785–793 (2010)
19. Kim, Y., Park, N.: Development and Application of STEAM Teaching Model Based on the Rube Goldberg's Invention. LNEE, vol. 203, pp. 693–698. Springer (2012)
20. Park, N., Cho, S., Kim, B., Lee, B., Won, D.: Security Enhancement of User Authentication Scheme Using IVEF in Vessel Traffic Service System. LNEE, vol. 203, pp. 699–705. Springer (2012)
21. Kim, K., Kim, B.-D., Lee, B., Park, N.: Design and Implementation of IVEF Protocol Using Wireless Communication on Android Mobile Platform. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 94–100. Springer, Heidelberg (2012)
22. Ko, Y., An, J., Park, N.: Development of Computer, Math, Art Convergence Education Lesson Plans Based on Smart Grid Technology. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 109–114. Springer, Heidelberg (2012)
23. Kim, Y., Park, N.: The Effect of STEAM Education on Elementary School Student's Creativity Improvement. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 115–121. Springer, Heidelberg (2012)

Improved Authentication Scheme with Anonymity for Roaming Service in Global Mobility Networks*

Youngseok Chung[1,2], Youngsook Lee[3], and Dongho Won[2,**]

[1] Electronics and Telecommunications Research Institute, Korea
[2] Department of Computer Engineering, Sungkyunkwan University, Korea
[3] Department of Cyber Investigation Police, Howon University, Korea
yschung11@ensec.re.kr, ysooklee@howon.ac.kr,
dhwon@security.re.kr

Abstract. Recently Chang, Lee, and Chiu proposed an enhanced authentication scheme with anonymity for roaming service in global mobility networks. Their scheme is suitable for mobile environments. This is because it uses only low-cost functions such as one-way hash functions and exclusive-OR operations. After that, Youn, Park, and Lim showed the weaknesses of Chang et al.'s scheme without any countermeasures. In this paper, we propose an improved authentication scheme with anonymity by basing on low-cost functions as Chang et al.'s scheme. The proposed scheme overcomes the security flaws demonstrated by Youn et al. by adopting two secret values and a virtual identity. Therefore, our scheme is more secure and still efficient when compared with Chang et al.'s scheme.

Keywords: authentication, anonymity, roaming service, mobility network.

1 Introduction

The more global mobility network (GLOMONET), which provides mobile users with global roaming services is becoming increasingly common; the more public concerns about the authentication scheme with anonymity are increasing. Especially, the authentication schemes which are suitable for battery-powered mobile devices in GLOMONET are becoming more essential in many aspects of efficiency.

In recent years, many authenticating schemes have been proposed for wireless environments [1-5]. In addition, several attacks against the proposed schemes with anonymity and countermeasures have been also produced. It is possible to categorize the proposed schemes into two groups: schemes using high-cost functions such as asymmetric and symmetric cryptosystems also schemes using low-cost functions such as one-way hash functions and exclusive-OR operations.

[*] This research was supported by the KCC(Korea Communications Commission), Korea, under R&D program supervised by the KCA(Korea Communications Agency) (KCA-2012-12-912-06-003).

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 752–760, 2013.

In 2004, Zu and Ma [1] proposed a new authentication scheme with anonymity for wireless environments using high-cost functions. Since then, the intertwining offensive and defensive suggestions, namely, proofs of the previous schemes' weaknesses and improvements have been followed [2-4]. On the other hand, Chang, Lee, and Chiu [6] proposed an enhanced authentication with anonymity by basing on low-cost functions that is suitable for mobile environments in 2008. Youn, Park, and Lim [7] presented that Chang et al.'s scheme not only fails to achieve the anonymity against both passive adversaries and malicious mobiles users but also is insecure against known session key attacks and side channel attacks. Unfortunately, they did not provide any countermeasures to resolve these vulnerabilities.

In this paper, we propose an improved authentication scheme with anonymity for roaming service in GLOMONET. Our scheme uses only one-way hash functions and exclusive-OR operations as Chang et al.'s scheme dose. The proposed scheme overcomes the security flaws demonstrated by Youn et al. by adopting two more secret values and a virtual identity. Therefore, our scheme which is an improved version of Chang et al.'s scheme is more secure and still efficient.

The remainder of this paper is organized as follows. In Section 2, we review Chang et al.'s scheme and discuss its weaknesses. An improved authentication scheme is presented in Section 3. In Section 4, we analyze the security of our scheme. Finally, a concluding remark is given in Section 5.

2 Review of Previous Works

In this section, we review Chang et al.'s scheme and Youn et al.'s proofs. Table 1. lists some notations used in Chang et al.'s scheme and they are also used throughout in this paper.

Table 1. Some notations

Notations	Descriptions
MN	A mobile user
HA	The home agent of a mobile user
FA	The foreign agent of a foreign network
ID_X	The identity of an entry X
PW_{MN}	A password of MN
n_X	A nonce generated by entry X
h(.)	A collision free one-way hash function
$\|\|$	A concatenation
$\oplus$	A XOR operation

2.1 Chang et al.'s Scheme

There are three phases in Chang et al.'s scheme: registration, authentication, and session key establishment. MN, HA, and FA are involved in these phases. It is assumed that each FA and HA share a long-term common secret key K_{FH} established by using key agreement method, such as the Diffie-Hellman key agreement protocol.

Registration Phase: In this phase, a new mobile user MN submits his/her identity ID_{MN} and the selected password PW_{MN} to HA for registration. Then, HA generates $R = h(ID_{MN}||x) \oplus PW_{MN}$ using its private key x and delivers a smart card containing $\{ID_{MN}, ID_{HA}, R, h(x), h(.)\}$ to MN through a secure channel.

Authentication Phase: When MN roams in a foreign network and tries to access service, FA needs to authenticate MN through $MN's$ home agent HA. MN, HA, and FA perform the authentication procedures as follows.

1. MN inserts his/her smart card into the device and enters password $PW^*{}_{MN}$. $MN's$ smart card generates a nonce n_{MN} and calculates $C = (R \oplus PW^*{}_{MN}) \oplus n_{MN}$.
2. MN sends a login message $m_1 = \{Login\,request, n_{MN}, ID_{HA}\}$ to FA for authentication where "$Login\,request$" is the header of the message that alerts a new session between MN and FA.
3. Upon receiving m_1, FA generates a nonce n_{FA} and sends an authentication message $m_2 = \{Authenticaton\,request, n_{FA}, ID_{FA}\}$ to HA , where "$Authentication\ \ request$" is the header of the message that notifies HA to authenticate the roaming user MN.
4. After receiving m_2, HA checks ID_{FA} to determine whether it is an ally. HA generates a nonce n_{HA} and sends $m_3 = \{n_{HA}, ID_{HA}\}$ to FA.
5. After receiving m_3, FA sends $m_4 = \{n_{HA}, n_{FA}, ID_{FA}\}$ to MN.
6. Upon receiving m_4 , MN generates $SID = ID_{MN} \oplus h(h(x)||n_{HA})$, $V_1 = h(n_{HA}||C)$, $SK = h(h(x)||ID_{MN}||ID_{FA}||n_{MN}||n_{FA})$, $V_2 = SK \oplus h(n_{HA}||ID_{MN})$, and $S_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$. Then, MN sends $m_5 = \{SID, V_1, V_2, n_{MN}, S_1, ID_{HA}\}$ to FA.
7. After receiving m_5, FA computes $S^*{}_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$ and checks whether $S^*{}_1$ and S_1 are equal or not. If the result is valid, FA computes $S_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$ and sends $m_6 = \{SID, V_1, V_2,\ n_{MN}, S_2, ID_{FA}\}$ to HA to verify whether MN is legal.
8. After receiving m_6, HA checks ID_{FA} to determine whether it is an ally. Then, HA computes $S^*{}_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$ to check whether $S^*{}_2 = S_2$. If the result is valid, the identity of FA is authenticated. HA obtains ID_{MN} by computing $ID_{MN} = SID \oplus h(h(x)||n_{HA})$, and then verifies the format of ID_{MN}. If the format is invalid, HA terminates the connection. Otherwise, HA computes $C^* = h(ID_{MN}||x) \oplus n_{MN}$ and $V^*{}_1 = h(n_{HA}||C^*)$ to check whether $V^*{}_1 = V_1$. If the result is valid, HA computes $SK = V_2 \oplus h(n_{HA}||ID_{MN})$, $K_1 = SK \oplus h(K_{FH}||n_{FA})$, $V_3 = h(ID_{FA}||h(x)$ $||n_{MN})$, and $S_3 = h(K_{FH}||n_{FA}||K_1||V_3)$. And then HA sends $m_7 = \{K_1, V_3, S_3\}$ to FA to inform that MN is a legal user.

Session Key Establishment Phase: If the authentication process finishes successfully, FA and MN generate a common session key in this phase.

1. With m_7, FA computes $S^*{}_3 = h(K_{FH}||n_{FA}||K_1||V_3)$ to check whether $S^*{}_3 = S_3$ using K_{FH} and n_{FA}. If they are equal, FA computes $SK = K_1 \oplus h(K_{FH}||n_{FA})$, $K_2 = SK \oplus h(SK||n_{MN})$ and sends $m_8 = \{V_3, K_2\}$ to MN.
2. After receiving m_8, MN computes $V^*{}_3 = h(ID_{FA}||h(x)||n_{MN})$ and $SK^* = K_2 \oplus h(SK||n_{MN})$ to checks whether $V^*{}_3 = V_3$ and $SK^* = SK$. If the results are valid, MN is sure that FA also has an authenticated session key. Then, MN records the authenticated session key SK for future communications.

2.2 Youn et al.'s Proof

Youn et al. pointed out the insecurities of the Chang et al.'s scheme by describing four attack strategies for recovering the identities of mobile users.

Anonymity against Passive Adversaries: The passive adversary who simply eavesdrops the messages transferred between MN and FA can recover the identity of MN using two messages V_2 and K_2. To find ID_{MN}, the adversary chooses a candidate identity ID'_{MN}, computes $SK' = V_2 \oplus h(n_{HA}||ID'_{MN})$, and tests whether K_2 is identical with $K'_2 = SK' \oplus h(SK'||n_{MN})$. If $K'_2 = K_2$, the guessed identity ID'_{MN} is equal to the identity of real MN.

Anonymity against Malicious Mobile User: It is assumed that MN' is a malicious mobile user who possesses a valid smart card issued by HA. MN' can obtain $SID' = ID_{MN'} \oplus h(h(x)||n'_{HA})$ by executing a legitimate run of the scheme. MN' eavesdrops the messages between MN and FA, and replaces $HA's$ random nonce by n'_{HA} in a message m_3 or m_4. After receiving n'_{HA}, MN returns $SID = ID_{MN} \oplus h(h(x)||n'_{HA})$ and MN' captures SID. Then, MN'can recover the identity of MN, $ID_{MN} = SID \oplus SID' \oplus ID_{MN'} = ID_{MN} \oplus h(h(x)||n'_{HA}) \oplus ID_{MN'} \oplus h(h(x)||n'_{HA}) \oplus ID_{MN'}$.

Security against Known Session Key Attacks: If the session key SK established between MN and FA is revealed to an adversary, ID_{MN} can be recovered by an adversary. An adversary computes $Flg = V_2 \oplus SK$ and searches an identity ID such that $Flg = h(n_{HA}||ID)$. Since $V_2 = SK \oplus h(n_{HA}||ID_{MN})$ and $h(.)$ is a collision free one-way hash function, $V_2 \oplus SK = h(n_{HA}||ID_{MN})$ only if $ID_{MN} = ID$. Since a user's identity is short and has a certain format, the adversary can find it by executing an exhaustive search within polynomial time.

Security against Side Channel Attacks: If the smart card issued by HA is damaged by the revelation of sensitive information, an adversary can recover the identity of a mobile user by extracting $h(x)$ from a smart card. The adversary can recover the mobile user's identity by computing $ID_{MN} = SID \oplus h(h(x)||n_{HA})$ after obtaining SID and n_{HA} from communication messages.

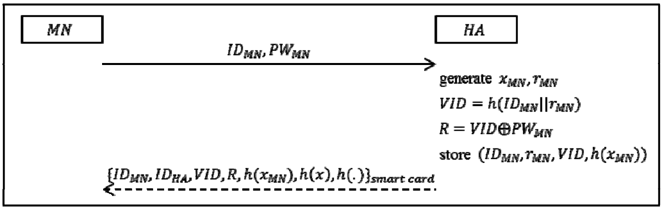
3 Proposed Scheme

In this section, we demonstrate an improved authentication scheme with anonymity overcoming security flaws showed by Youn et al. Our proposed is also consists of three phases: registration, authentication, and session key establishment. Basically, the number of communications among involved in our scheme is same as in Chang et al.'s scheme. And we also use just low-cost functions.

In the registration phase, HA adopts two more secret values for only MN and a random virtual identity of MN. They are key elements in guaranteeing the security of the messages for transmission and the anonymity of users. The improved version of Chang et al.'s scheme is as follows.

3.1 Registration Phase

In the registration phase, a mobile user MN who wants to register his/her home agent HA submits his/her identity ID_{MN} and the selected password PW_{MN} to HA. Then, HA computes a user's virtual identity $VID = h(ID_{MN}||r_{MN})$ and a parameter$R =$ ter$R = VID \oplus PW_{MN}$ after generating a secret key x_{MN} and a random secret parameter r_{MN} for him/her only. HA stores $(ID_{MN}, r_{MN}, VID, h(x_{MN}))$ secretly and delivers a smart card containing $\{ID_{MN}, ID_{HA}, VID, R, h(x_{MN}), h(x), h(.)\}$ to MN through a secure channel. The registration process between MN and HA is described in Fig. 1.



MN
HA
ID_{MN}, PW_{MN}
generate x_{MN}, r_{MN}
$VID = h(ID_{MN}||r_{MN})$
$R = VID \oplus PW_{MN}$
store $(ID_{MN}, r_{MN}, VID, h(x_{MN}))$
$\{ID_{MN}, ID_{HA}, VID, R, h(x_{MN}), h(x), h(.)\}_{smart\ card}$

Fig. 1. Registration phase of the proposed scheme

3.2 Authentication and Session Key Establishment Phases

We assume that MN who roams into the foreign network visits FA and FA needs to authenticate MN through HA. MN, HA, and FA perform the following steps. Fig. 2. represents the authentication and session key establishment processes.

3. MN inserts his/her smart card into the device and enters a password $PW^*{}_{MN}$. $MN's$ smart card generates a nonce n_{MN} and calculates $C = (R \oplus PW^*{}_{MN}) \oplus n_{MN}$. MN sends a login message $m_1 = \{Login\ req., n_{MN}, ID_{HA}\}$ to FA for authentication.
4. Upon receiving m_1, FA records n_{MN}, generates a nonce n_{FA} and sends an authentication message $m_2 = \{Authenticaton\ req., n_{FA}, ID_{FA}\}$ to HA.
5. After receiving m_2, HA checks ID_{FA} to determine whether it is an ally. If the result is valid, HA generates a nonce n_{HA} and sends a message $m_3 = \{n_{HA}, ID_{HA}\}$ to FA.
6. After receiving m_3, FA sends a message $m_4 = \{n_{HA}, n_{FA}, ID_{FA}\}$ to MN.

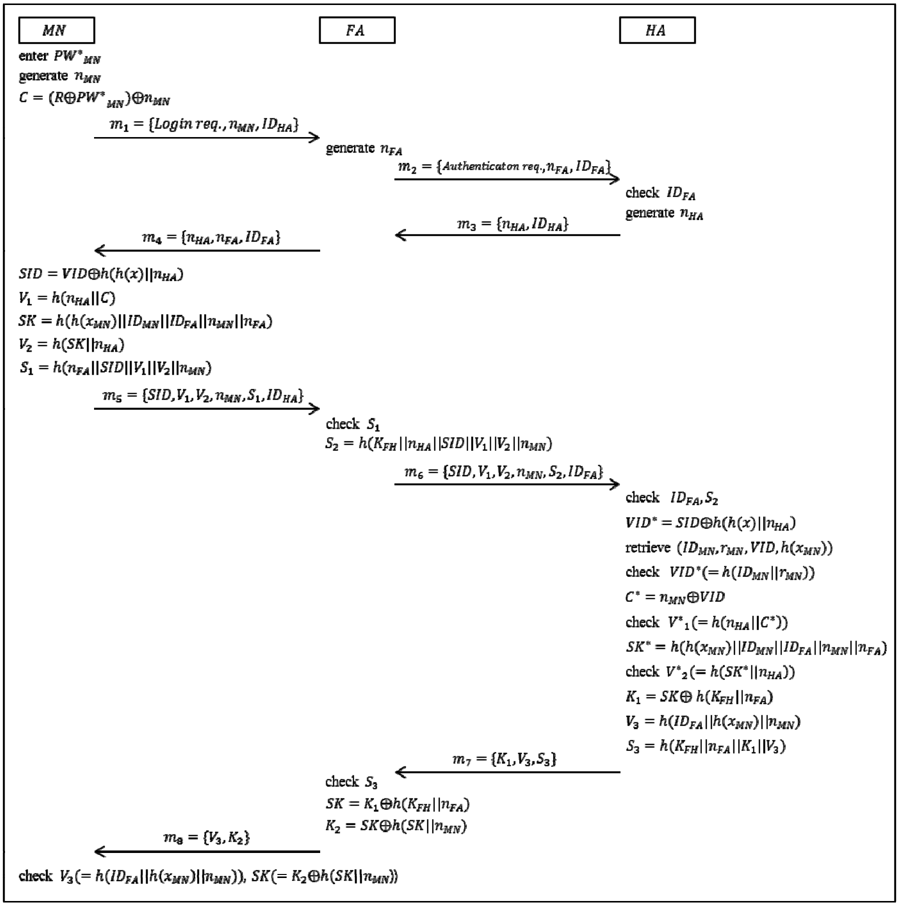

MN
FA
HA
enter PW^*_{MN}
generate n_{MN}
$C = (R \oplus PW^*_{MN}) \oplus n_{MN}$
$m_1 = \{Login\ req., n_{MN}, ID_{HA}\}$
generate n_{FA}
$m_2 = \{Authenticaton\ req., n_{FA}, ID_{FA}\}$
check ID_{FA}
generate n_{HA}
$m_3 = \{n_{HA}, ID_{HA}\}$
$m_4 = \{n_{HA}, n_{FA}, ID_{FA}\}$
$SID = VID \oplus h(h(x)||n_{HA})$
$V_1 = h(n_{HA}||C)$
$SK = h(h(x_{MN})||ID_{MN}||ID_{FA}||n_{MN}||n_{FA})$
$V_2 = h(SK||n_{HA})$
$S_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$
$m_5 = \{SID, V_1, V_2, n_{MN}, S_1, ID_{HA}\}$
check S_1
$S_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$
$m_6 = \{SID, V_1, V_2, n_{MN}, S_2, ID_{FA}\}$
check ID_{FA}, S_2
$VID^* = SID \oplus h(h(x)||n_{HA})$
retrieve $(ID_{MN}, r_{MN}, VID, h(x_{MN}))$
check $VID^*(= h(ID_{MN}||r_{MN}))$
$C^* = n_{MN} \oplus VID$
check $V^*{}_1(= h(n_{HA}||C^*))$
$SK^* = h(h(x_{MN})||ID_{MN}||ID_{FA}||n_{MN}||n_{FA})$
check $V^*{}_2(= h(SK^*||n_{HA}))$
$K_1 = SK \oplus h(K_{FH}||n_{FA})$
$V_3 = h(ID_{FA}||h(x_{MN})||n_{MN})$
$S_3 = h(K_{FH}||n_{FA}||K_1||V_3)$
$m_7 = \{K_1, V_3, S_3\}$
check S_3
$SK = K_1 \oplus h(K_{FH}||n_{FA})$
$K_2 = SK \oplus h(SK||n_{MN})$
$m_8 = \{V_3, K_2\}$
check $V_3(= h(ID_{FA}||h(x_{MN})||n_{MN}))$, $SK(= K_2 \oplus h(SK||n_{MN}))$

Fig. 2. Authentication and session key establishment phases of the proposed scheme

7. Upon receiving m_4, MN records n_{HA} and n_{FA}, generates the shadow identity $SID = VID \oplus h(h(x)||n_{HA})$, the parameter $V_1 = h(n_{HA}||C)$, the session key $SK = h(h(x_{MN})||ID_{MN}||ID_{FA}||n_{MN}||n_{FA})$, the parameter $V_2 = h(SK||n_{HA})$, and the hashing value $S_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$. And then MN sends a message $m_5 = \{SID, V_1, V_2, n_{MN}, S_1, ID_{HA}\}$ to FA.
8. After receiving m_5 , FA uses the nonce n_{FA} with the received $SID, V_1, V_2,$ and n_{MN} to compute the hashing value $S^*{}_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$.If $S^*{}_1$ and S_1 are equivalent, FA computes the hashing value $S_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$ and sends $m_6 = \{SID, V_1, V_2, n_{MN}, S_2, ID_{FA}\}$ to HA to verify whether MN is legal.
9. After receiving m_5, FA computes $S^*{}_1 = h(n_{FA}||SID||V_1||V_2||n_{MN})$, and checks whether $S^*{}_1 = S_1$. If $S^*{}_1$ and S_1 are equivalent, FA computes the hashing value $S_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$ and sends $m_6 = \{SID, V_1, V_2, n_{MN}, S_2, ID_{FA}\}$ to HA to verify whether MN is legal.

10. After receiving m_6, HA checks ID_{FA} to determine whether it is an ally. Then, HA computes the hashing value $S^*_2 = h(K_{FH}||n_{HA}||SID||V_1||V_2||n_{MN})$ to check whether $S^*_2 = S_2$ using the corresponding secret key K_{FH} and the nonce n_{HA}. If the result is valid, the identity of FA is authenticated, and HA continues to check the validities of VID, PW^*_{MN}, and SK as follows:
    (a) HA computes the virtual identity $VID^* = SID \oplus h(h(x)||n_{HA})$. Then, HA retrieves a set of $MN's$ secret information $(ID_{MN}, r_{MN}, VID, h(x_{MN}))$ using VID^* as a search word to check whether $VID^* = h(ID_{MN}||r_{MN})$.
    (b) If the result is valid, HA computes $C^* = n_{MN} \oplus VID$ and $V^*_1 = h(n_{HA}||C^*)$ to check whether $V^*_1 = V_1$. Note that, the equivalence between V^*_1 and V_1 implies that PW^*_{MN} equals PW_{MN}.
    (c) If they are equal, HA computes $SK^* = h(h(x_{MN})||ID_{MN}||\ ID_{FA}||n_{MN}||n_{FA})$ and $V^*_2 = h(SK^*||n_{HA})$, and checks the validity of V^*_2.
11. Then, HA computes $K_1 = SK \oplus\ h(K_{FH}||n_{FA})$, $V_3 = h(ID_{FA}||h(x_{MN})||n_{MN})$, and $S_3 = h(K_{FH}||n_{FA}||K_1||V_3)$ and sends a message $m_7 = \{K_1, V_3, S_3\}$ to FA to inform that MN is a legal user.
12. With a message m_7, FA computes the hashing value $S^*_3 = h(K_{FH}||n_{FA}||K_1||V_3)$ to check whether $S^*_3 = S_3$. If they are equal, FA computes $SK = K_1 \oplus h(K_{FH}||n_{FA})$, $K_2 = SK \oplus h(SK||n_{MN})$ and sends a message $m_8 = \{V_3, K_2\}$ to MN.
13. After receiving m_8, MN computes $V^*_3 = h(ID_{FA}||(x_{MN})||n_{MN})$ and checks where $V^*_3 = V_3$. If the result is valid, MN computes $SK^* = K_2 \oplus h(SK||n_{MN})$. If SK^* and SK are equal, MN is sure that FA also has an authenticated session key. Then, MN records the authenticated session key SK for future communications.

4 Security Analysis

In this section, we analyze the proposed scheme in several respects to remedy weaknesses provided by Youn et al.

Anonymity against Passive Adversaries: Although a passive adversary can simply eavesdrop any messages including V_2 and K_2 between MN and FA, he/she cannot recover the identity of MN. Note that, V_2 does not contain ID_{MN}, K_2 provides no clue to find the user's identity and even ID_{MN} included in SID cannot be found since SID is just a combination of two hashing values. Therefore, the proposed scheme always provides a user's anonymity under the passive attacks.

Anonymity against Malicious Mobile User: Even the malicious mobile user who possesses a valid smart card issued by HA cannot find a legitimate user's identity. A malicious user MN' eavesdrops the messages between MN and FA, and replaces $HA's$ random nonce by n'_{HA} in a message m_3 or m_4. After eavesdropping $MN's$ shadow identity, $SID = VID \oplus h(h(x)||n'_{HA})$, MN' tries to recover $MN's$ identity. However, MN' only can get the virtual identity of MN by computing $VID =$

$SID \oplus SID' \oplus VID' = VID \oplus h(h(x)||n'_{HA}) \oplus VID' \oplus h(h(x)||n'_{HA}) \oplus VID'$. As a result, malicious user's attacks to recover the identities of other users' are impossible.

Security against Known Session Key Attacks: The session key established between MN and FA contains the secret key for MN, x_{MN}, that only HA knows. And the session key is used to compute V_2 as one of the input values of a hash function. Therefore, it is impracticable the exhaustive search to find ID_{MN} using V_2 and SK by the adversary who has already obtained SK. Consequently, the proposed scheme is secure against known session key attacks.

Security against Side Channel Attacks: It is assumed that an adversary can extract $h(x)$ from a smart card and eavesdrop any messages between MN and FA. Then, he/she can obtain the $MN's$ virtual identity by computing $VID = SID \oplus h(h(x)||n_{HA})$. However, he/she cannot recover the session key established between MN and FA and the identity of MN using $h(x)$ and VID, because SK contains a secret value $h(x_{MN})$ and ID_{MN} is concatenated with a secret parameter r_{MN} in VID.

5 Conclusion

In this paper, we presented the improved version of Chang et al.'s scheme. Some modifications such as an addition of extra secret values and a virtual identity are accomplished to improve their scheme. In other words, no combinations of transmission messages reveal user's identity and secret values. The improved scheme not only provides anonymity against passive adversaries and malicious mobile users, but also is resistant to known session key attacks and side channel attacks. It is still efficient and suitable for mobile environments by using only low-cost functions such as one-way hash functions and exclusive-OR operations. Therefore, the proposed scheme is more secure and still efficient in global mobility networks.

References

1. Zhu, J., Ma, J.: A new authentication scheme with anonymity for wireless environments. IEEE Transactions on Consumer Electronics 50(1), 231–235 (2004)
2. Lee, C.C., Hwang, M.S., Lio, I.E.: Security enhancement on a new authentication scheme with anonymity for wireless environments. IEEE Transactions on Industrial Electronic 53(5), 1683–1687 (2006)
3. Wu, C.C., Lee, W.B., Tsaur, W.J.: A secure authentication scheme with anonymity for wireless communications. IEEE Communications Letter 12(10), 722–723 (2008)
4. Mun, H., Han, K., Lee, Y.S., Yeun, C.Y., Choi, H.H.: Enhanced secure anonymous authentication scheme for roaming service in global mobility networks. Mathematical and Computer Modelling 55(1-2), 214–222 (2012)
5. Nam, J., Kim, S., Park, S., Won, D.: Security analysis of a nonce-based user authentication scheme using smart cards. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E90-A(1), 299–302 (2007)

6. Chang, C.C., Lee, C.Y., Chiu, Y.C.: Enhanced authentication scheme with anonymity for roaming service in global mobility networks. Computer Communications 32, 611–618 (2009)
7. Youn, T.Y., Park, Y.H., Lim, J.: Weaknesses in an anonymous authentication scheme for roaming service in global mobility networks. IEEE Communications Letter 13(7), 471–473 (2009)

Cryptanalysis of an Authenticated Group Key Transfer Protocol Based on Secret Sharing[*]

Mijin Kim[1], Namje Park[2], and Dongho Won[1,**]

[1] College of Information and Communication Engineering, Sungkyunkwan University,
2066 Seobu-ro, Jangan-gu, Suwon, Gyeonggi-do, 440-746, Korea
{mjkim,dhwon}@security.re.kr
[2] Department of Computer Education Teachers College,
Jeju National University, Jeju, Korea
namjepark@jejunu.ac.kr

Abstract. In 2012, Sun et al. proposed an authenticated group key transfer protocol based on secret sharing instead of encryption algorithm. They claimed that their protocol provides mutual authentication to ensure that only the authorized group members can recover the right session key and all participants only need to store one secret share for all sessions. However, our analysis shows that Sun et al.'s protocol is vulnerable to outsider and insider attacks and does not provide mutual authentication. In this paper, we show a detailed analysis of flaws in Sun et al.'s protocol.

Keywords: key exchange protocol, group key transfer, secret sharing, attack, confidentiality.

1 Introduction

Group key exchange protocols are cryptographic algorithms that characterize how a group of parties can communicate with their common secret key over public networks. In order to build secure multicast channels over public networks, various group key exchange protocols have been proposed over the years in a variety of environments [1,2,3,4,5,6,7,8]. Key exchange protocols are often classified into two types: key agreement protocols and key transfer protocols. Key agreement protocols require each participant to contribute its part to the final form of the session key, whereas key transfer protocols allow one trusted entity to generate the session key and then transfer it to all participants.

Recently, Sun et al. presented a group key transfer protocol based on secret sharing instead of encryption algorithm [8]. Sun et al.'s protocol only needs the server to broadcast n+1 messages at once in a round of distribution and all of the legal users

[*] This research was supported by the KCC(Korea Communications Commission), Korea, under the R&D program supervised by the KCA(Korea Communications Agency) (KCA-2012-12-912-06-003).

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 761–766, 2013.

only need to store one secret share in all conversations regardless of new addition or someone's takeoff. That is, when refreshing a group key or performing a new conversation, the original entities who are also included in current session need not to change their existing secret shares in case of members have changed. In addition, instead of reconstructing the interpolating polynomial, a simple computation is enough for each user to obtain the key. However, due to a flaw in Sun et al.'s protocol design, the protocol fails to achieve authenticated key exchange. In this work, we provide a security analysis on Sun et al.'s group key transfer protocol. Our analysis shows that Sun et al.'s protocol has a flaw in the design and can be easily attacked. The attacks we mount on Sun et al.'s protocol are insider attack and outsider attack.

This paper is organized as follows: Section 2 reviews Sun et al.'s group key transfer protocol. Section 3 presents security analysis of the Sun et al.'s protocol. Finally, Section 4 concludes this work.

2 Sun et al.'s Group Key Transfer Protocol

This section reviews Sun et al.'s group key transfer protocol [8]. The protocol assumes a trusted key generation center (KGC) who provides key distribution service to its registered users, and consists of two phases: user registration, group key generation and distribution. Sun et al. expected the least mutual dependence on others, so they adopted the following derivative secret sharing scheme.

Derivative Secret Sharing
Phase 1: Secret sharing

1. KGC splits S into two parts n times: $S = s_1 + s_1' = s_2 + s_2' = \cdots = s_n + s_n'$.
2. KGC sends P_i the share s_i', i=1,2,…,n, respectively in a secure channel.

Phase 2: Reconstruction

1. KGC broadcasts the shares $s_i, i = 1,2, \dots, n$, at once when users want to recover the secret.
2. P_i regains S by computing $S = s_1 + s_1'$.

The derivative secret sharing greatly reduces the mutual dependence on others. Detailed steps of these phases are described as follows.

2.1 Sun et al.'s Protocol

User Registration: Each user is requested to login to KGC for subscribing the group key distribution service. During registration, KGC shares a secret s_i' with each user U_i. In the following process, KGC keeps tracking the actions of all registered users and removing any unsubscribed users.

Group key generation and distribution:

1. The initiator, a designated user of the group, appeals for a group key distribution service by sending KGC $\{u_1, u_2, \dots, u_n\}$, which contains the identities of the registered users $U_1, U_2, \dots, U_n$, in current session.

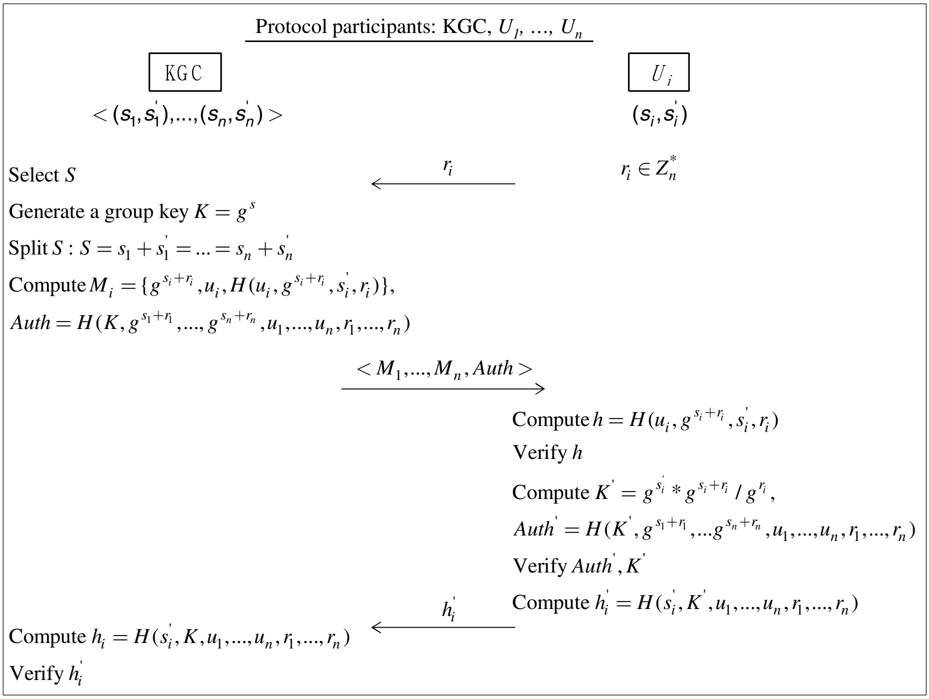

Protocol participants: KGC, U_1, ..., U_n
KGC
U_i
$<(s_1,s_1'),\dots,(s_n,s_n')>$
(s_i,s_i')
Select S
r_i
$r_i \in Z_n^*$
Generate a group key $K = g^s$
Split $S : S = s_1 + s_1' = \dots = s_n + s_n'$
Compute $M_i = \{g^{s_i+r_i}, u_i, H(u_i, g^{s_i+r_i}, s_i', r_i)\}$,
$Auth = H(K, g^{s_1+r_1}, \dots, g^{s_n+r_n}, u_1, \dots, u_n, r_1, \dots, r_n)$
$< M_1, \dots, M_n, Auth >$
Compute $h = H(u_i, g^{s_i+r_i}, s_i', r_i)$
Verify h
Compute $K' = g^{s_i'} * g^{s_i+r_i} / g^{r_i}$,
$Auth' = H(K', g^{s_1+r_1}, \dots g^{s_n+r_n}, u_1, \dots, u_n, r_1, \dots, r_n)$
Verify $Auth', K'$
h_i'
Compute $h_i' = H(s_i', K', u_1, \dots, u_n, r_1, \dots, r_n)$
Compute $h_i = H(s_i', K, u_1, \dots, u_n, r_1, \dots, r_n)$
Verify h_i'

Fig. 1. An execution of Sun et al.'s protocol (described from Step 3)

2. KGC broadcasts the list of all participants according to the above received message as a response.
3. Each U_i sends a random challenge r_i, i=1,2,…,n, to KGC.
4. KGC randomly selects S to generate the group key $K = g^S$ for current service and then invokes derivative secret sharing to split S into two parts n times such that $S = s_1 + s_1' = s_2 + s_2' = \cdots = s_n + s_n'$. KGC then computes: $M_i = \{g^{s_i+r_i}, u_i, H(u_i, g^{s_i+r_i}, s_i', r_i)\}$, i=1,2,…,n, and Auth=H(K, $g^{s_1+r_1}, \dots, g^{s_n+r_n}, u_1, \dots, u_n, r_1, \dots, r_n)$. At last, KGC broadcasts $\{M_1, \dots, M_n, \mathrm{Auth}\}$ to the users at once.
5. After receiving M_i and Auth, U_i firstly computes $h = H(u_i, g^{s_i+r_i}, s_i', r_i)$, where $g^{s_i+r_i}$ and u_i are from M_i, s_i' is the shared secret stored by U_i, r_i is the public value. And then U_i checks whether or not h is equal to the corresponding part in M_i. If any of the checks fails, U_i aborts; Otherwise, U_i then continues to compute $K' = g^{s_i'} * g^{s_i+r_i}/g^{r_i}$, $\mathrm{Auth}' = H(K', g^{s_1+r_1}, \dots, g^{s_n+r_n}, u_1, \dots, u_n, r_1, \dots, r_n)$ and checks whether or not K' can satisfy the condition that Auth' is identical to Auth. If so, then K' is just correct the group key K which is distributed by KGC.

6. Each user U_i returns a value $h_i' = H(s_i', K', u_1, \dots, u_n, r_1, \dots, r_n)$ to KGC. KGC computes $h_i = H(s_i', K, u_1, \dots, u_n, r_1, \dots, r_n)$ with its own s_i' and K, and checks whether or not $h_i' = h_i$. This review is to make sure that every user in current session has indeed obtained the correct group key.

3 Security Analysis

In this section, we analyze the security features of an authenticated group key transfer protocol based on secret sharing described in Section 2. One of the fundamental security goal of a key exchange protocol is key confidentiality which ensures that no one other than the intended users can compute the session key. In the Sun et al.'s protocol, this goal cannot be achieved. We reveal this security vulnerability of Sun et al.'s protocol.

3.1 Outsider Attacks

To an outside adversary, his motivation is to obtain the group key or share the group key with group participants. In the following analysis, we can see that his aim is fulfilled.

Method 1:

1. The adversary A can grasp $(M_1, \dots, M_2, \text{Auth})$ from the broadcast channel between KGC and authorized users U_i.
2. Since A knows $M_i = \{g^{s_i+r_i}, u_i, H(u_i, g^{s_i+r_i}, s_i', r_i)\}$ $(i = 1,2,\dots,n)$, A can obtain $H(u_i, g^{s_i+r_i}, s_i', r_i)$.
3. Then, A knows $(u_i, g^{s_i+r_i})$, and r_i is a public value, A can obtain s_i'.
4. A can obtain the session key K by calculating $K = g^{s_i'} * g^{s_i+r_i}/g^{r_i}$.

Method 2:

1. A can grasp $(M_1, \dots, M_2, \text{Auth})$ from the broadcast channel between KGC and authorized users U_i.
2. From $M_i = \{g^{s_i+r_i}, u_i, H(u_i, g^{s_i+r_i}, s_i', r_i)\}$ $(i = 1,2,\dots,n)$, A can obtain u_i and $g^{s_i+r_i}$.
3. Since r_i is the public value, A knows (Auth, $g^{s_1+r_1}, \dots, g^{s_n+r_n}, u_1, \dots, u_n, r_1, \dots, r_n$).
4. Thus, A can obtain the session key K From Auth=$H(K, g^{s_1+r_1}, \dots, g^{s_n+r_n}, u_1, u_n, r_1, \dots, r_n)$ by launching a guessing attack.

3.2 Insider Attacks

Every inside user in Sun et al. ' s protocol is able to reconstruct the group key but know nothing more extra information. For this purpose, the group key in Sun et al. ' s protocol is distributed to all authorized users U_i. Sun et al. care about the security on secret shares belonged to each participant. Sun et al. claimed that even if P_i can forge a random r_j' to impersonate another user P_j and receives corresponding response from KGC, P_i cannot forge the return response $H(s_j', K', u_1, \dots, u_n, r_1, \dots, r_n)$. However, our analysis shows that malicious inside user P_i can impersonate P_j as following.

1. A can grasp $(M_1, \dots, M_2, \mathrm{Auth})$ from the broadcast channel between KGC and authorized users U_i.
2. Since A knows $M_i = \{g^{s_i+r_i}, u_i, H(u_i, g^{s_i+r_i}, s_i', r_i)\}$ $(i = 1,2, \dots, n)$, A can obtain $H(u_i, g^{s_i+r_i}, s_i', r_i)$.
3. Then, A knows $(u_i, g^{s_i+r_i})$, and r_i is a public value, A can obtain s_i'.
4. Using the obtained s_j', malicious inside user P_i can forge the response message $H(s_j', K', u_1, \dots, u_n, r_1, \dots, r_n)$.

Thus, the inside adversary is able to obtain others' secret shares s_i' $(i = 1, \dots, n)$, and get the group session key without being detected.

Sun et al. claimed that the secret sharing protocol based on DLP guarantees the security of the secret shares. However, from the messages broadcasted between KGC and P_j, P_i can obtain $M_j = \{g^{s_j+r_j}, u_j, H(u_j, g^{s_j+r_j}, s_j', r_j)\}$, and r_j. Given $g^{s_j+r_j}$ and r_j, P_i is able to obtain s_j by computing $K = g^{s_j'} * g^{s_j+r_j}/g^{r_j}$. Since P_i knows $(h, u_j, g^{s_j+r_j}, r_j)$ where $h = H(u_j, g^{s_j+r_j}, s_j', r_j)$, P_i can launch a guessing attack in order to get s_j'. After getting the s_j', P_i is able to impersonate P_j.

4 Conclusion

In 2012, Sun et al. proposed a group key transfer protocol based on a special secret sharing scheme [8]. They claimed that their protocol provides mutual authentication to ensure that only the authorized group participants can recover the correct session key. However, our analysis shows that any inside or outside attacker can obtain the session key in Sun et al.'s protocol and be able to impersonate legal users. Therefore, Sun et al.'s protocol does not meet the fundamental security goal of a key exchange protocol. Future work could be undertaken to remedy Sun et al.'s protocol.

References

1. Shamir, A.: How to share secret. Communications of the ACM 22(11), 612–613 (1979)
2. Katz, J., Yung, M.: Scalable protocols for authenticated group key exchange. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 110–125. Springer, Heidelberg (2003)
3. Nam, J., Paik, J., Kim, U.M., Won, D.: Resource-aware protocol for authenticated group key exchange in integrated wired and wireless networks. Journal of Information Sciences 177, 5441–5467 (2007)
4. Hajyvahabzadeh, M., Eidkhani, E., Mortazavi, S.A., Pour, A.N.: A new group key management protocol using code for key calculation: CKC. Information Science and Applications, pp. 1–6 (2010)

5. Harn, L., Lin, C.: Authenticated group key transfer protocol based on secret sharing. IEEE Transactions on Computers 59(6), 842–846 (2010)
6. Nam, J., Paik, J., Won, D.: A security weakness in Abdalla et al.'s generic construction of a group key exchange protocol. Journal of Information Sciences 181(1), 234–238 (2011)
7. Nam, J., Kim, M., Paik, J., Won, D.: Security Weaknesses in Harn-Lin and Dutta-Barua protocols for group key establishment. KSII Transactions on Internet and Information Systems 6(2), 751–765 (2012)
8. Sun, Y., Wen, Q., Sun, H., Li, W., Jin, Z., Zhang, H.: An authenticated group key transfer protocol based on secret sharing. Procedia Engineering 9, 403–408 (2012)

Development of the STEAM-Based Media Education Materials for Prevention of Media Dysfunction in Elementary School[*]

Jaeho An and Namje Park[**]

Department of Computer Education, Teachers College, Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
{profirean,namjepark}@jejunu.ac.kr

Abstract. Indiscreet distribution of harmful content, excessive exposure to violent material and excessive use of the Internet are becoming an axis of social conflicts, and the damage is expanding to lower ages. At the introduction of media, we were in great expectation on its educational use. It has become, however, the most harmful environment from the aspect of education. Media education can be in two aspects: education to accept, product and utilize media to activate the positive functions; and protective and preventive education to minimize the negative functions. What is worthy of note is that media education can be carried out in the aspects of both the positive functions and the negative functions of media. This study suggests the media education program applicable to elementary schools, noting the aspects of both the positive functions and the negative functions of media.

Keywords: Media Education, STEAM, Media Dysfunction, Elementary School, Teaching Method.

1 Introduction

Indiscreet distribution of harmful content, excessive exposure to violent material and excessive use of the Internet are becoming an axis of social conflicts, and the damage is expanding to lower ages. At the introduction of media, we were in great expectation on its educational use. It has become, however, the most harmful environment from the aspect of education. However, we cannot overlook the positive functions of media. Then, can we approach this problem with the education which may minimize the negative functions and to utilize the positive functions of media? The answer is in the media education itself.

Media education can be in two aspects: education to accept, product and utilize media to activate the positive functions; and protective and preventive education to minimize the negative functions. What is worthy of note is that media education can

[*] This work was supported by the Korea Foundation for the Advancement of Science and Creativity(KOFAC) grant funded by the Korea government(MEST).

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 767–772, 2013.

be carried out in the aspects of both the positive functions and the negative functions of media. This paper suggests the media education program applicable to elementary schools, noting the aspects of both the positive functions and the negative functions of media.

2 Goal and Scope of Media Education Based on STEAM

Media education is a course of activities which promote critical understanding and evaluation capability, and furthermore, develop the capability of consumers to create own messages. In other words, media education is the way of developing capabilities to accept, produce and utilize media which were developed by the need of men. The media education to practice the integrated literacy to extend the media capability and to extend the communication capability can be performed in three areas: media acceptance, media production and media utilization.

Media acceptance is the area in which media consumers join a debate on media program and system, exploring the method to satisfy their desires through media, and the influence of media on their behavior. It also includes the process to analyze a media content to understand how the real world is constructed through media.

Media production is the activity of producing actively and directly the media contents. Starting with production of simple media contents, the media consumers go forward to produce a variety of thoughtful media contents. They also express through the media product their learning on media and their intentions to deliver in the technical aspect, and present and discuss on the production process, products and episodes.

Media utilization is the area in which consumers deliberate on how to utilize media contents. It is a course of asking, thinking, debating and utilizing the way of effective use of media contents. Critical acceptance of media contents and production of contents in a creative manner will help consumers to understand the problems of media contents and to find the solutions. By utilizing this course in everyday life, consumers can find the best way to identify and solve various problems.

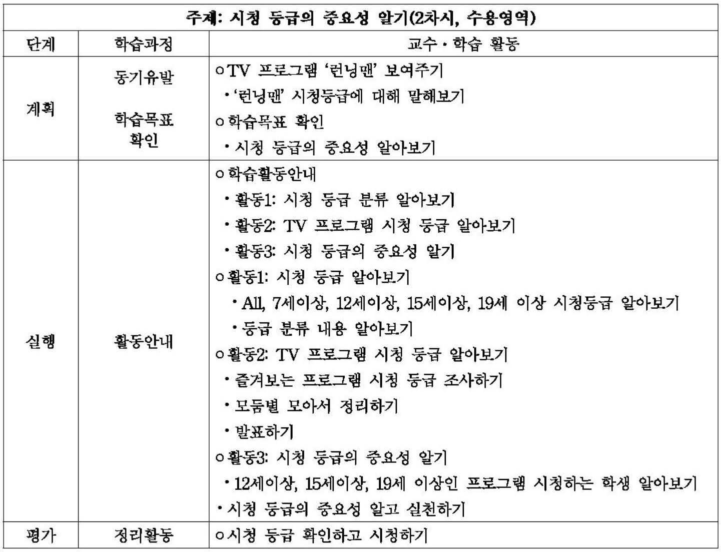
3 Method and Procedures of Proposed STEAM Program

Based on the goals and area of media education, this paper provides the 6 stages of programs applicable to the fifth/sixth year students.

In the area of acceptance, children can have the opportunities to reflect on their behavior for media as they think about the problems of media. In the area of production, children can produce public campaign advertisement on the problems of the rating system, developing the medial utilization capability. In the area of utilization, children can carry out campaign activities based on their production, learning how to deliver their idea effectively. In addition, schools can deliver the campaign advertisement produced by the children through the school newsletter, educating their parents.

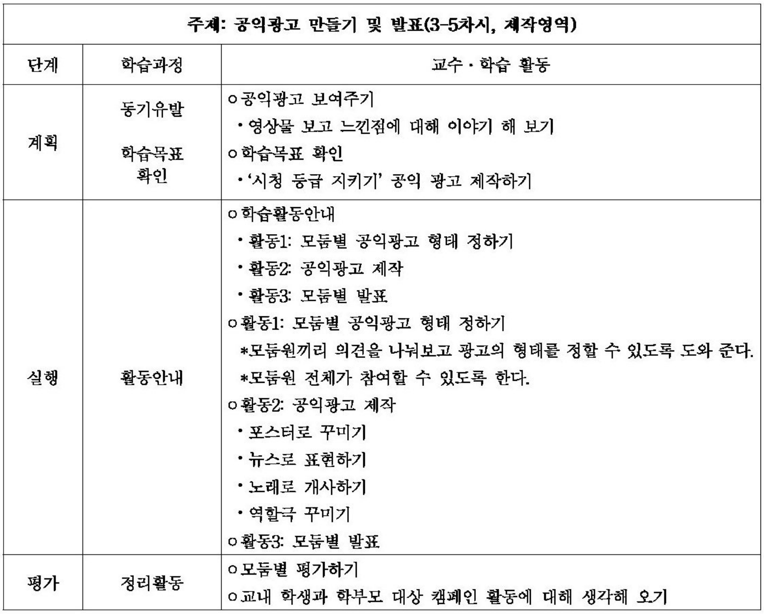
Table 1. STEAM-based Media Education Program

Message	Stage	Subject	Contents
Login	1	Understanding problems of media (acceptance)	- Understanding problems of media - Exploring our favorite media - Finding advantages and disadvantages of media
Login Response	2	Understanding importance of the rating system (acceptance)	- Understanding rates by media - Examining rates of favorite media - Understanding importance of rating system
Logout	3-4	Making public campaign advertisement (production)	- Understanding contents of public campaign advertisements - Drawing posters - Writing news - Changing lyrics - Role play
Ping	5	Presentation	- Group presentation
Pong	6	Campaign activities (activity)	- Participating in the campaign activity with the presented content - Evaluation



주제: 시청 등급의 중요성 알기(2차시, 수용영역)		
단계	학습과정	교수・학습 활동
계획	동기유발 학습목표 확인	ㅇTV 프로그램 ‘런닝맨’ 보여주기 ・‘런닝맨’ 시청등급에 대해 말해보기 ㅇ학습목표 확인 ・시청 등급의 중요성 알아보기
실행	활동안내	ㅇ학습활동안내 ・활동1: 시청 등급 분류 알아보기 ・활동2: TV 프로그램 시청 등급 알아보기 ・활동3: 시청 등급의 중요성 알기 ㅇ활동1: 시청 등급 알아보기 ・All, 7세이상, 12세이상, 15세이상, 19세 이상 시청등급 알아보기 ・등급 분류 내용 알아보기 ㅇ활동2: TV 프로그램 시청 등급 알아보기 ・즐겨보는 프로그램 시청 등급 조사하기 ・모둠별 모아서 정리하기 ・발표하기 ㅇ활동3: 시청 등급의 중요성 알기 ・12세이상, 15세이상, 19세 이상인 프로그램 시청하는 학생 알아보기 ・시청 등급의 중요성 알고 실천하기
평가	정리활동	ㅇ시청 등급 확인하고 시청하기

Fig. 1. Example of teaching guidance methods (Korea’s document format)



주제: 공익광고 만들기 및 발표(3-5차시, 제작영역)		
단계	학습과정	교수 · 학습 활동
계획	동기유발 학습목표 확인	ㅇ공익광고 보여주기 • 영상물 보고 느낀점에 대해 이야기 해 보기 ㅇ학습목표 확인 •'시청 등급 지키기' 공익 광고 제작하기
실행	활동안내	ㅇ학습활동안내 • 활동1: 모둠별 공익광고 형태 정하기 • 활동2: 공익광고 제작 • 활동3: 모둠별 발표 ㅇ활동1: 모둠별 공익광고 형태 정하기 *모둠원끼리 의견을 나눠보고 광고의 형태를 정할 수 있도록 도와 준다. *모둠원 전체가 참여할 수 있도록 한다. ㅇ활동2: 공익광고 제작 • 포스터로 꾸미기 • 뉴스로 표현하기 • 노래로 개사하기 • 역할극 꾸미기 ㅇ활동3: 모둠별 발표
평가	정리활동	ㅇ모둠별 평가하기 ㅇ교내 학생과 학부모 대상 캠페인 활동에 대해 생각해 오기

Fig. 2. Example of teaching guidance methods (Korea's document format)

4 Conclusion

In the information-oriented society of the 21st century, creative problem-solving capability knowledge will become the core element of competitiveness. In other words, media is the new foundation and basic tools for education in the 21st century. In the information-oriented society, in order to lead the successful learning, media education which is well-matched with the information society is required.

Media education shall provide systematically the opportunities for consumers to share and select various media experiences. Media education will develop the independent sense on media and the capability to produce creative and unique information for the digital generation. Education is a competitive power for the future. Education must aim at providing the capability to respond actively to the environment changed by new languages, new media and new culture. The program suggested above can be the typical example of the media education. This program helps children to think and debate on the negative functions of media, to produce campaigns by themselves, to apply them to the real world (campaign activity), and hence, to develop capability to respond proactively to the media environment. This may be the example of the program which considers both the negative functions and the positive functions of media.

References

1. Kang, E.-K., Park, N.: A Study on Media Education for Prevention of Media Disfunction in Elementary School's 5th & 6th Grade Students. In: Korea Information Processing Society Fall Conference 2012, vol. 19(2), pp. 1717–1719 (2012)
2. Lee, J.W., Park, N.: Individual information protection in smart grid. In: Kim, T.-h., Stoica, A., Fang, W.-c., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 153–159. Springer, Heidelberg (2012)
3. Kang, S.: Korean Morphological Analysis and Information Retrieval. Hongreung Science Press (2002)
4. Support Vector Machine, Wikipedia. the free Encyclopedia, `http://en.wikipedia.org/wiki/SVM`
5. Thesaurus, Wikipedia, The Free Encyclopedia (2008), `http://en.wikipedia.org/wiki/Thesaurus`
6. Frakes, W., Baeza-Yates, R.: Information Retrieval: Data Structures and Algorithms. Prentice Hall (1992)
7. Kim, Y., Park, N., Hong, D., Won, D.: Context-based classification for harmful web documents and comparison of feature selecting algorithms. Journal of Korea Multimedia Society 12(6) (2009)
8. Yang, Y., Pederson, J.: A Comparative Study on Feature Selection in text Categorization. In: Proceedings of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
9. Park, N., Kwak, J., Kim, S., Won, D., Kim, H.: WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 741–748. Springer, Heidelberg (2006)
10. Park, N.: Security scheme for managing a large quantity of individual information in RFID environment. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) ICICA 2010. CCIS, vol. 106, pp. 72–79. Springer, Heidelberg (2010)
11. Park, N.: Secure UHF/HF Dual-Band RFID: Strategic Framework Approaches and Application Solutions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 488–496. Springer, Heidelberg (2011)
12. Park, N.: Implementation of Terminal Middleware Platform for Mobile RFID computing. International Journal of Ad Hoc and Ubiquitous Computing 8(4), 205–219 (2011)
13. Park, N., Kim, Y.: Harmful Adult Multimedia Contents Filtering Method in Mobile RFID Service Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS, vol. 6422, pp. 193–202. Springer, Heidelberg (2010)
14. Park, N., Song, Y.: AONT Encryption Based Application Data Management in Mobile RFID Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS, vol. 6422, pp. 142–152. Springer, Heidelberg (2010)
15. Park, N.: Customized Healthcare Infrastructure Using Privacy Weight Level Based on Smart Device. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. CCIS, vol. 206, pp. 467–474. Springer, Heidelberg (2011)
16. Park, N.: Secure Data Access Control Scheme Using Type-Based Re-encryption in Cloud Environment. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods. SCI, vol. 381, pp. 319–327. Springer, Heidelberg (2011)
17. Park, N., Song, Y.: Secure RFID Application Data Management Using All-Or-Nothing Transform Encryption. In: Pandurangan, G., Anil Kumar, V.S., Ming, G., Liu, Y., Li, Y. (eds.) WASA 2010. LNCS, vol. 6221, pp. 245–252. Springer, Heidelberg (2010)

18. Park, N.: The Implementation of Open Embedded S/W Platform for Secure Mobile RFID Reader. The Journal of Korea Information and Communications Society 35(5), 785–793 (2010)
19. Kim, Y., Park, N.: Development and Application of STEAM Teaching Model Based on the Rube Goldberg's Invention. LNEE, vol. 203, pp. 693–698. Springer (2012)
20. Park, N., Cho, S., Kim, B., Lee, B., Won, D.: Security Enhancement of User Authentication Scheme Using IVEF in Vessel Traffic Service System. LNEE, vol. 203, pp. 699–705. Springer (2012)
21. Kim, K., Kim, B., Lee, B., Park, N.: Design and Implementation of IVEF Protocol Using Wireless Communication on Android Mobile Platform. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 94–100. Springer, Heidelberg (2012)
22. Ko, Y., An, J., Park, N.: Development of Computer, Math, Art Convergence Education Lesson Plans Based on Smart Grid Technology. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 109–114. Springer, Heidelberg (2012)
23. Kim, Y., Park, N.: The Effect of STEAM Education on Elementary School Student's Creativity Improvement. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech/CA/CES3 2012. CCIS, vol. 339, pp. 115–121. Springer, Heidelberg (2012)

Access Control Technique of Illegal Harmful Contents for Elementary Schoolchild Online Protection*

Namje Park** and Yeonghae Ko

Department of Computer Education, Teachers College, Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
{namjepark,smakor}@jejunu.ac.kr

Abstract. This paper is about access control of illegal and objectionable contents for child online protection coming through the combination of smart phone device, as the core technology of ubiquitous environments. To overcome the shortcoming of simple access control of illegal and objectionable contents on current internet, we suggest a framework for content-based classification and propose an access control of illegal and objectionable content's framework architecture using it. This paper suggests a solution to block the access of the illegal harmful contents by classifying the contents in detail instead of the existing simple user age classes for the multimedia contents.

Keywords: Access Control, Harmful Contents, schoolchild Online Protection, objection, Illegal contents.

1 Introduction

As the smartphone became the bare necessity of the modern people, many services using them are being offered and such trends are predicted to continue. However, in such an environment, the children are exposed to harmful contents, such as violent or pornographic materials online and it has become a main focus of the world to protect the children or the teenagers from such an exposure. However, to solve this problem, a national legal stance as well as a technological solution is required, which will only be possible with the mutual cooperation of the main users, such as the service providers, parents, and children, as well as the government.

The environment of new integrative services and smart tools, the control over harmful contents is crucial to continue to receive convenient services. In other words, if the smartphones owned by minors provide various pieces of online information, then the minors can be easily exposed to the adult contents, which necessitate a solution to prevent the approach to the illegal harmful contents. The current adult certification offered in the high speed internets utilize the Resident Registration Number, which includes a simple calculation method where the Resident Registration

* This work was supported by the Korea Foundation for the Advancement of Science and Creativity(KOFAC) grant funded by the Korea government(MEST).

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 773–780, 2013.

Number algorithm is used, inquiring to the financial information institution using the Resident Registration Number and the name, or using the real name when logging in without further entry. Other than these simple methods, the cell phone number, Resident Registration Number, and public certificate can be used in combination to confirm the age of the user in a systematic way. However, in the new integrative IT services and smart equipments, a new prevention for illegal harmful information contents is required to protect the children online.

This paper suggests a solution to block the access of the illegal harmful contents by classifying the contents in detail instead of the existing simple user age classes for the multimedia contents.

2 Suggested Applied Security Method for the Illegal File Sharing

This chapter suggests an applied security method on the illegal file sharing for the harmful information certificate service structure mentioned in the previous chapter. In many internet applications, the PKI-based services are offered to provide certificates or message perfection. In this case, the transmitted messages include the signatures of the sender and the receiver can certify the sender's signatures with the help of the CA. Currently, there are many applications that provide PKI-based services, but it has not been used in the P2P where the individuals directly connect to share files online. This is partially because of the various characteristics of the P2P and the inability to provide the compatibility among international PKI is another factor to make its application in global P2P more difficult. In addition, the establishment of PKI infrastructures, certificate issuance, and management are quite costly.

In the case of file sharing open source project, JXTA, the open keys, instead of the PKI, were used. In this structure, each peer creates his own certificate including an open key and distributes them in the peer advertise message transmission. This is relatively simple, but because there is no third trusted institution to certify the open keys, the MITM attacks will not be appropriate responded. For example, if an attacker in the middle alters the peer advertise message from peer A and replaces peer A's open keys, this action will not be sensed by anyone.

The method suggested in this study allows a safe file sharing application without using PKI by distributing self-made safe open keys. This suggestion has a basic structure of creating and distributing the open key/secret key pair for each peer and is protected against the MITM attacks.

2.1 Target Application Basic Action in Suggested Security Structures

The file sharing application targeted in this study has 2-layer structures of super peer basis. In this structure, each peer has a distinguishable ID. In addition, each peer needs to have their ID and password authenticated by the server to participate in the service network. In this stage, each peer receives the information of the super peer from the server and based on this information, a super peer is selected to send the Join message. If the Join message is successful, then the peer can deliver the meta information of the files possessed to the super peer. In this study, the self-made open key/secret key pairs are used and the authentication server is not the CA server from PKI, but only an authenticator for the ID and password.

The basic actions of the file sharing application targeted are shown in the picture below. The super peer in the picture possesses the meta information on the files of the each peer on the virtual domain. The peer that requests the resource search sends the request to the super peer and the super peer delivers the file search request to the other super peers to perform the search.

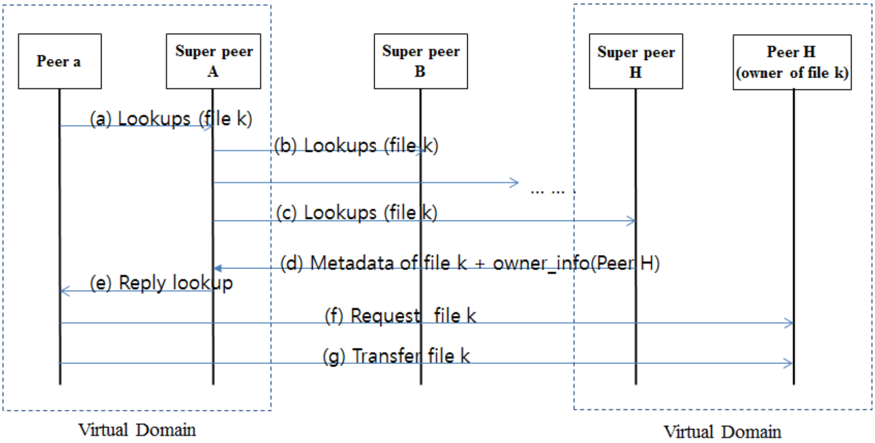


Fig. 1. File Share's Basic Process

2.2 Security Structure for Trusted File Sharing Application

The table 1 shows the symbols used in the security structure design. The super peer is assumed to be selected by the application. In addition, each super peer is assumed to have the open key of the server and the open keys of other super peers who participate in the service network currently.

In the framework suggested in this study, each peer creates an open key to register with the authenticating server. In the equation, Pa and Pb stand for the peers with the IDs a and b, respectively. S means the ID authenticating server. The detailed explanation of each stage is as follows:

Table 1. Symbols

	Contents
ID_k	Peer K's ID
IP_k	Peer K's IP IP Address
$K^u{}_k$	Peer K's Public Key
$K^r{}_k$	Peer K's Private Key
$K^u{}_s$	ID CA's Public Key
$K^r{}_S$	ID CA's Private Key
PW_k	Peer K's Password
$E_k(m)$	Encryption Function
$D_k(c)$	Decryption Function
$S_k(m)$	Digital Signatures

Stage 1: ID, Password Authentication and Open Key Registration

① ID, password authentication
② (Peer a -> Server) a's open key registration
③ (Server) Peer ID authentication
- Decode with the individual keys on the server
- authenticate through the comparison with the registered password saved in the message by the peer
- authenticate the signature through the peer's password

④ (server->peer a) return to success/failure of the open key registration
- transmit the list of super peers, IP, ID, open key information
- Transmit the coded open key of Peer a

⑤ (Peer a) Authenticate certifying server ID
- Decode with individual key
- Authenticate the signature with the open key of the server

Until stage 1, the peer will be authenticated based on the password, but after the peer's open key is registered in the server on stage 1, the peer's open key will be usable.

Stage 2: Join Request

① (Peer a -> super peer A) Join request
- transmit the open key when requesting to Join
- the super peer's open key will code and the individual key will sign to transmit

Stage 3: Peer a's Open Key Request and Response

① (Super peer A -> Server) request open key of peer a
② (server) authenticate super peer A's ID

- Authenticate super peer A’s ID
- Decode with the server’s individual key
- Authenticate the signature with the open key of super peer A

③ (Server-> super peer A) Return peer a’s open key

④ (super peer A) Authenticate server ID
- Decode with individual key
- authenticate the signature with the server’s open key

⑤ (super peer A) Authenticate peer a’s signature that is included in the message received in stage 2

Stage 4: Join Success or Failure

① Confirm Join success/failure

② (Peer a) Authenticate super peer A’s ID
- Decode with individual key
- Authenticate signatures with super peer A’s open key

Stage 5: Peer a’s File Sharing Meta Data Transmission

The 5 steps above are the service network Join steps. The safe file sharing security application based on these steps act as the following:

This mechanism reduces the server load as the management function of storing and providing each peer’s open key is distributed to each super peer. In addition, the open key is distributed naturally and safely in the message deliveries as each peer searches for files and returns the results. If the super peer experiences an error, then each peer can select another super peer from the list of super peers received during the ID authentication from the server and sends the Join message to register with another virtual domain.

The picture above shows the file search process after the network Join in the file sharing. The messages transmitted in each stage are shown in <Table 2>.

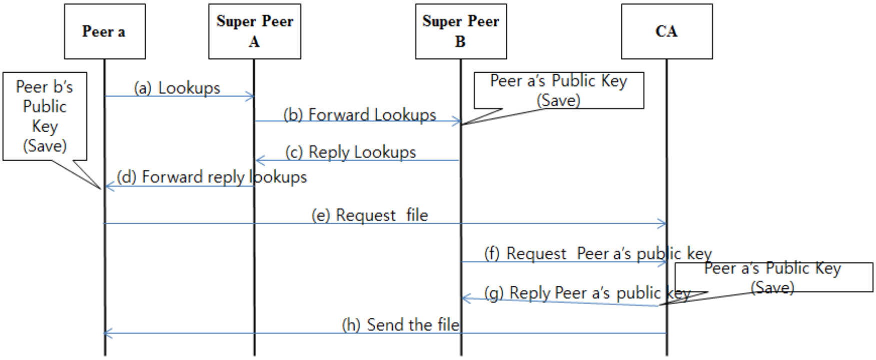

Peer a
Super Peer A
Super Peer B
CA
Peer b's Public Key (Save)
(a) Lookups
(b) Forward Lookups
Peer a's Public Key (Save)
(c) Reply Lookups
(d) Forward reply lookups
(e) Request file
(f) Request Peer a's public key
Peer a's Public Key (Save)
(g) Reply Peer a's public key
(h) Send the file

Fig. 2. File Share’s Basic Process

Table 2. Message Format

	Message Format
a	$E_{K^{u}_{A}}(\text{“looup\_req”} \mid \{\text{“beyonce”}\}) \mid S_{K^{r}_{u}}(m)$
b	$E_{K^{u}_{B}}(\text{“looupfile”} \mid \{\text{“beyonce”}\}) \mid \text{“requester”} \mid \{ID_a \mid IP_a \mid K^{u}_{a}\}) \mid S_{K^{r}_{A}}(m)$
c	$E_{K^{u}_{A}}(\text{“replylooup”} \mid \{\text{“beyonce”}\}) \mid fileid \mid \text{“owner”} \mid \{ID_b \mid IP_b \mid K^{u}_{b}\}) \mid S_{K^{r}_{B}}(m)$
d	$E_{K^{u}_{a}}(\text{“replylooup”} \mid \{\text{“beyonce”}\}) \mid \text{“owner”} \mid fileid \mid \{ID_b \mid IP_b \mid K^{u}_{b}\}) \mid S_{K^{r}_{A}}(m)$
e	$E_{K^{u}_{b}}(\text{“requestfile”} \mid fileid \mid S_{K^{r}_{a}}(m)$
f	$E_{K^{u}_{B}}(\text{“request\_pkey”} \mid fileid \mid ID_a \mid S_{K^{r}_{b}}(m)$
g	$E_{K^{u}_{b}}(\text{“reply\_pkey”} \mid \{ID_a \mid K^{u}_{a}\}) \mid S_{K^{r}_{B}}(m)$

3 Conclusion

The teenagers and children are exposed to many threats online, such as pornography, violent materials, and unhealthy information. To protect the children online, not only the technological security methods but also the legal measures, improved systems, and use education (service provider, parents, information, children, etc) are required. This study suggested a method of preventing the harmful information delivery t the teenagers in the network service environment. The suggested method carries all of the advantages of harmful material prevention methods using the classification of contents and also presents the service organization and harmful information authentication service based functions that can prevent the harmful contents in the new integrative service environment to be a reference for a practical solution.

References

1. Burges, C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
2. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys 43(1), 1–47 (2002)
3. Siolas, G., d'Alche-Buc, F.: Support Vector Machines based on a Semantic Kernel for Text Categorization. In: Proceeding of IJCNN 2000, vol. 5(1), pp. 205–209 (2000)
4. Lee, J.W., Park, N.: Individual Information Protection in Smart Grid. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 153–159. Springer, Heidelberg (2012)
5. Kang, S.: Korean Morphological Analysis and Information Retrieval. Hongreung Science Press (2002)
6. Support Vector Machine, Wikipedia. the free Encyclopedia, `http://en.wikipedia.org/wiki/SVM`
7. Thesaurus, Wikipedia, The Free Encyclopedia (2008), `http://en.wikipedia.org/wiki/Thesaurus`

8. Frakes, W., Baeza-Yates, R.: Information Retrieval: Data Structures and Algorithms. Prentice Hall (1992)
9. Kim, Y., Park, N., Hong, D., Won, D.: Context-based classification for harmful web documents and comparison of feature selecting algorithms. Journal of Korea Multimedia Society 12(6) (2009)
10. Yang, Y., Pederson, J.: A Comparative Study on Feature Selection in text Categorization. In: Proceedings of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
11. Park, N., Kwak, J., Kim, S., Won, D., Kim, H.: WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 741–748. Springer, Heidelberg (2006)
12. Park, N.: Security scheme for managing a large quantity of individual information in RFID environment. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) ICICA 2010. CCIS, vol. 106, pp. 72–79. Springer, Heidelberg (2010)
13. Park, N.: Secure UHF/HF Dual-Band RFID: Strategic Framework Approaches and Application Solutions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 488–496. Springer, Heidelberg (2011)
14. Park, N.: Implementation of Terminal Middleware Platform for Mobile RFID computing. International Journal of Ad Hoc and Ubiquitous Computing 8(4), 205–219 (2011)
15. Park, N., Kim, Y.: Harmful Adult Multimedia Contents Filtering Method in Mobile RFID Service Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 193–202. Springer, Heidelberg (2010)
16. Park, N., Song, Y.: AONT Encryption Based Application Data Management in Mobile RFID Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 142–152. Springer, Heidelberg (2010)
17. Park, N.: Customized Healthcare Infrastructure Using Privacy Weight Level Based on Smart Device. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. CCIS, vol. 206, pp. 467–474. Springer, Heidelberg (2011)
18. Park, N.: Secure Data Access Control Scheme Using Type-Based Re-encryption in Cloud Environment. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 319–327. Springer, Heidelberg (2011)
19. Park, N., Song, Y.: Secure RFID Application Data Management Using All-Or-Nothing Transform Encryption. In: Pandurangan, G., Anil Kumar, V.S., Ming, G., Liu, Y., Li, Y. (eds.) WASA 2010. LNCS, vol. 6221, pp. 245–252. Springer, Heidelberg (2010)
20. Park, N.: The Implementation of Open Embedded S/W Platform for Secure Mobile RFID Reader. The Journal of Korea Information and Communications Society 35(5), 785–793 (2010)
21. Kim, Y., Park, N.: Development and Application of STEAM Teaching Model Based on the Rube Goldberg's Invention. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) Computer Science and its Applications. LNEE, vol. 203, pp. 693–698. Springer, Heidelberg (2012)
22. Park, N., Cho, S., Kim, B.-D., Lee, B., Won, D.: Security Enhancement of User Authentication Scheme Using IVEF in Vessel Traffic Service System. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) Computer Science and its Applications. LNEE, vol. 203, pp. 699–705. Springer, Heidelberg (2012)

23. Kim, K., Kim, B.-D., Lee, B., Park, N.: Design and Implementation of IVEF Protocol Using Wireless Communication on Android Mobile Platform. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 94–100. Springer, Heidelberg (2012)
24. Ko, Y., An, J., Park, N.: Development of Computer, Math, Art Convergence Education Lesson Plans Based on Smart Grid Technology. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 109–114. Springer, Heidelberg (2012)
25. Kim, Y., Park, N.: The Effect of STEAM Education on Elementary School Student's Creativity Improvement. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 115–121. Springer, Heidelberg (2012)

The Concept of Delegation of Authorization and Its Expansion for Multi Domain Smart Grid System[*]

Mijin Kim[1] and Namje Park[2,**]

[1] College of Information and Communication Engineering, Sungkyunkwan University,
2066 Seobu-ro, Jangan-gu, Suwon, Gyeonggi-do, 440-746, Korea
mjkim@security.re.kr

[2] Department of Computer Education, Teachers College, Jeju National University,
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
namjepark@jejunu.ac.kr

Abstract. Current smart grid service protocol is not defined as a standard but is only material items for security procedures, message, and policies. In other words, this has no contents on the standards of elucidating delivery security for network safety and data perfection or any standards or requirements on the security for a safe information exchange. Therefore, sensitive information between nations needs to consider many different factors, which may require a complex security structure. Therefore, this paper suggests a security structure that can automatically be connected based on characteristics while sending security policy and security request messages when the smart grid users desire a connection with standard service protocol.

Keywords: Access Control, Multi domain, Smart grid Security, Authorization.

1 Introduction

Smart grid is a new electricity grid which transmits and distributes electricity intelligently by converging the information technology into the traditional electricity grid. Recently, the smart grid projects are promoted rapidly because the 'Green IT' becomes more and more interesting. However, Modernization of the grid will increase the level of personal information detail available as well as the instances of collection, use and disclosure of personal information. Thus, smart grid data users must consider carefully how they will protect the integrity, privacy, and security of the smart grid data obtained from consumer usage patterns, and the data collected must not be excessive. In addition, smart grid data must be gathered responsibly, securely, and with a measure of transparency and consumer control.

[*] This work (Grants No.C0033984) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2012.

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 781–787, 2013.

This paper suggests a security structure that can automatically be connected based on characteristics while sending security policy and security request messages when the smart grid users desire a connection with standard service protocol.

2 Proposed Service Structure and Security Architecture

2.1 Linking Service Structure among Domains

The linking service among domains (gateway service) is regionally or globally connected as shown in fig. 1.

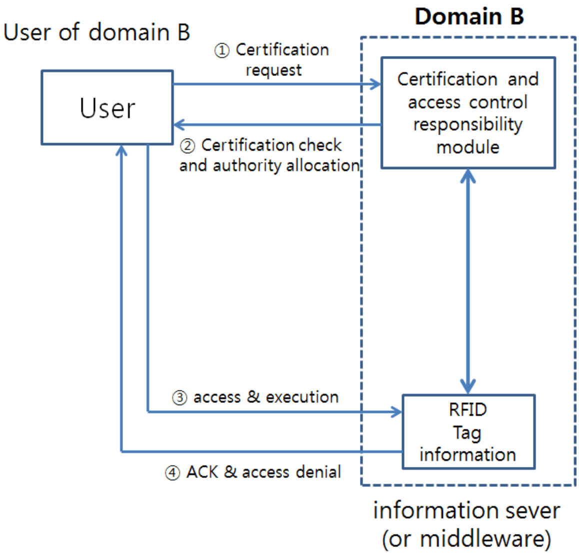

Domain B
User of domain B
① Certification request
User
Certification and access control responsibility module
② Certification check and authority allocation
③ access & execution
RFID Tag information
④ ACK & access denial
information sever (or middleware)

Fig. 1. The linking service among domains (gateway service)

The domain headquarters can manage the policies and authorities with other domains and is organized in a hierarchical manner by the domain headquarters to separate the roles of the control even with the centers under different departments through transfer of right to control.

2.2 Mutual Linkage Domain Security Requirements

In the interlinked domains, the mutual data provision, sharing and safe linkage service by authority requires the following requirements to consider the service aspect. This includes the basic security requirements of authentication, authorization, data protection as well as the service business aspect and physical security of the main facilities.

- Authentication : certification of the service users related to both service providers and users. In addition, the M2M (Machine to Machine) between the domains or the users for the IVEF services are included.
- Authorization : Authorization for the service. Only related to the service client. In addition, the system linkage for domain connection and the users for IVEF services are included.
- Data Protection : Safety of data sent and received by IVEF client. Related to actually exchanged data between systems or between system and user.
- Business Security for Service : Business security of the IVEF manager and users at the service provider, VTS center, and is determined by the business policy based on the negotiation between service users and businessmen or the provider's fees.
- Physical security : security against physical approach for the places where the IVEF client and server system exist or mutually connected control centers as well as the network access points

3 Domain Security Factor Definition and Management Flow

This clause defines the mutual security factors between domains and detailed procedures using the defined security messages.

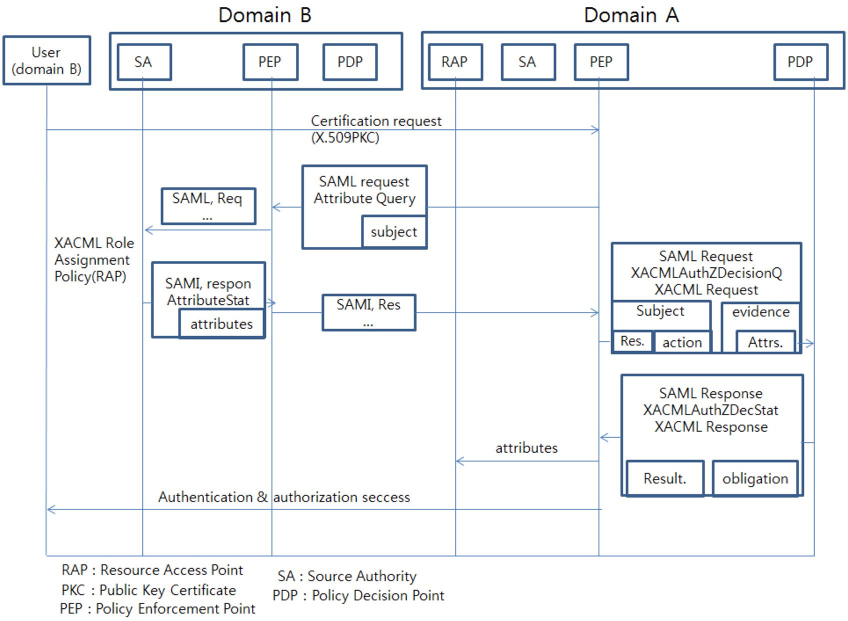

Domain B
Domain A
User (domain B)
SA
PEP
PDP
RAP
SA
PEP
PDP
Certification request (X.509PKC)
SAML, Req ...
SAML request Attribute Query
subject
XACML Role Assignment Policy(RAP)
SAMI, respon AttributeStat
attributes
SAMI, Res ...
SAML Request XACMLAuthZDecisionQ XACML Request
Subject
evidence
Res.
action
Attrs.
SAML Response XACMLAuthZDecStat XACML Response
attributes
Result.
obligation
Authentication & authorization seccess
RAP : Resource Access Point
PKC : Public Key Certificate
PEP : Policy Enforcement Point
SA : Source Authority
PDP : Policy Decision Point

Fig. 2. Domain Security management flow

In other words, fig. 2 shows the security management flow map on the linking areas with the security messages where the smart grid domain B approaches smart grid domain A. The basic security structure uses the XML based standard protocols and the characteristics for smart grid service are expanded using the smart grid security message characteristic exchange protocol. The approach management procedures according to the procedures and authorities for the policy management within a domain when the domains are linked are shown in Fig. 2. After the smart grid service between the domains is requested, the smart grid service basic certification mechanism based on ID/Password with the access limitation based authority function is as follows.

1) The user sends the access request to use the system resources or application service. At this time, the access request is same as the existing methods with user ID and password.
2) The PEP of the access control receives the access request and confirms the user's ID and password with the access control list. This is same as the previous method.
3) Once the PEP (Policy enforcement Point) confirms the user ID and password, it will transmit the user ID and the requested items (read, write, execute) to PDP (Policy Decision Point).
4) PDP loads the policy from PAP (Policy Administration Point) and determines whether the user has the appropriate authorities for the requested actions. For this, the user, resource, environmental characteristics, and policy are used to determine whether to approve.
5) PDP delivers the result to PEP. In other words, approval/denial is delivered to PEP. When it is "approved," the user certificate is examined and if it is valid, then the user request is approved.
6) PEP downloads the user certificate from the storage and checks for validity. If it is valid, it approves the access.

The smart grid basic authority management certificate mechanism procedures with access limitation do not have a huge advantage because the smart grid system becomes more complex and the costs for the access control increases compared to the existing system, but it makes the integration of the access control system easier. Because it uses the standard technology of XACML and X.509 PMI, if the newly added system requires access control, the access determination does not have to be realized again.

In other words, the additional open access control function in smart grid basic authority management certificate access control system only needs to add the communication between the user and PEP and the rest can be on the existing realizations. Even if it is an independent system, the access control function can be reused and it can reduce the costs for access control when adding the service system.

The conditional user access control mechanism for smart grid mutual linkage is used when smart grid domain B approaches the overall smart grid service system provided by the external system through the internal system and works as the following. The smart grid domain B user opens an account based on the overall smart grid service system certification and authority mechanism and also opens an account at the smart grid service system. Before the overall smart grid service system can approve the approach of the smart grid domain B user, the smart grid service system inquires the overall service related characteristics.

In addition, the smart grid service system can have its own PEP and PDP. In this case, PDP returns the authority decisions attached to the conditional tasks on the smart grid service characteristics. Therefore, PEP transmits the authority decision request to the external service. smart grid service receives the authority decision request from PEP and returns the characteristic token and this is used to examine the conditional tasks from PEP on the smart grid service system. The following is the conditional user access control mechanism procedures for smart grid domain linkage.

(1) smart grid domain B requests the smart grid service system access.
(2) smart grid service system transmits the authority decision request to service. This is performed by the PEP within the smart grid service system or the external PEP of smart grid service system.
(3) Once smart grid service receives the authority decision request from the PEP of smart grid service system delivers the authority decision request to PEP.
(4) PEP transmits the XAML characteristic decision request to PDP.
(5) PDP confirms the smart grid domain B account, characteristics, and policy to transmit the approval/denial decision to PDP.
(6) If the authority is approved, PDP will transmit the SAML characteristic token to PEP.
(7) PEP transmits SAML token.
(8) smart grid service transmits the token to smart grid service system.
(9) smart grid service system provides the smart grid domain A service to smart grid domain B.

4 Conclusion

This study designed the security structures on the smart grid system linkage based on mutual networking technology for future smart grid and suggested a mutually safe management structures, which brings the future direction of detailed results on practical application of function analysis and service platform.

References

1. Bellare, M., Rogaway, P.: Optimal asymmetric encryption. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (1995)
2. Kuwakado, H., Tanaka, H.: Strongly non-separable encryption mode for throwing a media away. Technical Report of IEICE 103(417), 15–18 (2003)
3. Siolas, G., d'Alche-Buc, F.: Support Vector Machines based on a Semantic Kernel for Text Categorization. In: Proceeding of IJCNN 2000, vol. 5(1), pp. 205–209 (2000)
4. Lee, J.W., Park, N.: Individual Information Protection in Smart Grid. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 153–159. Springer, Heidelberg (2012)

5. Kang, S.: Korean Morphological Analysis and Information Retrieval. Hongreung Science Press (2002)
6. Support Vector Machine, Wikipedia. the free Encyclopedia, `http://en.wikipedia.org/wiki/SVM`
7. Thesaurus, Wikipedia, The Free Encyclopedia (2008), `http://en.wikipedia.org/wiki/Thesaurus`
8. Frakes, W., Baeza-Yates, R.: Information Retrieval: Data Structures and Algorithms. Prentice Hall (1992)
9. Kim, Y., Park, N., Hong, D., Won, D.: Context-based classification for harmful web documents and comparison of feature selecting algorithms. Journal of Korea Multimedia Society 12(6) (2009)
10. Yang, Y., Pederson, J.: A Comparative Study on Feature Selection in text Categorization. In: Proceedings of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
11. Park, N., Kwak, J., Kim, S., Won, D., Kim, H.: WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 741–748. Springer, Heidelberg (2006)
12. Park, N.: Security scheme for managing a large quantity of individual information in RFID environment. In: Zhu, R., Zhang, Y., Liu, B., Liu, C. (eds.) ICICA 2010. CCIS, vol. 106, pp. 72–79. Springer, Heidelberg (2010)
13. Park, N.: Secure UHF/HF Dual-Band RFID: Strategic Framework Approaches and Application Solutions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 488–496. Springer, Heidelberg (2011)
14. Park, N.: Implementation of Terminal Middleware Platform for Mobile RFID computing. International Journal of Ad Hoc and Ubiquitous Computing 8(4), 205–219 (2011)
15. Park, N., Kim, Y.: Harmful Adult Multimedia Contents Filtering Method in Mobile RFID Service Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 193–202. Springer, Heidelberg (2010)
16. Park, N., Song, Y.: AONT Encryption Based Application Data Management in Mobile RFID Environment. In: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (eds.) ICCCI 2010, Part II. LNCS (LNAI), vol. 6422, pp. 142–152. Springer, Heidelberg (2010)
17. Park, N.: Customized Healthcare Infrastructure Using Privacy Weight Level Based on Smart Device. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. CCIS, vol. 206, pp. 467–474. Springer, Heidelberg (2011)
18. Park, N.: Secure Data Access Control Scheme Using Type-Based Re-encryption in Cloud Environment. In: Katarzyniak, R., Chiu, T.-F., Hong, C.-F., Nguyen, N.T. (eds.) Semantic Methods for Knowledge Management and Communication. SCI, vol. 381, pp. 319–327. Springer, Heidelberg (2011)
19. Park, N., Song, Y.: Secure RFID Application Data Management Using All-Or-Nothing Transform Encryption. In: Pandurangan, G., Anil Kumar, V.S., Ming, G., Liu, Y., Li, Y. (eds.) WASA 2010. LNCS, vol. 6221, pp. 245–252. Springer, Heidelberg (2010)
20. Park, N.: The Implementation of Open Embedded S/W Platform for Secure Mobile RFID Reader. The Journal of Korea Information and Communications Society 35(5), 785–793 (2010)
21. Kim, Y., Park, N.: Development and Application of STEAM Teaching Model Based on the Rube Goldberg's Invention. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) Computer Science and its Applications. LNEE, vol. 203, pp. 693–698. Springer, Heidelberg (2012)

22. Park, N., Cho, S., Kim, B.-D., Lee, B., Won, D.: Security Enhancement of User Authentication Scheme Using IVEF in Vessel Traffic Service System. In: Yeo, S.-S., Pan, Y., Lee, Y.S., Chang, H.B. (eds.) Computer Science and its Applications. LNEE, vol. 203, pp. 699–705. Springer, Heidelberg (2012)
23. Kim, K., Kim, B.-D., Lee, B., Park, N.: Design and Implementation of IVEF Protocol Using Wireless Communication on Android Mobile Platform. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 94–100. Springer, Heidelberg (2012)
24. Ko, Y., An, J., Park, N.: Development of Computer, Math, Art Convergence Education Lesson Plans Based on Smart Grid Technology. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 109–114. Springer, Heidelberg (2012)
25. Kim, Y., Park, N.: The Effect of STEAM Education on Elementary School Student's Creativity Improvement. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 115–121. Springer, Heidelberg (2012)

Security Requirement of End Point Security Software*

Hyun-Jung Lee[1], Youngsook Lee[2], and Dongho Won[1,**]

[1] School of Information and Communication Engineering, Sungkyunkwan University, Korea
hjlee@kosyas.com, dhwon@security.re.kr
[2] Department of Cyber Investigation Police, Howon University, Korea
ysooklee@howon.ac.kr

Abstract. Security vulnerabilities exist in end point devices such as PDA's, laptops, Blackberries, tablets, etc. Examples of endpoint software include anti spyware / malware software, encryption software, Data Loss Prevention system, Security USB and Device Control S/W, client operating system based firewalls, etc. It is important to note that leveraging company security policy and ensuring that it is enforced through stringent monitoring of in-house security breaches, further enhance the effectiveness of end point security. However, no criteria have been established as yet to evaluate whether such End point Security Software correctly provides the basic security functions needed by user and whether such functions have been securely developed. Therefore, this paper proposes security requirements of End point Security Software by modeling a threat and applying a security requirement engineering methodology based on Common Criteria.

Keywords: End Point Security S/W, Protection Profile, Common Criteria, PP, CC.

1 Introduction

Many of the previously published Protection Profile (PP) is written based on a particular product type (e.g. Firewall, Multi-Function Devices). There are about 220 PP's released through Common Criteria Portal, with additional PP's that are developed and/or released separately from different countries[5].

Many of the previously published Protection Profile (PP) is written based on a particular product type (e.g. Firewall, Multi-Function Devices). There are about 220 PP's released through Common Criteria Portal, with additional PP's that are developed and/or released separately from different countries. Advantage of Creating a PP can define the minimum security functions in its particular product type, and also can reflect the required security functions by the PP developer or consumer. But whenever new security product introduced and whenever each other's

* This research was supported by the KCC(Korea Communications Commission), Korea, under the R&D program supervised by the KCA(Korea Communications Agency)" (KCA-2012-12-912-06-003).

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 788–795, 2013.

requirement is difference, new PP is needed for consumer and development. As a result, Many PPs are developed and/or released. And there are a lot of similarities in previous PPs, such as threats, assumptions, organization's security policy and security functional requirement. By analyzing the similarities, creating a common PP to be absorbed in various products can decrease the excessive number of PP development, as well as enabling ST developers to add common PP's basic requirements thoroughly, even when developing a ST of a product that is without a PP.

From the various product types evaluated by the Common Criteria, the most commonly identified products are largely distinguished as Network Device (defined to be an infrastructure device that can be connected to a network) such as Firewall, IDS and VPN, and End Point Security S/W(installed in PC that prevents PC's security threats) such as PC Firewall and host-DLP. Both have same or similar threats because the products operating environments is same or similar. For example, for Network Device, security purpose and security functional requirements are required to confront corresponding threats such as TSF data exposure at communication channel, malicious "Update", prohibited access, TOE resource exhaustion and data damage. [8,9] Likewise Network Device, End Point Security S/W' operating environments are similar to each other, therefore even the type of products may differ, common threats may exist which leads to the same necessity of security objectives and security functional requirements.

This paper wish to analyze security objective and functional requirements through analyzing threats, assumptions, organization's security policy through a security environment of a product in the form of End Point Security S/W that is active in Windows operational environment. Apart from the facts derived from its PP where the ST is being developed, or a product in the form of End Point Security S/W developing PP, this can lead to time-effectiveness in developing PP/ST and omission of derivation items by deriving additional threats, assumptions, organization's security policy, security objective and security functional requirements.

Table 1. Protection Profiles - Statistics

Category	PPs
Access Control Devices and Systems	6
Biometric Systems and Devices	7
Boundary Protection Devices and Systems	28
Data Protection	5
Databases	7
Detection Devices and Systems	17
ICs, Smart Cards and Smart Card-Related Devices and Systems	56
Key Management Systems	11
Multi-Function Devices	4
Network and Network-Related Devices and Systems	22
Operating Systems	13
Other Devices and Systems	27
Products for Digital Signatures	13
Trusted Computing	5
Totals:	**221**

2 Security Problem Description

This section describes the security aspects of the environment in which the End Point Security S/W will be used and the manner in which the End Point Security S/W is expected to be employed. It provides the statement of the End Point Security S/W's security environment, which identifies and explains all:

- Known and presumed threats countered by either the End Point Security S/W or by the security environment
- Organizational security policies with which the End Point Security S/W must comply
- Assumptions about the secure usage of the End Point Security S/W, including physical, personnel and connectivity aspects

This Chapter identifies assumption as A.***assumption***, threats as T.***threat*** and Policies as P.***policy***. The following are the assets that End Point Security S/W should protect:

- Protective PC
- Data of PC user
- Key data related to product operation

A threat is the IT entity and user which threatens by access to prohibited protective assets or in an unusual way.

2.1 Threat

The following threats should be integrated into the threats that are specific to the technology by the PP/ST authors when including the requirements described in this document. Modifications, omissions, and additions to the requirements may impact this list, so the PP/ST author should modify or delete these threats as appropriate[6,7].

Table 2. Threats

NAME	DESCRIPTION
T.TSF disclosure	A malicious user may cause the TOE or its configuration data to be inappropriately viewed.
T.TSF Modify	A malicious user may cause the TOE or its configuration data to be inappropriately modified or deleted.
T.UnAuthorized Update	A malicious party attempts to supply the end user with an update to the product that may compromise the security feature of the TOE.
T.Undetected Actions	Malicious users may take actions that adversely affect the security of the TOE. These actions may remain undetected and thus their effects cannot be effectively mitigated.
T.UnAuthorized Stop	Using a way that can prohibit threats to stop TOEs security function.

Table 2. *(Continued)*

NAME	DESCRIPTION
T.TransferedData Disclosure	In a case when central management server for TOE's security policy is available, a threat can expose, change, or delete the key data such as passwords, configuration settings, happening within the communication of management server and TOE.
T.UnAuthorized Management	In a case when central management server for TOE's security policy is available, a threat can disguise into management server to applicate prohibited security policy setting into TOE.
T.Reused auth data	Threat can reuse the authentification data to try access to the security function.

2.2 Organizational Security Polices

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. PP/ST Authors should ensure that any policies that apply to their particular technology are captured in the following table, and that the policies listed below are applicable[6,7].

Table 3. Organizational Security Polices

NAME	DESCRIPTION
P.Accountability	The TOE shall generate and maintain a record of security-related events to ensure accountability. Records shall be reviewed.
P.Secure Management	The TOE shall provide its authorized user with a means to manage the TOE securely and keep the TSF data up to date.

2.3 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's Operational Environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE. PP/ST authors should ensure that the assumptions still hold for their particular technology; the table should be modified as appropriate[6,7].

Table 4. Assumptions

NAME	DESCRIPTION
A.OS Reinforcement	The TOE has a routine to remove unnecessary services or measures and to fix vulnerability of the OS(e.g. using patches) to ensure credibility and stability of the OS.
A.Timestamp	The IT environment provides the TOE with a reliable timestamp.

3 Security Objectives

This Chapter identifies TOE Objective as O.***TOE objective*** and Objectives for Operational Environment as OE. ***Objective for operational environment***.

Table 5. TOE Objective

NAME	DESCRIPTION
O.TSF Protection	The TOE shall protect itself from unauthorized access or tampering to its functionality and data in order to maintain the integrity of the system data and audit records. And The TOE will provide the capability to test some subset of its security functionality to ensure it is operating properly.
O.Secure Communication	TOE must protect the distributed IT entity through secure communication channel.
O.Server Verification	Legitimacy of the corresponding management server or update server must be checked when setting TOE policy or upgrading it
O.Residual Information Clearing	The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.
O.Audit	The TOE's IT environment will provide a reliable time source to enable the TOE to timestamp audit records.
O.Notice	The TOE shall raise an alarm according to the policy set for each event.
O.Secure Management	The TOE shall provide its authorized user with an efficient means to manage the TOE and keep the TSF data up to date.

The following table contains objectives for the Operational Environment. As assumptions are added to the PP, these objectives should be augmented to reflect such additions.

Table 6. Objective for operational environment

NAME	DESCRIPTION
OE.OS Reinforcement	The TOE shall have a routine to remove unnecessary services or measures and to fix vulnerability of the OS(e.g. using patches) to ensure credibility and stability of the OS.
OE.TimeStamp	The TOE's IT environment must provide a reliable time source for the TOE to provide accurate timestamps for audit records.

4 Security Requirements

The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4*, with additional extended functional components. The TOE Security Functional Requirements that appear below in Table 6 are described in more detail in the following subsections[2,3,4].

Table 7. Security Functional Requirements

Functional class	Functional component	
Security audit	FAU_ARP.1	Security Alarms
	FAU_GEN.1	Audit data generation
	FAU_SAA.1	Potentialviolation analysis
	FAU_STG.1	Protected audit trail storage
	FAU_STG.4	Prevention of audit data loss
Cryptographic support	FCS_CKM.1	Cryptographic key generation
	FCS_CKM.4	Cryptographic key destruction
	FCS_COP.1	Cryptographic operation
Security Management	FMT_MOF.1	Management of security functions behavior
	FMT_MTD.1	Management of TSF data
	FMT_MSA.1	Management of security attributes
	FMT_SMF.1	Specification of management functions
	FMT_SMR.1	Security roles
Identification and authentication	FIA_AFL.1	Authentication failure handling
	FIA_UAU.2	User Authentication before any action
	FIA_UID.2	User Identification before any action
Protection of the TSF	FPT_ITC.1	Inter-TSF confidentiality durinhg transmissions
	FPT_ITT.1	Basic internal TSF data transfer protection
	FPT_RCV.1	Manual recovery
	FPT_STM.1	Reliable time stamps
	FPT_TST.1	TSF testing

5 Case Study: End Point Security S/W PP Extended Package End Point DLP

This Extended Package describes security requirements for End Point DLP is intended to provide a minimal, baseline set of requirement s that are targeted and mitigating well defined and described threats. However this Extended Package is not complete in itself but rather extends the End Point Security S/W PP[6].

End point DLP monitors and mitigates unsafe data handling at the endpoint. End point DLP can watch for programs that may be passing confidential or sensitive data on the host, or review data in storage looking for violations of the DLP policies. They run on desktops and servers, and may be paired with Network DLP products to provide a complete monitoring solution.

5.1 Added Security Problem Description

Table 8. Added Security Problem Description

NAME	DESCRIPTION
T.Data Leakage	A threat agent can leak the user data without authorization.
T.Analysis Failure	The TOE may fail to detect a threat agent's inappropriate access to or action taken on the data that needs protection, which may result in the data modified or tampered with.
A.Access	The TOE can access all IT system data required to enforce its functionality.
P.Statistics	An authorized administrator shall be able to take statistics on the data of audit and intrusion detection.

5.2 Added Security Objectives

Table 9. Added Security Objectives

NAME	DESCRIPTION
O.DataCollection	The TOE shall collect from the managed system the program codes that can be allowed and objects that need to be protected.
O.DataAnalysis	The TOE shall have an analysis process to decide whether to allow or deny access of an object.
O.Tagging	The TOE shall be able to identify the data categorized by data analysis.
O.LeakageProtect	The TOE shall monitor itself to prevent leakage of assets.
O.Statistics	The TOE shall analyze and take statistics on all events according to the policy.
OE.Access	The TOE shall be able to access all IT system data required to enforce its functionality.

5.3 Added Security Requirements

Host DLP products' key features are sensitive contents browsing through disk analysis, cryptography, media control, identification and authentication, and audit trail. As the SFRs in the CC2 are insufficient, extended components are necessary as shown below.

Table 10. Added Security Requirements

Functional class	Functional component	
User data protection	EXT_FDP_COL.1	Monitored data collection
	EXT_FDP_ANL.1	Monitored data analysis
	EXT_FDP_MON.1	Real-time monitoring of data leakage
	EXT_FDP_PRV.1	Prevention of data leakage
Security Management	EXT_FMT_STA.1	Data statistics of audit and leakage detection

6 Conclusion

This paper propose security requirements which can be used as a request for a proposal to procure an Security S/W for PC, a guideline for developers to develop a secure Security S/W for PC and criteria with which evaluators can evaluate can completeness of a developed system. Thus, the Security S/W for PC was analyzed, a threat was modeled, and CC based security requirements was deduced.

And this report wishes to address a Protection Profile (PP) that can be used commonly by deriving Windows basic products' possible common threats as well as its security functional requirements. Consequently, by developing this commonly usable PP not only enables effectively to reduce the time needed to develop Windows basic products' PP or ST, but also to prevent an omission in threats, objectives and security functional requirements.

References

1. Lee, S.-Y., Shin, M.-C.: Protection Profile for Software Development Site. In: Gervasi, O., Gavrilova, M.L., Kumar, V., Laganá, A., Lee, H.P., Mun, Y., Taniar, D., Tan, C.J.K. (eds.) ICCSA 2005. LNCS, vol. 3481, pp. 499–507. Springer, Heidelberg (2005)
2. Common Criteria, Common Criteria for Information Technology Security Evaluation; part 1: Introduction and general model, Version 3.1 R1, CCMB-2006-09-001 (September 2006)
3. Common Criteria, Common Criteria for Information Technology Security Evaluation; part 2: Security functional components, Version 3.1 R2, CCMB-2007-09-002 (September 2007)
4. Common Criteria, Common Criteria for Information Technology Security Evaluation; part 3: Security assurance components, Version 3.1 R2, CCMB-2007-09-003 (September 2007)
5. Common Criteria Portal, `http://www.commmoncriteiaportal.org`
6. Lee, H.-J., Won, D.: Protection Profile for Data Leakage Protection System. In: Kim, T.-H., Adeli, H., Slezak, D., Sandnes, F.E., Song, X., Chung, K.-I., Arnett, K.P. (eds.) FGIT 2011. LNCS, vol. 7105, pp. 316–326. Springer, Heidelberg (2011)
7. Lee, H.-J., Won, D.: Protection Profile for Personal Information Security System. In: IEEE TrustCom 2011, pp. 806–811 (2011)
8. Information Assurance Directorate, Network Device Protection Profile (NDPP) Extended Package Stateful Traffic Filter Firewall Version 1.0 (December 2011)
9. Information Assurance Directorate: Protection Profile for Network Devices Version 1.1 (June 2012)

Collecting and Filtering Out Phishing Suspicious URLs Using SpamTrap System*

Inkyung Jeun[1], Youngsook Lee[2], and Dongho Won[3,**]

[1] Korea Internet & Security Agency, Korea
ikjeun@kisa.or.kr
[2] Department of Cyber Investigation Police, Howon University, Korea
ysooklee@howon.ac.kr
[3] School of Information and Communication Engineering, Sungkyunkwan University, Korea
dhwon@security.re.kr

Abstract. Recently, Phishing is a significant security threat to users and has been easy and effective way for trickery and deception on the internet. Phishing is an attempt to acquire our information as well as financial information without user's knowledge by making similar kind of website or sending e-mails to users. Some of the widely available and used phishing detection techniques include whitelisting, blacklisting, and heuristics. But, absolute and perfect anti-phishing solutions and techniques are hard to fine due to a variability of phishing site domain. This paper aims to collect and filter out phishing suspicious URLs before determine phishing sites using Spamtrap system which is a honeypot used to collect spam e-mail. Spam e-mail usually contain phishing site URLs, so we can collect phishing site URLs from spam e-mail of spamtrap system. After collect URLs that can be phishing sites, many kind of phishing site detection algorithm can be used in our paper.

Keywords: Phishing, Security, Cyber Attacks, Spam Trap.

1 Introduction

"Phishing" came from 'fishing', by replacing the letter 'f' with 'ph' to represent the act of deceiving users by faked e-mail or websites. Phishing is a form of online identity theft that aims to steal sensitive information from users such as online banking passwords and credit card information. Phishing attacks use a combination of social engineering and technical spoofing techniques to persuade users into giving away sensitive information that the attacker can then use to make a financial profit.

The earliest form of phishing attacks were email-based method. A hacker involved spoofed emails that were sent to users for sending their password and account

* This research was supported by the KCC(Korea Communications Commission), Korea, under the R&D program supervised by the KCA(Korea Communications Agency)" (KCA-2012-12-912-06-003).

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 796–802, 2013.

information. In these days, besides email, attackers have also started to use smart phone SMS (Short Message Service) or instance message to persuade and direct users to spoofed web sites. Smartphone is the most popular device that can program any application which is customized for needs [2]. In these days, new word like "Smishing" was born. Smishing is derived from "SMs phishing" and referring to a phishing attack sent via SMS.

According to KISA(Korea Internet & Security Agency), the number of phishing sites which has increased dramatically since 2011[11] as shown in Fig1. In spite of this rapid increase, the absolute and perfect anti-phishing solutions are not available yet. In order to avoid hacking attacks, many vendors and security companies have released a variety of defense mechanisms.

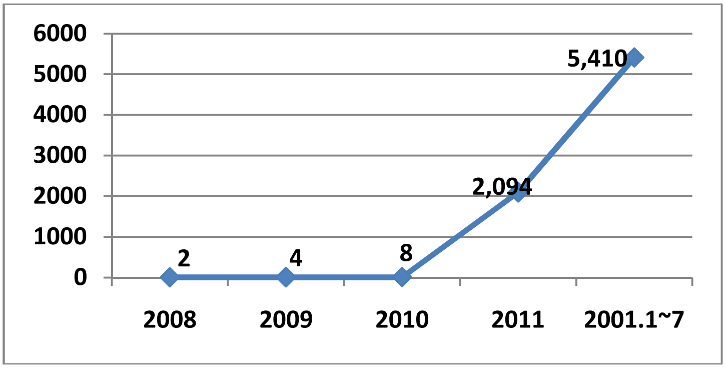


Fig. 1. Number of Phishing site in Korea

Also, in China, about 140 people per second visit phishing sites in the first-half of 2011. And new phishing sites which are founded by a security company in China, reached 35 million during the first half of the year.

In spite of phishing attack are so serious, the perfect anti-phishing solutions are not yet available. Although variety methods that can detect phishing sites are now introducing, it is difficult to detect all phishing sites because it changes its domain name on real-time. So, in this paper, we proposed a colleting method of phishing candidate sites effectively using Spam trap system.

This paper proceeds as follows: in Chapter 2, the background and related research about phishing detecting methods are described. In Chapter 3, the colleting of phishing candidate sites and filtering algorithm are described. In Chapter 4, we analyze the characteristics of our proposed method and conclusions are drawn In Chapter 5.

2 Background and Related Work

2.1 Phishing Detection Methods

Recently, anti-phishing has been studied intensively, and a variety of methods have been investigated as follows.

2.1.1 URL and Domain Verification

Generally speaking, the most widely used anti-phishing method is to use check the identity of URL and domain of web-sites. Internet browser can warn users whenever a phishing site is being accessed by using blacklisting, which matches a given URL with a list of URLs belonging to a blacklist. Blacklists hold URLs that refer to websites that are considered phishing.

They may use the public phishing blacklist sources like PhishTank [4], Google Safe Browsing API[5]. In here, the major problem with blacklists is incompleteness. To solve we can check IP address, abnormal request URL, abnormal DNS record, abnormal URL.

Pawan Prakash, et.al proposed PhishNet has a predicting malicious URLs component and an approximate matching component.[14] First component predicts new malicious URLs from existing blacklist entries by heuristic for generating new URLs and verification of it. Second component determines whether a given URL is a phishing site or not. It performs an approximate match of a given URL to the entries in the blacklist by first breaking the input URL into four different entities—IP address, hostname, directory structure and brand name - and, scoring individual entities by matching them with the corresponding fragments of the original entries to generate one final score. Jianyi Zhang et.al, proposed a prior-based transfer learning method for phishing detection engines.[3] They adjust the trained classifier and deploy the adaptive models to their corresponding regions according to the transfer learning. Thereafter, they construct a series of comprehensive experiments to test our proposed method.

2.1.2 Web Page Style and Contents

Another way to detect phishing sites is to check the web page style and its contents. It confirms spelling errors, copying sensate, using of "Submit" button and using of pop-up window. Also, we can check the images link because all images in the website including website logo should load from the same URL of the website not from another website. Therefore, we check the links to detect any external links inside the source code.

In [13], the authors came up with a total of 18 properties based on the page structure. Zhang et al proposed a content-based method using a linear classifier on top of eight features (the TF-IDF heuristic, age of domain, inconsistency of the logo image and domain name, suspicious page URL, suspicious links in the HTML, IP address, number of dots in URL, login forms).[8] Also, Guang Xiang and J.Hong suggested login form detection using HTML DOM properties((such as the page title, meta description field, etc.) .[6] Arel Cordero et al used website images to detect phishing attacks.[9] They proposed the use of rendered images as a basis for phishing detection and implemented initial prototypes.

E. Kidra and C.Kruegel suggested AntiPhish method that keep track of the sensitive information is typed into a form on a web site.[19] This tool has been implemented as a Mizilla Firefox plug-in and is free for public use.

2.1.3 Source Code

There are some characteristics in webpage source code that distinguish phishing websites from legitimate websites, so we can detect the phishing attacks by check the webpage and search for these characteristics in the source code file if it exists or not.

Phishing websites typically contain pages for the user to enter sensitive information, such as account number, password and so on. And the Phishing website uses logos found on the legitimate website to mimic its appearance. So phishers can load it from the legitimate website domain to their phishing websites (external domain). Also, Many Phishing pages have misspellings, grammatical errors, and inconsistencies. [10]

Also, Phishers use the iframe and make it invisible i.e. without frame borders, when the user goes to website, he/she cannot know that there is another page is also loading in the iframe window like phishing page.

2.2 Phishing Sites Domain

According to Global Phishing Survey by Anti-Phishing Working Group (APWG)[11], there were at least 112,472 unique phishing attacks worldwide on 1H2011. In here the attacks used 79,753 unique domain names and only 14,650 domains (18%) were registered maliciously by phishers. The other 65,103 domains were hacked or compromised on vulnerable Web hosting. Malicious registrations took place in 44 TLD (Top-Level Domain)s. 93% of the malicious domain registrations were made in just four TLDs : TK., INFO., COM., and NET.

As we can see from this data, attacker use sub-domain of hacked web site-domain as well as maliciously registered domain. Malicious use of sub-domain services continued to increase during in the first half of 2011, and accounted for the majority of phishing in many TLDs.

3 Our Approach

It is important issue how effectively collect phishing suspected sites in order to use phishing detecting algorithms. The phishing sites are changed on real-time and the new registered domain or second sub-domains are used as phishing sites domain. There is a limit that we find a newly registered domain. So, considering phishing sites are induced to access by e-mail, we proposed collecting phishing suspicious URLs using e-mail spams.

3.1 Collect of Phishing Candidate URL

We select phishing candidates group using e-mail spamtrap system which is a honeypot used to collect spam. E-mail spamtrap is system to collect analysis and preserve for evidences. For example, KISA operating spamtram system has been collected more than 11,000 e-mail account by using more than 1,000 web mail accounts from 9 portal-sites and install 1,000 e-mail addresses on it's operating mail server. More than 1 Million e-mails are collected by spamtrap system per one day.

On the e-mail body of collected e-mails, we should distinguish normal URLs, internal URL from suspicious URLs. E-mail is good tools to attack some one. It is used to send Phishing URL as well as a virus to a specific target [7].

Using this spam e-mail, we can collect phishing candidate URL, because almost phishing site is connected by e-mail. Only to analyze a spam e-mail, we can get a phishing candidates URL groups.

3.2 Filtering of Collected URLs

To enhance the reliability of collected URLs, we can give a priority to the top sub-domain services used for phishing attack. Use of sub-domain services continues to be a challenge, because only the sub-domain providers themselves can effectively mitigate these phish.

According to the Anti-Phishing Working Group (APWG), the top sub-domain used for phishing is co.cc.service, based in Korea. Over 30% of attacks using sub-domain services occurred on CO.CC, despite the fact that CO.CC is very responsive to abuse reports. T35.com, osa.pl, CN.la, Mu.la, altervista.org, co.tv, vv.cc, ce.ms are also top sub-domain refer to the APWG report.

3.3 Using Phishing Detection Algorithm

To detect phishing site among collected phishing candidate URLs, we can use man phishing detection algorithms which are mentioned in Section 2.1. We can check security, page style, source code and contents of web-site to decide phishing sites.

To confirm a security of phishing candidate URLs, we can check whether the website uses SSL certificate or encryption protocol. And, a page style like as pop-up window, disabling of right click or using of "Submit" button.

Fig2 is our proposed phishing detecting process using spamtrap system.

4 Evaluation

We now validate the effectiveness of our proposed approach using spamtrap system to collect phishing candidate URLs. \

- **Ease of Implementation:** To enhance detecting ability of phishing site, normal and public phishing detect solutions can use our approach, because it just need to be linked with spamtrap system. Of course, we need to implement URL filtering system from collected spam e-mails.
- **Efficacy:** Almost phishing sites are induced from e-mail, SMS, etc. In other words, by collecting phishing sites from e-mail which is normally used to induce phishing sites, we can increase the efficacy of detecting function.

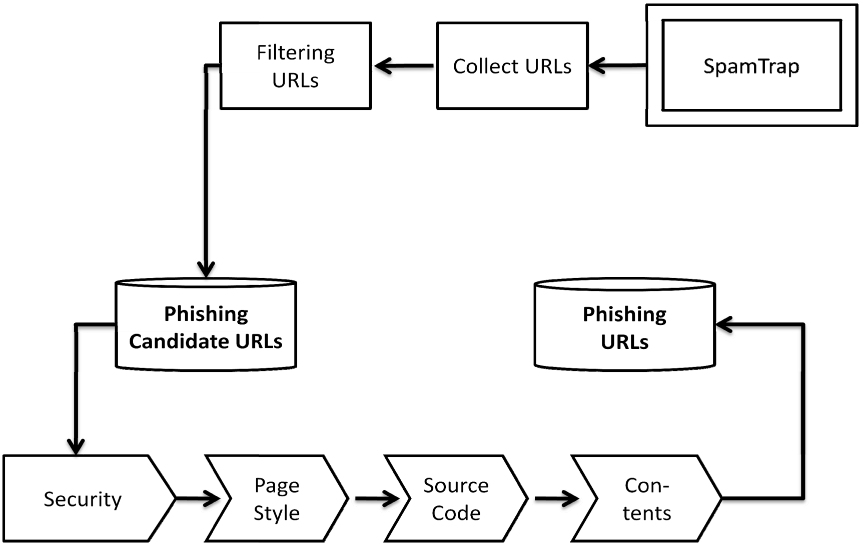


Fig. 2. Phishing Detect Process

- **Scalability:** Our major concern is to collect and filter our phishing URLs, not to detect and verify of phishing sites. So, phishing candidate URLs DB can be used any other anti-phishing solution and algorithm to detect phishing sites. Also, it can be added any URL search engines.

5 Conclusion

In this paper we analyze the current phishing detection method and proposed phishing detecting model using spamtrap system. Perfect and beforehand phishing detecting is almost impossible due to its variableness. So we proposed phishing candidate sites collecting method using spamtrap system. Many phishing sites are linked from e-mail as well as cell phone SMS. So we can effectively collect phishing candidate sites using e-mail spamtrap system.

To improve of trust of this paper, we need to implementation and performance evaluation in the future. To do this, we hope we can meet the time that the rate of phishing site can be reduced.

References

1. APWG, Global Phishing Survey : Trends and Domain Name Use in 1H 2011 (January-June 2011)
2. Jeon, W., Kim, J., Lee, Y., Won, D.: A Practical Analysis of Smartphone Security. In: Smith, M.J., Salvendy, G. (eds.) HCII 2011, Part I. LNCS, vol. 6771, pp. 311–320. Springer, Heidelberg (2011)

3. Zhang, J., Ou, Y., Li, D., Xin, Y.: A Prior-based Transfer Learning Method for the Phishing Detection. Journal of Networks 7(8) (August 2012)
4. OpenDNS Phishtank (2011), http://www.phishtank.com/
5. Google, Google Safe Browsing API Developer's Guide (v2) (2009), http://code.google.com/intl/zh-CN/apis/safebrowsing/developers_guide_v2.html
6. Xiang, G., Hong, J.I.: A Hybrid Phish Detection Approach by Identity Discovery and Keywords Retrieval. In: International Conference on World Wide Web (WWW) (2009)
7. Jeun, I., Lee, Y., Won, D.: A Practical Study on Advanced Persistent Threats. In: Kim, T.-H., Stoica, A., Fang, W.-C., Vasilakos, T., Villalba, J.G., Arnett, K.P., Khan, M.K., Kang, B.-H. (eds.) SecTech, CA, CES^3 2012. CCIS, vol. 339, pp. 144–152. Springer, Heidelberg (2012)
8. Zhang, Y., Hong, J., Cranor, L.: Cantina: a content-based approach to detecting phishing web sites. In: The 16th International Conference on World Wide Web (WWW 2007) (2007)
9. Cordero, A., Blain, T.: Catching Phish: Detecting Phishing Attacks From Rendered Website Images (2006)
10. Alkhozae, M.G., Maratfi, O.A.: Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code. International Journal of Information and Communication Technology Research (2011)
11. KISA, http://www.krecert.or.kr
12. Kidra, E., Kruegel, C.: Protecting Users against Phishing Attacks. The Computer Journal 49 (2006)
13. Ludl, C., McAllister, S., Kirda, E., Kruegel, C.: On the effectiveness of techniques to detect phishing sites. In: Hämmerli, B.M., Sommer, R. (eds.) DIMVA 2007. LNCS, vol. 4579, pp. 20–39. Springer, Heidelberg (2007)
14. Prakash, P., et al.: PhishNet: Predictive Blacklisting to Detect Phishing Attacks. In: IEEE INFOCOM (2010)

Improvement of a Chaotic Map Based Key Agreement Protocol That Preserves Anonymity*

Hyunsik Yang[1], Jin Qiuyan[1], Hanwook Lee[1], Kwangwoo Lee[2], and Dongho Won[1,**]

[1] Information Security Group Sungkyunkwan University Suwon, Korea
[2] Samsung Electronics Co., LTD Suwon, Korea
{hsyang,qyjin,hwlee,kwlee,dhwon}@security.re.kr

Abstract. In 2009, Tseng et al. proposed a key agreement protocol based on chaotic maps. Tseng et al. claimed that their protocol preserve user anonymity. However, Tseng et al.'s protocol is insecure against the insider attack. Nui et al. proposed a new anonymous key agreement protocol in 2011. Unfortunately, Nui et al.'s protocol cannot provide user anonymity and has computational efficiency problem. We introduce a new key agreement protocol based on Chebyshev chaotic map. Our protocol overcomes these security problems and provides user anonymity.

Keywords: chaotic map, key agreement protocol, user anonymity.

1 Introduction

Since 1976, lots of key agreement protocols were introduced, including Diffie and Hellman key agreement protocol[1], which is the first and most famous protocol. In 2009, Tseng et al. proposed a key agreement protocol based on chaotic maps[2]. Tseng et al. claimed that their protocol preserve user anonymity. However, Tseng et al's protocol is insecure against the insider attack[14]. Nui et al. proposed a new anonymous key agreement protocol in 2011[15]. Unfortunately, Nui et al.'s protocol cannot provide user anonymity and has computational efficiency problem[16][17]. We introduce a new key agreement protocol based on Chebyshev chaotic map. Our protocol overcomes these security problems and provides user anonymity.

The organization of the paper is as follows. In section 2, introduce Chebyshev chaotic map and hash function based on chaotic map. In section 3, we analyze Tseng et al.'s protocol and Niu et al.'s in section 4. In section 5, we describe our proposed key agreement protocol and performance and security analysis of our protocol. In the last section, we conclude this paper.

* This research was supported by the KCC(Korea Communications Commission), Korea, under the R&D program supervised by the KCA(Korea Communications Agency)"(KCA-2012-12-912-06-003).

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 803–810, 2013.

2 Related Work

In this section, we describe Chebyshev chaotic maps and a hard work.

Definition 1.
Chebyshev polynomial[4] is defined as follows:

The chebyshev polynomial $T_n(x)$ of the first kind is a polynomial in x of degree n, defined by the relation.

$$T_n(x) = \cos n\theta \text{ when } x = \cos\theta \quad (-1 \leq x \leq 1)$$

The initial conditions are $T_0(x) = 1$, $T_1(x) = x$ and few Chebyshev polynomials are

$$T_2(x) = 2x^2 - 1$$

$$T_3(x) = 4x^3 - 3x$$

$$T_4(x) = 8x^4 - 8x^2 + 1$$

With Definition 1, the fundamental recurrence relation is obtained.

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x),\ n = 2, 3, \ldots,$$

Chebyshev polynomials have two important properties.

The semi-group property

$$T_n(T_m(x)) = T_{nm}(x) = T_m(T_n(x))$$

The chaotic property

If the degree $n > 1$, T_n :[-1,1]→[-1,1] is a chaotic map.

This map has a unique absolutely continuous invariant measure

$\mu(x)dx = dx/(\pi\sqrt{1-x^2})$,

with positive Lyapunov exponent $\lambda = \ln p$.

Kocarev et al.'s system has security weakness against Bergamo et al. Zhang proposed enhanced Chebyshev polynomials, as follow:

$$T_n(x) = (2xT_{n-1}(x) - T_{n-2}(x))(\text{mod } N)$$

N is a large prime number and $x \in (-\infty, \infty), n = 2, 3, \ldots.$

Definition 2.

$T_n(x) = y$

When x and y are given, to find n is DLP(Discrete Logarithm Problem).

Definition 3.
When x, $T_r(x)$, $T_s(x)$ are given, to find $T_{rs}(x)$ is DHP(Diffie-Hellman Problem)

3 Analysis of Niu et al.'s Protocol

Niu et al. proposed a new anonymous key agreement protocol based on Chebyshev chaotic map[15]. Niu et al.'s protocol used TTP(Trusted Third Party) to preserve user anonymity. In this section, we describe Niu et al.'s protocol and show the problems[16][17].

3.1 Niu et al.'s Key Agreement Protocol

Niu et al.'s key agreement protocol is based on the chaotic one-way hash function, and uses Chebyshev chaotic maps.

(1) U_i → Server

U_i chooses randomly a large integer r, a large prime number N and a random number x ($x \in$ ($-\infty, +\infty$)). Next, U_i computes $T_r(x)$ and encrypts $(ID_i, T_r(x))$ with $E_{K_{TU}}$.

$C_1 = E_{K_{TU}}(ID_i, T_r(x))$

Finally, U_i sends N, x, C_1 to Server.

(2) Server → TTP

After receiving the message from U_i, Server choose a large integer s and compute $T_s(x)$. Then, Server encrypts $(T_s(x), n_s)$ with K_{TS}

$C_2 = E_{K_{TS}}(T_s(x), n_s)$

and sends ID_s, n_s, C_1, C_2 to TTP.

(3) TTP → U_i

After receiving the message, TTP decrypts C_1, C_2 and encrypt C_3, C_4 as follow:

$C_3 = E_{TU}(ID_s, T_s(x), T_r(x), n_s)$, $C_4 = E_{K_{TS}}(ID_i, T_r(x), n_s)$

Then, TTP sends C_3, C_4 to U_i.

(4) U_i → Server

U_i decrypt C_3 to get $(ID_s, T_s(x), T_r(x), n_s)$ and check the validity of $T_r(x)$. Then, U_i computes

$SK_i = T_r(T_s(x))$, $AU_i = h(ID_s, n_s, SK_i)$

Finally, U_i sends (AU_i, C_4) to Server.

(5) Server → U_i

Server decrypts C_4 to get $(ID_i, T_r(x), n_s)$ and computes

$SK_i = T_s(T_r(x))$, $AU_i' = H(ID_s, n_s, SK_i)$

Then, Server verifies $AU_i' = AU_i$?. If AU' and AU are not equal, Server stop here; otherwise Server computes as follows:

$AU_s = H(ID_i, n_s, SK_i)$

Finally, Server sends AU_s to U_i.

(6) U_i

U_i computes $AU_s' = H(ID_i, n_s, SK_i)$ and verifies $AU_s' = AU_s$? . If they are equal, the session key SK_i is agreed.

3.2 Weakness of Nui et al.'s Key Agreement Protocol

Nui et al. claimed that Nui et al.'s protocol was more secure than Tseng et al.'s protocol. Unfortunately, Nui et al.'s protocol has some problems[16][17].

- **Efficiency problem**

Niu et al.'s protocol has six steps, which has two steps more than Tseng et al.'s protocol. And Niu et al.'s protocol has two symmetric encryption /decryption more than Tseng et al.'s protocol. Nui et al.'s protocol needs more communication overlay and more computing overlay than Tseng et al.'s protocol.

- **Security weakness**

In Nui et al.'s protocol step (3), TTP decrypts C_1 with the U_i 's secret key. However, Nui et al. didn't explain how to verify the identity of U_i. In order to find the secret key used to encryption the message C_1 , TTP decrypt C_1 with all secret keys stored in TTP or receive the plaintext user information from U_i. Therefore Nui et al.'s protocol cannot provide the user anonymity.

4 Our Protocol

We propose a new key agreement protocol.

4.1 The New Key Agreement Protocol

- **Initialization phase**

Before performing the key agreement protocol, User i and the server share a one-way hash function and Chebyshev chaotic map

- **Registration phase**

(1) U_i → Server

U_i selects ID, PW and b where b is a large random number. U_i computes $h(PW \oplus b)$ and sends (ID, $h(PW \oplus b)$) to Server.

(2) Server → U_i

After receiving the user's data, Server selects a large random number C and compute as follow:

$$VID = h(ID \oplus C),\ \ \mathrm{Re}\,g = C \oplus h(PW \oplus b),\ \ C_{en} = Enc_s(C)$$

A symmetric key encryption algorithm(Enc_s) is used by the server. Then, Server stores ($C_{en}, VID, ID, h(PW \oplus b)$) in its database and sends $\mathrm{Re}\,g$ to U_i. A symmetric encryption scheme is used by the server.

- **Authentication phase**

(1) U_i → Server

A user U_i inputs ID and PW into the user's device. Then, U_i computes $C = \mathrm{Re}\,g \oplus h(pw \oplus b)$, $VID = h(ID \oplus C)$ and choose a random number $r, x(x \in (-\infty, \infty))$. Next, U_i computes follow as:

$$C_1 = C + 1,\ U_1 = h(ID \oplus C_1),\ U_2 = U_1 \oplus T_r(x).$$

Finally, U_i sends VID, U_2, x to Server.

(2) Server → U_i

After receiving the message, Server checks the validity of VID and get ID and C_{en} . Server selects a random number s where s is a large integer. Then, Server computes as follows:

$C = Dec_s(C_{en})$, $C_1 = C + 1$, $U_1 = h(ID \oplus C_1)$, $T_r(x) = U_2 \oplus U_1$, $SK = T_s(T_r(x))$, $S_1 = s \oplus U_1$, $AU_s = h(U_1, T_r(x), s, SK)$

Finally, Server sends (S_1, AU_s) to U_i .

(3) U_i → Server

After receiving the message, U_i computes as follows:

$$s = S_1 \oplus U_1,\ SK = T_{s'}(T_r(x)),\ AU_i = h(U_1, T_r(x), s, SK)$$

Then, U_i checks $AU_i = AU_s$? .

If they aren't equal, U_i stops the session. Otherwise, U_i sends AU_i to Server.

(4) Server

After receiving the message AU_i , Server checks $AU_i = AU_s$? . If they are equal, the session key checks SK is agreed.

(5) U_i , Server

U_i computes $\mathrm{Re}\,g_2 = ((C_1 \oplus h(pw \oplus b))$ and changes $\mathrm{Re}\,g$ to $\mathrm{Re}\,g_2$. Server changes VID, C to $h(ID \oplus (C_1 + 1))$, $Enc_s(C_1 + 1)$.

4.2 Security Analysis

In this section, we show that the improved protocol is secure against above attacks.

- **Security analysis of user anonymity**

The attacker *A* can intercept $VID = h(ID \oplus C)$ used for user identity. However, user i will use $h(ID \oplus (C + 2))$ in next authentication phase. So, after our protocol finish, VID is not valuable and the attacker *A* cannot get the user identity. Therefore, our protocol can provide the user anonymity.

- **Security analysis of inside attack**

The inside attacker A computes to get the counter number C .

$C = \mathrm{Re}\,g \oplus h(PW \oplus b)$

However, Every user uses the different random counter number C. So the attacker A cannot get any information about other users.

- **Security analysis of replay attack**

For each full rum of our proposed protocol, U_i and Server change the random counter number C. Communication data in the current key agreement protocol run would be not useful in next protocol run. Therefore, our key agreement protocol can resist replay attack.

- **Security analysis of off-line password guessing attack**

The attacker A can intercept the messages $VID, U_2, S_1, AU_i, AU_s$. However, the messages have no information about user password. Therefore, off-line password guessing attack cannot work in our protocol.

- **Provide the mutual authentication**

In Step 3 and Step4, the server and the user U_i compute $AU_* = h(U_1, T_r(x), s, SK)$ and check whether AU_i and AU_s are equal. If they are equal, the user and the session key is authenticated. Thus, the mutual authentication is provided in our protocol.

- **Security analysis of perfect forward secrecy**

If an attacker A get user i 's password and user stored data, A can compute $\mathrm{Re}\,g \oplus h(pw \oplus b)$ to get C. A computes $U_1 = h(ID \oplus (C-1))$. Then, A can get $T_r(x)$ and $T_s(x)$ with U_2 and S_1. However, A cannot find $SK = T_s(T_r(x))$, since A has to compute $T_s(T_r(x))$ from $T_r(x)$ and $T_s(x)$, A faces with DHP(Diffie-Hellman Problem) and DLP(Discrete Logarithm Problem).

Table 1. Comparison of security properties

	Tseng et al.'s protocol	Niu et al.'s protocol	Our proposed protocol
User anonymity	No	No	Yes
Inside attack	No	Yes	Yes
Without TTP	Yes	No	Yes
Replay attacks	Yes	Yes	Yes
Off-line password guessing attack	Yes	No use password	Yes
mutual authentication	Yes	Yes	Yes
perfect forward secrecy	Yes	Yes	Yes

4.3 Performance Analysis

In this section, we show performance analysis of our improved key agreement protocol for the authentication phase in Table 2. We define some notations as follows:

T_{hash} : The time of executing the hash function.

T_{Cheb} : The time of executing the Chebyshev chaotic map.

T_{Symm} : The time of executing the symmetric encryption or decryption.

According to Table 2, Server, User and TTP operate the symmetric encryption and decryption in Tseng et al.'s protocol and Nui et al.'s protocol. However, our protocol uses only XOR and one-way hash functions. Therefore our protocol is more efficient than other protocols.

Table 2. Comparison of computation overhead

	Tseng et al.'s protocol	Niu et al.'s protocol	Our proposed protocol
User *i*	$5T_{hash}+2T_{Cheb}+2T_{Symm}$	$2T_{hash}+2T_{Cheb}+2T_{Symm}$	$4T_{hash}+2T_{Cheb}$
Server	$2T_{hash}+2T_{Cheb}+2T_{Symm}$	$2T_{hash}+2T_{Cheb}+2T_{Symm}$	$3T_{hash}+2T_{Cheb}+2T_{Symm}$
TTP	-	$2T_{hash}+2T_{Cheb}+2T_{Symm}$	-

5 Conclusion

In this paper, we show the vulnerability of Tseng et al.'s protocol against the inside attack and some problems of Niu et al.'s protocol. To overcome these weaknesses, we proposed a new improved anonymous protocol based on chaotic map. Not only we provide the security analysis of the proposed scheme and but we also show performance analysis.

References

1. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Transactions on Information Theory 22(6), 644–654 (1976)
2. Tseng, H.-R., Jan, R.-H., Yang, W.: A chaotic maps-based key agreement protocol that preserves user anonymity. In: Proceedings of the 2009 IEEE International Conference on Communications, ICC 2009, Dresden, Germany, June 14-18, pp. 1–6 (2009)
3. Xiao, D., Liao, X., Deng, S.: One-way hash function construction based on chaotic map with changeable-parameter. Chaos, Solitons & Fractals 24(1), 65–71 (2005)
4. Mason, J.C., Handscomb, D.C.: Chebyshev polynomials. Chapman & Hall/CRC, Boca Raton (2003)
5. Kocarev, L., Tasev, Z.: Public-key encryption based on Chebyshev maps. In: Proceedings of the International Symposium on Circuits and Systems, ISCAS 2003, vol. 3, pp. III-28–III-31 (May 2003)
6. Bergamo, P., D"Arco, P., Santis, A., Kocarev, L.: Security of public key cryptosystems based on Chebyshev polynomials. IEEE Transactions on Circuits and Systems-I 52(7), 1382–1393 (2005)
7. Han, S.: Security of a key agreement protocol based on chaotic maps. Chaos, Solitons & Fractals 38(3), 764–768 (2008)

8. Yoon, E.-J., Yoo, K.-Y.: A new key agreement protocol based on chaotic maps. In: Nguyen, N.T., Jo, G.-S., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2008. LNCS (LNAI), vol. 4953, pp. 897–906. Springer, Heidelberg (2008)
9. Lee, Y., Kim, S., Won, D.: Enhancement of two-factor authenticated key exchange protocols in public wireless LANs. Elsevier Computers and Electrical Engineering 36(1), 213–223 (2010)
10. Nam, J., Paik, J., Kim, U.M., Won, D.: Security Enhancement to a Password-Authenticated Group Key Exchange Protocol for Mobile Ad-hoc Networks. IEEE Communications Letters 12(2), 127–129 (2008)
11. Lee, C., Park, S., Lee, K., Won, D.: An Attack on an RFID Authentication Protocol Conforming to EPC Class 1 Generation 2 Standard. In: Lee, G., Howard, D., Ślęzak, D. (eds.) ICHIT 2011. LNCS, vol. 6935, pp. 488–495. Springer, Heidelberg (2011)
12. Nam, J., Lee, K., Paik, J., Paik, W., Won, D.: Security Improvement on a Group Key Exchange Protocol for Mobile Networks. In: Murgante, B., Gervasi, O., Iglesias, A., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2011, Part IV. LNCS, vol. 6785, pp. 123–132. Springer, Heidelberg (2011)
13. Nam, J., Paik, J., Lee, B., Lee, K., Won, D.: An Improved Protocol for Server-Aided Authenticated Group Key Establishment. In: Murgante, B., Gervasi, O., Iglesias, A., Taniar, D., Apduhan, B.O. (eds.) ICCSA 2011, Part V. LNCS, vol. 6786, pp. 437–446. Springer, Heidelberg (2011)
14. Yang, H., Lee, K., Lee, C., Kwak, J., Won, D.: A Security Weakness in Tseng et al Key Agreement Protocol. In: Proc. of JWIS 2011, The 6th Joint Workshop on Information Security (October 2011)
15. Niu, Y., Wang, X.: An anonymous key agreement protocol based on chaotic maps. Commun. Nonlinear Sci. Numer. Simulat. 16(4), 1986–1992 (2011)
16. Yoon, E.-J.: Efficiency and security problems of anonymous key agreement protocol based on chaotic maps. Commun. Nonlinear Sci. Numer. Simulat. 17, 2735–2740 (2012)
17. Xue, K., Hong, P.: Security improvement on an anonymous key agreement protocol based on chaotic maps. Commun. Nonlinear Sci. Numer. Simulat. 17, 2969–2977 (2012)
18. Oh, S., Kwak, J., Lee, S., Won, D.: Security analysis and applications of standard key agreement protocols. In: Kumar, V., Gavrilova, M.L., Tan, C.J.K., L'Ecuyer, P. (eds.) ICCSA 2003, Part II. LNCS, vol. 2668, pp. 191–200. Springer, Heidelberg (2003)
19. Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. J. Mol. Biol. 147, 195–197 (1981)
20. Bergamo, P., D'Arco, P., Santis, A., Kocarev, L.: Security of public key cryptosystems based on Chebyshev polynomial. IEEE Trans. Circ. Syst.-I 52(7), 1382–1393 (2005)
21. Zhang, L.: Cryptanalysis of the public key encryption based on multiple chaotic systems. Chaos Soliton Fract. 37(3), 669–674 (2008)

Solving Router Nodes Placement Problem with Priority Service Constraint in WMNs Using Simulated Annealing*

Chun-Cheng Lin[1,**], Yi-Ling Lin[1], and Wan-Yu Liu[2]

[1] Dept. of Industrial Engineering and Management,
National Chiao Tung University, Hsinchu 300, Taiwan
[2] Dept. of Tourism Information, Aletheia University, New Taipei City 251, Taiwan
cclin321@nctu.edu.tw

Abstract. The QoS performance of wireless mesh networks (WMNs) is measured by the topology connectivity as well as the client coverage, both of which are related to the problem of router nodes placement, in which each mesh client is served as equal. In practice, however, mesh clients with different payments for the network services should be provided by different qualities of network connectivity and QoS. As a result, to respond to the practical requirement, this paper considers the router nodes placement problem in WMNs with service priority constraint in which each mesh client is additionally associated with a service priority value, and we constrain that the mesh clients with the top one-third priority values must be served. Our concerned problem inherited from the original problem is computationally intractable in general, and hence this paper further proposes a novel simulated annealing (SA) approach that adds momentum terms to search resolutions more effectively. Momentum terms can be used to improve speed and accuracy of the original annealing schedulers, and to prevent extreme changes in values of acceptance probability function. Finally, this paper simulates the proposed novel SA approach for different-size instances, and discusses the effect of different parameters and annealing schedulers.

Keywords: Wireless mesh networks, simulated annealing, router nodes placement, annealing schedule.

1 Introduction

Based on Wi-Fi technology, wireless mesh networks (WMNs) [1, 2] are the communication networks made up of radio nodes organized in a mesh topology. This paper considers the problem of router nodes placement (RNP) for the WMNs consisting of mesh routers and mesh clients [3], in which an optimal deployment of mesh routers is determined so that the network connectivity and the client coverage are maximized. In the previous work [4,8,9,10], the RNP problem only considered fixed and simple network environments, in which each mesh client is served as equal.

* Research supported in part by NSC 101-2219-E-009-025.

** Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 811–818, 2013.

In practice, however, each mesh client should be served by different quality of network connectivity as well as QoS [6] according to the user's payment for the service. To respond to the practical requirement, this paper extends the original RNP problem to the router nodes placement problem with service priority constraint in WMNs (WMN-RNPSP), in which each mesh client is associated with a service priority value that represents its service priority in this WMN, and we constrain that the mesh clients with the top one-third priority values must be served. The WMN-RNPSP problem is challenging due to the following three additional characteristics: (a) the locations of mesh routers are not predetermined, (b) mesh routers are assumed to have different radio coverage area sizes, and (c) each mesh client is associated with a different priority value. The last characteristic is designed for our practical requirement for providing users different service qualities. Our objective is to find an optimal placement of mesh routers in the deployment area to maximize both the network connectivity and the client covering.

Like the original RNP problem, the WMN-RNPSP problem cannot be solved by an efficient deterministic polynomial-time algorithm [7]. Hence, we propose a novel simulated annealing (SA) approach by analogy with [5] to solve the WMN-RNPSP problem, which provides an efficient promising solution. Our novel SA approach improves speed and accuracy of annealing schedulers and makes the algorithm become faster by adding momentum terms. In addition, we propose two types of neighbor selection mechanisms, called random scheme and local scheme, for comparing the original neighbor selection mechanism.

The rest of the paper is organized as follows. Section 2 introduces the basic original router nodes placement problem and define the router nodes placement problem with service priority constraint in WMNs. Then, the SA approach phases with momentum terms for constructing a WMN and its detail application phases to WMN-RNPSP problem is presented in Section 3. In Section 4, we present environment setting, simulation results and discussion. Finally, we discuss the future network and make some conclusion in Section 5.

2 Problem Description

This section first gives the basic environmental settings as well as concepts for the RNP problem, and then formulates the RNPSP problem.

2.1 The Router Nodes Placement Problem

An instance for the RNP problem [3, 8] consists of:

(a) m mesh routers each of which has a different-size radio coverage;
(b) a two-dimensional rectangular grid area of size $W \times H$ in which m mesh routers are deployed;
(c) n mesh clients located in arbitrary points of the deployment grid.

Figure 1 gives an instance for the RNP problem, in which according to the locations of mesh routers in the rectangular deployment grid, we can establish a network topology graph. Let the graph denoted by $G=(V, E)$, in which V is the set of all mesh routers and mesh clients, and E is the set of edges, which include two types of connections as follows. First, if the radio coverage of two mesh routers are overlapped, we create an edge between the two mesh routers. Second, if a mesh client is located within the radio coverage of a mesh router, we create an edge between the mesh client and the mesh router. There are two measure for the performance of the WMN. The first measure is the *network connectivity*, which is defined as the size of the greatest graph component of graph G, while the second measure is the *client coverage*, which is defined as the number of covered mesh clients.

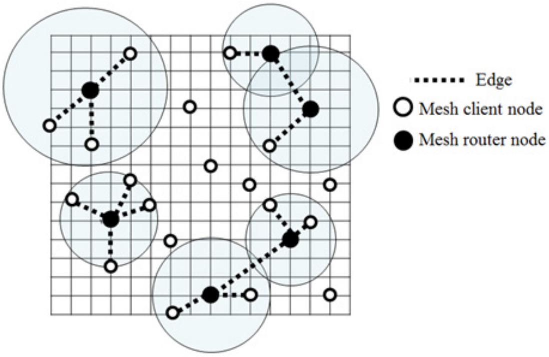


Fig. 1. An instance of WMN

2.2 The Router Nodes Placement Problem with Priority Service Constraint

An instance for the WMN-RNPSP problem consists of:

(a) m mesh router nodes each of which has a different-size radio coverage;
(b) a two-dimensional grid area of size $W \times H$ where m mesh routers are deployed;
(c) n mesh clients located in arbitrary points of the deployment grid each of which is associated with a service priority value.

In light of the above, the WMN-RNPSP problem can be stated as follows:

The WMN-RNPSP Problem: We are given a graph underlying a WMN distributed in a two-dimensional $W \times H$ grid area where the locations of mesh clients located in arbitrary collations of the grid area and each of mesh clients has a priority value, while the locations of mesh routers need be assigned. The objective of the problem is to find a placement X of the mesh routers so that the network connectivity and the client coverage are maximized while the mesh clients with the top one-third service priority values must be served.

3 Our Novel SA Approach to the WMN-RNPSP Problem

This section focuses on the annealing schedule and the acceptance probability module of our proposed novel SA, by analogy with [5]. Finally, we present in detail the key steps of our proposed novel SA.

3.1 Simulated Annealing Algorithm

Simulated annealing (SA) is a metaheuristic algorithm used for solving combinatorial optimization problems. The basic idea of SA is to simulate the cooling process of metals by heating and cooling of a material to increase the size of crystals and reduce defects. Initially, a feasible solution for the problem is represented a state of the metals. Heating causes the metals to change and rearrange their current state, while cooling finds a state with lower energy than the previous one. Note that the cooling process follows an annealing schedule. In each iteration of the annealing schedule, the SA considers a neighboring state of the current state, and bases the Metropolis rule to probabilistically decide whether the system moves to the neighboring state or stays at the current state. Those steps are repeated until the system reaches a state that is good enough, or the maximal number of iterations is achieved. The final state would be associated with a locally optimal solution of the concerned optimization problem.

The SA algorithm contains two main phases: annealing schedule and Metropolis rule. The annealing specifies "when and what temperature must be decreased", and the Metropolis rule considers a probability function and specifies "whether to replace the current state by a neighboring state". The probability is used to overcome the local optimal problem and lead the system to move the optimal solution of lower energy gradually. This paper considers three types of annealing modules: Geometric, Logarithmic, and Boltzmann. Unless stated otherwise, we use the most popular Boltzmann acceptance probability function.

3.2 A Novel Simulated Annealing Algorithm Using Momentum Terms

The novel SA approach is similar to the SA, and it speeds up the system time and enhances the accuracy of solution greatly on SA by adding momentum terms. Momentum terms are used to improve cooling speed and prevent extreme changes in values on acceptance probability function. This section summarizes three newly annealing modules: Hybrid, Extended logarithmic and Extended Boltzmann and one acceptance probability function: Extended Boltzmann function as follows. Note that T_i is the temperature of the i-th iteration, and ΔT is the difference between current temperature and previous temperature. Readers are referred to [5] for more details of those designs.

- **Hybrid:** $T_{k+1} = T_k - \alpha T_k - k \cdot \Delta T / e^k$ where α is similar to that used in the Geometric annealing module, and k is the number of iterations.
- **Extended logarithmic:** $T_k = C / \log(T_0 + k) - k / e^k - (\log(k))^{1/2}$ where C is constant.
- **Extended Boltzmann:** $T_k = T_0 / \log(1 + k) - \log(1 + k)$.

- **Extended Boltzmann function:** $P(\Delta E)$ = $e^{-\Delta E / bt}$ where ΔE is calculated as follows: $\Delta E = (E_i - E_j) - \alpha b T_i (E_i - E_j)^{1/2}$ where α is a running time parameter, and b is the Boltzmann constant.

3.3 Our Novel SA Approach to the WMN-RNPSP Problem

This section gives in detail my novel SA approach to the WMN-RNPSP problem: solution representation of each candidate solution, fitness function, scheme of neighboring solution selection and acceptance criteria.

3.3.1 Solution Representation

The (x, y)-coordinates of the routers should be determined as a candidate solution, which is expressed by two vectors (current and current best solutions) and two fitness values (current and current best fitness).

3.3.2 Fitness Function

The objective $f(X)$ for a placement X of our concerned problem is to maximize the network connectivity $\phi(G)$ and the client coverage $\psi(G)$ at the same time. Note that G is the topology graph underlying the placement X. The fitness function is calculated as follows:

$$f(X) = \lambda \cdot \frac{\phi(G)}{n+m} + (1-\lambda)\frac{\psi(G)}{m}$$

where λ is the weighting scale in the range [0, 1]. Note that the denominator of each term of the equation is used for normalization.

3.3.3 Neighbor Selection

The implementation of SA considers three types of moving schemes as follows:

- **Standard:** Choose a router randomly and place it in a new position randomly.
- **Random:** All of the mesh routers are reconfigured randomly.
- **Local:** Choose a router randomly and place it in a new position within the specified range randomly.

4 Implementation and Experimental Results

Based on the proposed SA approach described in the previous section, we implemented our proposed novel SA approach to the WMN-RNPSP problem. This section is divided into three subsections mainly. We first give the parameter setting, and then present the type of optimal neighbor selection on SA and novel SA in the individual various cases. Second, we use the result of the first one, compare all annealing schedules mentioned in Section 3 with Boltzmann and extended Boltzmann probability. Finally, we summarize all the previous results to give the experimental results in a variety of cases.

4.1 Data and Simulation Environment

Similar to [8], we consider the following three cases:

Case 1: There are 16 mesh routers and 48mesh clients on a 32 × 32 area.
Case 2: There are 32 mesh routers and 96mesh clients on a 64 × 64 area.
Case 3: There are 64 mesh routers and 192 mesh clients on a 128 × 128 area.

Table 1. Performance of neighbor selection on the original and our novel SA approaches for 32 × 32, 64 × 64 and 128 × 128 grid area

CASES	SA/NSA	Standard	Random	Local
32 × 32 grid size	Original SA	0.955385	0.744469	0.743010
	Novel SA	0.982656	0.783781	0.788635
64 × 64 grid size	Original SA	0.923776	0.876479	0.873375
	Novel SA	0.999229	0.871833	0.879516
128 × 128 grid size	Original SA	0.884500	0.860487	0.859797
	Novel SA	0.981529	0.868177	0.866550

One important aspect of the SA process is to study the performance under different neighboring selection methods. Table 1 shows the statistics results of the fitness values under different selection schemes for original SA and novel SA. We can see that the Standard scheme of neighbor selection on original or novel SAs can generate better solutions for all cases.

4.2 Annealing Schedule Method and Acceptance Probability Function

We give in Table 2 the computational results of six types of annealing schedule methods with Boltzmann and extended Boltzmann probability acceptance functions. Due to page limitation, we only put the results of case 1, 32 × 32 grid size. In Table 2 it is illustrated that the proposed acceptance function of novel SA has better results than original SA and almost all annealing schedule methods showed high quality performance under novel extended acceptance probability function.

Table 2. Comparison of annealing schedules with Boltzmann and extended Boltzmann probability for 32 × 32 grid size

Annealing schedule	Boltzmann	Extended Boltzmann
Geometric	0.950729	0.982292
Logarithmic	0.767517	0.981375
Boltzmann	0.759705	0.985792
Hybrid	0.769045	0.978687
Extended logarithmic	0.765486	0.983083
Extended Boltzmann	0.755781	0.982406

4.3 Experimental Results

After the fine tuning of above parameters was done, we measured the performance of the novel SA algorithm for all the problem instances. The statistics of all the problem instances are given in Table 1, in which four columns stores best fitness, average fitness, worst fitness, and the standard deviation of fitness values; ten rows indicates each 5 instances of clients distributions. We observe that our novel SA approach performs high efficiency and almost achieves to maximum both network connectivity and client coverage.

Table 3. The statistics of all cases

Instance	Case 1				Case 2				Case 3			
	Best	Mean	Worst	SD	Best	Mean	Worst	SD	Best	Mean	Worst	SD
uniform_1	1.0000	0.9823	0.9474	0.0155	1.0000	1.0000	1.0000	0.0000	0.9952	0.9860	0.9711	0.0089
uniform_2	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.0000	0.9904	0.9846	0.9711	0.0047
uniform_3	1.0000	0.9953	0.9615	0.0090	1.0000	1.0000	1.0000	0.0000	0.9952	0.9888	0.9855	0.0025
uniform_4	1.0000	0.9978	0.9672	0.0068	1.0000	1.0000	1.0000	0.0000	0.9952	0.9852	0.7978	0.0276
uniform_5	1.0000	0.9691	0.9423	0.0152	1.0000	0.9992	0.9615	0.0055	1.0000	0.9955	0.9904	0.0015
nniform_1	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.0000	0.9952	0.9935	0.9855	0.0025
normal_2	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.0000	0.9952	0.9907	0.9904	0.0012
normal_3	1.0000	0.9963	0.8172	0.0259	1.0000	1.0000	1.0000	0.0000	0.9952	0.9929	0.9855	0.0026
normal_4	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.0000	0.9952	0.9941	0.9855	0.0026
normal_5	1.0000	0.9965	0.8271	0.0243	1.0000	1.0000	1.0000	0.0000	0.9952	0.9928	0.9892	0.0025
average	1.0000	0.9937	0.9463	0.0097	1.0000	0.9999	0.9961	0.0005	0.9952	0.9904	0.9652	0.0057

5 Conclusion and Future Work

A novel simulated annealing approach for optimizing the placement of mesh router nodes for mesh clients with service priority constraint in wireless mesh networks has been proposed and implemented. The experimental results showed the efficient implementation of our proposed novel SAs for the WMN-RNPSP problem. The results also confirmed that our proposed novel SA is an effective method for the problem as it achieved the network connectivity of almost all mesh router nodes and covered almost all mesh client nodes in variety of grid sizes. In addition, the performance of our proposed novel SA is always better than original SA.

In the future, we intend to solve the dynamic version of the WMN-RNPSP problem or consider the optimization of other objectives at the same time, so that the problem is more realistic and can be used in the community.

References

1. Akyildiz, I.F., Wang, X., Wang, W.: Wireless mesh networks: A survey. Journal on Computer Networks 47, 445–487 (2005)
2. Aoun, B., Kenward, G., Boutaba, R., Iraqi, Y.: Gateway placement optimization in wireless mesh networks with QoS constraints. IEEE Journal on Selected Areas in Communications 24(11), 2127–2136 (2006)
3. Barolli, A., Sánchez, C., Xhafa, F., Takizawa, M.: A study on the performance of search methods for mesh router nodes placement problem. In: Proc. of 2011 IEEE International Conference on Advanced Information Networking and Applications (AINA 2011), pp. 756–763. IEEE Press (2011)
4. Barolli, A., Sánchez, C., Xhafa, F., Takizawa, M.: A Tabu search algorithm for efficient node placement in wireless mesh networks. In: Proc. of 3rd International Conference on Intelligent Networking and Collaborative Systems (INCoS 2011), pp. 53–59. IEEE Press (2011)
5. Keikha, M.M.: Improved simulated annealing using momentum terms. In: Proc. of Second International Conference on Intelligent System, Modeling and Simulation (ISMS 2011), pp. 44–48. IEEE Press (2011)
6. Napieralski, A., Wojtanowicz, K., Zabierowski, W.: Quality of service in wireless networks. In: Proc. of 10th International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM 2009), pp. 168–170. IEEE Press (2009)
7. Rabiner, L.: Combinatorial optimization: Algorithms and complexity. IEEE Transactions on Acoustics, Speech and Signal Processing 32(6), 1258–1259 (1984)
8. Sánchez, C., Xhafa, F., Barolli, L., Miho, R.: An annealing approach to router nodes placement problem in wireless mesh networks. In: Proc. of 2010 International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2010), pp. 245–252. IEEE Press (2010)
9. Sánchez, C., Xhafa, F., Barolli, L.: Locals search algorithms for efficient router nodes placement in wireless mesh networks. In: Proc. of International Conference on Network-Based Information Systems (NBIS 2009), pp. 572–579. IEEE Press (2009)
10. Sánchez, C., Xhafa, F., Barolli, L.: Genetic algorithms for efficient placement of router nodes in wireless mesh networks. In: Proc. of 24th IEEE International Conference on Advanced Information Networking and Applications (AINA 2010), pp. 465–472. IEEE Press (2010)

Topology Information Based Spare Capacity Provisioning in WDM Networks*

Hoyoung Hwang[1] and Seung-Cheon Kim[2,**]

[1] Dept of Multimedia Engineering, Hansung University, Korea
hyhwang@hansung.ac.kr
[2] Dept of Information Communication Engineeging, Hansung University, Korea
kimsc@hansung.ac.kr

Abstract. For networksurvivability, recovery methods from link or node failures should be provided and spare capacity to perform recovery should be prepared on the network links. The efficiency of spare capacity provisioning is a key issue in survivable network design. In this paper, we studied topology information based capacity provisioning methods for WDM optical networks which are widly used as a backbone architecture of current Internet.In the methods, the spare wavelengths are reserved to perform optical link protection using only topology information of a network without need to calculate the amount of on-going traffic of the network, thus provide simple and efficient spapre capacity planning. The basic idea of the topology information based methods is embedding virtual cycles to perform recovery on the network topology graphs.We suggest a multiple ring-cover based spare capacity provisioning scheme and compare it with two other topology information based schemes called cyclic-double-cover and p-cycle. We provide performance analysis of the topology information based schemes by the numerical calculation using cut-sets of the topology graphs, and compare it with computer simulation results.

Keywords: WDM, capacity, topology, cycle, cut-set.

1 Introduction

The requirements for network recovery methods includespeed of recovery, efficiency in resource utilization,robustness against multiple failures, and etc.The resource efficiency can be obtained by sharing spare resources needed fornetwork recovery. To improve the sharing of spare resources in WDM networks,methods to share backup paths as well as spare capacity together should be studied since the routing and the capacity assignment are tightly coupled in WDM networks via wavelength channels.

In this paper, we studied simple and fast network recovery methods using only the topology information of networks. We proposed a cycle-based recovery method for WDM optical mesh networks.The proposed method centersaroundmultiple ring-cover

[*] This work is financially supported by Hansung University.

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 819–825, 2013.

where each network link is included in number of *m* backup cycles and each cycle protects 1/*m* of the link capacity. Distributed link restoration is performed using preplanned cycles, and both the backup paths and the spare capacity can be shared.The pre-configuration of the cycles and the spare capacity placement arederived directly from the network topology in off-line, which is independent of the working traffic status or its dynamic changes over time.The proposed method provides efficiency and simplicity to survivable network design and management.

We first examine several topology information based network recovery methods. Then we present the provisioning of backup paths and the recovery procedurefor the proposed method. The performance of the proposed method is presented by computer simulationsand also by calculation using the concept of cut-set. The performance results show that the proposed topology information based methodprovides improved resource efficiency and robustness.

2 Topology Information Based Recovery Methods

The basic idea of the topology information based methods is embedding virtual cycles to perform recovery on the network topology graphs. The motivation behind a cycle-based backup configuration in mesh networks is that only cycles that include the primary path can contribute to find alternative paths in a graph that represents a network topology.Cycles can provide backup paths that are independent of the routing of primary connections, and show simple and fast recovery operation. In addition, carefully designed cycles can provide backup paths sharing and spare capacity sharing together to the links included in the cycle. One of the important features of WDM networks is the tightly-coupled route and capacity relationship based on wavelength channels, in contrast to packet based networks such as IP or ATM networks where the routing function and the capacity allocation function are separated.Therefore, sharing of backup paths as well as sharing of spare capacity is an important requirement for an efficient recovery method in WDM networks. Cycle-based backup configuration is a promising approach to meet the requirement of resource sharing for WDM networks.

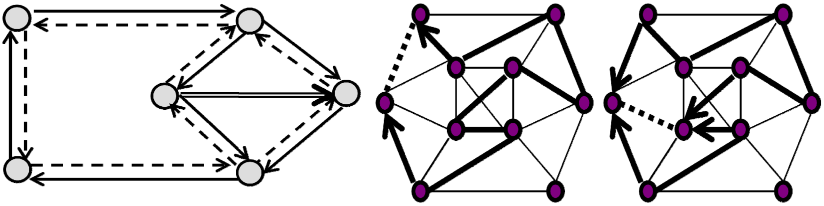


Fig. 1. Topology information based methods using embedded cycles:generalized loopback(left)and p-cycle

Configuration of rings such that each link is included in at least one ring is called ring-cover or single ring-cover, and several approaches to use a ring-cover as a backup path configuration method were studied[1, 2]. The drawback of single ring-cover is the high spare capacity redundancy which is more than 100% inprotection techniques and also very high in shared restoration techniques as will be presented in this paper.

An alternative method is that using cyclic-double-cover (CDC) conjecture.The CDC is a well-known conjecture in graph theory: a set of cycles exists in a two-connected graph G that each edge of G is included exactly two of the cycles.A protection technique using CDC conjecture has been proposedwhich performsfiber protection with 100% of spare capacity redundancy[3]. A problem on the CDC configuration is that some cycles may be toolong in a large network.A long backup cycle needs more spare capacity and time to complete restoration than shorter one,thus decreases the QoS of restored connections and also the robustness in the event of multiple failures.

A protection cycle configuration method using generalized loopback is studied[4], and the tradeoff between spare capacity and robustness has been observed.This method requires less than 100% of spare capacity redundancy for single link failure,since it is possible that not all the links are included in the protection cycles, however, which results in decreased robustness against multiple failures.A strong point of this configuration method is that it does not need to be globally reconfiguredfor a small change of a network topology.

As another alternative to ring covers, p-cycle configuration has been proposed[5]. This method can provide efficient spare capacity redundancy which is very close to optimal spare capacity placement for local link restoration. However, this method may also suffer from long protection cycles.

In this paper, we present a new cycle-based recovery method for WDM optical mesh networks. Objectives of the proposed method include simple design and management, efficient spare resource utilization, and robustness for multiple failures.

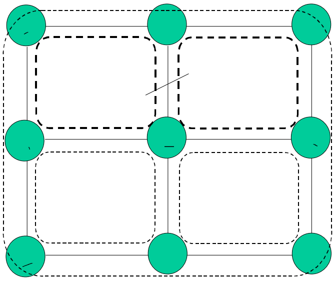


Fig. 2. An example of virtual backup cycles (*m*=2)

3 Multiple Ring Covers

Fig. 2 presents a configuration of multiple backup cycles.In this case, the number of backup cycles per link (*m*) is 2.In the figure, physical links are shown as solid lines and five shared backup cycleshave been found shown as dashed lines.Each network link is assigned two backup cycles, and each cycle is responsible forrestoration of half (1/2) of the link capacity. Primary capacity of a link, i.e., the available wavelengths of a link are partitioned into two even restoration units so that one restoration unit covers half of the link capacity. Then, each unit can be restored by one backup cycle as preconfigured. For example, if link (2-5) failed, one restoration unit on link (2-5) can be restored using the backup path (2-1-4-5), and the other restoration unit can be restored using another backup path (2-3-6-5),as preconfigured by the backup cycles. It is obvious that this basic idea can be easily extended to other values of m, i.e., the number of backup cycles and the number of restoration units per link can be 1, 3, 4, or more.

The pre-configuration of the backup paths and placement of spare capacity is performed at network design phase. The multiple backup cycles are found by searching k-shortest paths between the end nodes of a link with preference of disjoint shortest paths, and joining them with the target link. Backup cycles are determined only once for a given network topology G=(*N, E*), where *N* is the set of nodes and *E* is the set of edges. Our first goal is to find a set of cycles that covers each link at least *m* times to configure *m* backup cycles per link. To perform efficient spare capacity planning, the backup cycles of a link should have the least number of shared links which would reduce the sharability of spare capacity. Therefore, the first goal should be updated to reflect this fact that each link should be included in *m* cycles that have the least number of shared links. If we consider the restoration speed and the QoS of restored connections, short backup cycles are preferred to long backup cycles.

4 Spare Capacity Provisioning Using Cut-Set

The set of links to be eliminated to make partition of a topology graph into two separated parts is called cut-set. A cut-set may have various numbers of links as shown in Fig. 3. A cut-set with n links is denoted as CS(n).In multiple ring covers, the spare capacity needed to restore a communication link is distributed to *m* backup cycles. To have *m* disjoint backup routes, a link should be included in a cut-set with more than m+1 links;CS(*m*+1).If one of the link in CS(*m*+1) has failed, then the capacity on the failed link can be restored using the other *m* linksincluded in *m* disjoint backup cycles, and each link restore 1/*m* of the required spare capacity.

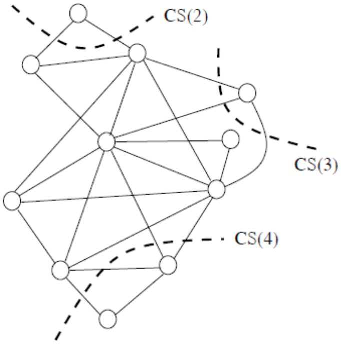


Fig. 3. Cut-sets in a network topology

For example, if a link in CS(4) has failed when the backup cycle multiplicity *m*=3,then the remaining threelinks can accommodate1/3 of the failed capacity respectively. If a link in CS(3) has failed, however, one of the two remaining links should restore 2/3 of the failed capacity while the other remaining link restore 1/3 of the failed capacity. If a link in CS(2) has failed, then the only remaining link should restore all of the failed capacity. The failed capacity of a link in CS(1) cannot be restored.

The spare capacity assignment for given number of *m* and the type of cut-set are presented in table 1. This means that we can calculate the spare capacity requirement using only the topology information such as cut-sets. For example, in the topology graph G(100, 180) shown in Fig. 4, there are 8 links included in 4 CS(2) type cut-sets. Thus if *m*=2, the 8 links should accommodate 100% of link capacity while other 172 links accommodate 1/2 of the link capacity, assuming all the links have the same wavelength capacity.In the next section, we will compare the calculation results using cut-sets with the computer simulation results.

Table 1. Spare capacity assignment for *m* and cut-sets

Cut-Set	$M = 1$	$M = 2$	$M = 3$	$M = 4$	$M = 5$
CS(1)	X	X	X	X	X
CS(2)	1+1	1+1	1+1	1+1	1+1
CS(3)	$\cdots$	$\frac{1}{2} \times 3$	$\frac{2}{3} \times 2 + \frac{1}{3} \times 1$	$\frac{2}{4} \times 3$	$\frac{3}{5} \times 2 + \frac{2}{5} \times 1$
CS(4)		$\cdots$	$\frac{1}{3} \times 4$	$\frac{2}{4} \times 2 + \frac{1}{4} \times 2$	$\frac{2}{5} \times 3 + \frac{1}{5} \times 1$
CS(5)			$\cdots$	$\frac{1}{4} \times 5$	$\frac{2}{5} \times 1 + \frac{1}{5} \times 4$
CS(6)				$\cdots$	$\frac{1}{5} \times 6$

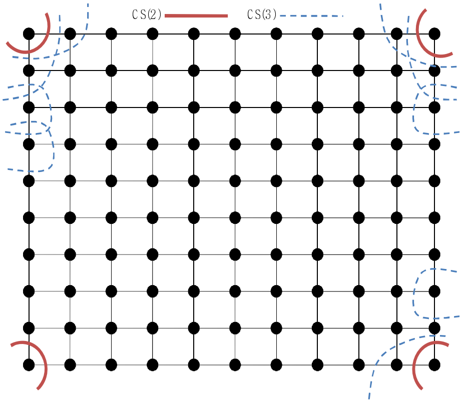


Fig. 4. Cut-sets in a10x10 grid network topology

5 Performance Analysis

We performed simulations to estimate the performance of the proposed method with 10 example network topologies. We assumed that each fiber contains 60 wavelengths and all the network links have the same number of fibers.Table2 presents the spare capacity overhead versus *m*. In this paper, the spare capacity overhead is defined as the ratio of the amount of required spare capacity to the amount of total primary capacity for 100% restoration of any single link failure.As we can see in each row, the spare capacity ratio can be substantially improved by using multiple backup cycles compared with that of the fiber or link-based single backup cycle (*m*=1).We can also realize that the spare capacity overheads of dense networks are better than that of sparse networks.

Table 3 shows the spare capacity calculation results for network 2 and network 10, when m=2, 3 respectively. The calculation results are the same as the simulation results shown in Table 2 for the same network topology and *m*. Therefore, we can see that the spare capacity ratio calculated using only topology information can be feasible and high accuracy.

Table 2. Spare capacity overhaed of multiple ring-covers

Networks	N	L	D	1-cycle	2-cycle	3-cycle	4-cycle
1	10	22	4.40	**91.9 %**	**50.0 %**	**39.0 %**	**36.4 %**
2	11	23	4.18	**82.6 %**	**63.0 %**	**54.5 %**	**55.4 %**
3	14	21	3.00	**90.5 %**	**59.5 %**	**67.5 %**	**73.8 %**
4	15	28	3.73	**96.4 %**	**57.1 %**	**54.2 %**	**60.7 %**
5	20	32	3.20	**93.8 %**	**50.0 %**	**55.6 %**	**67.8 %**
6	28	47	3.35	**97.9 %**	**62.8 %**	**58.2 %**	**62.8 %**
7	20	31	3.10	**96.8 %**	**71.0 %**	**68.1 %**	**68.5 %**
8	30	59	3.93	**93.2 %**	**53.4 %**	**44.7 %**	**51.3 %**
9	53	79	2.98	**98.7 %**	**75.3 %**	**76.0 %**	**80.4 %**
10	100	180	3.60	**100 %**	**52.2 %**	**41.0 %**	**56.1 %**
Average				***94.2 %***	***59.4 %***	***55.8 %***	***61.3 %***

Table 3. Spare capacity calculation using cut-sets

Network Topology	Spare Capacity Calculation	
	2-cycles (*m*=2)	3-cycles (*m*=3)
Net. 2 (11, 23)	(1×6) + (1/2×17) = 14.5 (spare capacity) 14.5/23 = **63.04%**	(1×6) + (2/3×3) + (1/3×14) = 12.66 (spare capacity) 12.66/23 = **55.07%**
Net. 10 (100, 180)	(1×8) + (1/2×172) = 94 (spare capacity) 94/180 = **52.22%**	(1×8) + (2/3×28) + (1/3×144) = 74.66 (spare capacity) 74.66/180 = **41.48%**

6 Conclusion

In this paper, a cycle-based backup path provisioning method is presented for WDM optical mesh networks. The proposed cycle configuration design can be derived directly from the network topology and applicable to networks with arbitrary two-connected topologies. We can calculate the spare capacity ratio of a network using only the topology information, and the results shows high accuracy and similarity compared with computer simulation results.

Acknowledgements. This work is financially supported by Hansung University.

References

1. Ahn, S.: A Fast VP Restoration Scheme using Ring-Shaped Sharable Backup VPS. In: Proceedings of Globecom 1997, pp. 1383–1387 (November 1997)
2. Gardner, L.M., et al.: Techniques for Finding Ring-Covers in Survivable Networks. In: Proceedings of Flobecom 1994, pp. 1862–1866 (November 1994)
3. Ellinas, G., Hailemariam, A.G., Stern, T.E.: Protection Cycles in Mesh WDM Networks. IEEE Journal on Selected Areas in Communications 18(10), 1924–1937 (2000)
4. Lummetta, S., Medard, M., Tseng, Y.C.: Capacity versus Robustness: A tradeoff for link restoration in mesh networks. Journal of Lightwave Technology 18(12), 1765–1775 (2000)
5. Grover, W.D., Stamatelakis, D.: Cycle-oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration. In: Proc. of ICC 1998, pp. 537–543 (June 1998)
6. Chaudhuri, S., Hjalmtysson, G., Yates, J.: Control of Lightpathsin an Optical Networks. In: Optical Internetworking Forum OIF 2000, 04 (January 2000)
7. Bakri, M., Koubaa, M., Bouallegue, A.: An iterative Partial Path Protection-based approach for routing static D-connections in WDM transparent networks with SRLG constraints. In: Proc. of ICOIN 2012 (February 2012)
8. Hwang, H., Lim, S.: WDM Optical Network Restoration and Spare Resource Planning using Multiple Ring-Cover. Journal of Korea Information Processing Society 12-C(6) (October 2005)
9. Jang, S., Kang, D.: A Wire/Wireless Convergence System for Ubiquitous network using Optical WDM-PON. In: Proc. of Summer Conference of Korea Institute of Information Technology (June 2006)

A Contents Service Profit Model Based on the Quality of Experience and User Group Characteristics

Goo Yeon Lee, Hwa Jong Kim, Choong Kyo Jeong, and Yong Lee

Department of Computer and Science Engineering,
Kangwon National University, Chuncheon, Korea
{Leegyeon,hjkim,ckjeong}@kangwon.ac.kr, yleehyun@gmail.com

Abstract. Generally, it is known that only quality-assuring services can provide a reasonable profit model. However until now a practical profit model considering the service cost and quality simultaneously has not been introduced yet. Recently, the Quality of Experience (QoE) was suggested to measure user's real satisfaction level. The QoE is expected to be used for efficient service provisioning and criteria for accurate satisfaction measuring. This paper introduces a profit model for the contents providers considering the costs for quality services and the QoE together. Especially, we assume that the QoE with user's feedback can be interpreted as the intention to pay for the services. We take into account that QoE is dependent on service area, demographic information and user group characteristics. The proposed profit model can be used for contents providers to find an optimum investment which maximizes the profit.

Keywords: quality of experience, characteristics of user group, quality of service, profit model, contents service.

1 Introduction

Recently, multimedia contents over the Internet is vastly increasing especially due to the social network service (SNS) and P2P traffic. Along with this, the quality of multimedia service becomes a critical issue because of the limited server capacity or network bandwidth. For the contents providers it is inevitable to increase the server and network capacity in order to satisfy more users, which however needs more investments. In order to handle the predicted huge multimedia traffic in the future, we need a useful model which can optimize the investment and maximize the profit of the contents providers.

Traditionally, network service quality management was performed using Quality of Service (QoS) parameters by controlling the traffic priority or guaranteeing bandwidth to specific services. The QoS parameters, e.g., delay, jitter, bandwidth and error rates, have been used well to represent network level performances. However the QoS parameters have limit to correctly indicate the real service satisfaction level. In order to measure the user's satisfaction more precisely, Quality of Experience (QoE) was introduced [1]. The QoE was expected to represent well user's subjective satisfaction level and to be used effectively for network service quality management [2]-[5].

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 826–831, 2013.

Network users can be classified into various groups; more sensitive users such as early adapters living in cities and average users living in rural area. Even in a region, some users may expect higher service quality and others may be satisfied with ordinary quality. In order to set up a reasonable business model, the contents providers should understand that the QoE values depend on the service area and demographics.

2 QoE Standadization

Various QoE measurement methods have been studied in many institutes. The QoE researches at ITU-T mainly focused on QoE metrics [1] and measurement schemes [6], defining QoE-based service quality criteria and extracting related service quality factors. In [7] the connection of QoS with QoE is studied to find QoE indicators from QoS parameters and the relationship between QoS and QoE factors is analyzed. They also obtained a formula that calculates QoE value from QoS parameters.

The ITU-T FG IPTV defined IPTV QoS/QoE metrics in three layers: Perceptual Quality Metrics, Video Stream Metrics, and Transport Metrics [3]. The Perceptual Quality Metrics provides QoE for video and audio signals. The Video Stream Metrics is related with performance of encoded video stream. The Transport Metrics gives performance information of transport protocols such as IP, UDP, RTP and so on.

The DSL forum defines QoE requirements for quality management on triple service [5]. It classified quality indicators into service layer, application layer and transport layer. The service layer measures the service quality level from user's experience. The application layer manages various system parameters, and the transport layer manages network delay, jitter and loss of packets. Each layer has control plane and data plane for handling control message and data transfer respectively.

3 Design and Analysis of Profit Model with QoE

Contents providers want to make higher profit from customers possibly with less investment. Generally, increasing server capacity or network bandwidth will improve the QoE and thereby profit from users, but it also needs more cost. Therefore, it is important to find a reasonable profit model for the contents providers to maximize their profit.

In the paper, we suggest a QoE-based profit model (QPM) of the contents providers considering the costs for quality services and the QoE characteristics of user groups. We assumed that the QoE can be interpreted as the intention to pay for the services. However, it is noted that QoE actually differs for every user because it depends on subjective measurement. Therefore, it is difficult to find a general QPM that handles various user groups with different characteristics. In the paper, we assumed the users can be grouped with similar sensitivity to the quality for the same contents.

Parameters used in the proposed QPM are as follows.

- QoE level (Q): Q represents the user satisfaction level, having value of 0 for minimum quality and 1 for maximum quality. Since the level of QoE changes with technology progress and user characteristics, Q must be measured periodically to be realistic.
- Server bandwidth (B): Conventionally, it has been assumed that QoS or QoE depends mainly on bandwidth, error rate, delay and jitter. However, in the current high speed network environment (especially in Korea), error rate, delay or jitter gives insignificant effect to the QoE because of efficient buffering technology. In the paper, it is assumed that user access network has enough bandwidth. On the other hand, we assume that the channel bandwidth of the content server is restricted and might be a critical bottleneck for service quality.
- Server bandwidth cost (C): Costs invested on the server bandwidth usage, paid by contents providers to network service providers.
- Server cost (S): Costs invested to server installation and maintenance for guaranteed service quality.
- Profit from contents delivery (P): Profit earned from users with the contents delivery. P can be made from direct payment by users or from advertisement with free contents. It is assumed that P increases as QoE improves.

Relations between the parameters explained above are as follows.

- Relation between QoE level (Q) and profit (P) : In the paper, we assumed that Q is proportional to P. With higher QoE, the number of interested users increases and the contents providers get more profit directly from users or through advertisement. For simple model of the QPM, we have $P=aQ$, where a is a proportional constant. For maximum value of Q, i.e., when Q is 1, maximum possible number of users are assumed to be involved in the service.
- Relation between QoE level (Q) and server bandwidth (B): QoE level Q will increase as the server bandwidth B increases. In the paper, we used a logarithmic relation between Q and B as $Q = 1 - e^{-kB}$. The rationale of the equation is: for higher value of Q, we need much more incremental bandwidth for the same satisfaction increment. With zero bandwidth (i.e., B=0), Q has minimum value of 0, and with infinite bandwidth, Q will be maximum value of 1. In the equation parameter k represents the service sensitivity of a specific user group to give the same Q with a given B. For example, if a user group has small sensitivity value k, they need higher bandwidth B in order to get same Q. On the contrary, if a user group have larger sensitivity k they need low bandwidth B for the same level Q. Figure 1 shows the relations between B and k when Q is 0.5, 0.7 and 0.9, respectively. In the figure, for example, when bandwidth is SD level (e.g., 1.5Mbps), a user group with 70% satisfaction (i.e., Q=0.7) has sensitivity parameter k=0.8026. For 10Mbps bandwidth, k will be 0.1204, and for 20Mbps (e.g., full HD level), k becomes 0.06.
- Relation between server bandwidth (B) and server bandwidth cost (C): In the paper, we assume that contents providers pay in flat rate for the bandwidth usage. We have $C=bB$ where b is a proportional constant. For a different rate policy, the relation between C and B would be changed in the following analysis.

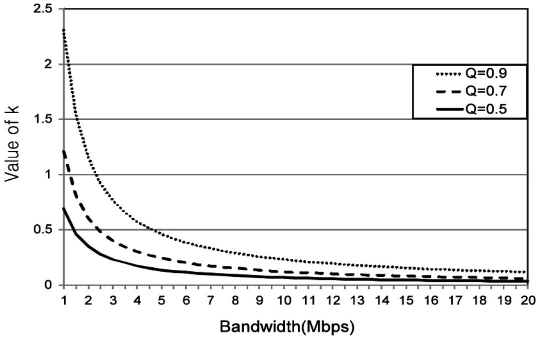


Fig. 1. Value of k as bandwidth increases

- Relation between server cost (S) and server bandwidth (B): Server cost S is composed of many factors such as server hardware, operational cost and maintenance fee. In the paper, we simply assume that server cost is proportional to server bandwidth. When d is a proportional constant, we can have $S=dB$.

From the above assumptions and parameters, we have the total profit T as follows.

$$T = P - C - S = aQ - (b+d)B \ = aQ + \frac{(b+d)}{k} \cdot \ln(1-Q) \qquad (1)$$

Contents providers may decide how much satisfaction they will offer to a target user group with service sensitivity parameter k. Contents providers also seek for a greater profit, so they want T in (1) to have a maximum value. Differentiating (1) with respect to Q and setting it to 0, we have Q_{opt} for (1) as $Q_{opt} = 1 - \frac{b+d}{ak}$. Q_{opt} is the user satisfaction level that makes maximum profit for the user group with sensitivity parameter k. The maximum profit can be obtained by substituting Q_{opt} to (1).

For a discussion, an example case with the following parameters will be used.

- Maximum number of users in the group : 100,000
- Profit from one user in one unit period (i.e., one month): 1,000KRW(Korean Won). Therefore, we have a=100,000,000.
- b+d : 4,000,000KRW per 1Mbps and per one unit period.

Figure 2 shows total profit as QoE varies when k is 0.1204. In the figure, the maximum profit is 30,166,437KRW when QoE is 0.67. The profit increases while QoE is increasing from 0, but when it reaches to the maximum point it starts to decrease. It means that increasing investments to improve QoE finally results in decreased profit due to the overinvestment. From figure 2, we see that when QoE reaches 0.95, the total profit becomes 0. So, the analysis in the paper can be used for the contents providers to find an optimum investment.

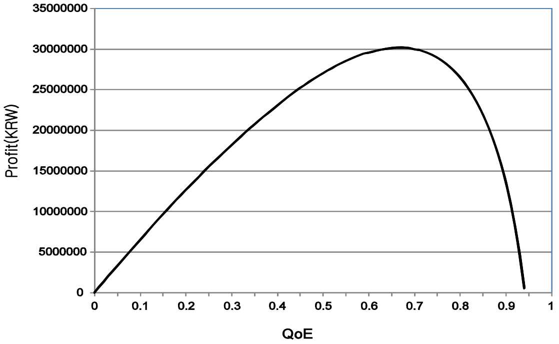


Fig. 2. Profit of contents service provider as QoE increases

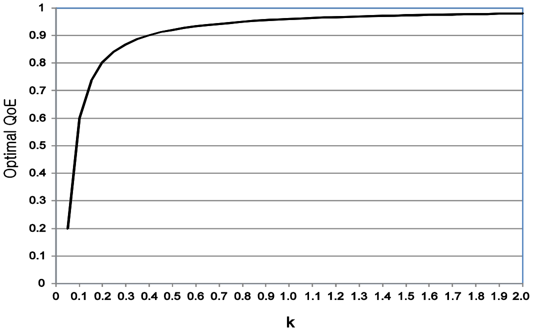


Fig. 3. Optimal QoE as group characteristics k varies

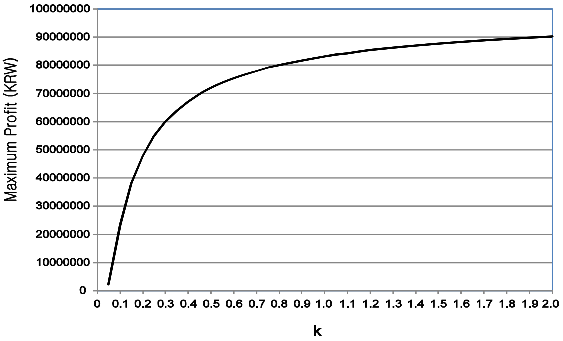


Fig. 4. Maximum profit as group characteristics k varies

Figure 3 shows optimum QoE for various values of user group sensitivities *k*, and figure 4 shows the maximum profit at the points of optimum QoE obtained in figure 3. From figures 3 and 4, we can find that the optimum QoE and maximum profit fall when *k* decreases. Lower *k* indicates that the user group is more sensitive and expects higher bandwidth, i.e., higher service quality for the same satisfaction. We may call this group premium users. It is noted that the total profit from the premium group is smaller than the group with larger *k* (i.e., less sensitive group or standard group). From figure 3, we also find that when *k* is higher, optimum QoE comes close to 1.

4 Conclusion

Recently, QoE concept was introduced to measure user's real satisfaction level of service based on the user's feedback. In the paper, we assumed that the QoE can be interpreted as the user's intention of paying to the services, very important information to the contents providers. In the paper, we proposed a profit model of contents providers taking into account the QoE level and sensitivity parameters of different user groups. The proposed profit model can be used for the contents providers to find an optimum investment which maximizes the profit by using the QoE levels of each user group.

Acknowledgements. This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2012-0004625) and also supported by the MKE(The Ministry of Knowledge Economy), Korea, under the CITRC(Convergence Information Technology Research Center) support program (NIPA-2012-H0401-12-1002) supervised by the NIPA(National IT Industry Promotion Agency).

References

1. ITU-T Recommendation G.1080: Quality of Experience requirements for IPTV services (2008)
2. ITU-T G.1010: End User Multimedia QoS Categories (2008)
3. ITU-T FG IPTV-C-0411: IPTV QoS.QoE Metrics (2007)
4. ATIS-IIF, ATIS-0800004: A Framework for QoS Metrics and Measurements supporting IPTV Services (2006)
5. DSL Forum TR-126: Triple-play Services Quality of Experience(QoE) Requirements (2006)
6. ITU-T Recommendation J.247: Objective perceptual multimedia video quality measurement in the presence of a full reference (2008)
7. Kim, H.-J., Lee, D.-H., Lee, J.-M., Lee, K.-H., Lyu, W., Choi, S.-G.: The QoE Evaluation Method through the QoS-QoE Correlation Model. In: Networked Computing and Advanced Information Management IEEE CNF, pp. 719–725 (2008)

Toward Hybrid Model for Architecture-Oriented Semantic Schema of Self-adaptive System

Jin-Hong Kim and Seung-Cheon Kim

Department of Information and Communication Engineering,
Hansung University, 116, Samseongyoro-16gil, Seongbuk-gu, Seoul, Korea
{jinhkm,kimsc}@hansung.ac.kr

Abstract. Self-adaptive software is an essential approach to manage the challenges of establishing system that autonomously responds to a variety of context-aware situation. In addition, self-adaptive software use a lot of policies and explicitly or implicitly between rules and cases to decide how to react to monitored events. For theses, Self-adaptive systems persistingly develop and modify behavioral properties to meet changing demands. Most of important, specification of adaptation policy is based on element in the construction of architecture-based hybrid model. However, several rules are usually scattered in different procedures, which makes procedures more complex, as well as cases are merely recognized situation by the rule and case. Accordingly, in this paper, we presents what is hybrid model including architecture-centric semantic schema. Also, a core element in architectural self-adaptive systems is the specification of adaptation policy: the mapping between hybrid observer indicating the need for an adaptation and hybrid diagnosis of properties with regulator in this need, along with an expression of self-adaptive schema algorithm.

Keywords: Self-adaptive Software, Context-aware, Hybrid Model.

1 Introduction

In recent years, software computing highest technology, and next generation computing paradigms are getting more and more toward building, running, and managing of self-adaptive software [2][3]. Nevertheless, the specific consequence of these two dimensions, as well as their fundamental approaches are significantly divergent, or to a certain individual exclusive. The primary concern for the self-adaptive software researchers must become more flexible, dependable, robust, and configurable by adapting to changing operational contexts, environments or system characteristics [5]. It is important to often provide dynamic event at run time, as well as emphasize that in all the many initiatives to explore self-adaptive behavior, the common event that enables the provision of self-adaptivity is usually software [1].

This applies to the research in several application areas and technologies such as pervasive computing, grid computing, service computing, and autonomous smart platform. In all these case self-adaptive software's flexibility allows evolutionary applications; however, appropriate realization of the self-adaptive software substantially remains an

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 832–837, 2013.

important approach and inquire cautiously in designing self-adaptive system with restrict application domains [4]. Although various self-adaptive software use rules and cases in the long run, we need to define the foundations that enable the explicitly or implicitly to decide how to establish the systematic development of next generation of self-adaptive system, and what is supported to react to monitoring events, adaptation and architectural model within hybrid model. The goal of this research paper is to propose concept of Hybrid Model, which is used to point out the current state-of-the-art, as well as to identify the commonalities and autonomic element structure for architectural self-adaptive system. Specifically, we present a hybrid model-based approach for designing architectural self-adaptive system that is capable of accommodating during system operation. We'd like to call this approach semantic challenge [6]. To present and motive this challenge, this paper introduces the hybrid modeling paradigms for surely view of self-adaptation we have identified. Section 2 presents Hybrid Model Architecture. In Section 3 Hybrid model schema in our system. Section 4 shows Schema Evolution for SSA and conclusions finally.

2 Hybrid Model Architecture

The entire Hybrid Model Architecture approach, illustrated in Figure 1, is focused on artificial intelligence-based approaches for self-adaptive system, which typically are complex dynamical processes that are handled by software process. It is often provides explicitly "Knowledge element" in autonomic events and rules, which is likely organizes into high-level business logics and policies, thus facilitate self-adaptive mechanism for architectural system [7].

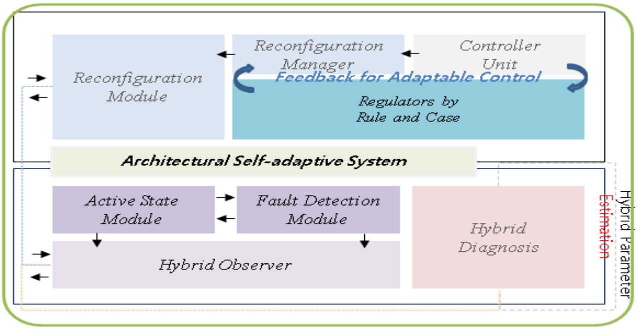


Fig. 1. Hybrid Model Architecture

Accordingly, our systems are modeled using hybrid approaches to improve the reconfiguration, thus is based on ideas from separation of concepts as followings;

(1) Active State and Fault Detection Module for Hybrid Observer. It is necessary and sufficient conditions for a coupled design Activate State Module (ASM) and Fault Detection Module (FDM) for hybrid observer achieving exponential convergence.

Coupled ASM and FDM is based on the current-state observable if there exists an integer K such that (1) for any known initial state $q(0)$, and (2) for any input sequence $\sigma(K)$, the state $q(i)$ can be determined for every $i>k$ from the observation sequence $\Psi(k)$ up to i. We call this Hybrid Observability.

(2) Hybrid Diagnosis. It is cases we are triggered by internal variables of the plant hybrid model, and attributed to control commands. These hybrid diagnosis variables evolve continuously by estimated hybrid parameter in time. But, some mode changes can result in discontinuous changes in variable values. To solve the problem, the plant hybrid model is denoted by hybrid observer scheme as following Figure 2 by each of expression: *i)* $q(k+1) \in \Psi\ (q(k), \sigma(k+1))$, *ii)* $q(k+1) \in \Psi\ (q(k), \sigma(k+1))$, *iii)* $\sigma(k+1) \in \Phi\ (q(k), x(t^{-}_{k+1}), u(t^{-}_{k+1})$, *and iv)* $\Psi(k+1) \in л(q(k), \sigma(k+1), q(k+1))$.

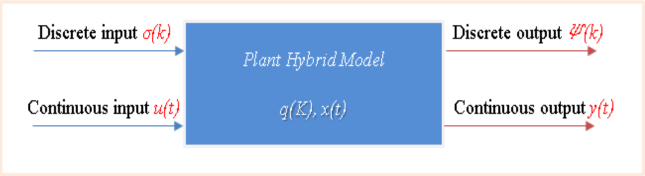


Fig. 2. Hybrid Observer Schema (HOS) diagnosis

(3) Feedback for Adaptive Control. It should be Regulator by rule and case between controller unit and reconfiguration manager, which consider only event by involved in adaptive rule and case. If environment changing by hybrid parameter from resulted on diagnosis, hybrid observer provides mechanisms to capture reconfiguration module.

3 Schema for Hybrid Model

As the first step towards the self-adaptive schema, a schema is created, which from definition is exactly an object-oriented schema. However, it is not a schema by itself, since in the diagnosis, so called symptom database, or used for data typing in real applications. The schema can be also interpreted from a semantic perspective: each class in the schema corresponds to a semantic concept, and the inheritance relationship indicates that one concept is specialization of another [10-12]. Viewed as a whole, the sematic schema starts with concepts at higher layers and extends downwards to more specialized concepts at lower layers, which forms an intuitive classification that captures the commonsense. As we known that described our model, we are defined based on the schema by introducing self-adaptive schema algorithm (SSA) within hybrid parameter as following expression of example. When an application is delivered, it usually composes only procedure by each steps, and hybrid model architecture. Accordingly, we should add hybrid model both rule and case to application, to assign self-adaptive ability to application.

```
Step 1. Set $V_{cp}$ to C //$V_{cp}$ is variables that can used to denote any class
While $V_{cp}$ does not exist in S then, // S mean a self-adaptive schema
Set $V_{cp}$ to the super class of $V_{cp}$

Step 2. Set $\{V_{1}...........\ V_{N}\}$ as the children nodes of $V_{cp}$
for (i=1) to N
set $V_{cq}$ as the lowest common ancestor class of Cai and C
if $V_{cq}$ = C, then go to Step 4
else if $V_{cq}$ <> $V_{cp}$ , then go to Step 5

Step 3. Insert C into S as a child node of $V_{cp}$
Exit ()

Step 4. Insert C into S as a child node of $V_{cp}$ and parent node of C
Exit ()

Step 5. Insert $V_{cp}$ into S as a child node of $C_{cp}$
Insert C into S as a child node of $V_{cq}$
Reconnect $C_{ai}$ as a child node of $V_{cq}$
Exit ()
```

4 Schema Evolution for SSA

The Self-Adaptive Schema Algorithm (SSA) is defined based on hybrid model architecture that must have to provide this structure of self-adaptive schema with the basic principle of their requirement through a set of procedures. The algorithm for schema cause by this model defines the applications that are managed in a rule. These applications have associated properties with various types. A rule is integrated into diagnosis of hybrid observer schema. Accordingly, we can define hybrid model to create different property types and add them to the rule as following Figure 3.

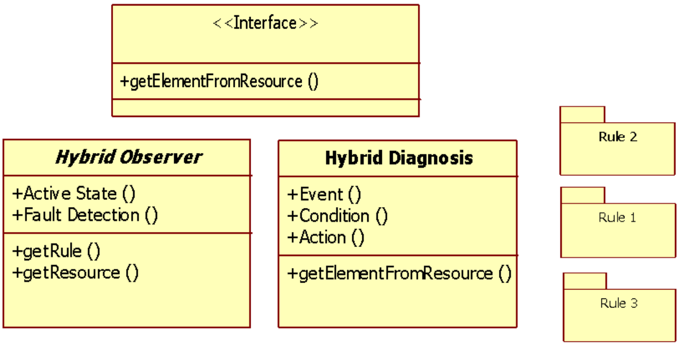


Fig. 3. Schema Evolution for Hybrid Model

The hybrid model is evaluated by two perspectives as following Figure 4; (1) the performance of the system when it runs generally, (2) the performance of the system when it operates by hybrid model from hybrid observer schema.

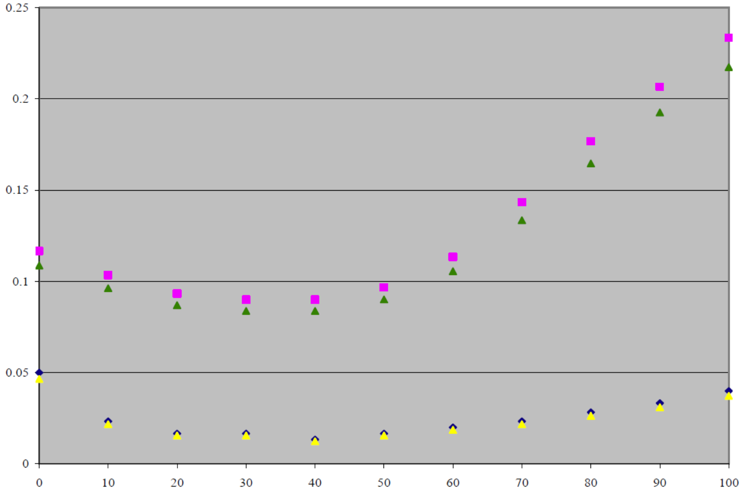


Fig. 4. Schema Evolution for Hybrid Model

5 Conclusions

In our research paper, we has presented a hybrid model for architectural self-adaptive system using hybrid observer schema and algorithm which take advantages of several techniques including software as a service and aspect-oriented software within object-oriented modeling. Our architecture features a self-adaptive mechanism, which exploits a semantic schema to automatically optimize towards the requirement of a specific application. For hybrid model based architectural this system as shown in Figure 1 are given some benefits; (1) it helps to organize coupled between active state and fault detection module by hybrid observer in a suitable scope. (2) From the hybrid parameter by diagnosis, it gives regulators by rule and case from the feedback for adaptable control more apparent meanings, which make it easier to concepts in rule and case to operation in reconfiguration module. As the response to the shortcoming mentioned in Section 3 and 4, we proposed to enable to extend the architecture-centric hybrid modeling by semantic schema coupled between active state and fault detection module with several applicable semantic concepts. Most of important thing, we will be necessary to thoroughly studied and critical challenge for the systematic software engineering of self-adaptive systems which is identified essential views of self-adaptation: evolutionary and explorer modeling dimensions, requirements, and assurances. Although, there is nothing scenario-based prototype in our research, we are necessary to keep going on research eventually.

Acknowledgements. This research was financially supported by Hansung University.

References

1. Perry, D., Wolf, A.: Foundations for the Study of Software Architecture. ACM SIGSOFT Software Engineering Notes 17(4), 40–52 (1992)
2. Dashofy, E., Hoek, A., Taylor, R.: Towards architecture-based self-healing systems. In: Workshop on Self-Healing Systems (WOSS 2002), November 18-19 (2002)
3. Kephart, J., Chess, D.: The vision of Autonomic Computing. IEEE Computer, 41–51 (January 2003)
4. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.-M., Irwin, J.: Aspect-Oriented Programming. In: Akşit, M., Matsuoka, S. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997)
5. Sampath, M., Sengupta, R., Lafortune, S., Sinnamohideen, K., Teneketzis, D.: Failure diagnosis using discrete-event models. IEEE Trans. Cont. Syst. Tech. 4(2), 105–126 (1996)
6. Kim, J., Lee, E.-S.: Semantic web recommender system based personalization service for user XQuery pattern. In: Deng, X., Ye, Y. (eds.) WINE 2005. LNCS, vol. 3828, pp. 848–857. Springer, Heidelberg (2005)
7. Alur, R., Courcoubetis, C., Henzinger, T.A., Ho, P.-H.: Hybrid Automata: an algorithmic approach to the specification and verification of hybrid systems. In: Grossman, R.L., Ravn, A.P., Rischel, H., Nerode, A. (eds.) HS 1991 and HS 1992. LNCS, vol. 736, pp. 209–229. Springer, Heidelberg (1993)

Optimal Channel Sensing in Cognitive Radio Network with Multiple Secondary Users

Heejung Yu

Dept. of Inform. and Commun. Eng., Yeungnam University, Gyeongsan, Korea
heejung@yu.ac.kr

Abstract. The optimal channel sensing problem in cognitive radio networks with multiple secondary users sharing a channel with primary users based on channel sensing is considered. Based on the previous system model and results in [1], we extend to cases with multiple secondary users. The characteristics of a sum rate and the optimal sensing are investigated. It is shown that the optimal sensing point is determined depending on the primary activity factor, primary and secondary link qualities.

Keywords: Cognitive radio, Optimal channel sensing, Receiver operation characteristics, Multi-user, Multiple Access.

1 Introduction

Recently, the scarcity of frequency resource is considered as one of most critical problems in wireless communication networks. To mitigate this problem, dynamic frequency utilizations like cognitive radio communication have been studied [2], [3]. A secondary network can share the spectrum with a primary user if the interference from the secondary network does not cause harmful effects on the primary operation in cognitive radio networks. To this end, several approaches are proposed in [3], [4]. In an interweave approach, secondary transmitters access the channel based on their sensing results. Therefore, the network performance of cognitive radio dominantly depends on the capability of channel sensing for the secondary users. To avoid the dependency on the sensing performance, a geo-locational database can be used as in IEEE 802.11af standard operating in TV white spaces [5]. On the other hand, secondary transmitters can avoid the interference to the primary users by transmitting signal with very low power spectral density and making interference lower than a certain threshold which is called interference temperature.

In this paper, we focus on channel sensing based cognitive radio networks. Hence, the operating point which is given by the optimal false alarm and detection probabilities of channel sensing as well as the sensing performance is the critical factor to determine the performance. As in [1], the optimal operating point on the receiver operating characteristics (ROC) has interesting properties depending on a system rate, a primary activity factor and so on. In addition to these results, this paper further investigates the optimal sensing characteristics in cognitive radio networks with

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 838–845, 2013.

multiple secondary users to access the channel according to the centralized round-robin and decentralized p-ALOHA protocols. The rate gap between two multiple access manners is also examined.

2 System Model

A cognitive radio network with one primary transmitter-receiver pair and N secondary transmitter-receiver pairs is considered as shown in Fig. 1. This is an extension of system model in [1] to the case with multiple secondary users. The synchronized transmissions of primary and secondary packets are assumed with a slot interval T. At each slot the primary transmitter sends its packet to the corresponding receiver with a primary activity factor, i.e., a transmission probability, $\gamma \in [0,1]$. The received signal at the primary receiver is given by

$$y_p[n] = h_p s[n] + w[n], \tag{1}$$

where $s[n]$, h_p, and $w[n]$ denote a primary symbol, a channel gain between a primary transmitter and receiver, and a white Gaussian noise with zero mean and variance of σ^2. When a secondary transmitter gets a chance to transmit its own signal, the received signal at the secondary receiver can be expressed as

$$y_{s,i}[n] = h_{s,i} s_i[n] + w_i[n] \tag{2}$$

where $s_i[n]$ and $w_i[n]$ stand for the transmit signal of the i-th secondary transmitter and white Gaussian noise with zero mean and variance of σ^2 at the i-th secondary receiver, respectively. $h_{s,i}$ denotes a channel gain for the i-th secondary link. Secondary transmitters should sense the transmission of the primary transmitter to check the channel availability. Considering the sensing channel links between the primary transmitter and the secondary transmitters, the received signal at the i-th secondary transmitter is given by

$$y_i[n] = \begin{cases} h_i s[n] + v_i[n], & \textit{if the priamry TX tranmits,} \\ v_i[n], & \textit{otherwise,} \end{cases} \tag{3}$$

where h_i is the channel gain between the primary transmitter and the i-th secondary transmitter and $v_i[n]$ is white Gaussian noise with zero mean and variance of σ^2 at the i-th secondary transmitter.

Each secondary sender employs a detector to sense the primary transmission. It is assumed that all detectors are of the same type and their ROCs are given by $\{(\alpha_i, \beta(\alpha_i))\}$ where α_i and $\beta(\alpha_i)$ are the false alarm probability and the detection probability, respectively, of the detector at the i-th secondary sender. In this paper, it is assumed that all secondary users employ a matched filtering method for channel sensing. Two types of multiple access schemes are considered as a means to resolve the collision among secondary users. They are the round-robin and p-ALOHA schemes. In the round-robin scheme, all secondary users sense the same channel and report their sensing results to a central controller. When a secondary user detects the primary packet, the secondary user does not transmit their packets and wait until next time slot. If all secondary users detect the idle channel, they transmit packets one by one. In the p-ALOHA scheme, a secondary user senses the channel first and then decides to transmit a packet with probability p even when channel is idle. In the secondary transmitters, a channel is sensed during the initial T_s symbols for each slot as in [1]. If the channel is sensed to be idle, a packet is sent by a secondary transmitter to the corresponding receiver for the remaining $T-T_s$ symbols. Otherwise, it waits for the next time slot to sense the channel again. Here, it is assumed that the secondary transmitters always have packets to send to the corresponding receivers. The primary and secondary data rates under perfect sensing ($\alpha_i = 0$ and $\beta(\alpha_i) = 1$) are given by $C_p = \log\left(1+\frac{|h_p|^2}{\sigma^2}\right)$ and $C_{s,i} = \frac{(T-T_s)}{T}\log\left(1+\frac{|h_{s,i}|^2}{\sigma^2}\right)$, respectively. We consider the sum rate of all users (both primary and secondary users). When the sensing is perfect, the sum rate is given by

$$R_{s,rr} = \gamma C_p + (1-\gamma)\left\{\frac{1}{N}\sum_{i=1}^{N} C_{s,i}\right\} \tag{4}$$

for the round-robin scheme and

$$R_{s,al} = \gamma C_p + (1-\gamma)\sum_{i=1}^{N} p(1-p)^{N-1} C_{s,i} \tag{5}$$

for the p-ALOHA scheme, where γ is the primary activity factor. In practice, the sum rates in (4) and (5) are decreased due to imperfect sensing. False alarm prevents the secondary sender from transmitting its data and miss-detection causes collision among packets. When such collision occurs, we assume that no transmission is successful. In the round-robin cases, a collision occurs when one of secondary users miss-detects the channel. In the p-ALOHA scheme, a secondary user does not transmit its packet when it detects the primary packet. Incorporating the false alarm probability α and the detection probability $\beta(\alpha)$ into (4) and (5), the sum rates are rewritten as

$$R_{s,rr}(\alpha_1,\cdots,\alpha_N) = \gamma\prod_{i=1}^{N}\beta(\alpha_i)C_p + (1-\gamma)\left\{\frac{1}{N}\sum_{i=1}^{N}(1-\alpha_i)C_{s,i}\right\} \tag{6}$$

for the round-robin and

$$R_{s,al}(\alpha_1,\cdots,\alpha_N) = \gamma\prod_{i=1}^{N}\beta(\alpha_i)C_p + (1-\gamma)\sum_{i=1}^{N}p(1-p)^{N-1}(1-\alpha_i)C_{s,i} \quad (7)$$

for the *p*-ALOHA.

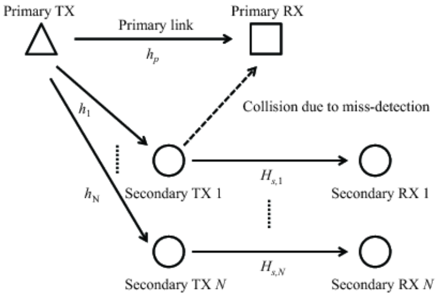


Fig. 1. Schematic of calibration

3 Optimal Channel Sensing: Single Secondary Link [1]

In this section, we review the previous results on the optimal channel sensing in a cognitive radio network with a single primary and single secondary links. In this case, we sum rum is given by a special case of (6) with *N*=1, i.e,

$$R_{sum}(\alpha,\gamma) = \gamma\beta(\alpha)C_p + (1-\gamma)(1-\alpha)C_s. \quad (8)$$

Here, we omit the subscript *i* since the number of secondary users is one. Even in the *p*-ALOHA, the optimal *p* to maximize the sum rate is given by 1/*N*=1 regardless of the operating point of channel sensing when *N*=1. Therefore, (7) is also expressed as (8) when *N*=1. The optimal operating point in ROC and the optimal sum rate with the assumption of exact knowledge of the noise variance and channel gain can be summarized as follows:

- For any value of $\gamma \in (0,1)$, there exists an optimal operating $\alpha^{opt}(\gamma)$ when the ROC curve of the channel sensor is concave, i.e., $\beta(\alpha)$ is a concave function of α. Furthermore, $\alpha^{opt}(\gamma)$ is non-decreasing in this case as the primary activity factor increases. In the case of strict concavity, $\alpha^{opt}(\gamma)$ increases monotonically, and the optimal value is unique.
- The optimal sum rate $R_{sum}^{opt}(\gamma)$ (optimized over α for each γ) is a convex function of γ for any type of ROC curve.

To examine the loss in sum rate due to imperfect sensing, with a given primary activity factor γ, we define the rate loss as

$$L(\gamma) = \frac{R_{perf}(\gamma) - R_{sum}^{opt}(\gamma)}{R_{perf}(\gamma)} \tag{9}$$

where $R_{perf}(\gamma)$ denotes the sum rate with perfect sensing, i.e., $\alpha_i = 0$ and $\beta(\alpha_i) = 1$. It is also shown that the rate loss can be greater than 1/2 regardless of the value of and the sensing SNR, i.e.,

$$\max_{\gamma} L(\gamma) \leq \frac{1}{2}. \tag{10}$$

Moreover, it is also shown that the sum rate loss with consideration of uncertainties in noise variance and channel gain is no greater than 1/2.

4 Optimal Channel Sensing: Multiple Secondary Links

In this section, without loss of generality, we assume that the secondary link qualities are ordered, i.e., $C_{s,1} \geq C_{s,2} \geq \cdots \geq C_{s,N}$. In this paper, we assume that the sensing SNR values of all secondary transmitters are identical. First, we consider the round-robin scheduling. In this case, the sum rate is given by (8). For the successful transmission of the primary user, all the secondary users should detect correctly. Since each secondary user is given equal priority in the time slot, we expect that the secondary link with better quality should operate with lower false alarm probability to be more aggressive to access the channel and increase the total sum rate, which is indeed true as stated in the following proposition.

Proposition 1. *For a fixed γ, the false alarm probabilities to maximize $R_{s,rr}$ with the round-robin scheduling are given by*

$$\alpha_{rr,1}^{opt} \leq \alpha_{rr,2}^{opt} \cdots \leq \alpha_{rr,N}^{opt},$$

if $C_{s,1} \geq C_{s,2} \cdots \geq C_{s,N}$. Additionally,

$$\frac{f'(\alpha_{rr,i}^{opt})}{f'(\alpha_{rr,i+1}^{opt})} = \frac{C_{s,i}}{C_{s,i+1}}, \quad i = 1, \cdots, N-1,$$

where $f(\alpha) = \log(\beta(\alpha))$.
Proof: See Appendix.

In the case of the decentralized p-ALOHA scheme, the sum rate is given by

$$R_{s,al} = \gamma \prod_{i=1}^{N} \beta(\alpha_i) C_p + (1-\gamma) \sum_{i=1}^{N} p(1-p)^{N-1} (1-\alpha_i) C_{s,i}. \tag{11}$$

The optimal access probability p is easily found to be the well-known solution $p^{opt} = \frac{1}{N}$ by maximizing $R_{s,al}$ with respect to p. This optimality is guaranteed regardless of α_i. With this optimal access probability, the optimal false alarm probability $\alpha_{al,i}^{opt}$ is obtained by solving $\frac{\partial R_{s,al}}{\partial \alpha_i} = 0$, i.e.,

$$\gamma\beta'(\alpha_i)\prod_{j\neq i}\beta(\alpha_j)C_p = (1-\gamma)\frac{1}{N}\left(1-\frac{1}{N}\right)^{N-1}C_{s,i}. \tag{12}$$

In this case, we can also show that the secondary user with better link quality operates with a lower false alarm probability as in the round-robin scheme. Comparing the operating false alarm rates in both cases, we obtain the following result.

Proposition 2. *Given the same primary and secondary link capacities and the primary activity factor, the optimal false alarm probability of the round-robin scheduling is lower than that of the p-ALOHA scheme*

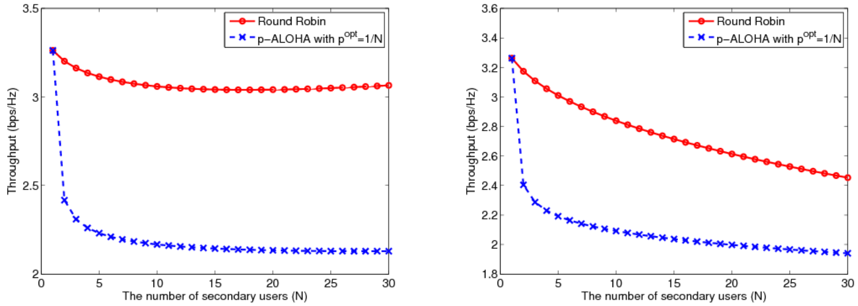


Fig. 2. Sum rates of round-robin and p-ALOHA: (left) secondary link SNR = 10:0.2: 10+0.2 (N-1) and (right) 10: -0.2:10-0.2(N-1) (in matlab notation)

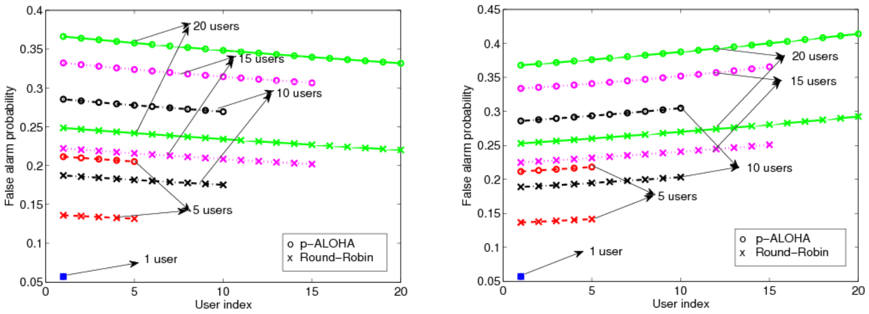


Fig. 3. Optimal false alarm probabilities with a different number of secondary users with SNR = (left) 10: 0.2: 10+0.2(N-1) and (right) 10: -0.2:10-0.2(N-1)

Proof: See Appendix.

This result is intuitive. Since there are more secondary users who try to access the channel at each slot in the decentralized scheme, the collision by secondary users is mitigated by the increased false alarm rate for a higher sum rate. Fig. 2 shows the throughput for the multiple secondary link case. The SNR of the primary link is 10dB. The SNR value of secondary links is increased or decreased by 0.2dB from 10dB as the number of users increases. The sensing SNR and primary activity factor are -10dB and $\gamma = 0.5$, respectively. Fig. 3 shows the optimal false alarm probability achieving the above sum throughput. As shown in the figure, the false alarm probability of a secondary user with higher capacity is lower than that of another secondary user with lower capacity. When a user with higher capacity is added, therefore, the optimal false alarm probability decreases. Note that the optimal false alarm probability increases as the total number of secondary users increases. It is also seen that the optimal false alarm probability of the *p*-ALOHA scheme is higher than that of the round-robin scheme, as expected.

5 Conclusion

We have considered the problem of optimal channel sensing in cognitive radio networks with multiple secondary users accessing the channel with round-robin and *p*-ALOHA approaches. With defined multiple access scenarios, we have formulated the sum rates and investigated the properties of optimal operating point of channel sensing.

Acknowledgement. This work was supported by the 2012 Yeungnam University Research Grant.

A Appendix

Proof of Proposition 4.

With the sum rate given by (6), the optimal operating point of the *i*-th user is the solution to $\frac{\partial R_{s,rr}}{\partial \alpha_i} = 0$. Therefore, the optimal false alarm probability $\alpha_{rr,i}^{opt}$ can be obtained by solving the following equation:

$$\gamma\beta'(\alpha_i)\prod_{j\neq i}\beta(\alpha_j)C_p = (1-\gamma)\frac{1}{N}\left(1-\frac{1}{N}\right)^{N-1} C_{s,i}. \tag{13}$$

In the above equation, $\beta'(\alpha_i)$ is a non-increasing function of α_i since $\beta(\alpha_i)$ is concave with respect to α_i. If we consider *i*-th and (*i*+1)-th users with $C_{s,i}$ and $C_{s,i+1}$, we can find the following equation:

$$\gamma\beta'(\alpha_i)\prod_{j\neq i}\beta(\alpha_j)C_p = (1-\gamma)\frac{1}{N}\left(1-\frac{1}{N}\right)^{N-1}C_{s,i}. \tag{14}$$

Because $\frac{C_{s,i}}{C_{s,i+1}} \geq 1$,

$$\frac{\beta'(\alpha_{rr,i}^{opt})}{\beta(\alpha_{rr,i}^{opt})} \geq \frac{\beta'(\alpha_{rr,i+1}^{opt})}{\beta(\alpha_{rr,i+1}^{opt})}. \tag{15}$$

By the definition of $f(\alpha)$, $f'(\alpha)=\frac{\beta'(\alpha)}{\beta(\alpha)}$. $f(\alpha)$ is a concave and non-decreasing functions of α because of the concave and non-decreasing properties of $\log(\cdot)$ and $\beta(\alpha)$. Hence, $\alpha_{rr,i}^{opt} \leq \alpha_{rr,i+1}^{opt}$ when $C_{s,i} \geq C_{s,i+1}$.

Proof of Proposition 4.

The only difference between (13) and (12) is $(1-1/N)^{N-1}$ in front of the secondary user capacity. When $N > 1$, $(1-1/N)^{N-1} < 1$. Therefore, the solution to (13) is less than that to (12) because $\beta'(\alpha)$ is a non-increasing function of α.

References

1. Yu, H.: Optimal Channel Sensing for Maximizing System Rates in Cognitive Radio: An Analytical Approach. Submitted to ETRI Journal (2012)
2. Haykin, S.: Cognitive radio: Brain-empowered wireless communications. IEEE J. Sel. Areas Commun. 23(2), 201–220 (2005)
3. Zhao, Q., Sadler, B.: A survey of dynamic spectrum access. IEEE Signal Process. Mag. 24(3), 79–89 (2007)
4. Zhang, R., Liang, Y.-C.: Exploiting multi-antenna for opportunistic spectrum sharing in cognitive radio networks. IEEE J. Sel. Topics in Signal Process. 2(1), 88–102 (2008)
5. IEEE Std. P802.11af D2.0: Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 3: TV White Spaces Operation. IEEE, Piscataway, NJ (2012)

H.264 Video Delivery over Wireless Mesh Networks Based on Joint Adaptive Cross-Layer Mapping and MDCA MAC

Byung Joon Oh[1] and Ki Young Lee[2]

[1] Dept. of Engineering, Link Communications, Ltd., Annapolis Junction, MD 20701, USA
byungjoonoh@lnkcom.com
[2] Dept. of Info and Telecom Engineering, University of Incheon, Incheon 406-749, Korea
kylee@incheon.ac.kr

Abstract. This paper presents a QoS-guaranteed transmission of H.264 video over Wireless Mesh Networks (WMNs) based on an adaptive cross-layer mapping of IEEE 802.11e MAC strategy. We call this MDCA (Mesh Distributed Channel Access) MAC strategy as it is based on 802.11e standard with adaptive mechanism in response to dynamic nature of mesh networks. This novel MDCA strategy employs the channel reservation control packets at the MAC layer to exchange timely Channel State Estimation information for an optimal FEC at the application layer as well as the QoS-centric GEDSR model [1] for an optimal adaptation. The proposed scheme offers an optimized transmission to guarantee the minimum packets delay and drop rate needed for video over WMNs. In this research, we resolve the problem associated with 802.11e standard by designing an integrated scheme that allows the system to achieve the optimal transmission via a FEC implemented in the application layer. We evaluate the proposed scheme based on network-level metrics, including bit rate, packets delay and drop rates in comparison with the static cross-layer mapping scheme based on 802.11e WMNs. We can confirm that the adaptive cross-layer mapping strategy MDCA outperforms the static cross-layer mapping scheme by a significant margin.

1 Introduction

In recent years, WMNs have gained massive research interest [2]-[5]. Due to their fast configuration and low cost, they can be easily deployed for multimedia delivery, such as IPTV, VOD, and mobile digital video recorder systems [3]. However, it is difficult to guarantee the QoS for video streaming over WMNs because of their dynamic nature. In particular, the QoS issue has not yet been adequately investigated based on the recently finalized H.264 video coding standard [4]-[7]. Therefore, there are still several research challenges that need to be addressed in all protocol layers such as physical layer[1] and MAC [5][7][11][12], network and transport layer [8], application layer [1], and cross-layer design [8]-[10] for WMNs to support H.264 video streaming applications. The issue of QoS has been addressed in WMNs applications. Shen et al. proposed in [5] an admission control based on available bandwidth estimation for WMNs. It was shown that admission control algorithm at the MAC layer

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 846–853, 2013.

could resolve the QoS issue for both real-time and non-real-time traffic. However, they considered only throughput, delay, and jitter, not packet loss rate which is crucial QoS factors that significantly affect the performance of video streaming [10]. In [7], an Enhanced Distributed Channel Access scheme with resource reservation (EDCA/RR) that provides deterministic, contention-free medium access is proposed. However, only the QoS of EDCA/RR MAC, not the QoS of video streaming is studied. In [2], we addressed the network-level performances including throughput, packet loss rate and delay for robust H.264 video transmission over WMNs. We developed an Opportunistic Multi Rate MAC that can be viewed as static cross-layer framework without adaptation. Two key innovations of the proposed joint adaptive scheme are: (1) a novel MDCA (Mesh Distributed Channel Access) scheme based on 802.11e standard of MAC layer [11] and (2) an adaptive FEC implemented in application layer based on effective QoS (GOP-level Estimation Decodable Slice Rate (GEDSR)) Model [1]. Based on channel state estimation and GEDSR model, the joint adaptation based on MDCA and adaptive FEC is designed to improve the quality of the link under error-prone transmission conditions. We apply an unequal error protection for H.264 video traffic through an adaptive cross-layer mapping strategy in order to dynamically adapt Access Category (AC) [10]. This adaptive strategy is able to overcome unnecessary transmission delays and packet losses as we encountered in static cross-layer mapping in [2].

2 Architecture of Adaptive Cross-Layer Mapping Strategy

2.1 Analysis of Adaptive Cross-Layer Mapping Scheme

As MDCA supports different precedence AC queues according to video coding significance, encoded H.264 data is also allocated accordingly. When the mapping scheme is static and non-adaptive, the video data mapped to lower priority AC such as AC[1] and AC[0] may cause packet loss and unnecessary transmission delays even when the network load is light. Therefore, when the AC[2] queue is empty (which indicates the video traffic load is light), the static mapping algorithm will lead to high packet losses as well as unnecessary transmission delays if both AC[1] and AC[0] are almost full simultaneously. Figure 1 illustrates the proposed architecture for adaptive cross-layer mapping policy.

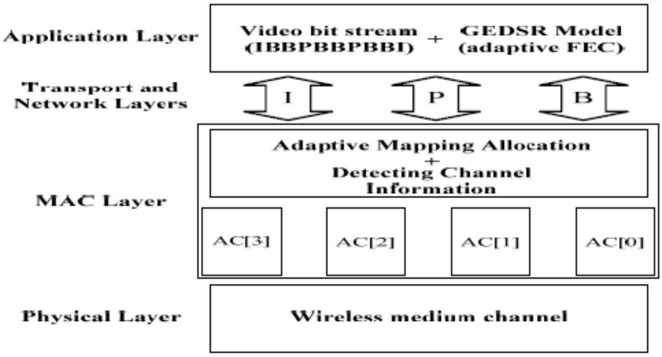

Application Layer
Video bit stream (IBBPBBPBBI) + GEDSR Model (adaptive FEC)
Transport and Network Layers
I
P
B
Adaptive Mapping Allocation + Detecting Channel Information
MAC Layer
AC[3]
AC[2]
AC[1]
AC[0]
Physical Layer
Wireless medium channel

Fig. 1. Architecture of Adaptive Cross-layer Mapping Scheme

Based on the significance of video type and the current load of network traffic, the proposed mapping algorithm dynamically distributes the video data into the most appropriated AC so as to guarantee the QoS metrics as well as the visual quality of delivered video at the MAC layer. At 802.11e MAC layer, we allocate an important video data (I-slice) into higher priority AC queue and we defined different mapping probabilities as *P(Type)* to different video slice types according to its coding significance. Less important video slice types will be assigned larger *P(Type)*.Therefore, for H.264 codec, the downward mapping probability relationship of these three video slice types is *P(B)>P(P)>P(I)*, and these probabilities are between 0 and 1. When transmitted over an 802.11e WMNs, H.264 video packets are placed in AC[2] category will have better opportunity to admit to the channel than lower priority ACs. The proposed mapping algorithm reschedule most recently received video packets into other available lower priority queues, while the AC[2] queue is getting filled. To predicatively avoid the upcoming congestion by performing a queue supervision in advance, we define two parameters, $Threshold_{low}$ and $Threshold_{high}$. To incorporate these two parameters into the algorithm, the integrated function will be:

$$P(New) = P(Type) \times \frac{Qlen_{AC[2]} - Threshol_{low}}{Threshold_{high} - Threshold_{low}} \tag{1}$$

The original predefined downward mapping probability of each type of video slice in this equation *P(Type)* will be adjusted according to the current queue length and threshold values. The result is a new downward mapping probability *P(New)*. The higher the value of *P(New)*, the greater the chance for a packet to be mapped into a lower priority queue. Table 1 presents the notations used in the proposed adaptive cross-layer mapping algorithm.

Table 1. Parameter Notations in Proposed Adaptive Mapping Algorithm

Term	Definitions
$P(Type)$	Download mapping probability of each type video packet ($P(I)$, $P(P)$, $P(B)$)
$P(New)$	New computed downward mapping probability
$Threshold_{low}$	The lower threshold of queue length
$Threshold_{high}$	The lower threshold of queue length
$Qlen_{AC[2]}$	The queue length of Access Category 2

The pseudo code of mapping policy is shown in Figure 2. When a video packet arrives, the queue length of AC2 ($Qlen_{\mathrm{AC[2]}}$) is checked. If the queue length is lower than the lower threshold value, $Threshold_{low}$ (light load), the video data is mapped into AC[2]. However, if the queue length is greater than the upper threshold value, $Threshold_{high}$ (heavy video traffic load) the video data is straightforwardly mapped to lower priority queues, AC[1] or AC[0]. However, when the queue length of AC[2] decreases between $Threshold_{high}$ and $Threshold_{low}$, the mapping decision is made based on both mapping probability (*P(Type)*) and the current buffering size of the queue as given by (1). Hence, based on the estimated downward mapping probability, the video data packet will be mapped to either AC[2], AC[1] or AC[0]. By exploiting such a priority scheme and queue length management strategy of MDCA MAC, the video transmission is prioritized and the drop rate of video can be minimized to enable efficient utilization of network resources.

```
When a video data slice arrives:
If (Qlen AC[2]   < Threshold low )
     Video packet → AC[2];
Else if (Qlen AC[2]   < Threshold high ) {
            $P(New) = P(Type) \times \frac{Qlen_{AC[2]}-Threshol_{low}}{Threshold_{high}-Threshold_{low}}$
RN = a random number generated from Uniform function (0.0, 1.0);
If (RN > P(New) )
                Video slice → AC[2];
Else
                Video slice → AC[1];
}
Else If(Qlen AC[2]   > Threshold high ) {
                If(RN > P(Type) ) {
                Video slice → AC[1];
                Else
                Video slice → AC[0]; }
```

Fig. 2. The Proposed Adaptive Cross-Layer Mapping Strategy

2.2 GEDSR Model for Adaptive FEC

To characterize and estimate the dependence and sensitivity of video streams, we adopt the GEDSR model. This is a network-level metric and is defined as the fraction of decodable slice rate, which is the total number of decodable slices over the total number of slices transmitted by the sender as follows:

$$GEDSR=N_{dec}/(N_I + N_P + N_B) \qquad (2)$$

where N_{dec} is the summation of $N_{I\text{-slice dec}}$, $N_{P\text{-slice dec}}$ and $N_{B\text{-slice dec}}$. It is clear that, the larger the GEDSR value, the better the video quality as received by the receiver. If we denote the probability that a slice α is regarded as decodable by *P(α)*, then, the probability *P(I)* that the I-slice in GOP_i is decodable is simply as follows:

$$P(I)=(1-\xi I)^{Avgpacket_I} \qquad (3)$$

$$N_{I\text{-slide dec}} = P(I) \times N_{GOPi} \qquad (4)$$

where ξI stands for packet loss rate, $Avgpacket_I$ is the average number of packets to carry the data of each type of I-slice and N_{GOPi} represents total number of GOPs. The probability of the P-slice can be obtained as:

$$P(P_{Np})=(1-\xi I)^{Avgpacket_I} \quad (1-\xi P)^{Avgpacket_P * Np} \qquad (5)$$

With all these derivations, the expected number of decodable P-slices for the entire video will be:

$$N_{P-slice\ dec} = P(I) \times \sum_{j=1}^{Np}(1-\xi\mathrm{P})^{j\times Avgpacket_P} \times N_{GOPi} \qquad (6)$$

where ξP represents packet loss rate, $Avgpacket_P$ is the average number of packets to carry the data of each type of P-slice. It can be observed that the channel state feedback and adaptive FEC can be incorporated with GEDSR. Especially, as shown in Figure. 3, with CSE information, we can design adaptive FEC that allows N_{dec} to achieve higher value in order to improve the GEDSR parameter and the received video quality. The channel state estimation algorithm was illustrated in [1] in detail.

$$\alpha = 1 - \frac{queue_{high\_threshold\_of\_retry} - retry}{queue_{high\_threshold\_of\_retry} - queue_{low\_threshold\_of\_retry}} \tag{7}$$

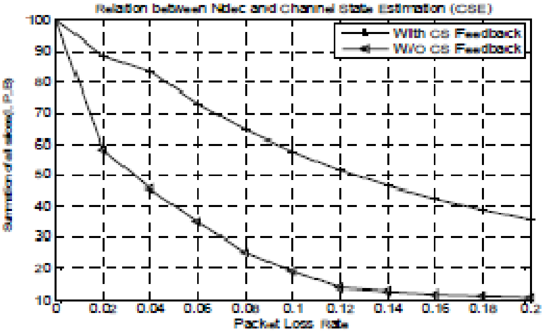


Fig. 3. Relation between N_{dec} and CSE

MP reduces the number of redundant FEC packets based on the current retry time: *retry (weighted moving average retry time)* = (1- rweight) * current_*retry* (*current retry time*) + rweight * *retry*. Note that $queue_{low\_threshold\_of\text{-}retry}$ and $queue_{high\_threshold\_of\text{-}retry}$ are 5 and 15, respectively. Consequently, the number of adaptive FEC is implemented as shown in Figure 4.

```
If ( retry < queue low_ threshold of retry )
        FEC no (number of redundant FEC) = 0;
Else if ( retry < queue high _ threshold of retry )
        FEC no = FEC no ×α (adaptive factor) ;
Else
        FEC no = FEC no ;
```

Fig. 4. Pseudo code of adaptive FEC Algorithm

3 Experimental Results

In this research, simulations have been carried out to compare the performance of static cross-layer and adaptive cross-layer mapping algorithm for video streaming over wireless mesh networks. Specifically, we implement the hybrid mesh mode simulation topology that consists of 14 mobile stations with 4 mesh clients, 4 conventional clients, and 6 mesh points. The bit rate is at 1Mbps, and several system parameters are based on physical layer parameters used in the 802.11b standard. In addition, in order to implement more complicated channel model that is close to practical network setting, we adopt the Rayleigh fading statistical channel in combination with the Finite-state Markov chain channel models [13]. This is more realistic than existing approaches in which they have not considered the impact of wireless fading channel on video transmission quality over wireless mesh networks. Figure 5(a) illustrates the performance of the throughput in destination nodes based on the simulation topology described above using NS-2. Both static and adaptive cross-layer mechanism have the similar throughput improvements. Figure 5(b) shows the dropping rate performance comparison. As expected, adaptive cross-layer algorithm outperforms static cross-layer scheme resulting because of full CSE information feedback and higher total number of decodable slices as shown in Figure 3.

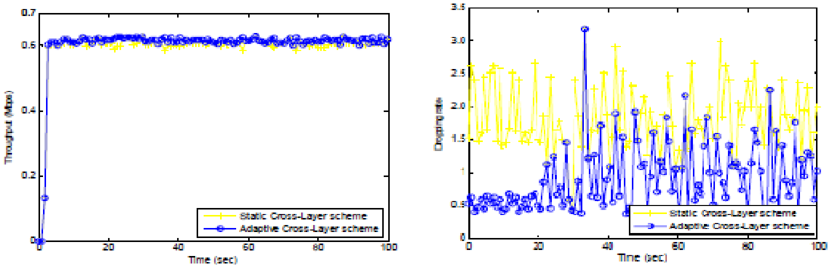


Fig. 5. Throughput and dropping rate comparison

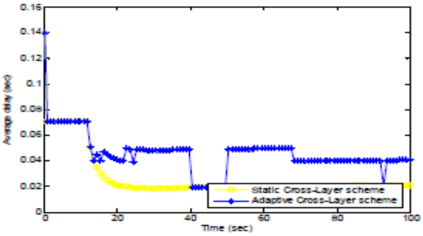
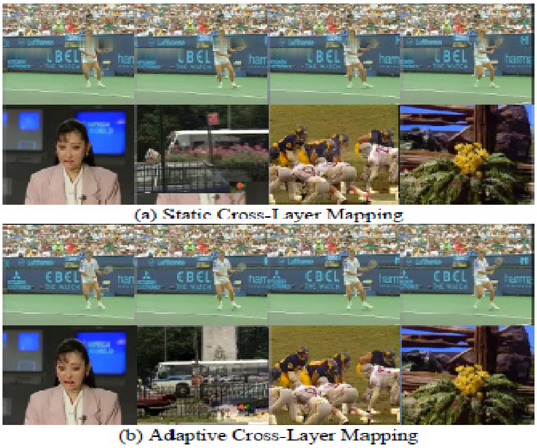


Fig. 6. Delay comparison

Figure 6 shows the average delay performance comparisons. The static cross-layer mechanism actually outperforms the proposed adaptive cross-layer scheme. With CSE, adaptive cross-layer scheme needs additional time to accommodate the feedback. However, the delay is well under the acceptable range. Overall, we can observe that the performance of our proposed model is significantly better than that of static cross-layer mapping, especially in terms of packet drop rate. In addition to the simulation results of relevant QoS metrics, we have also estimated the subjective quality of this H.264 video transmission for both static cross-layer and adaptive cross-layer mapping schemes as shown in Figure 7. It is clear that the overall subjective quality of the proposed adaptive scheme is noticeably better than that of the static scheme.



(a) Static Cross-Layer Mapping

(b) Adaptive Cross-Layer Mapping

Fig. 7. Evaluation of Video Streaming Transmissions

4 Conclusions

We have described in this paper a novel adaptive cross-layer mapping strategy for MAC protocol to achieve reliable delivery of the H.264 video streaming over wireless mesh networks. Based on the unique dynamic characteristic of wireless mesh networks, we developed an adaptive cross-layer mapping based MDCA MAC, making full use of the GEDSR model for the adaptation and application of adaptive FEC to combat wireless channel impairments. We adopt both network-level QoS metrics as well as received video quality at the receiving node to evaluate the proposed adaptive cross-layer mapping scheme against the static cross-layer mapping scheme for H.264 video over WMNs. The simulation results have confirmed that the proposed scheme is able to substantially outperform the static cross-layer mapping scheme with an average of 1.5dB in reconstructed video quality. Future extension of the proposed research include the adaptation of H.264 scalable video coding standard in order to meet the desired scalability requirements for a wider range of MAC configurations.

Acknowledgments. This work was supported by the University of Incheon Research Grant in 2010.

References

1. Oh, B.J., Hua, G., Chen, C.W.: Seamless Video Transmission over Wireless LANs based on an effective QoS Model and Channel State Estimation. In: Proc. of IEEE ICCCN, pp. 1–6 (2008)
2. Oh, B.J., Chen, C.W.: An Opportunistic Multi Rate MAC for Reliable H.264/AVC Video Streaming over Wireless Mesh Networks. In: Proc. of IEEE ISCAS, pp. 1241–1244 (May 2009)

3. Mobile Digital Video Recorder (MDVR) of Link Communications, Ltd., `http://www.lnkcom.com`
4. Athanasiou, G., Korakis, T., Ercetin, O., Tassiulas, L.: A Cross-Layer Framework for Association Control in Wireless Mesh Networks. IEEE Trans. on Mobile Computing 8, 65–80 (2009)
5. Shen, Q., Fang, X., Li, P., Fang, Y.: Admission Control Based on Available Bandwidth Estimation for Wireless Mesh Networks. IEEE Trans. on VT 58, 2519–2528 (2009)
6. Mogre, P.S., Hollick, M., Steinmetz, R.: QoS in Wireless Mesh Networks: Challenges, Pitfalls, and Roadmap to its Realization. In: Proc. of ACM NOSSDAV (June 2007)
7. Hamidian, A., Korner, U.: QoS Provisioning in Wireless Mesh Networks. In: Proc. of EuroN-GI/FGI Workshop on Wireless and Mobility, Barcelona, Spain (January 2008)
8. Moleme, N.H., Odhiambo, M.O., Kurien, A.M.: Improving Video Streaming Over IEEE 802.11 Mesh Networks through a Cross-Layer Design Technique. In: Proc. of IEEE BroadCom, Pretoria, South Africa, pp. 50–57 (November 2008)
9. Oh, B.J., Chen, C.W.: A Cross-Layer Oriented Multi-Channel MAC Protocol Design for QoS-Centric Video Streaming over Wireless Ad Hoc Networks. In: Proc. of IEEE ICME, New York, USA, pp. 774–777 (June 2009)
10. Oh, B.J., Chen, C.W.: A Cross-Layer Approach to Multi-Channel MAC Protocol Design for Video Streaming over Wireless Ad Hoc Networks. IEEE Trans. Multimedia 11, 1052–1061 (2009)
11. Oh, B.J., Chen, C.W.: Energy Efficient H.264 Video Transmission over Wireless Ad Hoc Networks based on Adaptive 802.11e EDCA MAC Protocol. In: Proc. of IEEE ICME, Hanover, Germany, pp. 1389–1392 (June 2008)
12. Hiertz, G., Max, S., Zhao, R., Denteneer, D., Berlemann, L.: Principles of IEEE 802.11s. In: Proc. of IEEE ICCCN, Honolulu, Hawaii, USA, pp. 1002–1007 (August 2007)
13. Oh, B.J.: Supporting Multimedia Quality of Service (QoS) in Wireless Networks. Ph. D. Dissertation, Dept. of ECE, Florida Institute of Technology, Melbourne, FL (December 2008)

Automatic Tracking Angle of Arrival of Bandpass Sampling OFDM Signal by MUSIC Algorithm

Xin Wang and Heung-Gyoon Ryu

Department of Electronic Engineering, Chungbuk National University, Cheongju, Korea
wxzf007@naver.com, ecomm@cbu.ac.kr

Abstract. In this paper, a combined OFDM system and bandpass sampling method using Multiple Signal Classification(MUSIC) algorithm for automatic (angle of arrival) AOA tracking is discussed. And we propose a new method that adding (time division multiplexing)TDM with bandpass sampling in the same time to avoid interference due to RF filter characteristics. Also, we consider Doppler effect for the targets' movement and after compensating the Doppler effect with a valid range , the system performances well. Computer simulation shows that the performances of MUSIC spectrum for AOA due to various conditions and demonstrates the accuracy of AOA estimations.

Keywords: MUSIC, AOA, Bandpass sampling, OFDM, Doppler effect.

1 Introduction

Angle of arrival estimation technology play an important role in enhancing the performance of adaptive arrays for mobile wireless communications[1]. A number of angle of arrival estimation algorithms have been developed. For the most recent ones being MUSCI[2] and ESPRIT[3] algorithms, who both utilizing subspace-based on exploiting the eigen structure of the input covariance matrix and thus requires a higher computation effort. Although ESPRIT needs less computation, the MUSIC algorithm is found to be more stable and accurate[4]. In this paper, we use the MUSIC algorithm combine the OFDM bandpass sampling signal model to perform the antennas sensing to allow accurate azimuth. The accuracy of the estimation in azimuth increases proportional to the number of antenna elements utilized.

Bandpass sampling can be used for direct down conversion without analog mixers. In practice, the required sampling rate for ADC can be too high to be achieved if the Nyquist sampling theorem is to be satisfied[5]. So we use bandpass sampling which is a technique that samples high data rate signals with smaller sampling rate than Nyquist sampling rate to relax the demand for ADCs. After down-sampling about over two band signals using bandpass sampling, the signals are digitized and then two band signals can be received [6].

In this paper, we propose a bandpass sampling technique with time division multiplexing (TDM). In previous system, although over two signals can be down-sampling without interference between signals, it is possible to generate interference due to RF filter characteristics. RF filter cannot cut adjacent band signals so the

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 854–859, 2013.

remaining adjacent band signals (undesired signals) can affect desired signals. So we propose bandpass sampling with TDM that can avoid previous problems to separate over two signals timely.

2 System Model

In this paper we consider two signals that have different center frequency. Transmitted signals are based on OFDM. Eq. (1) is the signals in time domain. Assume that there are two received bands. $X_{k,m}^{A}$ and $X_{k,m}^{B}$ are transmitted signals respectively. As Eq. (1), the signals is represented after IFFT in time domain.

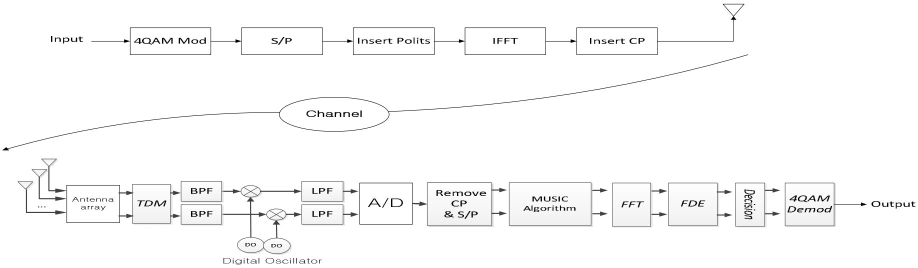


Fig. 1. System Model

$$x(t)=\begin{cases}\dfrac{1}{\sqrt{N}}\sum_{k=0}^{N-1}X_{k,m}^{A}e^{j(\frac{2\pi k}{N}+f_A)t}, x_A(t)\\ \dfrac{1}{\sqrt{N}}\sum_{k=0}^{N-1}X_{k,m}^{B}e^{j(\frac{2\pi k}{N}+f_B)t}, x_B(t)\end{cases} \tag{1}$$

Assume that there are P(P <M) uncorrelated narrowband signals received by ULA from different direction θ_p ,corrupted by AWGN, where p=1,2… P. The observation is given as

$$X(t)=\sum_{p=1}^{P}a(\theta_p)*x_p(t)+n(t) \tag{2}$$

where $a(\theta)$ is the array steering vector given by

$$a(\theta)=[1\ \ \mathrm{e}^{-j2\pi d\sin\theta/\lambda}\cdots\mathrm{e}^{-j2\pi d\sin\theta(M-1)/\lambda}]^T \tag{3}$$

where d is the inter element spacing, λ is the signal wavelength. When we take snapshot at time k=1,2…K, we can get

$$X(k)=\sum_{p=1}^{P}a(\theta_p)*x_p(k)+n(k) \tag{4}$$

where noise $n(k)$ is assumed to be both temporally and spatially white, and uncorrelated with signal $s_p(k)$.

MUSIC stands for MUltiple SIgnal Classifacation. The covariance matrix, R ,is the collected data for each of the array receivers in the time domain. The correlation matrix is given as[7]

$$R = E\left[XX^{H}\right] = AR_{s}A^{H} + \sigma^{2}I \tag{5}$$

where R_s is the $P \times P$ signal correlation matrix. σ^2 is the white noise power. The noise subspace E_N used in MUSIC can be obtained from eigenvalue decomposition of R , and the spatial spectrum of MUSIC is given by

$$P(\theta) = \frac{1}{a(\theta)^{H} E_{N} E_{N}^{H} a(\theta)} \tag{6}$$

3 Doppler Effect and Compensation

The orthogonally among subcarriers is often destroyed by the CFOs due to oscillator mismatches. So Doppler effect was generated and degrades performance. Doppler effects cause shifting in frequency domain and phase rotation in time domain. Signal x(t) is like (7) due to Doppler effect.

$$y_n = \sum_{k=0}^{N-1} H_k \cdot X_k \cdot e^{i2\pi\frac{k+\varepsilon}{N}} + z_n \tag{7}$$

Signal x(t) is like (8) due to Doppler effect in time domain. Channel H is represented as product of X. Doppler effect is represented phase rotation in frequency domain. k , n, ε are sub-carrier, symbol, normalized Doppler frequency respectively in (7).

$$\begin{aligned} Y_p &= \sum_{m=0}^{N-1}\sum_{k=0}^{N-1} H_{k,m} \cdot X_{k,m} \cdot e^{i2\pi\frac{(k+\varepsilon)}{N}} \cdot e^{-i2\pi\frac{m}{N}} + Z_p \\ &= H_p \cdot X_p\, e^{i2\pi\varepsilon p} + \sum_{\substack{m=0\\ m\neq k}}^{N-1}\sum_{k=0}^{N-1} H_{k,m} \cdot X_{k,m} \cdot e^{i2\pi\frac{(k-m)}{N}} \cdot e^{i2\pi\frac{\varepsilon}{N}} + Z_p \end{aligned} \tag{8}$$

In (8), first stage is phase rotation and second stage is ICI. Where p is symbol in frequency domain and k, m are sub-carrier before IFFT in transmitter and sample before FFT in receiver.

$$\varepsilon = \frac{f_d}{carrier\ spacing}, \quad f_d = \frac{v \cdot f_c}{c} \tag{9}$$

We compensate those problems with synchronization signal and block type pilot and assume that the receiver speed is constant. fd, c, v are Doppler frequency, the velocity of light, the speed of receiver respectively.

$$Y_p = H_p \cdot X_p\, e^{i2\pi\varepsilon p} + Z_p \tag{10}$$

Phase rotation is estimated using received pilot signals.

$$P(i) = \sum_{i=1}^{N} mean\left\{ \sum_{n=1}^{64} Block\_Pilot(i+n-1) \right\} \tag{11}$$

$$\frac{angle\{P(i)\} - angle\{P(i+1)\}}{pilot\_interval} \cdot ([1 : pilot\_interval - 1]) \tag{12}$$
$$i = 1, 2, \ldots$$

P is average of block type pilot. Eq. (12) represents linear interpolation using P, so the symbols that have not pilots is estimated.

4 Proposed Bandpass Sampling Method

4.1 Existing Structure

Existing multi-band system with bandpass sampling finds sampling frequency that doesn't overlap signals between multi-band signals according to (7). For RF filter can't remove all adjacent signals, the remaining adjacent signal is able to be overlap when multi-band signals are converted at low frequency band.

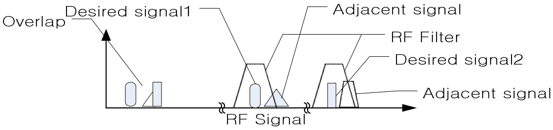


Fig. 2. The problem when signals are to be sub-sampling from RF band.System Model

Bandpass sampling about multi-band of over 2 bands meet condition like (13) [7]. To convert the two signals in low frequency band without interference between signals, $F_{IF,A}$ and $F_{IF,B}$ have to meet (13).

$$\begin{aligned}
&0 < F_{IF,A} - BW_A/2, \quad F_S > F_{IF,A} - BW_A/2 \\
&0 < F_{IF,B} - BW_B/2, \quad F_S > F_{IF,B} - BW_B/2 \\
&if\ F_{IF,B} > F_{IF,A} \\
&\qquad F_{IF,B} - BW_B/2 > F_{IF,A} + BW_A/2 \\
&if\ F_{IF,A} > F_{IF,B} \\
&\qquad F_{IF,A} - BW_A/2 > F_{IF,B} + BW_B/2
\end{aligned} \tag{13}$$

4.2 Proposed Structure

We propose a method that adds TDM method in bandpass sampling method.

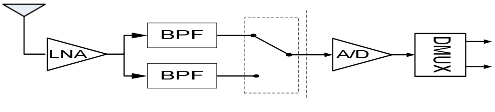


Fig. 3. A multi-band receiver structure that joint bandpass sampling and TDM method

The proposed structure is like figure 4.

$$\begin{aligned} 0 &< F_{IF,A} - BW_A / 2, \quad F_S > F_{IF,A} - BW_A / 2 \\ 0 &< F_{IF,B} - BW_B / 2, \quad F_S > F_{IF,B} - BW_B / 2 \end{aligned} \tag{14}$$

The signals that are received with TDM has no interference between receiving singles because the signals is divided in time. Therefore, the converted signals just satisfy (14) instead of (13). So it is possible to give an low sampling frequency.

5 Simulation and Discussion

Figure 4 indicates BER performance when Doppler effect occurs. We can see the performance according to Doppler scale. The two bands have no difference according to Doppler scale. The two bands have no difference due to TDM. In the case of Doppler effect ε =0.01 ,for that both A band and B band without compensating, we can't communicate because of the phase rotation. And after compensating phase rotation, there is small performance degradation comparing to the theory curve because of existing ICI. And when the Doppler effect is give ε=0.05, we can't communicate as we use block type pilot and do linear interpolation which is difficult to estimate fast phase rotation.

Table 1. Simulation Parameters

OFDM system	
The number of Subcarriers	64
The number of Sensors	8
Pilot Type	Block Type Pilot
Modulation	QAM
Channel	AWGN

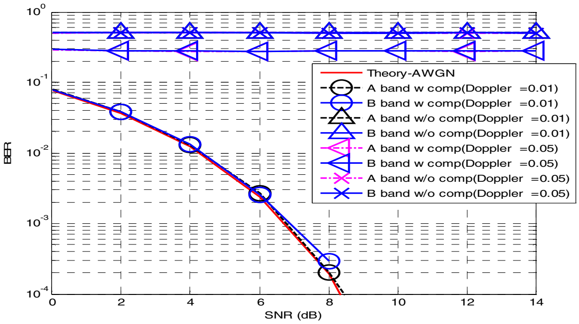


Fig. 4. BER performance with Doppler effect

Figure 5 shows spatial spectrum of MUSIC with 4 receiver antenna arrays due to different SNRs. We can see that both target A and target B are tracked with accurate angle whose are 10°and 50°. And we can also see that more higher the SNR is, the shaper the spectrum pointing the angle performances.

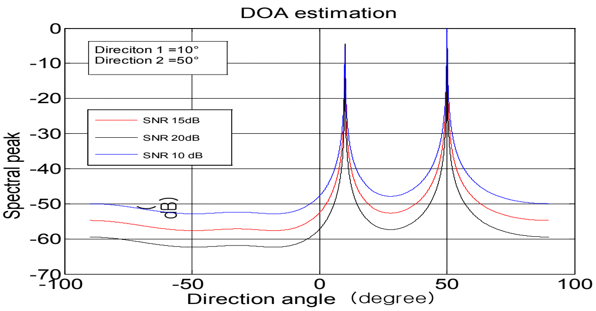


Fig. 5. AOA estimation due to SNR change

6 Conclusion

In this paper, we discussed and performance a automatic AOA tracking method using MUSIC algorithm by bandpass sampling method. And we also proposed a adding TDM with bandpass sampling method which can avoid interference. By considering the Doppler effect and compensating the effect, system using proposed method performances well. And simulation shows using MUSIC algorithm to estimate the AOA under different conditions.

Acknowledgements. This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2012017339).

References

1. Schmidt, R.O.: Multiple emitter location and signal parameter estimation. IEEE Trans. Antennas Propag. AP-34, 276–280 (1986)
2. Paulraj, A., Roy, R., Kailath, T.: A subspace rotation approach to signal parameter estimation. Proceedings of the IEEE 74(7), 1044–1046 (1986)
3. Lavate, T., Kokate, V., Sapkal, A.: Performance analysis of MUSIC and ESPRIT DoA estimation algorithms for adaptive array smart antenna in mobile communication. International Journal of Computer Networks (IJCN) 2(3), 152–172 (2010)
4. Walden, R.H.: Performance trends for analog-to-digital converters. IEEE Commun. Mag. 37(2), 96–101 (1999)
5. Akos, D.M., Stockmaster, M., Tsui, J.B.Y., Caschera, J.: Direct bandpass sampling of multiple distinct RF signals. IEEE Trans. Commun. 47(7), 983–988 (1999)
6. Wang, J., Zhao, Y.J., Wang, Z.G.: A MUSIC like DOA estimation method for signals with low SNR. In: GSSM 2008, pp. 321–324 (2008)
7. Tseng, C.-H., Chou, S.-C.: Direct Downconversion of Multiband RF Signals Using Band Pass Sampling. IEEE Trans. Commun. 5(1), 72–76 (2006)

A White-List Based Security Architecture (WLSA) for the Safe Mobile Office in the BYOD Era

Jaeho Lee[1], Yongjin Lee[1], and Seung-Cheon Kim[2]

[1] NIA (National Information Society Agency), Seoul, Korea
{jaeho,leeyj}@nia.or.kr
[2] Dept. of Information and Communication Eng., Hansung Univ., Seoul, Korea
kimsc@hansung.ac.kr

Abstract. The BYOD (Bring Your Own Device) based mobile office services become popular as the rapid growth of smartphone users. However, malicious codes are also widespread, therefore security threats become serious problems. This paper suggests WLSA (White-List based Security Architecture) for the better mobile office security and presents required procedures and the analysis of the expected security enhancement.

Keywords: WLSA, Whitelist, BYOD, Smartphone Security.

1 Introduction

The smartphone users are increasing at a high speed. A lot of government bodies and private companies are trying to utilize personal smartphones as for a business purpose. Therefore BYOD (Bring Your Own Device) becomes a new business culture of smart-work [1]. The Mobile office supports real-time communication and fast decision making process by connecting mobile devices to internal MIS systems. However, malicious codes have increased with smart phones diffusion. In 2013, about 1 million sorts of malicious codes are expected in android applications [2]. Basically BYOD is exposed to security vulnerabilities, because it uses personal owned private devices which usually company cannot control. If some devices are infected with malicious codes, internal MIS system may be contagious as well. Google-Play [3] doesn't adopt pre-verification systems, security vulnerabilities may exist. Therefore only safe applications, which are proved by trusted organizations, have to be allowed to install in BYOD devices. This paper proposes WLSA (White-List based Security Architecture) which can enhance the overall security level. The trusted authority organizations offer their application lists which are verified and a government office or neutral organization makes and maintains a WL. The remainder of this paper is organized as follows: Chapter 2 surveys the related research works and activities regarding WL systems; Chapter 3 proposes the WLSA scheme and procedure; Chapter 4 presents the analysis and expected advantages; and, finally, the paper concludes with a summary and suggestions for future work.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 860–865, 2013.

2 Related Work

A WL is defined as the trusted and safe applications list for smartphones. The WL is made by trusted or authorized organizations or companies, such as telecom companies, major smartphone manufacturers, government bodies and etc. On the other hands, there is a black-list which can identify malicious codes by containing applications list. Usually vaccine programs utilize blacklist based systems. WL based methods of blocking malicious codes were suggested by analyzing security threats [4,5]. A WL DB is an essential information container for enhancing security strength and user's convenience. Furthermore with previous research works, more detailed architecture regarding building and managing the WL is required.

3 Proposed Architecture

In order to make and manage a WL, the participation of trusted organizations is necessary. A basic WL component diagram is suggested as Fig. 1. Trusted organizations, such as telecom companies or smartphone manufacturers, provide their own WL to the WL manager who operates main WL systems. The WL manager sends the list to the malicious verification system in order to recheck them. MDM (Mobile Device Management) systems utilize the WL for allowing or blocking application running on smartphones. Usually MDM agent software on mobile devices and MDM servers exchange WL information frequently in order to maintain update status. A MDM agent program checks the application on the smartphone when a user runs mobile office software. If MDM agent software finds applications which are absent from WL, then the MDM agent take over the root authority from OS and stops mobile office applications as well as sends the application information to the WL manager system in order to check the safety.

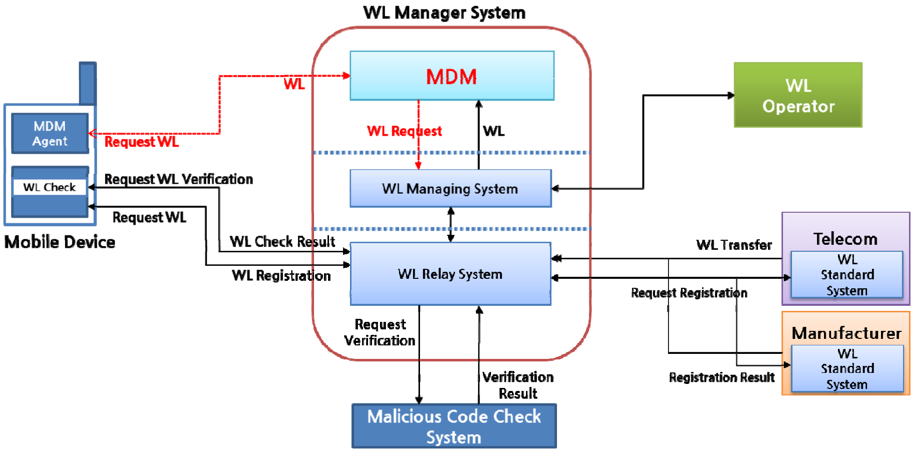


Fig. 1. WLSA Diagram

The MDM agent which is installed in BYOD devices can reduce the range of application download as well as utilization. Therefore WL manager has to expand WL-DB to the enough range of application usages. Usually app markets of trusted organizations verify applications before upload on markets, therefore these markets are safer than usual android markets which are most frequently used. Fig. 2. shows the information exchange procedures between WL providers and the manager. These procedures must be operated with very safe way

1. Trusted organizations make WL message information, such as application name, maker information, hash information and etc, by extracting data from mobile applications from their app markets.
2. Trusted organizations transfer the message information to the WL standard system.
3. WL standard system performs electronic signatures and encrypts WL message information.
4. WL standard system connects WL relay system and transfers the encrypted message.
5. WL relay system connects to the WL manager system.
6. WL relay system is ready to transfer the message to the WL manager system.
7. WL manager system decrypts the transferred message and verifies electronic signatures as well as hash data. WL manager system inserts received WL to the WL data base.
8. WL manager system notifies the result of the transaction to the WL relay system.
9. WL relay system registers the transaction history and closes the connection.
10. WL relay system transfers the transaction result to the WL standard system and finishes the procedures.

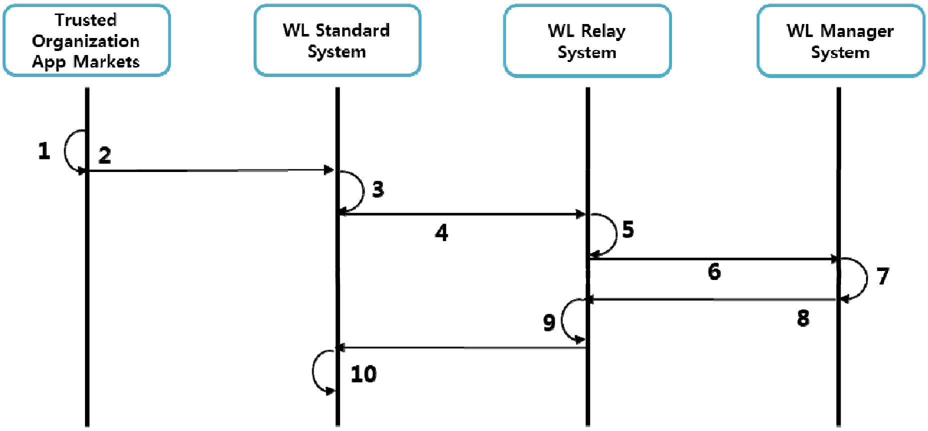


Fig. 2. WL Transaction Procedures

4 Analysis

Mobile Applications are categorized into three parts as Fig. 3, ① White-List ② Grey-List ③ Black-List. Applications in white-list are considered safe, whereas applications in black-list are unsafe. Applications in grey-list are not identified whether they are safe or unsafe. Therefore reducing the range of grey-list is very important for enhancing safety as well as the convenience of users.

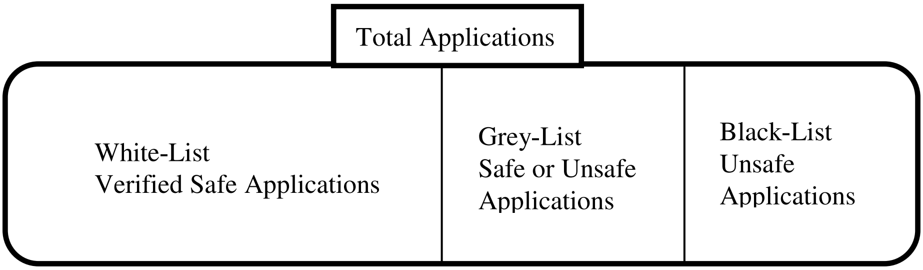


Fig. 3. Application Category According to Safety

The total applications of Fig. 3 can be presented as Eq. (1).

$$T_{APP} = N_{WL} + N_{GL} + N_{BL} \tag{1}$$

- T_{APP} *: Total number of mobile applications*
- N_{WL} *: Number of White-List applications*
- N_{GL} *: Number of Grey-List applications*
- N_{BL} *: Number of Black-List applications*

When a mobile user downloads applications from app-markets, the probability of infection by malicious codes is suggested as Eq. (2). First, if all applications in white-list are safe, then the probability of infection can be zero. Second, if applications in black-list are unsafe, then the probability of infection is 100%. Third, if applications in grey-list are not sure, then the probability of infection can be changed according to the app markets.

$$P_{MAL\_Down}=(N_{GL} * P_{MAL} + N_{BL}* 100\%) / T_{APP} \tag{2}$$

- P_{MAL_Down} *: Probability of downloading malicious apps*
- P_{MAL} *: Probability of infection in Grey-List apps*

Most applications in black-list can be blocked by mobile vaccine programs, therefore N_{BL}* can be omitted as Eq. (3).

$$P_{MAL\_Down}=(N_{GL} * P_{MAL}) / T_{APP} \tag{3}$$

We analyzed the probability of infection by download applications in grey-list. We assumed that the total applications (T_{APP}) in markets are 500,000 and the domain of grey-list (N_{GL}) may be 10%, 20% and 30% respectively. We supposed that the probability of malicious code infection (P_{MAL}) can be varied from 0.1% ~ 10%. Table. 1 is parameters and values for the analysis.

Table 1. Analysis Parameters and Values

Analysis Parameters	Values
T_{APP}	500,000
P_{MAL}	0.1% ~ 10%
N_{GL}	T_{APP} * 10% or 20% or 30%

The result of the analysis is presented as Fig. 4. and Table 2. In this analysis, we can have the range of possible infection from 5 to 15,000. It means that companies or organizations which use BYOD based mobile office services can prevent possible infection from malicious codes. As the domain of grey-list increases, the possibility of infection and inconvenience of users also become higher. Eventually it is highly desirable that all applications have to be categorized into white-list or black-list, but it is very hard because of the quantity and complexity of whole mobile application codes.

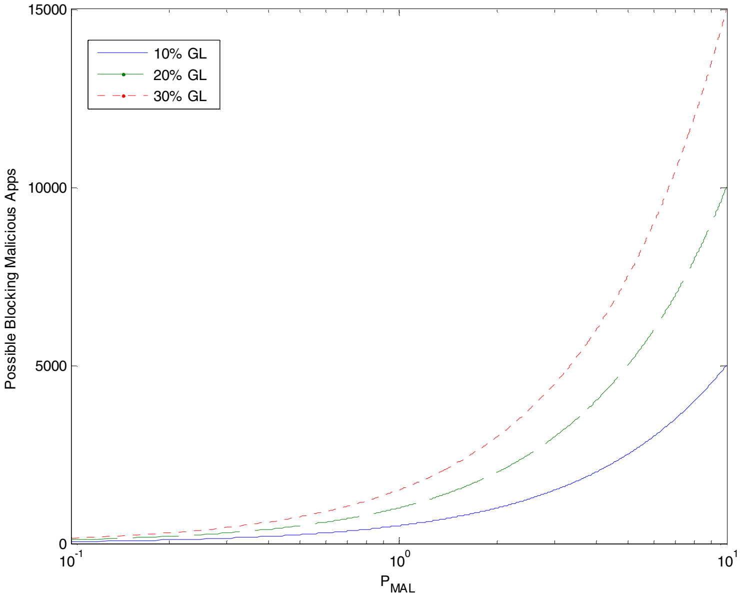


Fig. 4. Number of Possible Malicious Apps According to P_{MAL}

Table 2. Analysis Parameters and Values

T_{APP}	Possible Blocking Malicious Code by WL
T_{APP} 10%	5 (P_{MAL} 0.1%) ~ 5,000 (P_{MAL} 10%)
T_{APP} 20%	10 (P_{MAL} 0.1%) ~ 10,000 (P_{MAL} 10%)
T_{APP} 30%	15 (P_{MAL} 0.1%) ~ 15,000 (P_{MAL} 10%)

5 Conclusion

The security architecture is very important for the safe mobile office environment using BYOD. However, there are considerable malicious codes in application markets because of post verification systems. Internal MIS systems can be exposed to various malicious codes via BYOD mobile devices. In order to reduce the possibility of infection, an application download policy from only safe application list is required. In this paper, we proposed WLSA (White-List based Security Architecture) for BYOD users as well as presented the analysis of possible expectations by using WLSA. WLSA can be a very complicated and large scale system, therefore WLSA should be driven only by government or major companies. If a white-list covers only small domain of whole applications, then it will cause user's inconvenience. On the other hand, if a white-list covers most domain except black-list, it can be utilized by most government officers, company staffs as well as citizen for public services. As a result, white-list can enhance the safety level and reduce the anxiety of infection.

References

1. Miller, K.W., Voas, J., Hurlburt, G.F.: BYOD: Security and Privacy Considerations. IT Professional 14(5), 53–55 (2012)
2. Shankar, S.: As BYOD caches on, IT sector gets ready for 1M malicious apps (December 25, 2012), http://mydigitalfc.com
3. http://play.google.com
4. Lee, K., Tolentino, R.S., Park, G.-C., Kim, Y.-T.: A Study on Architecture of Malicious Code Blocking Scheme with White List in Smartphone Environment. In: Kim, T.-h., Chang, A.C.-C., Li, M., Rong, C., Patrikakis, C.Z., Ślęzak, D. (eds.) FGCN 2010. CCIS, vol. 119, pp. 155–163. Springer, Heidelberg (2010)
5. Stueckle, J.D.: Android Protectoin System: A Signed Code Security Mechanism for Smartphone Applications, Air force institute of technology, Thesis (March 2011)
6. Kim, K.Y., Kang, D.H.: Smart Phone Security Technology in Opened Mobile Environment. Korea Institute of Information Security & Cryptology 19(5) (2009)

A Study of Vessel Deviation Prevention Scheme Using a Triangulation in a Seaway

Shu Chen[1], Rashid Ahmad[1], Byung-Gil Lee[2], Byung-Doo Kim[2], and Do-Hyeun Kim[1]

[1] Dept. of Computer Engineering, Jeju National University, Jeju, Republic of Korea
wlstj870211@naver.com, Rashid141@gmail.com, kimdh@jejunu.ac.kr
[2] Eletronics & Telecommunication Research Institute, Republic of Korea
{bdkim,bglee}@etri.re.kr

Abstract. Recently, the marine accidents happened due to the ship deviation from the established route or seaway. Due to the sensitivity of the problem, the Automatic Identification System (AIS) has been studied to improve the safety in seaway traffic. In this paper, we propose the seaway deviation prevention scheme that makes use of the distance of ship from a seaway based on a triangulation method for preventing ship collision. This scheme devises a control strategy for ships which keeps the ship in the route range and prevents the ship deviation from the normal sea route with a route range. This scheme could be used to prevent marine accidents and increase the ship's safety degree.

Keywords: Automatic Identification System, Vessel Traffic System.

1 Introduction

Marine accidents have been increased gradually in the past few years. The International Maritime Organization cites human error as the casual fact in 80% of ship accidents. It is necessary to support preventive methods because it is a potential risk to economy and human life. VTS (Vessel Traffic System) is provided to prevent serious ship accidents. VTS has been defined to manage and supervise ship displacements and states. Typical VTS systems use radar, closed-circuit television (CCTV), VHF radiotelephony and AIS (Automatic Identification System) to keep track of ship movements and provide navigational safety in a limited geographical area. The AIS supports processing the receiving related information of ship, including position, heading, speed that sending by sailing ship automatically, and sending the information to other ship in order to make sure of the marine traffic safety of the coastal sea. VTS also supporting the ship traffic service in the harbor and distressed ship search and relief operation effective. One of the most important functions of VTS is to alert and prevent deviation from route among ships using ship's course deviation indicator.

In this paper, we propose a ship deviation prevention scheme based on distance from seaway by using a triangulation in a seaway to support safety navigation of ship in the VTS system. In the first step, a basis of distance from a seaway is calculated from triangulation. Next, the alerts of deviation from route are planned. In the third step, the ship is controlled effectively to return to the seaway safely.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 866–872, 2013.

The rest of this paper is structured as follows. In section 2, we discuss in detail the proposed prevention procedure of course deviation in a seaway. In section 3, we describe our proposed ship deviation distance and show our calculation method. Finally we conclude in section 4.

2 Proposed Prevention Procedure of Course Deviation in a Seaway

The route on which a ship sails from one port to another is called a seaway. The coordinates of the port are passed point through ship via node. There are more than two ship via node exists from one port to another port. The ship via node can be added between any two ports. The seaway set by connecting the ship via nodes using straight line segments. Between the two ship via node coordinates is ship via range and ship via range can be calculated according to the coordinate of the ship. The seaway is shown in Fig. 1 that adds ship via point between departure port and arrival port.

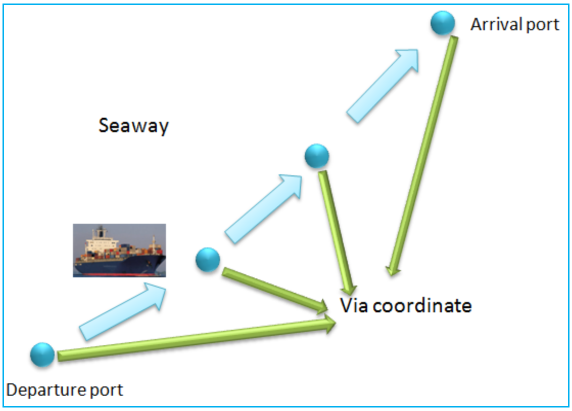


Fig. 1. Conceptual diagram of seaway

Users can set the ship via node between any two ports. Departure port will be the first ship via node coordinate and arrival port will be the last coordinate. The position of the ship will update in real time, and use straight line to connect the ship and the already set ship via node coordinates. We will compare all the straight line's value and select the shortest straight line and second shortest straight line.

For calculating triangle between the ship coordinate and ship via node coordinates. The proposed procedure is divided into five different functions that are 1) seaway creation function; 2) first/second warning range set function, 3) ship's deviation judgment function, 4) deviation warning function, and 5) ship control function. Table 1, shows the explanation of these functions. The seaway is created by connecting all ship via nodes with the departure and arrival ports. The First/Second

Warning range is set in order to set the desired safety seaway range for deviation prevention. The ship position is used to make the triangle between the ship and the nearby nodes (reference points). We use the triangle height as a distance between the ship and the seaway range. The ship's deviation status is then checked against the triangle height. If the ship deviation from the route, the warning function sends a warning message to user in real time. Ship control function has four buttons left, right, up and down to control the ship move on the map.

Table 1. The functions of ship deviation prevention

Step	Function	Description
1	Ship via node coordinates creation	Select and create any ship via node coordinate on the start-up screen
2	Seaway creation	Connect already set ship via node coordinates to create a seaway
3	Set first/second warning range	User input the value of first and second warning range
4	Deviation judgment	Calculate the height of the triangle and check the ship is deviation or not
5	Deviation warning	Send the current navigation information of the ship
6	Ship control	Control ship's navigation

Fig. 2 shows the Sequence diagram that is the process of ship via range set. The ship will obtain the position, navigation direction, and speed information from the GPS device. The latitude of the position is X, and the longitude of the position is Y. The ship will get this information from GPS periodically and update it.

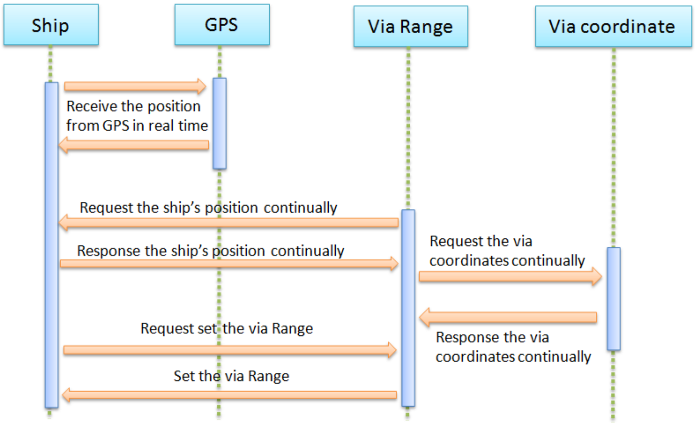

Ship
GPS
Via Range
Via coordinate
Receive the position from GPS in real time
Request the ship's position continually
Response the ship's position continually
Request the via coordinates continually
Response the via coordinates continually
Request set the via Range
Set the via Range

Fig. 2. The sequence diagram of ship via range set in a seaway

The ship via range will be set based on the position of ship received from GPS. User will set ship via node coordinate and set the departure port. In the process of deviation detection, the ship will get the current position from GPS in real time and will find the first straight line of ship via node coordinate and second straight line of ship via node coordinate. After make a triangle through use the ship and two straight lines ship via node coordinates the height of the triangle will be calculated. The height of triangle will be compared with the first and second warning range and the ship deviation status will be checked.

The first/second warning range of ship's deviation detection flow chart is shown in Fig. 3. Firstly users will set the departure port, arrival port, ship via node coordinates and the range of the seaway in the start-up screen and use straight line to connect them for make a seaway. Secondly, a triangle will be formed according to the position of ship and any two adjacent ship via node. The height of the triangle will be calculated and compare with the already set first/second warning range and check the ship deviation or not according to the comparing results. This process will continue until the ship arrives at the arrival port.

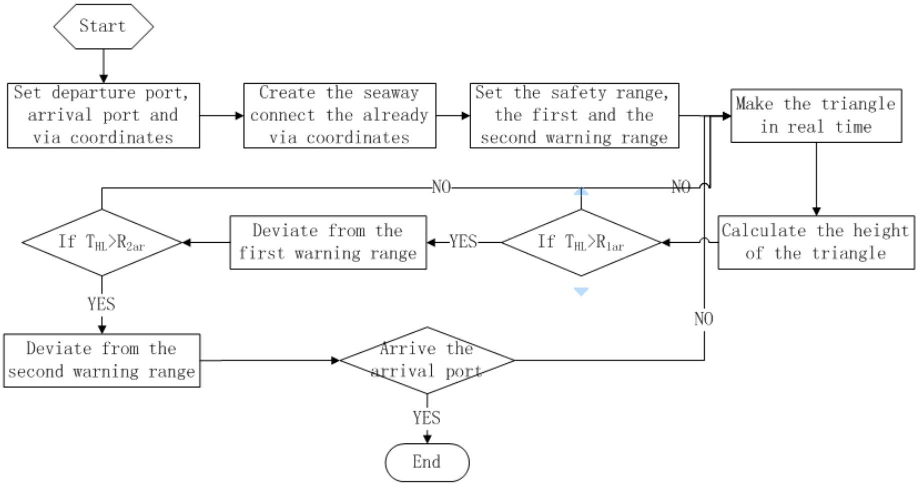

Start
Set departure port, arrival port and via coordinates
Create the seaway connect the already via coordinates
Set the safety range, the first and the second warning range
Make the triangle in real time
Calculate the height of the triangle
If T_HL>R_1ar
YES
NO
Deviate from the first warning range
If T_HL>R_2ar
NO
YES
Deviate from the second warning range
Arrive the arrival port
NO
YES
End

Fig. 3. The flow chart of proposed prevention procedure of course deviation

3 Calculation Method of Course Deviation Distance in a Seaway

The Euclidean distance between all the ship via nodes and ship is calculated using equation 1 and equation 2. The triangle is made between the ship coordinate and the two nearest via nodes.

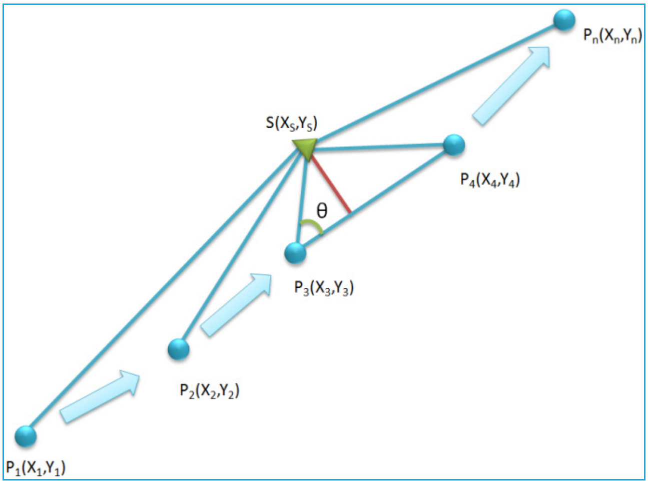

P_n(X_n,Y_n)
S(X_S,Y_S)
P₄(X₄,Y₄)
θ
P₃(X₃,Y₃)
P₂(X₂,Y₂)
P₁(X₁,Y₁)

Fig. 4. The coordinate of triangulation

Fig. 4 shows the coordinates of ship are $S(X_S, Y_S)$, and the two ship via node coordinates are $P_3(X_3, Y_3)$, $P_4(X_4, Y_4)$.We connect the three coordinates each other and make a triangle.

$$SP_3 = \sqrt{(X_S - X_3)^2 + (Y_S - Y_3)^2} \tag{1}$$

$$SP_4 = \sqrt{(X_S - X_4)^2 + (Y_S - Y_4)^2} \tag{2}$$

$$P_3P_4 = \sqrt{(X_3 - X_4)^2 + (Y_3 - Y_4)^2} \tag{3}$$

$$\cos\theta = \frac{SP_3^2 + P_3P_4 - SP_4^2}{2SP_3 \cdot P_3P_4} \tag{4}$$

$$\sin\theta = \sqrt{1 - \cos^2\theta} \tag{5}$$

$$d = SP_3 \cdot \sin\theta \tag{6}$$

Here, SP_3 is the distance that is connected by the thirdly ship via node coordinate P_3 and ships coordinates. SP_4 is the distance connects by the fourth ship via node coordinate P_4 and ship's coordinate. P_3P_4 is the distance that connect by the thirdly ship via node coordinate and the fourth ship via node coordinate. θ is the degree between straight line SP_3 and P_3P_4. After the triangle making, equation 6 is used to calculate its height. The height of triangle is used to find the ship's position whether it is in the specified seaway range or outside.

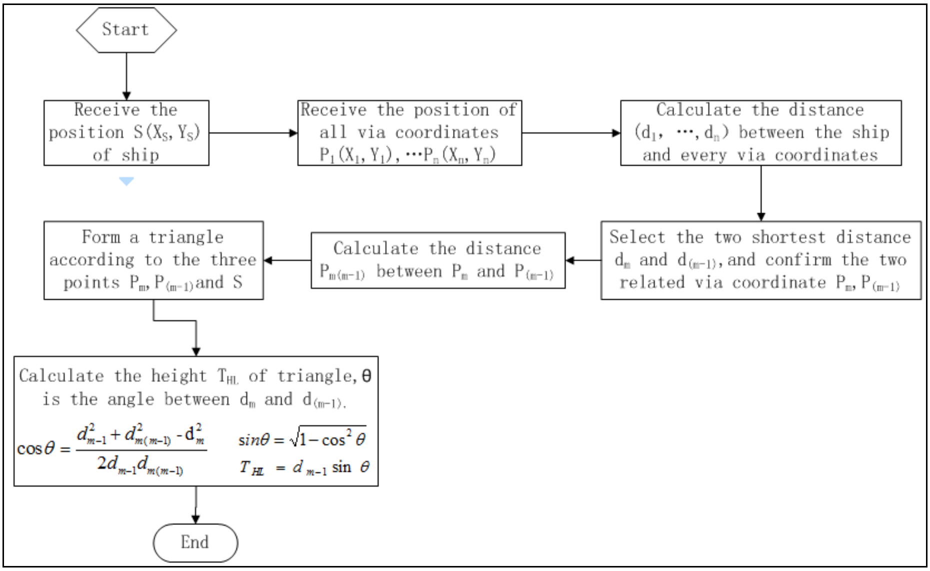

Start
Receive the position S(X_S, Y_S) of ship
Receive the position of all via coordinates P_1(X_1, Y_1), …P_n(X_n, Y_n)
Calculate the distance (d_1, …, d_n) between the ship and every via coordinates
Select the two shortest distance d_m and d_(m-1), and confirm the two related via coordinate P_m, P_(m-1)
Calculate the distance P_m(m-1) between P_m and P_(m-1)
Form a triangle according to the three points P_m, P_(m-1) and S
Calculate the height T_HL of triangle, θ is the angle between d_m and d_(m-1).
cos θ = (d²_m-1 + d²_m(m-1) - d²_m) / (2d_m-1 d_m(m-1))
sinθ = √(1 − cos² θ)
T_HL = d_m-1 sin θ
End

Fig. 5. The flow chart of calculate the height of triangle

Fig. 5 shows the process of finding ships navigation status using triangulation method. The triangle is made between the ship's coordinates and the nearest two ship via node coordinates. Equations 1 to 6 are used to calculate the distance between the ship's coordinates and all the ship via node coordinates. The ship via nodes having the shortest distance will be used to calculate the triangle. The height of this triangle will be the distance of ship from the center of normal seaway route. Fig. 6 elaborates the triangle making process in detail.

The warning messages are dependent on the ship's position. Suppose the height of triangle between ship via nodes and ship is T_{HL}, the T_{HL} will be compared with the specified seaway range, if the T_{HL} is greater than the specified range, a warning message will be generated and displayed the warning message on the message window show that the ship already deviated from the first warning range. Depend on this condition, compare the T_{HL} and already set second warning range (R_{2ar}), it will be calculated that the ship is deviated from the second warning range or not. If the T_{HL} is greater than the second warning range, a warning message will be generated and displayed on the message window showing that the ship is already deviated from the second warning range. Here, T_{HL} is the height of the triangle, R_{1ar} is the first warning range and R_{2ar} is the second warning range.

4 Conclusions

In this paper, we propose a ship deviation prevention scheme based on distance from seaway using triangulation method. This scheme supports safety navigation of ship in the VTS system. Firstly, we present a calculation method of course deviation distance by using triangulation. And, we present alerting method for course deviation prevention. According to proposed methods, it will send the information of ship deviation to user in advance, and prevent the ship break away from the seaway range.

Acknowledgments. This work was supported by ETRI through Maritime Safety & Maritime Traffic Management R&D Program of the MLTM/KIMST (D10902411H360000110 ; Development of u-VTS for Maritime Safety). Corresponding author ; DoHyeun Kim (e-mail: kimdh@jejunu.ac.kr).

References

1. Journée, J.M.J.: Prediction of Speed and Behaviour of a Ship in a Seaway. Technology report, Delft University of Technology ISP, vol. 23(265) (1976)
2. Barrett, S., Ashton, I., Lewis, T., Smith, G.: Spatial & Spectral Variation of Seaways. In: Proceedings of the 8th European Wave and Tidal Energy Conference, Uppsala, Sweden (2009)
3. Ying, S.: Ship Route Designing for Collision Avoidance Based on Bayesian Genetic Algorithm. In: IEEE International Conference on Control and Automation (2007)

Analysis of Energy Consumption in Edge Router with Sleep Mode for Green OBS Networks

Wonhyuk Yang[1], Mohamed A. Ahmed[1], Ki-Beom Lee[1], and Young-Chon Kim[2,*]

[1] Department of Computer Engineering, Chonbuk National University, Jeonju, Korea
{whyang,mohamed,aresys}@jbnu.ac.kr
[2] Department of IT, Chonbuk National University, Jeonju, Korea
yckim@jbnu.ac.kr

Abstract. In this paper, we analyze the energy consumption of edge router with sleep mode in OBS networks. The edge router with sleep mode consists of multiple line cards, multiple OBS line cards, a SCU(Switch Control Unit) and an electronic switch fabric. The OBS line card, which is the main part of edge router, performs the functions of edge router such as BCP(Burst Control Packet) and burst assembly, BCP scheduling and sleep mode. In OBS line card, it is possible to reduce energy consumption by controlling PHY/Transceiver module from active state to sleep state for burst assembling by using sleep mode. In order to evaluate the energy saving performance of the OBS edge router with sleep mode, the power consumption is analyzed according to the datasheet of packet router and optical device. And, simulation by using OPNET is also performed in terms of sleep time and average queuing delay.

Keywords: OBS, Green IT, Router, Sleep mode.

1 Introduction

The Green IT has been continuously studied for reducing energy consumption and preserving environment of world in IT field. The number of Internet users and the demands on useful bandwidth has been increased continuously. And, the power usage for operation in network equipment is increased due to increasing performance of network equipments. Indeed, energy consumption of Internet and network equipments are estimated about 74TWh per year which means that it can be converted to the cost $6 billion in USA [1]. And, network devices such as NICs, router and switch consume about 5.3TWh in United States [2]. For this reason, many researchers in network field have tried to develop energy saving scheme for reducing energy consumption. As a result, IEEE Std 802.3az was approved on September 30, 2010. In this standard, sleep mode is used for reducing the energy consumption of network equipment such as NICs, routers, switches, hubs and etc [3].

The sleep mode is operated as ACTIVE and SLEEP mode to control energy consumption. ACTIVE mode consumes same energy to traditional network equipment to

[*] Correspondingauthor.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 873–879, 2013.

transmit the data when there is data to be processed in the network equipment (BUSY state). On the other hands, if there is no data to be processed in the network equipment (IDLE state), Sleep mode that consumes low power in some modules of network equipment is activated in network equipment. When the network equipment is activated, energy consumption of it can be about 10 % that of ACTIVE mode, generally.

Most studies of sleep mode are focused on LAN or access network like Ethernet or PON(Passive Optical Network). However, it should apply to backbone network in order to construct future energy saving network. An OBS [4, 5], which has the advantage of OCS and OPC, is one of promising backbone technology. In OBS, we can have many chances to save the energy due to characteristic of OBS. Data bursts and Burst Control Packets(BCP) in OBS network are separately transmitted into destinations. Therefore, we can save the energy consumed to transmitters during assembling the bursts. And, before the bursts arrive at core router, it receives BCP to schedule and control the switch. For this reason, OBS core router can change the own state from active to sleep mode.

In this paper, we analyze energy consumption of edge router with sleep mode in OBS networks. The edge router with sleep mode is comprised of line card, SCU, electronic switch and OBS line card. The first one transmits packets from access to core network or vice versa. The second controls electronic switch fabric, and the third one connects input port to output port. The OBS line card, which is main part of edge router, performs main function of edge router such as BCP(Burst Control Packet) and burst assembly, BCP scheduling and sleep mode. In OBS line card, as a control information processing engine performs the sleep mode, it is possible to reduce energy consumption by controlling PHY/Transceiver from active to sleep mode for assembling the burst. To evaluate energy saving performance of the OBS edge router, the power consumption is analyzed by using datasheets of realistic packet router and optical devices. And, simulation is performed in terms of sleep time and average queuing delay.

2 OBS Edge Router Architecture with Sleep Mode

In this section, we introduce the OBS edge router with sleep mode to reduce energy consumption in OBS networks. To do this, we modify the OBS edge router architecture.

Fig. 1 show block diagram of OBS edge router with sleep mode. It consists of four parts: line card OBS line card, electronic switch and SCU. A line card in OBS edge router transmits and receives packet from access to core or vice versa. A line card is same architecture to the one in packet router. The role of Electronic switch and SCU(Switch Control Unit) are to connect line card with OBS line card, appropriately. To do this, the SCU is comprised of switch controller and routing engine and it controls the switch fabric according to packet information from access network and BCP and routing information from core network. The OBS line card is a main device of edge router. It performs burst assemble and disassemble to transmit and receive the data. Inside of OBS line card is shown in fig. 2.

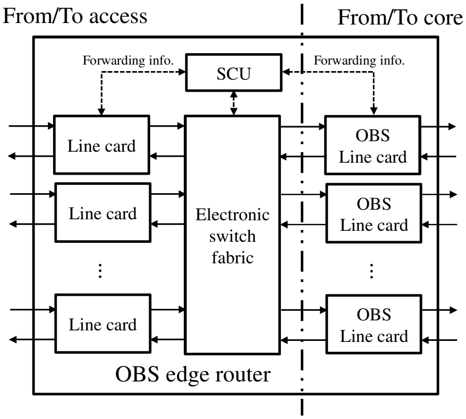


Fig. 1. Block diagram of OBS edge router with sleep mode

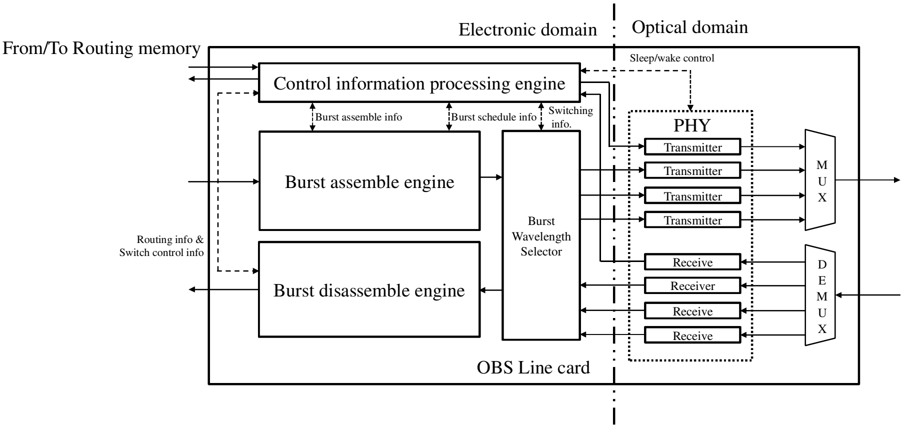


Fig. 2. Block diagram of OBS line card and modules

OBS line card consist of control information processing engine, burst assemble and disassemble engine, burst wavelength selector and PHY/Transceiver. Control information processing engine has responsibility for generating BCP and processing routing information packet and received BCP. The burst assemble engine creates and transmits the burst to core router when burst assemble memory satisfies conditions for generating burst such as time and length threshold. When edge router receives the bursts through PHY/Transceiver, Burst disassembler divides it into origin packets to transmit to its own destination. Burst wavelength selector chooses a proper wavelength for burst transmission and receives the burst from a specific wavelength. The detail module of each block is shown in Table 1.

To support sleep mode mechanism in OBS network, the edge router has sleep/wake controller in control information processing unit. This controller change state of PHY/Transceiver from active to sleep or from sleep to active while the burst assemble processing is not finished. Through this function, we can reduce energy consumption in OBS edge router. Moreover, since the edge router can accommodate high capacity over at least 40Gbps and be connected with multiple core routers, one or more OBS line card can be installed in an OBS edge router. In this case, if offered load is very low at an OBS edge router, it is possible to change some line card state into sleep sate to save energy.

Table 1. Component and modules for OBS edge router with sleep mode

Block	Sub-block	Module	Functions
OBS line card	Control information processing unit	BCP processor BCP memory BCP scheduler Sleep/Wake controller Forwarding engine	BCP creation, BCP processing and scheduling, Routing information control, Sleep and wake control for PHY/Transceiver
	Burst assemble unit	Burst assemble memory, burst assembler burst tx memory	Burst creation
	Burst disassemble unit	Burst disassembler packet memory traffic processor	Burst disassembling
	Burst wavelength selector		Selecting proper wavelength
	PHY/Transceiver		Transmitting and receiving burst

3 Performance Evaluation

3.1 Comparing Power Consumption of Routers

In order to evaluate energy saving performance, the edge router with sleep mode is analyzed and compared with packet router in terms of power consumption. Power consumption of packet router refers to data sheet of Cisco CRS-1 multi-shelf system [8] and several router systems [9] while the edge router with sleep mode use power consumption of packet router, partially, and some optical devices refer to data sheet of conventional product[10] and [11, 12].

Table 2 shows power consumption of each router. Routing engine(38.5%) and forwarding engine(28.1%) consume the most power in each router. The routing processor sets up many routes to interconnect ports or line cards, and manages the routing table for route decision to transfer the packets or bursts, continuously. The forwarding processor looks up the routing table for finding the matched output port. In this progress, since it searches many data in the routing table to find proper output port, processor consumes a lot of power.

Compared to packet router, The OBS edge router consumes more power than it because of the additional modules such as burst assemble and disassemble unit. Thus, the OBS edge router consumes power about 1.3 times. However, when it transits the state from active to sleep, we can reduce power consumption. As a result, we expect that the power consumption in OBS networks can be reduced by using the edge router with sleep mode.

Table 2. Power consumption comparison of packet router and OBS edge router

Components types \ Router			Packet router	OBS edge router with sleep mode
Block	Sub-block	Module name	Power(%)	Power(%)
OBS line card	Control information processing unit	Packet(BCP) processor	3.3	3.2
		Packet(BCP) memory	1.5	1.5
		Packet reconstructor (BCP schduler)	3.7	4.3
		Sleep/wake controller	0.3	0.4
		switch controller	-	0.4
		Forwarding engine	28.1	28.7
	burst assemble unit	burst assemble memory	-	1.5
		burst assembler	-	3.2
		burst tx memory	-	1.5
	burst disassemble unit	traffic processor	-	3.2
		packet memory	-	1.5
		burst disassembler	-	3.2
	burst wavelength select switch		-	12.8
	O/E		-	0.2
	E/O		-	0.2
	PHY/Transceiver(40Gbps)		1	1.1
SCU	Routing engine		38.5	39.4
	switch controller		0.3	0.4
Electornic switch(320Gbps)			23.3	23.9
Total			100	130.6

3.2 Simulation Results

To evaluate energy saving performance of edge router with sleep mode in OBS networks, we perform simulation in terms of sleep time and average queuing delay by using OPNET modeler. In this simulation, it is possible to sleep to PHY/Transceiver while burst assemble is executed in burst assemble unit. Simulation parameter used in this simulation is shown in Table 3.

Figure 3 shows total sleep time and average queuing delay of OBS edge router for simulation time. In all period of the sleep time graph, the edge router can have chance of energy saving. Especially, at 0.1, the PHY/Transceiver can sleep for 180sec. Moreover, length based burst assemble algorithm show more good performance than time based burst assemble algorithm at low utilization. It is because the time of burst creation in length based algorithm is longer than time based algorithm. In average queuing delay, length based algorithm has little higher queuing delay. This is also same reason with the result of sleep time. Although time based algorithm show lower delay, length based algorithm can keep QoS boundary. This means that sleep mode can not only reduce energy consumption, but also can guarantee QoS for OBS network.

Table 3. Simulation parameters

Parameters	Value
Link capacity	40Gbps
Active to sleep time	None
Sleep to active time	2us
Traffic pattern	Exponential distribution
Simulation time	200

Burst assemble scheme	Time(Th=1ms)
	Length(Th=3Mbit)

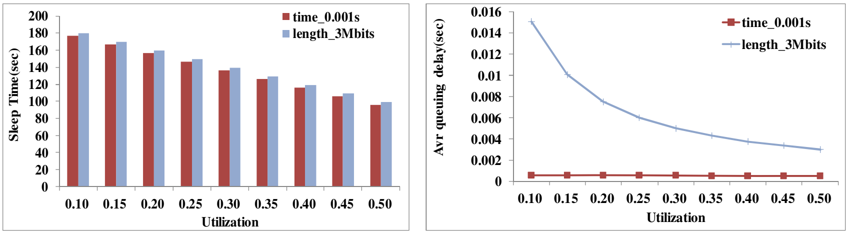


Fig. 3. PHY/Transceiver sleep time and average queuing delay of OBS edge router

4 Conclusions

In this paper, we analyzed energy consumption of edge router with sleep mode in OBS networks. The edge router with sleep mode is comprised of multiple line cards, an electronic switch, a SCU and multiple OBS line cards. The OBS line card performed main OBS function and sleep mode. The sleep/wake controller at control information processing engine in an OBS line card changed the state of PHY/Transceiver from active to sleep or vice versa for assembling the bursts. Moreover, if the proper algorithm or mechanism is executed in edge router, an OBS line card could have the chance to sleep at low load to save the energy.

To evaluate the energy saving performance of the OBS edge router, it was compared with packet router in terms of power consumption. And, the simulation was performed in terms of sleep time and average queuing delay. The edge router with sleep mode consumed more energy about 1.3 times than packet router. However, Simulation results showed that PHY/Transceiver module could sleep about 90% for simulation time at low utilization. Therefore, although the edge router with sleep mode consumed 1.3 times energy in normal state, it could reduce energy consumption by using sleep mode in the PHY/Transceiver. Moreover, the edge router with sleep mode could guarantee QoS performance such as average queuing delay.

Acknowledgment. This research was supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No.2010-0013822). And it was also supported by the second stage of Brain Korea 21 and the Human Resource Training Project for Regional Innovation funded by the MEST and Korea Industrial Technology Foundation(KOTEF).

References

1. Kawamoto, K., Koomey, J., Nordman, B., Brown, R., Piette, M., Ting, M., Meier, A.: Electricity used by office equipment and network equipment in the US: Detailed report and appendices, Technical Report LBNL-45917, Lawrence Berkeley National Laboratory (2001)
2. Nordman, B., Christensen, K.: Reducing the Energy Consumption of Network Devices. Tutorial Presented at the July 2005 IEEE 802 LAN/MAN Standards Committee Plenary Session (2005)
3. Reviriego, P., Hernandez, J.A., Larrabeit, D., Maestro, J.A.: Performance Evaluation of Energy Efficient Ethernet. IEEE Communications Letters 13(9), 697–699 (2009)
4. Qiao, C., Yoo, M.: Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet. Journal of High Speed Networks 8(1), 69–84 (1999)
5. Chen, Y., Qiao, C., Yu, X.: Optical Burst Switching (OBS): A New Area in Optical Networking Research. IEEE Networks 18(3), 16–23 (2004)
6. Aweya, J.: On the design of IP routers Part1: Router architecture. Journal of Systems Architecture 46(6), 483–511 (2000)
7. CISCO: Cisco Router Architecutre, `http://www.cisco.com/networkers/nw99_pres/601.pdf`
8. CISCO: CRS-1 Mult-self system description, `http://www.cisco.com/`
9. CISCO: CISCO Catalyst series datasheet, `http://www.cisco.com/`
10. CISCO: Cisco DWDM SFP Module, `http://www.cisco.com/`
11. Alecksić, S.: Analysis of Power Consumption in Future High-Capacity Network Nodes. Journal of Optical Communications and Networking 1(3), 245–258 (2009)
12. Yamada, M., Yazaki, T., Matsuyama, N., Hayashi, T.: Power Efficient Approach and Performance Control for Routers. In: IEEE International Conference on Communications Workshops 2009, pp. 1–5 (2009)

VLC Based Multi-hop Audio Data Transmission System

Le The Dung[1], Seungwan Jo[2], and Beongku An[2]

[1] Dept. of Electronics & Computer Engineering in Graduate School, Hongik University, Korea
thedung_hcmut@yahoo.com
[2] Dept. of Computer & Information Communications Engineering, Hongik University, Korea
wh7923@gmail.com, beongku@hongik.ac.kr

Abstract. In this paper, we propose a multi-hop transmission system using visible light communication to transmit audio data. In our proposed transmission system, at the transmitter we encode audio data based on S/PDIF standard – a popular standard for digital audio signal, and transmit the encoded audio signal via general LED. At each relay, digital audio signal is improved and amplified before sending. At the receiver, encoded audio signal from photodiode (PD) is decoded, amplified and coverted to analog audio signal. We evaluate our proposed transmission system in a room with flourescent light source. The audio signals obversed at the receiver show that with the support of relays, our proposed transmission system can provide high quality audio transmission from transmiter to receiver via multi-hop relays at a long distance.

Keywords: VLC based system, audio transmission, S/PDIF digital audio signal, multi-hop communication.

1 Introduction

Nowadays, optical communication has been widely used around us in various applications due to its advantages compared with conventional radio frequency communication (i.e. high speed, harmlessness to health). Optical communication can be classified into two categories: wired optical communication (or guided optical communication), in which fiber optic cable is used as communicating channel and wireless optical communication, in which free space is used as communicating channel.

Visible Light Communication (VLC) based system belongs to wireless optical communication. The development of LED technology brings new opportunities for energy savings and reduces maintenance cost in illumination system. Unlike incandescent and fluorescent light bulbs, LEDs have significantly low thermal inertia. Moreover, LEDs are able to switch between on and off state at high frequency. This characteristic can be explored to send data at high speed using visible light waves.

In [1], the authors develop a prototype for sending data between two devices. However, that prototype only provides moderate data rate in one hop communication. The authors in [2] demonstrate a visible communication link for audio transmission. In that system, transmitter sends audio data to receiver directly in a short distance. Also, digital audio signals in [2] need to be improved. In our previous work [3], we proposed a VLC based multi-hop transmission system for sending text data. To the

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 880–885, 2013.

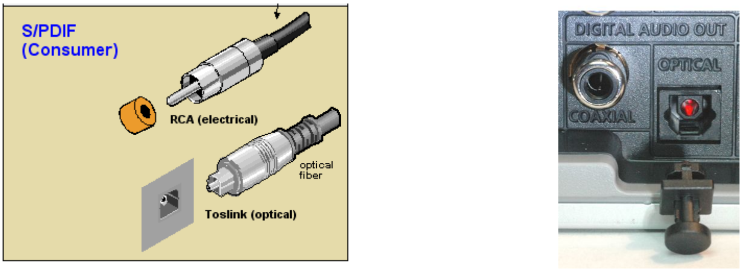
best of our knowledge, there hasn't been any implementation and performance evaluation of multi-hop audio data transmission system in practice. In this paper, we propose a multi-hop system to transmit high quality audio data at high speed from transmitter to receiver via several relays.

The rest of this paper is organized as follows. Section 2 explains in detail signal format and the functions of all modules used in our system. Section 3 presents experimental setup and results achieved with implemented system prototype. Finally, section 4 concludes the paper.

2 Our Proposed Multi-hop Audio Data Transmission System

2.1 Signal Format

The format of digital audio signal used in our transmission system is S/PDIF. This name stands for Sony/Philips Digital Interconnect Format. SPIDF is a data link layer protocol and physical layer protocol for carrying digital audio signal between various devices. S/PDIF is standardized by International Electrotechnical Commission (IEC) [4] in IEC 60958 as IEC 60958 type II (IEC 958 before 1998). SPDIF has several small differences with AES/EBU [5] and can be considered as minor update of the original AES/EBU. Digital audio signal in S/PDIF format is transmitted over either a coaxial cable with RCA connectors or a fiber optical cable with TOSLINK connector as in Figure 1(a). Nowadays, many modern devices have digital audio output ports which use S/PDIF format as in Figure 1(b).



(a) (b)

Fig. 1. (a) Types of cables used in S/PDIF digital audio signal transmission, (b) Digital audio output port of a device

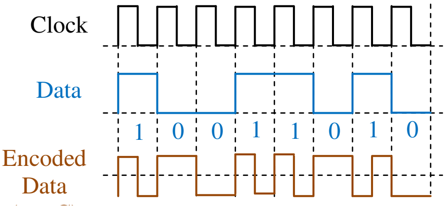


Fig. 2. Bi-phase Mark Coding (BMC) scheme in S/PDIF

In S/PDIF, original digital data stream is encoded using Differential Manchester encoding, also called Bi-phase Mark Code (BMC) or FM1 as in Figure 2. According to this code, in one data period, two zero crossings of the signal mean a logical 1 and one zero crossing means a logic 0. Therefore, the frequency of the clock is twice the bit rate of original data.

The advantages of BMC code are as follows:

- Signal transition happens at least once every bit, allowing receiving device to perform clock recovery and synchronization.
- To be less error-prone in noisy environment compared with other codes.

Due to the above advantages, in our system, we will use BMC code for transmitting digital audio signal through visible light communication channel.

2.2 System Descriptions

Our proposed multi-hop audio data transmission system is illustrated in Figure 3. In this system, audio data source is fed to transmitter module through USB port. The original digital audio signal is encoded to S/PDIF format, shifted, amplified, and sent to LED. At each relay, digital audio signal is corrected and amplified before sending to next relay. At the receiver, the encoded audio signal received at photodiode (PD) is amplified, removed shifted DC component, decoded to original digital audio signal, and converted to analog audio signal. Finally, analog audio signal is sent to speakers.

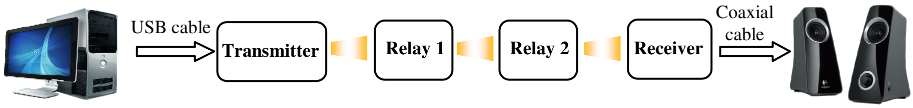


Fig. 3. Our proposed multi-hop audio data transmission system

We will describe in detail transmitter module and receiver module by showing their block diagrams and explaining the function of each block.

A. Transmitter Module

The main functions of transmitter module are to get audio signal (e.g. analog signal or digital signal) from a device, convert to S/PDIF audio digital signal and transmit to next hop. Figure 4 shows the block diagram and real design of transmitter module.

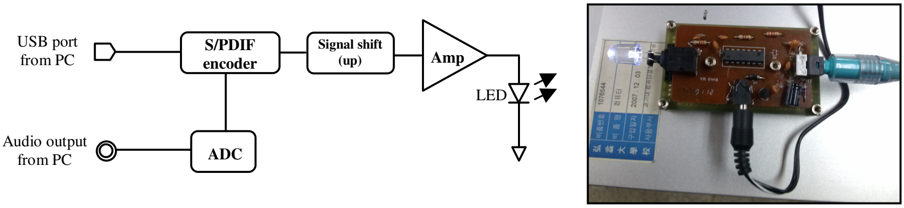


Fig. 4. Transmitter module in our proposed VLC based multi-hop audio data transmission system

If the audio signal from a device is analog one, then it will go through ADC block to be converted to digital signal before encoding to S/PDIF digital signal. Since S/PDIF signal is a bipolar signal and LED will clip this bipolar signal, S/PDIF digital audio signal should be shifted up by adding appropriate DC component before sending to LED. Then the signal is amplified to drive LED.

B. Relay Module

The main function of relay module are to improve the quality of digital audio signal due to interference from external light source and to compensate the power loss of signal due to long distance transmission via visible light. The block diagram and real design of relay module are shown in Figure 5.

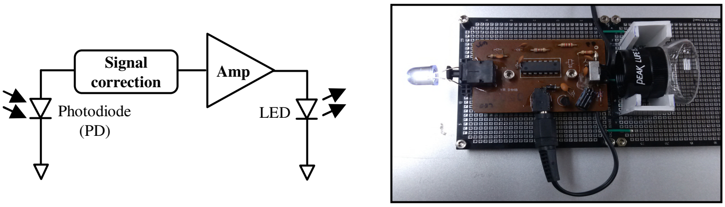


Fig. 5. Relay module in our proposed VLC based multi-hop audio data transmission system

Signal correction block improves the quality of S/PDIF square signal received from photodiode. Then the improved S/PDIF signal is amplified before sending to LED to transmit to next hop.

C. Receiver Module

The main function of receiver module is to convert received digital audio signal to analog audio signal. Figure 6 shows the block diagram and real design of receiver module.

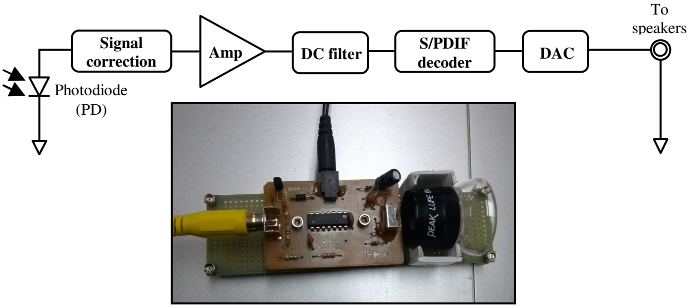


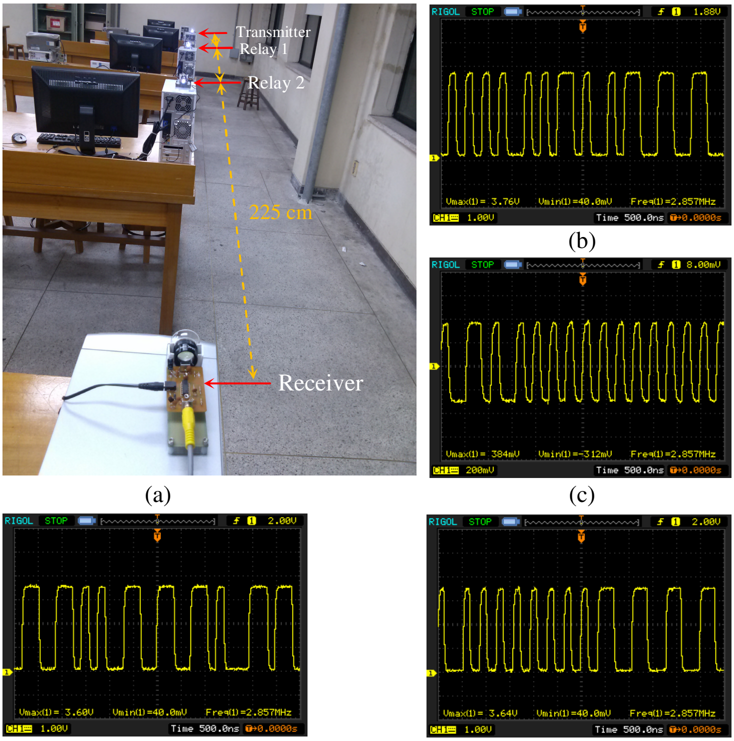
Fig. 6. Receiver module in our proposed VLC based multi-hop audio data transmission system

First two blocks in receiver module have the same function as those in relay module. Since the digital audio signal was shifted up at transmitter module by adding DC component to avoid signal clipping, at receiver module the DC part of this digital signal should be filtered to obtain S/PDIF digital signal. Then S/PDIF signal is sent to DAC block to reconstruct analog audio signal.

3 Performance Evaluation

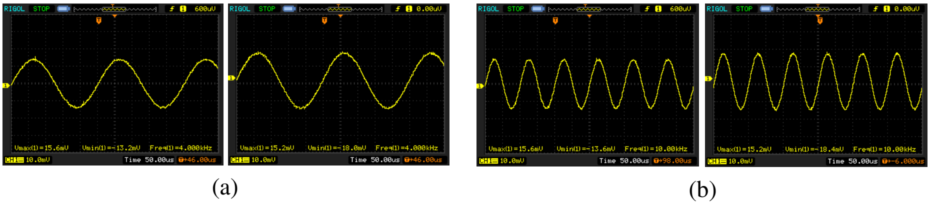
We evaluate the performance of our proposed VLC based multi-hop audio transmission system in a laboratory in the present of normal room illumination from fluorescent lamps on the ceiling as in Figure 7 (a). The audio data is sent from a PC to speakers at long distance via two relays. The distance between each module is 225 cm and the total distance from transmitter to receiver is 705 cm.

To test the degree of signal distortion in our proposed multi-hop audio data transmission system, we use sample sounds (i.e. sine waveform with different frequencies) sending from PC as input analog audio signal to transmitter. Figure 7(b) – Figure 7(e) show the S/PDIF digital audio signal that we measure by oscilloscope at the output of transmitter and receiver, relay 1, and relay 2, respectively. As we can see in those figures, all digital signals still remain square shapes and their frequencies are unchanged because received signals are improved at each module before sending to the next one. As in Figure 7(b) – Figure 7(e), S/PDIF digital audio signal is shifted by adding DC component of around 1.8V to prevent the signal from clipping by LEDs. At the receiver, DC component of that digital audio signal is filtered out as in Figure 7(c) before converting to analog audio signal.


Transmitter
Relay 1
Relay 2
225 cm
Receiver
RIGOL STOP 1.88V
Vmax(1)= 3.76V Vmin(1)=40.0mV Freq(1)=2.857MHz
CH1 1.00V Time 500.0ns T+0.0000s
RIGOL STOP 8.00mV
Vmax(1)= 384mV Vmin(1)=-312mV Freq(1)=2.857MHz
CH1 200mV Time 500.0ns T+0.0000s
RIGOL STOP 2.00V
Vmax(1)= 3.60V Vmin(1)=40.0mV Freq(1)=2.857MHz
CH1 1.00V Time 500.0ns T+0.0000s
RIGOL STOP 2.00V
Vmax(1)= 3.64V Vmin(1)=40.0mV Freq(1)=2.857MHz
CH1 1.00V Time 500.0ns T+0.0000s

(a) (b) (c) (d) (e)

Fig. 7. (a) Setup of our proposed VLC based multi-hop audio transmission system; S/PDIF digital audio data signal observed on oscilloscope at (b) transmitter, (c) receiver, (d) relay 1, (e) relay 2



(a)

(b)

Fig. 8. (a) 4 kHz and (b) 10 kHz analog sample sound observed on oscilloscope at transmitter/receiver

Figure 8(a) and Figure 8(b) show the shapes of 4 kHz and 10 kHz audio analog signals observed at transmitter and receiver, respectively. The results confirm that our multi-hop transmission system has good frequency response. Thus, it can provide high quality audio transmission.

4 Conclusions

In this paper, we propose a VLC based multi-hop audio data transmission system. With the support of relays, our proposed system can provide audio data transmission at a long distance. The results of experiment show that audio signals remain good quality without signal distortion when traveling through each hop. The received audio sound is clear when playing on speakers. Therefore, we believe that our proposed multi-hop audio data transmission system can be used to build VLC based networks for sending high quality audio data among devices.

Acknowledgement. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0007119).

References

1. T.D.C. Little, P. Dib, K. Shah, N. Barraford, B. Gallagher: Using LED Lighting for Ubiquitous Indoor Wireless Networking. In: IEEE International Conference on Wireless & Mobile Computing, Networking & Communication, pp. 373-378. (2008)
2. Do Ky Son, Eun Byeon Cho, Chung Ghiu Lee: Demonstration of visible light communication link for audio and video transmission. In: Photonic Global Conference (PGC 2010), pp. 1-4. (2010)
3. Seungwan Jo, Le The Dung, Beongku An: LED Communication-based Multi-hop Wireless Transmission Network System. In: The Journal of The Institute of Webcasting, Internet and Telecommunication (IWIT), vol. 12, no. 4, pp. 37-42. (2012)
4. `http://www.iec.ch/standardsdev/publications/is.htm`
5. `http://en.wikiaudio.org/AES_EBU`

A Practical Adaptive Scheme for Enhancing Network Stability in Mobile Ad-Hoc Wireless Networks

Le The Dung[1], Sue Hyung Ha[1], and Beongku An[2]

[1] Dept. of Electronics & Computer Engineering in Graduate School, Hongik University, Korea
thedung_hcmut@yahoo.com, xtempthotx@nate.com
[2] Dept. of Computer & Information Communications Engineering, Hongik University, Korea
beongku@hongik.ac.kr

Abstract. The performance of mobile ad-hoc wireless networks is highly sensitive to node-to-node connection caused by node movement. Thus, to create robust mobile ad-hoc networks against node mobility, stable routing paths and routing refresh interval should be selected adaptively based on instantaneous network parameters. In this paper, we present a practical adaptive scheme to improve network stability in mobile ad-hoc networks by adaptively selecting most stable routing paths and optimal routing refresh interval. We validate our proposed adaptive scheme via simulation using OPNET. The simulation results in different scenarios demonstrate our proposed scheme remarkably enhances the network stability, providing robust mobile ad-hoc wireless networks.

Keywords: Mobile ad-hoc networks, Network stability, Node mobility.

1 Introduction

Mobile ad-hoc wireless networks have shown their advantages in specific situations where wireless communication is needed for a short time, in decentralized manner. In mobile ad-hoc wireless networks, all nodes are free to move during communicating with other node. However, due to node mobility, the network topology may change rapidly. Thus, it is challenging to obtain robust mobile ad-hoc wireless networks against node mobility so that they can provide flexible, reliable services.

From the knowledge of how node mobility impacts the performance of mobile ad-hoc networks in [1], in this paper, we propose a practical adaptive scheme to select stable connections in mobile ad-hoc networks. The contributions of our proposed scheme are as follows:

- *Simplicity* - in our proposed scheme, we exploit the information obtained at each node (i.e. node location, node mobility) to derive other useful information without using additional control overheard which helps to reduce network congestion.
- *Efficiency* - with the information of node mobility, our proposed scheme can select most stable routes. Moreover, route lifetime can be calculated exactly and used as a guideline for other routing parameters (i.e. routing refresh interval, routing table update interval) to work in adaptive manner.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 886–892, 2013.

- *Compatibility* - our proposed scheme can be applied to any routing protocols in mobile ad-hoc wireless network by adding some necessary information in RREQ and RREPs during the process of routing path creation.

The rest of this paper is organized as follows. In section 2, we review recent research in network stability improvement. In section 3, we present in detail our proposed adaptive scheme. We evaluate its performance in different settings of node mobility and node density in section 4. Finally, section 5 concludes the paper.

2 Related Works

Some approaches have been done in order to increase the link stability and path stability in mobile ad-hoc wireless networks. In [2], the authors propose a method which uses link stability factor (LSF) and path stability factor (PSF) to select a stable path. These two factors are calculated by indirect measurement of the distance between two mobile nodes based on reception power threshold at receiver antenna. The authors define stable zone and caution zone of a mobile node. The link between two mobile nodes is considered to be stable if the distance between them is less than radius of stable zone and vice versa. This method didn't explicitly consider node's mobility-aware link stability and path stability. Also, the distance between two mobile nodes cannot fully model the impact of node mobility on link stability and path stability. The authors in [3] propose link stability prediction based routing algorithm. This algorithm uses stochastic model to calculate average link duration calculated as a metric to select the stable link. However, in this algorithm, each node has to periodically send HELLO packets. Also, the calculation of link duration does not consider the effect of node pause. In [4], the authors use a model to calculate the amount of time two mobile nodes stay connected. From this information, a proposed algorithm classifies node's neighbors into eight different zones based on the heading directions of those neighbors. The model used in [4] may not predict link duration accurately because it does not use mobility information to calculate long-term traveling pattern of mobile nodes.

3 Proposed Practical Adaptive Scheme

In this section, we propose an adaptive scheme to create stable routing paths which are robust against high node mobility in mobile ad-hoc wireless networks. The proposed scheme consists of adaptive selection of stable link which is used in route discovery phase and adaptive routing refresh interval which is used in route maintenance phase.

3.1 Route Discovery: Adaptive Selection of Stable Link

In traditional routing protocols in mobile ad-hoc wireless networks, several metrics such as minimum hop count [5], ETX [6] and minimum transmission time [7] are used to identify “*best routing path*”. Minimum hop count metric is easy to implement but does not reflect network environment. ETX metric is based on link lost ratio. Thus, it can be considered as passive metric which means the routing paths have to suffer packet lost due to link disconnection and then give feedback later to adjust routing parameters. Minimum transmission time does not reflect the potential of link break due to node mobility. Since in those metrics the simultaneous information of node mobility is not considered, the selected forwarding nodes may not form stable routing paths. Motivated by above issues, in this paper, we propose a practical adaptive scheme to enhance network stability in mobile ad-hoc wireless networks by inserting mobility information of mobile nodes into RREQs.

To implement the calculation of link duration of the link between two mobile nodes by using Eq. (1), we use RREQ packet with the format in Figure 1(a).

Node’s mobility	Weakest link’s information	Route record	Sequence number	TTL

(a)

Route record	Weakest link’s information

(b)

Fig. 1. The format of (a) RREQ packet, (b) RREP packet

A triplet “node’s mobility” (node’s current location, ending waypoint location, node speed) is used to fully describe the moving segments and the velocities of mobile nodes in vector forms in every moving segments. Weakest link’s information is a set of two items, i.e. the duration of the weakest link, the time that this duration is recorded. Route record has the same function as in DSR routing protocol [7]. Sequence number is used to detect duplicate RREQs. TTL prevents RREQs from flooding through many nodes.

The RREP packet has the format in Figure 1(b). Route record in RREP consists of mobile node’s ids of intermediate nodes in the most stable routing path, copied from RREQ received at destination node. Weakest link’s information consists of two items (i.e. duration of the weakest link, the time that this duration is recorded) of the weakest link in that most stable routing path.

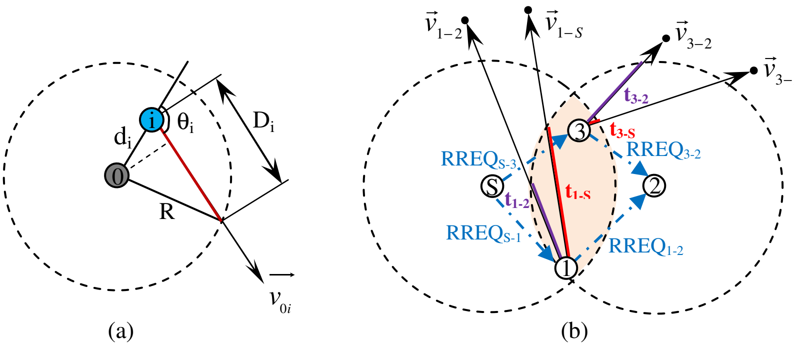


Fig. 2. (a) Calculation of link duration. (b) Stable link selection based on the information of link duration between two mobile nodes.

When source node has data packet to send, it performs stable route discovery process. The stable route discovery process has the following steps:

Step 1. At the initial time, source node S broadcasts the information of its location and its waypoint in RREQ to its neighbor nodes.

Step 2. After receiving RREQ from source node, node calculates the link duration (t_{iS}) of itself and the source node as in Figure 2(a) by using equation (1), and then adds this information to next RREQ.

$$t_{link} = D_i / v_{0i} = (\sqrt{R^2 - d_i^2 \sin^2(\pi - \theta_i)} + d_i \cos(\pi - \theta_i)) / 2v_{fix} \sin(\frac{\alpha}{2}) \qquad (1)$$

Figure 2(b) shows the relationship between link duration and link stability of the link between two mobile nodes.

Step 3. When next node receives unduplicated RREQ, it calculates the link duration of itself and pre-hop node ($t_{i,\ i\text{-}1}$) by using the same Equation (1). Then it compares this link duration with the one stored in Routing Cache. If the link duration calculated from the information in received RREQ is greater than the link duration stored in Routing Cache, this mobile node stores this link duration into Route Cache, adds its updated mobility information in RREQ, inserts its id in route record field, and then forwards this RREQ packet. Otherwise, it does nothing. This process repeats for all mobile nodes which receive unduplicated RREQs in the networks.

Step 4. Upon receiving RREQs, destination node chooses the most stable one from the available routing paths. It copies route record and weakest link's information in RREQ traversed through the most stable path. Destination node sends RREP back to source node by using the path specified in RREQ's route record.

Step 5. After receiving RREP, source node sends data packets via this stable path.

3.2 Route Maintenance: Adaptive Routing Refresh Interval

Route maintenance of any routing protocols in mobile ad-hoc wireless networks is an important task to keep a continuous connection between source node and destination node. For mobile ad-hoc networks, in most of cases, route disconnection of a multi-hop path happens when any link of it breaks due to node mobility. A good route maintenance method should determine when that event happens then creates new update routing path by resending RREQ packets. The duration between two routing path updates is called *routing refresh interval*. Obviously, the routing refresh interval of a multi-hop path depends on the lifetime of weakest link in that path.

In our proposed scheme, as shown in Figure 3, the RREP sent from destination node to source node has the information of the weakest link on the selected path. After source node receives RREP, it uses the routing path described in RREQ to forward data packets to destination node. At the same time, it set timer equal to the link duration of the weakest link. After this timeout, source node sends RREQ to build a new routing path.

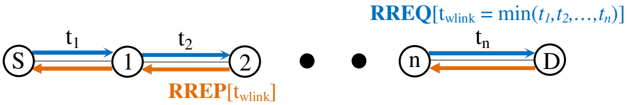
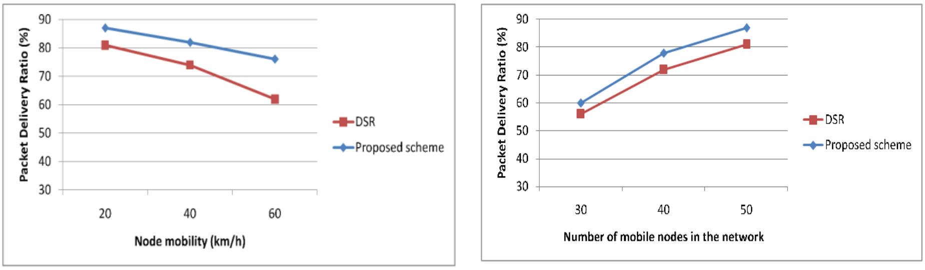


Fig. 3. Adaptive routing refresh interval for a routing path is calculated from weakest link information of that path

4 Performance Evaluation

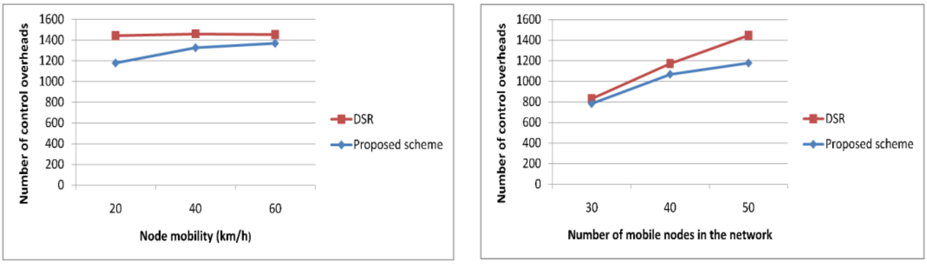
We simulate with a routing path between a random pair of source node and destination node in a network size of 1000m×1000m. Each mobile node has a radio range of 250m, moving under Random Waypoint mobility model with pause time uniformly distributed in [0; 10sec]. Source node sends 512 byte-data packets with constant rate of 20 packets per second. Source node begins to send data packets right after it receives the first RREP and continues sending data packets to destination node throughout the simulation without using DATA_ack between mobile nodes in the routing path. We evaluate how our proposed scheme can improve network stability of mobile adhoc networks compared with conventional routing protocols by changing node speed and the number of mobile nodes in different scenarios.

Figure 4 shows Packet Delivery Ratio (PDR) as the functions of node mobility and node density, respectively. In Figure 4(a), we simulate with 50 mobile nodes inside the network and vary node mobility. As we can see in Figure 4(a), as node mobility increases, our proposed scheme gives remarkably higher PDR than DSR routing protocol because it selects stable routing paths to forward data packets. In Figure 4(b), when number of mobile node is low, the connectivity of each node is low. Therefore, the PDR values obtained from our proposed scheme and DSR are not much different. But as the number of mobile nodes in network increases, by using our proposed scheme a mobile node can select the most stable link among many available links with its neighbors, thus our proposed scheme gives higher PDR.



(a) (b)

Fig. 4. Packet Delivery Ratio (PDR) as functions of node mobility and number of mobile nodes in the network



(a) (b)

Fig. 5. Total number of control overheads as a function of node mobility and number of mobile nodes in the network

Figure 5 shows the total number of control overheads (i.e. RREQs and RREPs) used in the network as the functions of node mobility and node density, respectively. In Figure 5(a), we simulate with 50 mobile nodes and vary node mobility. In Figure 5(a), since source node in DSR periodically sends RREQs to update routing path and destination node reply with RREPs, the total number of control overhead does not change much with node mobility. By contrast, in Figure 5(b), the total number of control overheads significantly changes when we keep node mobility and vary the number of mobile nodes in the network. In our proposed scheme, we use adaptive routing refresh interval based on the stability of routing paths, thus our proposed scheme uses lower number of control overheads compared with that in DSR.

5 Conclusions

In this paper, we propose a practical adaptive scheme to select stable routing paths and determine routing update interval in mobile ad-hoc wireless networks dynamically based on the information of node mobility. The simulation results from different settings of node speed and node density show that our proposed scheme can create reliable routing paths with lower route maintaining effort in mobile ad-hoc wireless networks.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0007119).

References

1. Dung, L.T., An, B.: A modeling framework for supporting and evaluating performance of multi-hop paths in mobile ad-hoc wireless networks. Computer and Mathematics with Application 64(5), 1197–1205 (2012)
2. Sun, J., Liu, Y., Hu, H., Yuan, D.: Link Stability Based Routing in Mobile Ad hoc Networks. In: IEEE 5th ICIEA, pp. 1821–1825 (2010)

3. Hu, X., Wang, J., Wang, C.: Link Stability Prediction and its Application to Routing in Mobile Ad Hoc Networks. In: IEEE 2nd PEITS, pp. 141–144 (2009)
4. Al-Akaidi, M., Alchaita, M.: Link stability and mobility in ad hoc wireless networks. IET Communications 1(2), 173–178 (2007)
5. Meghanathan. N.: Location Prediction Based Routing Protocol for Mobile Ad Hoc sNetworks. In: IEEE GLOBECOM 2008, pp. 1–5 (2008)
6. De Couto, D., Aguayo, D., Bicket, J., Morris, R.: A high-throughput path metric for multihop wireless routing. In: ACM MOBICOM 2003, pp. 134–146 (2003)
7. Johnson, D., Maltz, D., Broch, J.: DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks. In: Ad Hoc Networking, ch. 5, pp. 139–172. Addison-Wesley (2001)

A Geomulticast Routing Protocol Based on Route Stability in Mobile Ad-Hoc Wireless Networks

Sue Hyung Ha[1], Le The Dung[1], and Beongku An[2]

[1] Dept. of Electronics & Computer Engineering in Graduate School, Hongik University, Korea
xtempthotx@nate.com, thedung_hcmut@yahoo.com
[2] Dept. of Computer & Information Communications Engineering, Hongik University, Korea
beongku@hongik.ac.kr

Abstract. Geomulticast is a specialized location-dependent multicasting technique, where messages are multicasted to some specific user groups within a specific area [1]. In this paper, we propose a routing protocol for supporting geomulticast service based on route stability in mobile ad-hoc wireless networks. The main features and contributions of the proposed method are as follows. First, we present a direction guided routing method that can reduce control overhead for construction of routes and improve data transmission efficiency. Second, we introduce how to calculate and evaluate link stability between two nodes as well as route stability of multi-hop quantitatively by using nodes' mobility. Third, we can establish the most stable path by using those two information, link stability and route stability, between a source node and a representative node as well as a representative node and candidate destination nodes within some specific region in order to deliver packets with reliability, reduced overhead, and improved data transmission efficiency. Fourth, a route stability model is presented. We evaluate the proposed routing protocol by using OPNET. The results show that the proposed routing can support the geomulticast services effectively in mobile ad-hoc wireless networks.

Keywords: Geomulticast, Location-based service, Multicast, Routing, Mobile ad-hoc networks, Route stability, Mobility.

1 Introduction

Geomulticast is a specialized location-dependent multicasting technique, where messages are multicasted to some specific user groups within a specific area [1]. While conventional multicast protocols define a multicast group as a set of nodes with a multicast address and geocast defines a geocast group as all the nodes within a specified zone at a given time, a geomulticast group is defined as a set of nodes of some specific groups within a specified area [1-3]. In this paper, we propose a routing protocol for supporting geomulticast service based on route stability in order to deliver packets with reliability and reduced overhead to the destinations within some specific user groups in mobile ad-hoc wireless networks. Some considerations for developing

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 893–898, 2013.

of technologies in mobile ad-hoc wireless networks are more critical issues because these networks present dynamic, sometimes rapidly changing, random, multi-hop topologies and the mobile nodes communicate with each other over wireless links. Currently, there are a lot of active research works [1-6] for location-dependent services such as location-based routing, location-based multicast, geocast etc.

The contents of this paper are as follows. Chapter 2 presents in detail the proposed geomulticast routing protocol with basic concepts, algorithm, and theoretical analysis. In chapter 3, we will discuss the performance evaluation. Finally, chapter 4 concludes this paper with some discussions.

2 The Proposed Geomulticast Routing Protocol

In this section, we explain in detail the basic concepts, protocol, and theoretical analysis of the proposed geomulticast routing protocol. Figure 1 presents the basic concepts of the proposed geomulticast routing protocol while Figure 2 illustrates the route stability theoretical analysis model for supporting geomulticast routing services.

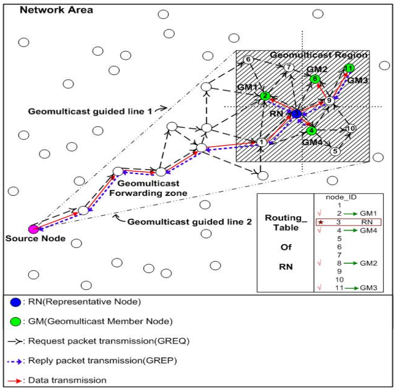

Network Area
Geomulticast Region
GM1
GM2
GM3
GM4
RN
1
2
3
4
5
6
7
8
9
10
11
Geomulticast guided line 1
Geomulticast Forwarding zone
Geomulticast guided line 2
Source Node
Routing_ Table Of RN
node_ID
1
√ 2 → GM1
★ 3 RN
√ 4 → GM4
5
6
7
√ 8 → GM2
9
10
√ 11 → GM3
●: RN(Representative Node)
●: GM(Geomulticast Member Node)
→: Request packet transmission(GREQ)
→: Reply packet transmission(GREP)
→: Data transmission

Fig. 1. The basic concepts

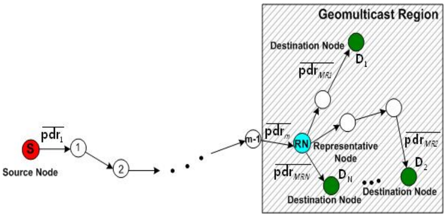


Fig. 2. The route stability analysis model

2.1 Basic Concepts

The basic concepts of the proposed method are as follows. First, the motivation is to use multicast approach by utilizing location-dependent information to support QoS service, save resources, and finally improve system performance. Second, a direction guided routing strategy is utilized by using direction guided line information to reduce control overhead for construction of routing route and improve the efficiency of data transmission. Third, calculation of link stability between two nodes as well as route stability of multi-hop quantitatively such as between a geomulticast source node and a representative node (RN), between a RN and a geomulticast member nodes (GM nodes) in geomulticast region by using nodes' mobility to find stable routing routes and improve packet delivery ratio. Fourth, establishment of the most stable path by using those two information, link stability and route stability, between a geomulticast source node and a representative node (RN) as well as between a RN and GM nodes in order to deliver packets with reliability and reduced overhead to the GM nodes within some specific user group destinations in the geomulticast region.

2.2 The Routing Protocol for Geomulticast Services

The detail explanations of the proposed geomulticast routing protocol are as follows. In this work, we assume that all nodes know their position information via GPS.

Step 1: Sender creates GREQ (Geomulticast Request packet) and sends to neighbor nodes. GREQ consists of source node ID, multicast ID, the information of Direction guided line, the information of geomulticast region.

Step 2: The node who receives GREQ compares its position information with the geomulticast guided line information in the GREQ.

- ✓ If its position is within the guided line, then the node calculates link stability and route stability by using Equation (1) between itself and previous node. We will in detail explain how to derive Equation (1) in section 2.3. After these operations, the node saves this information in Routing_Stability_Table (RS_Table).
- ✓ If its position is outside of the guided line, then the node doesn't send the GREQ message anymore.

Step 3: If one node receives GREQs from many nodes, then it calculates average stability by using each path's route stability and saves in RS_Table in Table 1. Then it compares the route stability with average stability. If route stability value is less than average stability, the node doesn't transmit GREQ with path information to the next node. This process will be repeated continuously until a representative (RN) node receives the GREQ.

Step 4: When a destination node, called representative node (RN) as geomulticast member, receives some GREQ within geomulticast region, the RN sends the GREQ message to the neighbor nodes within geomulticast region to find geomulticast member nodes. The RN is the geomulticast member node (GM node) that in the middle area of the geomulticast region with the most stable mobility. The link stability and route stability by using equation (2) between RN and GM node are calculated as step 3. We will explain in detail how to derive the equation (2) in section 2.3. Finally, the GM nodes join the RN as geomulticast group members.

Step 5: Then the RN creates GREP (Geomulticast Reply packet) and sends to the geomulticast source node via the path which has the highest Route Stability. If the geomulticast source node receives GREP, then routing route will be set up.

Step 6: When the geomulticast source receives the GREP, it will transmits data packets to the RN within the geomulticast region via the most stable route, and the RN also transmits the data packets to the geomulticast members nodes via the most stable route between the RN and member nodes.

2.3 The Theoretical Analysis: Route Stability Model

Figure 2 presents the theoretical analysis model for the route stability calculation of proposed geomulticast routing protocol from a geomulticast source node to GM nodes.

In Figure 2, D_1 is geomulticast member node 1, …, D_N is geomulticast member node N, and N is the number of geomulticast member nodes, respectively.

The route stability between a geomulticast source and RN can be obtained as in Equation (1) while the route stability between RN and N geomulticast member destinations can be obtained as in Equation (2), respectively.

$$pdr_m = \sum_{m=1}^{\left\lceil \frac{L}{R} \right\rceil} \left\{ \prod_{i=1}^{m} \overline{pdr_i} \bullet P_m(M) \right\} \tag{1}$$

where $\lceil \alpha \rceil$ is the ceiling function (the smallest integer not less than α) of α, L is maximum distance between source node and RN, and R is radio range, m is hop counts from source to RN, $\overline{pdr_i}$ is average pdf (probability density function) of link lifetime between two nodes and $P_m(M)$ is pmf (probability mass function) of hop counts in a link. By using Equation (1) we can calculate the stability from RN to each GMs noted in Equation (2).

$$pdr_{MR_k} = \sum_{A=1}^{\left\lceil \frac{\gamma\sqrt{2}}{R} \right\rceil} \left\{ \prod_{i=1}^{a} \overline{pdr_i} \bullet P_a(A) \right\} \quad (k = 1,2,\cdots,\mathrm{N}) \tag{2}$$

where N is the number of GMs, $P_a(A)$ is the pmf of hop counts from RN to GMs, γ indicates the length of geomulticast region's side. By using Equation (1) and (2), we can obtain final PDR (i.e., route stability) between a geomulticast source node and several GM nodes as in Equation (3).

$$PDR_{GM} = pdr_m \times \frac{pdr_{MR_1} + pdr_{MR_2} + \cdots + pdr_{MR_N}}{N} \tag{3}$$

where PDR_{GM} is total route stability of proposed geomulticast routing protocol.

3 Performance Evaluation

3.1 Simulation Environment

In this simulation, we assume that all nodes already know their position via GPS (Global Positioning System). Table 1 shows simulation environment.

Table 1. Simulation Environment

Simulation Tool	OPNET (Optimized Network Engineering Tool)
Network size	5 km × 5 km
Geomulticast Region	2 km × 2 km
Number of Mobile nodes in Network Area	500
Number of Mobile nodes in Geomulticast Area	50, 100, 150
Number of Geomulticast nodes	5, 10, 15
Radio range	250m
	0～2π
Mobility model	RWP(Random WayPoint)

3.2 Performance Measurement Parameter

The measurement parameters for performance evaluation are as follows. **i) PDR (Packet Delivery Ratio):** data packet transmission rate that source node's sending to GM nodes' receiving, **ii) Scalability:** PDR as a function of number of GM, **iii) Control Overhead:** the average number of control overhead signal for construction of routing routes, **iv) Delay:** average time for construction of routing routes.

3.3 Simulation Results

Figure 3 shows the Packet delivery Ratio (PDR) as a function of node mobility. As we can in Figure 3, the results of simulation and analysis are almost the same. Figure 4 presents the scalability (i.e., PDR) as a function of GM nodes. As we can in Figure 4, the results of simulation and analysis have similar patterns. However, as node mobility increases, the PDR decreases a little bit. And if the GM nodes are increased, the possibility to find stable routes between RN and GM nodes is lower, leading to decreased PDR.

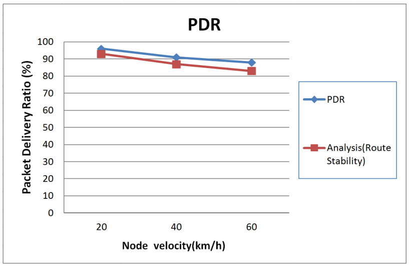


Fig. 3. Packet Delivery ratio (PDR) as a function of node mobility

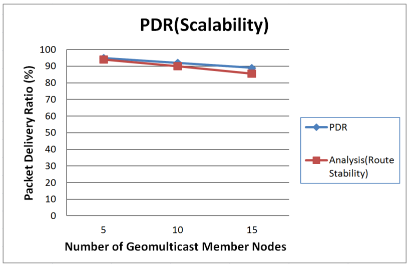


Fig. 4. Scalability as a function of GM nodes

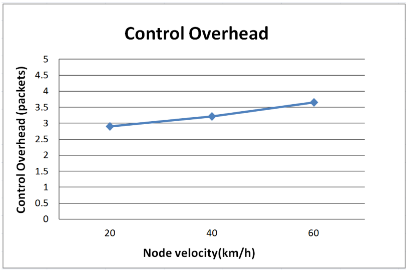


Fig. 5. Control overheads as a function of node mobility

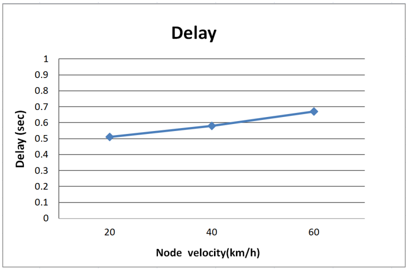


Fig. 6. Delay as a function of node mobility

Figure 5 and Figure 6 show control overhead and delay for construction of routing routes as a function of node mobility, respectively. As we can see in Figure 5 and Figure 6, the control overhead and delay are increased if the node mobility is increased. The reason is that network environment will be much more dynamic if the nodes' mobility is increased.

4 Conclusions

In this paper, we propose a geomulticast routing protocol based on route stability in mobile ad-hoc wireless networks. By combining two concepts of geocast and multicast, we can formulate a geomulticast strategy that can transmit the data messages to some candidate nodes in special region with improved system performance. We can select the most stable path among candidate routes between a geomulticast source and GM nodes in geomulticast region by calculating both link stability and route stability together. Especially, we present a theoretical analysis model to calculate both link stability and route stability routes between a geomulticast source and GM nodes in geomulticast region. The performance evaluation of the proposed geomulticast routing protocol is implemented by OPNET. The results of performance evaluation show that the patterns of simulation results and analysis results are very similar.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0007119).

References

1. An, B., Papavassiliou, S.: Geomulticast: architectures and protocols for mobile ad-hoc wireless networks. Journal of Parallel and Distributed Computing 63, 182–195 (2003)
2. An, B., Papavassiliou, S.: MHMR: mobility-based hybrid multicast routing protocol in mobile ad-hoc wireless networks. Wireless Communications and Mobile Computing 3, 255–270 (2003)
3. Beongku, A., Tekinay, S., Papavassiliou, S., Akansu, A.: A Cellular Architecture for Geocast Services. In: IEEE VTC 2000, vol. 3, pp. 1452–1459 (2000)
4. Shiraish, Y., Takahashi, O., Miki, R.: A geocast-based Multicast Method for Continuous Information Delivery in MANET. In: IEEE International Conference in P2P, Parallel, Grid, Cloud and Internet Computing, pp. 511–516 (2010)
5. Dung, L.T., An, B.: A Modeling Framework for Supporting and Evaluation Performance of Multi-hop Paths in Mobile Ad-hoc Wireless Networks. Computers and Mathematics with Applications 64(5), 1197–1205 (2012)
6. Hu, Wang, J., Wang, C.: Link Stability Prediction and its Application to Routing in Mobile Ad Hoc Networks. In: 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS), pp. 141–144 (2009)

RFID-Based Indoor Location Recognition System for Emergency Rescue Evacuation Support

Dae-Man Do, Maeng-Hwan Hyun, and Young-Bok Choi

Tongmyong University, 535 Yongdang-Dong, Nam-Gu, Busan, Korea 608-711
dmdo1224@naver.com, mang93@nate.com, ybchoi@tu.ac.kr

Abstract. Disasters such as fires, earthquakes, and acts of terror in public places such as subway stations, airports, and department stores can lead to tragic consequences. Although a number of studies are being conducted on indoor location recognition systems, they are not appropriate for emergency rescue evacuation support system (ERESS), which requires building new infrastructures in the indoor environment of all public buildings. This paper proposes an RFID-based indoor location recognition system for the ERESS. The proposed indoor location recognition system uses RFID readers and active tags to collect position coordinates in real time, allowing a single tag to monitor actual locations. Performance evaluation based on experiments indicates that the proposed indoor location recognition system is effective for monitoring indoor pedestrians in ERESS applications.

Keywords: LPS, RFID, location recognition, ERESS.

1 Introduction

A number of accidents and disasters occur in public places each year. Fires, natural disasters, and acts of terror in indoor public facilities such as subways stations, airports, and department can lead to tragic consequences for many people [1]. In addition to 9/11 attacks in the United States, the number and the scale of terrorist acts are increasing in other regions, including Afghanistan and Southeast Asia. Therefore, in order to respond to large scale emergency situations, there is a need for a reliable system capable of supporting evacuation quickly and clearly. Over the past 20 years, the United States and Great Britain have conducted studies on indoor location recognition systems and published a number of research results findings to resolve problems, elevating the accuracy of the systems to a very high level. However, these systems are not appropriate for emergency rescue evacuation support system (ERESS), which requires building new infrastructures in the indoor environment of all public buildings.

This paper proposes an RFID-based indoor location recognition system for ERESS applications that monitors people’s movement in panic situation in a building caused by fire, natural disaster, or terrorist act, quickly detecting emergency situations and allowing quick, accurate, and flexible measures to be taken according to the circumstances.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 899–906, 2013.

2 Emergency Rescue Evacuation Support System

Local positioning system (LPS) is a service that uses mobile devices to monitor positions of people and objects for various applications [2]. Unlike GPS, which uses absolute position data such as latitude and longitude, Indoor LPS uses relative positions to calculate the distance from a reference position. The LPS technology can be categorized into triangulation, image recognition, proximity recognition, and navigation monitoring. Major LPS services include Active Badge, Active Bat, Cricket System, RADAR (radio detection and ranging), LANDMARC (land-mine detection advanced radar concept), and Easy Living [2].

ERESS uses terminals such as mobile phones, smart phones, and tablet PCs (ERESS terminals) to provide real-time information that can be accessed by indoor pedestrians and to help evacuation during emergency [3][4][5]. The objective of ERESS is to evacuate indoor pedestrians in panic situations to reduce the number of casualties caused by a disaster. ERESS monitors occurrence of disasters in real time, enabling prompt evacuation to a safe place.

3 Indoor Location Recognition System

Choosing the RFID system is an important factor for implementing an ERESS. The RFID system must have a wide area of recognition, be robust against adverse effects from the surrounding environment, and use omnidirectional antennas[6]. In this study, 433 Mhz active reader RX-1000 manufactured by WAVETREND in Great Britain was used for the ERESS.

3.1 Indoor Location Recognition System

3.1.1 Moving Average Filter

Due to RF signal characteristics, RF signal is affected by the surrounding environment, and the received signal strength indicator (RSSI) value between the RFID reader and active tag becomes unstable . A moving average filter is deployed to reduce some of the problems.

A typical moving average of n data points is expressed as follows [7]:

$$\overline{X}_k = \overline{X}_{k-1} + \frac{X_k - X_{k-n}}{n} \tag{1}$$

Calculating a moving average requires the average value of the prior stage, the most recent data, and the most recent data that does not exceed the buffer size. A moving average can be effectively used when RSSI is measured according to the number of buffers.

3.1.2 RSSI Compensation System

Even if an stable RSSI between RFID reader and active tag is achieved through the moving average filter, there remains the problem of a little delay from the original signal due to filter characteristics. Therefore, an RSSI compensation system is necessary to resolve the problem. The compensation system is applied by taking into account the increase and decrease of the RSSI value from the time perspective. This study uses an RSSI compensation system that limits the number of averages to 3 and compensates a specific signal when the RSSI value increases.

3.1.3 RSSI Balancing Algorithm

Balancing of RSSI for application in the indoor location recognition system is achieved with the moving average filter and the RSSI compensation system. The RSSI balancing algorithm is executed according to the number of tags identified using the moving average filter and the RSSI compensation system.

3.2 Map Matching Using Coordinate Points

3.2.1 RSSI Leveling by Distance

While RSSI can be stabilized using compensation and the moving average filter, there is the problem of having to constantly modify the measured position coordinates in order to monitor the RSSI value in real time and express the position in coordinates. Therefore, in order to mitigate the problem in map matching, the measured RSSI needs to be divided into three ranges, and a value that falls within a certain range be modified to an assigned absolute value. The three ranges of RSSI values with reference to the measurement average are shown in Fig. 1.

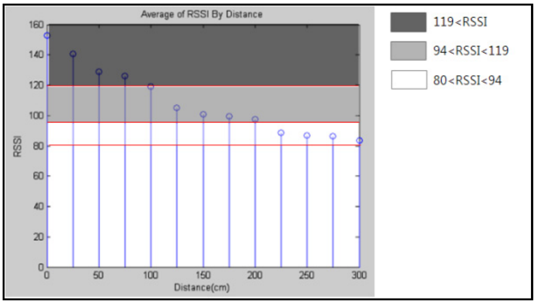


Fig. 1. RSSI Leveling by distance

3.2.2 Map Matching

The current position has to be shown on a map in order to enable real-time monitoring in an indoor location recognition system. For map matching, the indoor location need to be depicted and reduced by a consistent ratio to assign absolute coordinate values [8]. The map matching method proposed in this paper determines the position by comparing map data with the coordinate values around the tag. Whereas triangulation, a technique widely used for indoor location recognition, must recognize three or more tags, the proposed matching method uses only a single tag. The proposed map matching method is shown in Fig. 2.

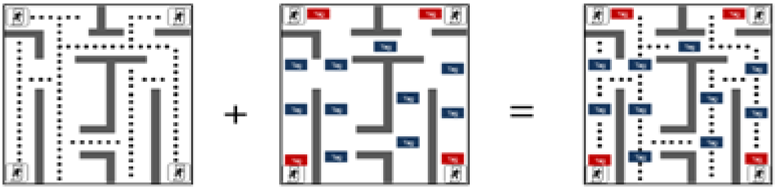


Fig. 2. Map matching

3.2.3 Coordinate Analysis Algorithm

The RFID reader compares its current position and the coordinate points received from active tag and moves the current position to the nearest coordinate point to modify its position in real time.

The coordinate analysis algorithm performs analysis using the coordinate data of a single selected tag. All coordinate points around the tag is compared with the current position. After the comparison, the current coordinate is modified to the point with the least amount of difference. The coordinate analysis algorithm is shown in Fig. 3.

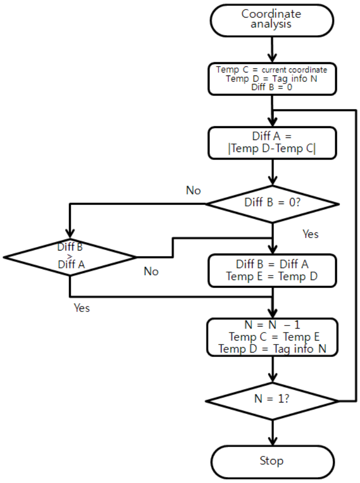


Fig. 3. Coordinate analysis algorithm

3.3 Indoor Location Recognition Algorithm

When the reader finds a tag, the indoor location recognition algorithm balances RSSI according to the number of tags. Then RSSI searches for the largest tag. The coordinate is analyzed using the data of a single tag found from the search. After coordinate analysis, the current position is modified. The algorithm of the indoor location recognition system is shown in Fig. 4.

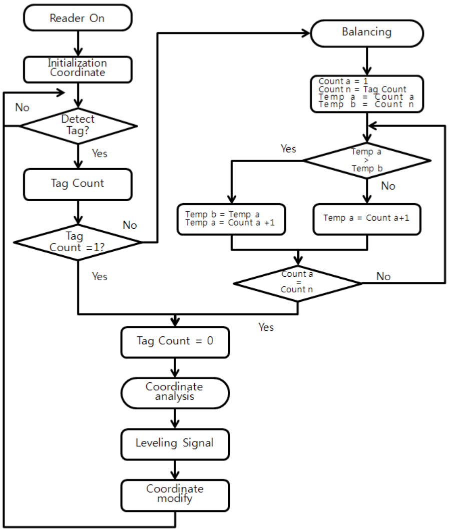


Fig. 4. Indoor location recognition algorithm

4 Performance Evaluation

In this chapter, the RFID-based indoor location recognition system is constructed and the performance of the RSSI balancing algorithm is evaluated using experiments. The on-site experimental environment is shown in Table 1.

Table 1. On-site experimental environment

Simulation Experiment Environment	Experiment 1	Experiment 2	Experiment 3
Place	4th floor of Academia-Industry Collaboration Center, Tongmyong University		
Area	22m X 15m		
Movement Speed	10 cm/s	30 cm/s	40 cm/s
Direction of Movement	Straight ahead	Straight ahead	Straight ahead, Left, Right
Number of Tags	6	7	7
Measurements per second	2	2	2
Number of Measurements	10	10	10
Coordinate Gap	-	100cm	100cm
Measurement Time	100sec	100sec	140sec

4.1 Experiment 1

In Experiment 1, the performance of the proposed RSSI balancing algorithm was evaluated for the RSSI values detected by the RFID reader. In Fig. 5, it can be confirmed that a specific amount of RSSI was compensated to the area where values were shifted and reduced due to the moving average filter, compensating the overall signal. Fig. 6 shows the process of comparing the RSSI values continuously received from each tag, selecting a high value, and measuring the RSSI of the single tag. The balanced RSSI was more stable and stronger than that of the measured signal.

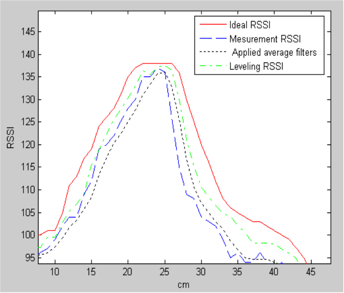


Fig. 5. Comparison of balanced RSSI

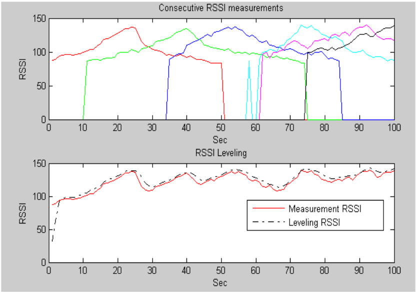


Fig. 6. Continuous RSSI measurement

4.2 Experiment 2

Experiment 2 was performed for specific directions of the pedestrian. Fig. 7 shows the coordinates of pedestrian's actual movement and Fig. 8 displays the coordinates estimated using the tag.

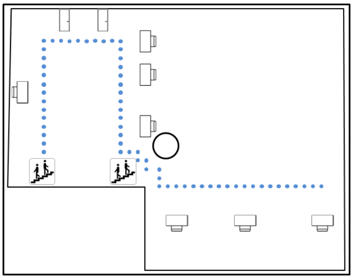


Fig. 7. Actual movement

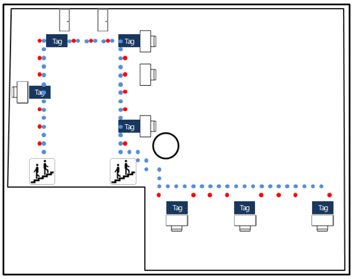


Fig. 8. Coordinates recognized by the tag

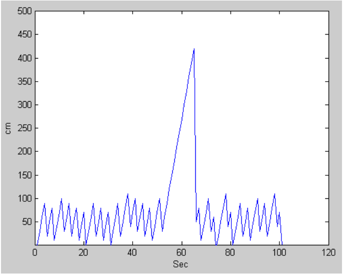


Fig. 9. Distance error by time

Fig. 9 shows the distance error by time. The coordinate is modified every time the tag is recognized. Since the coordinates according to the signal level was set in 1m intervals, the error is maintained within 1m. Between 50 and 65 seconds, error increased to 4m because although the tag was recognized, an identical coordinate was recognized before and after the coordinate point for each tag for a specific period.

4.3 Experiment 3

Experiment 3 was performed with the pedestrian walking in irregular directions. Fig. 10 shows the coordinates of pedestrian's actual movement and Fig. 11 displays the coordinates estimated using the tag.

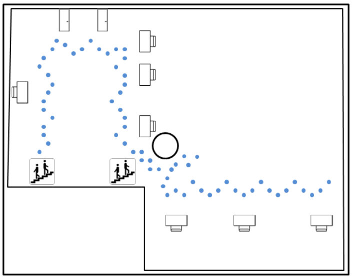


Fig. 10. Actual movement

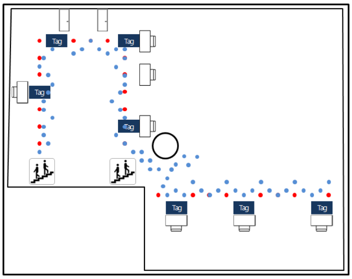


Fig. 11. Coordinates recognized by the tag

Fig. 12 shows the distance error by time. Unlike Experiment 2, which was performed with a straight walking path, Experiment 3 added the patterns of the pedestrian changing directions to left and right, and the error range increased to 3.5m.

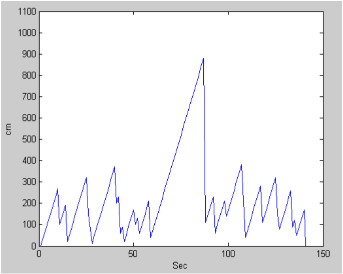


Fig. 12. Distance error by time

The results of performance evaluation indicate that balancing stabilized RSSI values from the conventional method. The proposed RFID-based indoor location recognition system showed an average error of 62cm when the tag was recognized continuously and 168cm when the tag was recognized irregularly.

Table 2. Error ranges in three experiments

		Momentary Error Range	Average Error	Remark
Experiment 1	Balanced RSSI	0.83～7.67(dBm)	3.64(dBm)	
	Measured RSSI	1.66～13.34(dBm)	7.55(dBm)	
Experiment 2		0～100(cm)	62(cm)	Excludes areas where coordinates were not recognized
Experiment 3		0～350(cm)	168(cm)	

5 Conclusions

This paper proposed a location recognition system for detecting positions inside a building using a single active tag. Since RF signal characteristics are sensitive to interference from the surrounding environment, an RSSI balancing algorithm based on a moving average filter and a compensation system was also proposed to stabilize the continuous signal. The results of performance evaluation based on experiments indicate that the proposed indoor location recognition system is effective for locating pedestrians in ERESS applications.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(NRF-2011-0024976).

References

1. Bong Chan, K., et al.: The features and research of domestic and foreign city fire trend analysis. In: The Korea Institute of Fire Science and Engineering Spring Conference of the Institute (April 2010)
2. Lee, W.H., et al.: Positioning System technology for ubiquitous Environment. Information Science and Technology 22(12) (December 2004)
3. Hayakawa, Y.: Analysis system development of pedestrian's behavioral for disaster event detection of ERESS. Kansai Univ. (March 2012)
4. Tsunemine, T., et al.: Emergency Urgent Communications for Searching Evacuation Route in a Local Disaster. In: IEEE Consumer Communications and Networking Conference 2008 (CCNC 2008), pp. 1196–1200 (January 2008)
5. Okada, C., et al.: A Novel Urgent Communications Technologies for Sharing Evacuation Support Information in Panic-type Disasters. In: The Sixth International Conference on Networking and Services (ICNS 2010), pp. 162–167 (March 2010)
6. Dong Ho, J., et al.: RTLS Design and Implementation by using active RFID. Journal of Korea Information and Communications Society 31 (December 2006)
7. Su Hwan, S., et al.: Higher-order time Moving Average Filter by using active-weighted charge sampling. Electronic Engineering Institute of Article 49(2), SD (February 2012)
8. Tae Kyum, K., et al.: Mapping algorithms for Error Compensation of Indoor Positioning System. Institute of Electronics Engineers of Article 47(4), CI (July 2010)

A Symmetric Hierarchical Clustering Related to the Sink Position and Power Threshold for Sensor Networks

Joongjin Kook

Realistic Platform Research Center,
Korea Electronics Technology Institute, Seoul, Korea
tipsiness@gmail.com

Abstract. Most existing clustering protocols have been aimed to provide balancing the residual energy of each node and maximizing lifetime of wireless sensor networks. In this paper, we present the symmetric hierarchical clustering strategy related to sink position and energy threshold in wireless sensor networks. This protocol allows networks topology to be adaptive to the change of the sink position by using symmetrical clustering that restricts the growth of clusters based on depth of the tree. In addition, it also guarantees each cluster has the equal lifetime, which may be extended compared with the existing clustering protocols. We evaluated the performance of our clustering scheme comparing to existing protocols, and our protocol is observed to outperform existing protocols in terms of energy consumption and longevity of the network.

Keywords: Sensor networks, Clustering protocol.

1 Introduction

In the sensor networks, wireless transmission is the most energy consuming operation. In addition, each sensor node has very limited batteries, and it is very hard to recharge them. Therefore, energy-efficient transmission protocol is required to maximize network lifetime of the entire sensor networks.

Many kinds of efforts have been done on developing energy-efficient transmission protocols for wireless sensor networks. Those can be categorized into routing, and clustering protocols. Particularly, the clustering protocols can significantly reduce energy consumption by aggregating multiple sensed data to be transmitted to the sink node. However, every existing clustering protocol assumes the sink node is fixed, and they only consider 'How can we configure clusters more energy-efficiently by size and/or count of clusters?' without concentrating on adaptive clustering related to the position of the sink node. Because the change of the sink node position is not considered, the existing networks topology might cause the residual energy of clusters out of balance. It is possible to configure network topology using simple clustering methods when the sink node that collects data is fixed. However, if the sink node is not fixed and dynamically changed, cluster reconstruction and cluster head selection must change according to the distance to the sink node.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 907–913, 2013.

In this paper, we present a symmetric hierarchical clustering protocol related to the sink position and energy threshold in wireless sensor networks. This protocol allows networks topology to adaptive by symmetrical clustering that restricting the growth of clusters based on depth of the tree, and this protocol guarantees the lifetime of the entire networks can be extended compared with the existing clustering protocols.

2 Related Works

Clustering protocols for the WSN can be categorized into two classes; hierarchical clustering protocols and chain-based aggregation protocols. LEACH[1] includes distributed cluster formation, local processing to reduce global communication, and randomized rotation of cluster-heads. Together, these features allow LEACH to achieve the desired properties. However, there is no guarantee that nodes selected as cluster head are evenly dispersed throughout the network because procedure to select cluster head is based on the random cluster formation method with local probability. To solve this problem, an improved version of LEACH was proposed, named LEACH-C[2]. In LEACH-C, cluster formation is made by a centralized algorithm at the base station. In 2005, Li et al. proposed EEUC[3], which is an energy-efficient unequal clustering protocol. In EEUC, nodes are partitioned into different-sized clusters; clusters closer to the base station have smaller sizes than those farther away from the base station. Thus cluster heads closer to the base station can preserve energy for the inter-cluster data forwarding.

TEEN[4] is similar to LEACH except that sensor nodes do not have data being transferred periodically. In TEEN, each sensor node decides to transmit their sensed data using a threshold value. Cluster heads broadcast the value, and if a sensed data is bigger than the threshold value, each node transmits data.

3 A Symmetric Hierarchical Clustering Protocol Based on the Sink Location and Energy Threshold

3.1 Impact of Clustering Protocol on Energy Consumption

The degree of energy consumption on each sensor node changes according to the distance between cluster head and sink, and transmission method by multi-hop or by direct. In case of 1-hop(e.g. LEACH), because cluster head transmits data to sink directly, the degree of energy consumption vary whether the cluster is the nearest to the sink or the farthest away from the sink. Generally, the farthest cluster away from sink consumes more energy than other clusters. In case of multi-hop, because of the data transmitted from all clusters by relay, the cluster nearest to the sink consumes large amount of energy.

Consequently, in cluster tree topology, the degree of energy consumption vary based on the roles of sensor nodes, thus, cluster head which need large amount of power must be distributed for energy-efficient networks.

The FND and LND[5] can be used to represent the barometers of energy-efficient network, because the infection of withdrawn arbitrary node spreads out to whole network. Consequently, the interval between FND and LND time must be minimized to be the greatest energy-efficient network.

If the position of the sink is fixed, cluster construction related with the distance from the sink node. Otherwise, it needed further study for clusters to construct energy-efficiently. We have researched focusing on this point.

3.2 Cluster Head Replacement According to the Energy Threshold

In traditional protocols that the number of cluster heads be reduced to decrease energy consumption or that energy efficiency-based optimal cluster size be constructed to extend the survival time of the network.

To calculate the whole energy consumption of the networks, we have to consider two parts. One is quantity of energy as roles of sensor nodes. Another is a volume of energy when role of sensor nodes is exchange. There is a significant disparity of energy consumption between cluster heads and member nodes.

All member nodes are transmitting perceived data to cluster head on allocated time slot periodically. And then cluster head transmit data aggregated in the cluster. Due to figure of network lifetime, it must be included that energy degree when cluster head replacing because of that.

P_{Tx} represents amounts of energy consumed for 1 byte data transmission, P_{Rx} the amounts of energy consumed for receiving 1 byte data. Whole network consists of n nodes, and if it is comprised of $C\%$ clusters, and it happens R count of cluster head replacement. pk_{Tx}, pk_{Rx} are size of packet when transmission and reception happen. In this time, Eq. 1 represents of whole energy degree that network has consuming.

$$E_{CHR} = \{pk_{Tx} \cdot P_{Tx} + pk_{Rx} \cdot P_{Rx} \cdot (nC - 1)\} \cdot R \cdot nC \tag{1}$$

In the Eq. 1, nC represent the number of nodes per each cluster. If receive cost is three times of transmit cost, Eq. 1 is replaced Eq. 2 as follows.

$$E_{CHR} = R \cdot nC(3nC - 2)P_{Tx} \cdot pk \tag{2}$$

In this Eq. 2, it seems that total amount of consumed energy of cluster head replacement is commensurate to R which is the number of count of cluster head replacement. In T-LEACH[6], decision whether to perform rounding is made based on additional residual energy in each sensor node to replace cluster heads. In other words, when current cluster heads maintain residual energy at a level above the pre-established threshold, heads are not replaced even when it is time to replace them and the time of rounding is delayed until the level falls below the threshold, thus making it possible for nodes to continuously play the roles of cluster heads. Also we know that whole energy consumption can be reduced by minimizing the number of R in Eq. 2.

One cluster consists of n nodes, firstly, the member of nodes transmit data to the cluster head consuming amounts of $P_{Tx} \times n \times l$(bit) and keep sleep mode until next round's time slot. Secondly, there are two times of energy consumption stages in the cluster heads; One is aggregation stage, in this stage, cluster head consumes amounts of $P_{Rx} \times n$(bit) × $(N\text{-}1)$ to aggregate data. N denotes the number of nodes per each cluster. Another is the transmission stage to the sink. In this stage, cluster head expenses amounts of P_{Tx} × l(bit) × (N-1) to transmit data. Obviously, cost of transmission stage can increased caused of growing distance to the sink.

To acquire threshold value, we also have to know how many times of round to active as member node in a cluster. The threshold must be set to amounts of C_{rnd} × P_{Tx} × l(bit) for accomplishment of member nodes until network extinguishes its own energy. C_{rnd} representing times of round, is calculated as follows:

$$C_{rnd} = \frac{E_{CHR}}{E_{PC}} \cdot 100 \tag{3}$$

E_{CHR} denotes power consumption of head replacement, E_{PC} denotes whole energy of each cluster and is represented Eq. 5. In this equation, E_{PC} denotes total amount of energy granted each cluster unit.

$$E(i)_{CHR} = n(5N_i - 3)P_{Tx} \tag{4}$$

$$E_{PC} = \frac{N}{k} \cdot E_{init} \cdot n_i \tag{5}$$

Eq. 4 represents of the amounts of energy cost for a round. Here, E_{CHR} composed of two cost value; one is the energy cost when the node is role of cluster head, and the other is the energy cost when the node actives member node. With Eqs. 3 and 4, what the threshold value when the cluster head can be replaced:

$$E_{Th} = \frac{C_{rnd} \cdot lP_{Tx}}{E_{init} \cdot n_i} \tag{6}$$

In Eq. 6, ETh represents threshold level of cluster head replacement. As we apply it to LEACH protocol, we could not only improve lifetime of LEACH but also make network to balance as shorten interval between FND and LND time. Figure 1 shows simulation result with tinyviz.

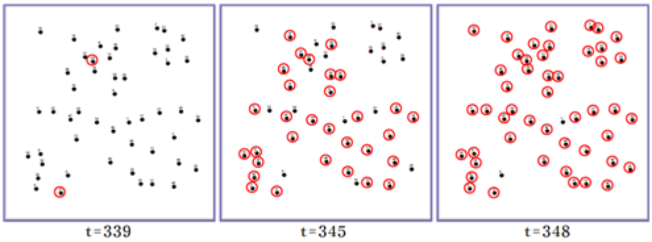


Fig. 1. Energy Threshold Based Cluster Head Replacement

3.3 Symmetric Clustering Based on the Location of the Sink

Our symmetric hierarchical clustering protocol, the cluster near the sink is the top level cluster. Because an upper cluster must transmit the data from a lower cluster to the sink, more energy is required. When we assume the level of the cluster tree is L, the forwarding energy consumed by the cluster head of the corresponding level can be presented as Eq. 8. In this equation, N is the number of all nodes and k is the number of clusters. The other parameters are the values from [1-2].

$$E_{fw} = lE_{elec}\left(\frac{N}{k} - 1\right) + lE_{DA}\frac{N}{k} + lE_{elec} + l\varepsilon_{fs}\left(\frac{d}{L}\right)^2 \tag{7}$$

The total forwarding energy consumption by the L level cluster is:

$$E_{totalfw} = E_{fw} \cdot \sum_{i=0}^{L-1}(L-i)(2^{L-i}) \tag{8}$$

If the level is 7.5 or above, the forwarding energy consumption gets more than the other case. If it is assumed that each node has the same initial energy and all nodes are distributed with equal spacing, in order for a certain cluster to have much more energy than the other cluster, the cluster size must be bigger. When the cluster size is calculated considering the energy consumption for data forwarding.

The cluster size increases 4/3 times as the level increases. In this case, the difference between the top level cluster and the lowest cluster steeply increases. The size of the top level cluster is too big so adequate clustering is difficult. Accordingly, when the level of the cluster tree increases over the specified amount, the trees can be divided into two or more to restrict the top level cluster size.

To meet the requirement for the optimal number of clusters that is calculated in LEACH. However, it is based on the distance between each cluster and the sink so it is not appropriate in the symmetric clustering that uses the multi-hop forwarding. The symmetric clustering must calculate the optimal number of clusters considering how many hops are used for data transmission from each cluster to the sink.

$$E_{total} = k \cdot E_{cluster} + E_{fw} \cdot \sum_{i=0}^{L-1}(L-i)(2^{L-i}) \tag{9}$$

In Eq. 9, the optimal number of clusters can be calculated. It makes the total energy E_{total} 0. When the cluster tree level is 3, the lowest energy consumption is presented in case the number of clusters is 7. With level 4, the lowest energy consumption is implemented in case the number of clusters is 15. When the number of clusters exceeds (2^L-1), the tree level increases. At this time, the forwarding energy consumption steeply increases.

When same-sized clusters are constructed, the cluster further away from the sink requires more energy for forwarding data to the sink. Thus, when compared with the clusters near from the sink, its life span gets shorter.

4 Experimental Results

In symmetric clustering protocol, the size of the farthest cluster from the sink is the smallest, and that of the nearest cluster is the largest. When the sensor nodes are assumed to be located evenly in the sensor field, the number of member nodes of each cluster will be proportional to the size of the cluster, which will determine the amount of the initial energy of each cluster. In order to make sure that symmetric cluster works well, we need to compare the FDC (First Died Cluster) and LDC (Last Died Cluster) to the energy consumption of the nearest cluster and the farthest cluster from the sink.

When the symmetric cluster configuration is made depending on the location of the sink in a cluster and the number of round is set to C of Eq.3 for the replacement of the energy threshold-based cluster head, the energy consumption of the entire network is Eq. 10.

$$E_{total} = \sum_{\mathrm{i}=0}^{\mathrm{rnd}}\{k \cdot E_{cluster} + E_{fw} \cdot \sum_{i=0}^{L-1}(L-i)(2^{L-i}) - k \cdot C \cdot E_{rnd}\} \tag{10}$$

In addition, the energy required to pass data to any cluster of n hops away from the sink is Eq. 11.

$$E_{(CH(n),BS)} = \sum_{i=0}^{n+1}\left\{lE_{elec}\left(\frac{N}{k}-1\right) + lE_{DA}\frac{N}{k} + lE_{elec} + l\varepsilon_{fs}d_{toPC}^{2}\right\} \tag{11}$$

Parameters for the simulation are equal to [4], and the basic energy consumption is calculated from Eqs. 10 and 11.

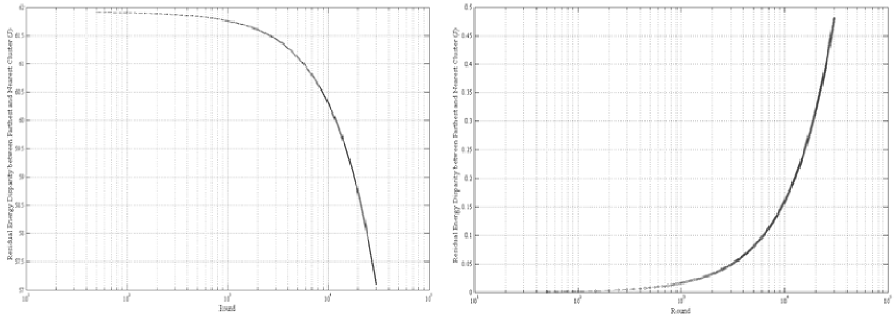


Fig. 2. Residual Energy Disparity between Farthest and Nearest Cluster (left: Proposed Clustering Protocol, right: Same-sized Clustering Protocol)

The left side of Fig. 2 is a graph showing the difference between the residual energy of the nearest cluster and the farthest cluster from the sink. In this graph, the residual energy of the two clusters shows big difference in the early stage, but you can see the difference decreases as time passes. Therefore, difference of the FDC(First Died Cluster) and the LDC(Last Died Cluster) can ultimately be minimized. The other hand, right side of Fig. 2 shows the difference between the residual energy of the nearest cluster and the farthest cluster when you build a cluster of the same size. Also you can see that the difference of the residual energy of the two clusters grows over time, which can cause an imbalance in the network with the difference of the FDC time and the LDC time increased.

5 Conclusions

This paper showed the study on adaptive clustering method to minimize energy consumption and guarantee the same life span for all sensor nodes considering the change of sensor location. Because the forwarding energy consumption is concentrated at the cluster nearest from the sink, the clustering has to be implemented considering the forwarding energy consumption. Because all clusters are constructed in hierarchical trees, the tree level has to be set considering the cluster size to prevent the cluster size from getting bigger and to block concentrated energy consumption in a specific cluster. For an ideal clustering mechanism considering the sink location change, relating to the time cycle to detect the change of the sink location, the cluster resetting interval and the energy consumption required to reconstruct clusters must be additionally considered.

References

1. Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: Proc. of 33rd Hawaii International Conference on System Science, vol. 8, p. 8020 (2000)
2. Heinzelman, W.B.: An Application-Specific Protocol Architectures for Wireless Networks. IEEE Transactions on Wireless Communications 1(4), 660–670 (2002)
3. Li, C., Ye, M., Chen, G., Wu, J.: An energy-efficient unequal clustering mechanism for wireless sensor networks. In: Proceedings of the 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2005 (2005)
4. Manjeshwar, A., Agrawal, D.P.: TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: Proc. of 15th International Conference on Parallel and Distributed Processing Symposium, pp. 2009–2015 (2001)
5. Handy, M.J., Haase, M., Timmermann, D.: Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In: 4th IEEE International Conference on Mobile and Wireless Communications Networks, pp. 368–372 (2002)
6. Jiman, H., Joongjin, K., Sangjun, L., Dongseop, K., Sangho, Y.: T-LEACH: The method of threshold-based cluster head replacement for wireless sensor networks. Information Systems Frontiers 11, 513–521 (2009)

A Study of Fire Refuge Guide Simulator Based on Sensor Networks

Jun-Pill Boo[1], Sang-Chul Kim[2], Dong-Hwan Park[3],
Hyo-Chan Bang[3], and Do-Hyeun Kim[1]

[1] Dept. of Computer Engineering, Jeju National University, Jeju, Republic of Korea
kimdh@jejunu.ac.kr
[2] School of Computer Science, Kookmin University, Seoul, Republic of Korea
sckim7@kookmin.ac.kr
[3] Dept. of Office, Eletronics & Telecommunication Research Institute, Republic of Korea
{bangs,dhpark}@etri.re.kr

Abstract. Recently, fire accidents in large buildings usually cause in tragic consequences. Many buildings are currently equipped with modern fire detection systems for preventing such accidents, but these support no clues as to how to escape. A lot of evacuation systems are aiming at providing more efficient means for alarming and guiding people. The evacuee guidance simulator provides to give clear navigation to nearest exit. This simulator must generate virtual objects such as sensors and guidance lights, connects it with nodes, inputs necessary information, and sets up virtual building environments. In this paper, we present the architecture for creating virtual objects to develop an evacuation guidance simulator. This simulator supports to evacuate people from the disaster area or the outbreak of fire with rapidity and safety for the virtual condition that a disaster or fire occurs in a large scale building.

Keywords: Fire evacuation guidance, Virtual sensor, Simulator.

1 Introduction

Recently, there has been an increasing trend in human-centric, ubiquitous, subliminal computing environments, which pervade everyday life and enable them to enjoy an easy and convenient life. The ubiquitous computing technology has been combined with various fields including medicine, education, architecture and environment, and thus has provided ubiquitous services. Particularly in architecture, the ubiquitous computing technology has been applied to the development of new intelligent system. The existing system is to monitor the states of buildings or bridges in real time, but the new intelligent system is expected not only to be aware of emergencies but to cope with them.

As buildings have been larger and taller, it has been more difficult to cope with emergencies. Actually, disasters in such buildings have inflicted huge losses of both life and property. In this regard, it is urgent to develop an effective evacuee guidance system. This paper designs virtual sensors and guidance lights for an evacuee

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 914–919, 2013.

guidance simulator, which aims at enabling the simulator to locate the spot a fire broke out, making a connection with sensor network, as well as to lead people to make a detour to avoid the fire and take shelter in a safe place. Here at, it needs to develop user interface as regards the node generator and the node connection.

The rest of this paper is structured as follows. In Section 2, we describe our proposed architecture for creating virtual sensor and guidance light, and show how our design addresses. Finally we conclude in Section 3.

2 Architecture for Creating Virtual Objects in Fire Refuge Guidance Simulator

It generates building environment-related information to execute the evacuee guidance simulation, and structure-related information to guide evacuees into the safe route. Node-related information shows guidance light and sensors. The guidance light nodes are divided into leading lights, emergency exits on respective floors, the final emergency door and the safety zone. A leading light is four directions (upper, lower, right and left). Sensor nodes are divided into sensor and sink. The guidance light consists of a structure of a building, which provides information necessary for the execution of the evacuee guidance algorithm. Fig. 1 shows building environment-related information that is necessary for the fire refuge guidance simulation.

▪ Input data for building layer ▪ Floor number and name ▪ Building floor blueprint	▪ Floor blueprint ▪ Floor number ▪ Floor name
▪ Input data for building node ▪ Id, Name, Coordination, Type ▪ Floor information, Direction ▪ Fire status ▪ Fire alert range	▪ Leading light ▪ Emergency exit ▪ Final exit ▪ Safety zone ▪ Fire Sensor
▪ Connection data for light and sensor ▪ Light and sensor connection ▪ Sensor and sensor connection ▪ Exit and exit connection ▪ Exit and light connection	▪ Connection information ▪ Distance information ▪ Fire information

Fig. 1. Information related to fire refuge guidance simulation in building environment

The creating module for virtual sensor is a structure of a building, which provides information necessary for the execution of the evacuee guidance algorithm. It does not provide information directly for the algorithm, but the sensor cannot be aware of fire if it is not connected with the guidance light. Fig. 2 shows the architecture for creating virtual sensor for monitoring fire in building.

Information related to guidance light is the identification number of every guidance light, their coordinate (x, y), information to show the shortest route (upper, lower, right and left), information on the types of guidance light (leading lights, emergency exits, the final emergency door and the safety zone), image information, floor-related information to build up a network with the building, fire information, administrator information to manage guidance light, and registration information to set up guidance light. Fig. 3 shows the information related to guidance light.

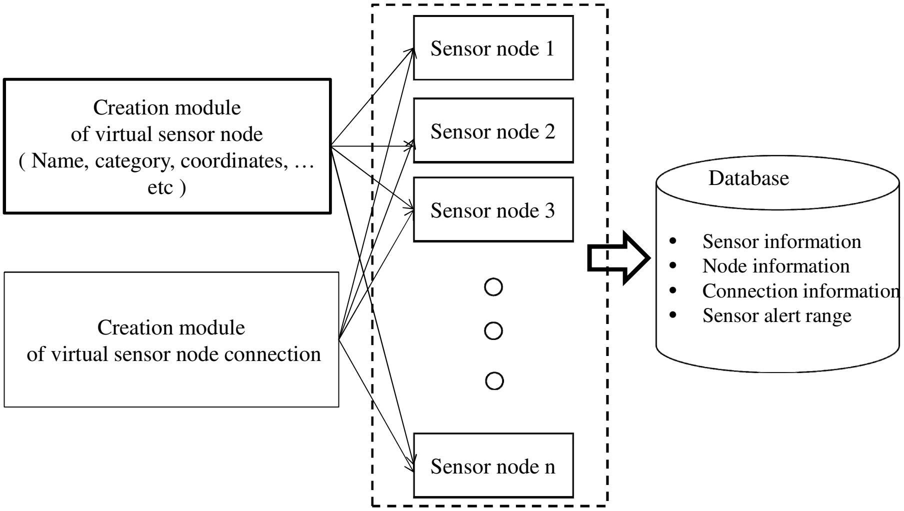


Fig. 2. Architecture for creating virtual sensors

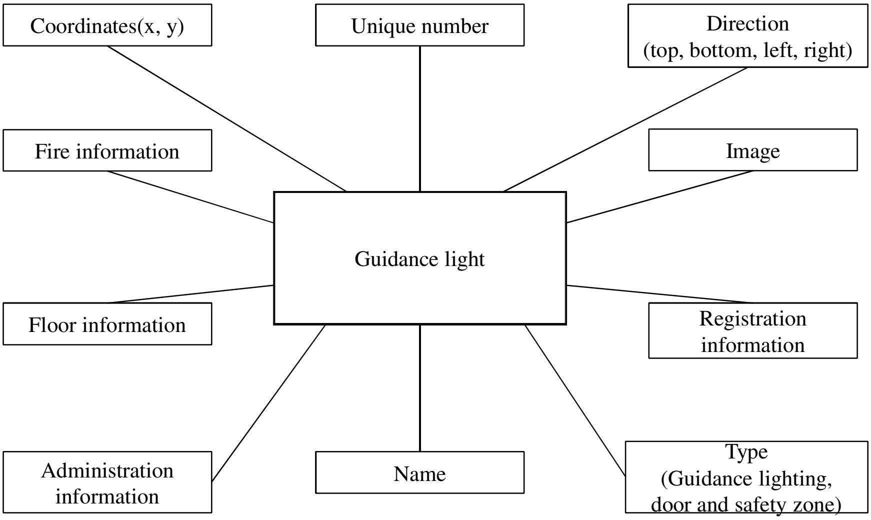


Fig. 3. Information related to virtual guidance light

Guidance light-related information is generated as the creation module of virtual guidance light nodes. In this case, the information shows the names of guidance light, their directions and coordinates, floors and the outbreak of fire. In the building-related information generating module, there is ‘floor selection’ on the menu bar, and information on the chosen floor is provided for the guidance light. In respect to the outbreak of fire, the guidance light is so initialized that it may give an answer “No.” But it gives an answer “Yes” when the sensor, connected to the guidance light, is aware of a fire. Fig. 4 shows the architecture for creating virtual guidance lights.

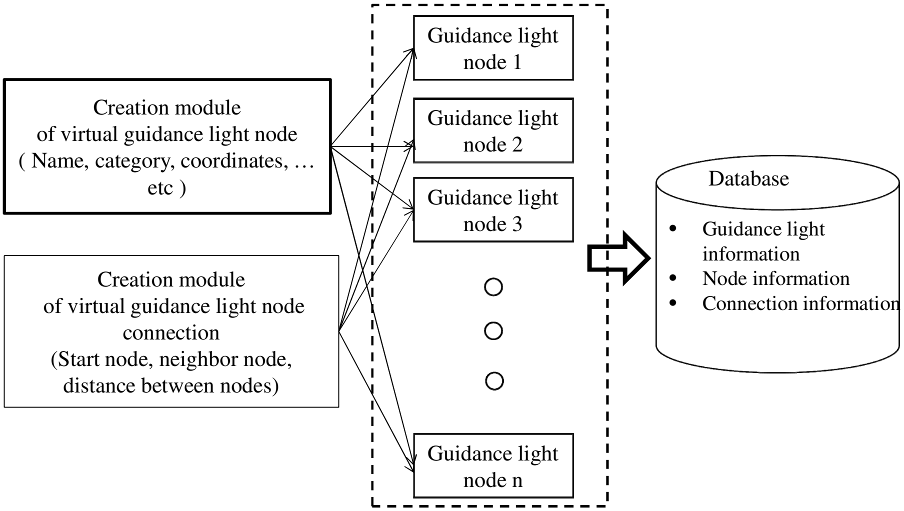

Creation module
of virtual guidance light node
(Name, category, coordinates, … etc)
Creation module
of virtual guidance light node connection
(Start node, neighbor node, distance between nodes)
Guidance light node 1
Guidance light node 2
Guidance light node 3
Guidance light node n
Database
• Guidance light information
• Node information
• Connection information

Fig. 4. Architecture for creating virtual guidance lights

The execution of the evacuee guidance algorithm needs information on the connection between guidance light, and the judgment of fire outbreak needs information on the connection between the guidance light and the sensor.

Information, related to the connection between guidance light, shows the identification numbers of a starting guidance light node and a neighboring one and on their distance. As regards information on the connection between guidance light, two guidance lights are chosen by a double click on ‘guidance light control’ in the building-related information creating module. The information shows the identification numbers of the starting node and the neighboring one and their distance. A guidance light can be connected not only with another one but with many ones at the same time. Fig. 4 shows the architecture for creating virtual guidance lights.

Information, related to guidance light-sensor connection, is generated by a double click on ‘guidance light control’ and on ‘sensor control’ in the structure-related information generating module. The information shows the identification numbers of the guidance light and the neighboring sensor. A guidance light can be connected not only with a sensor but with many sensors, and vice versa. Fig. 5 shows the architecture for connecting between virtual guidance light and sensor nodes. The information related to guidance light-sensor connection, i.e., their identification numbers, is saved in the mapping table between virtual guidance light and sensor nodes.

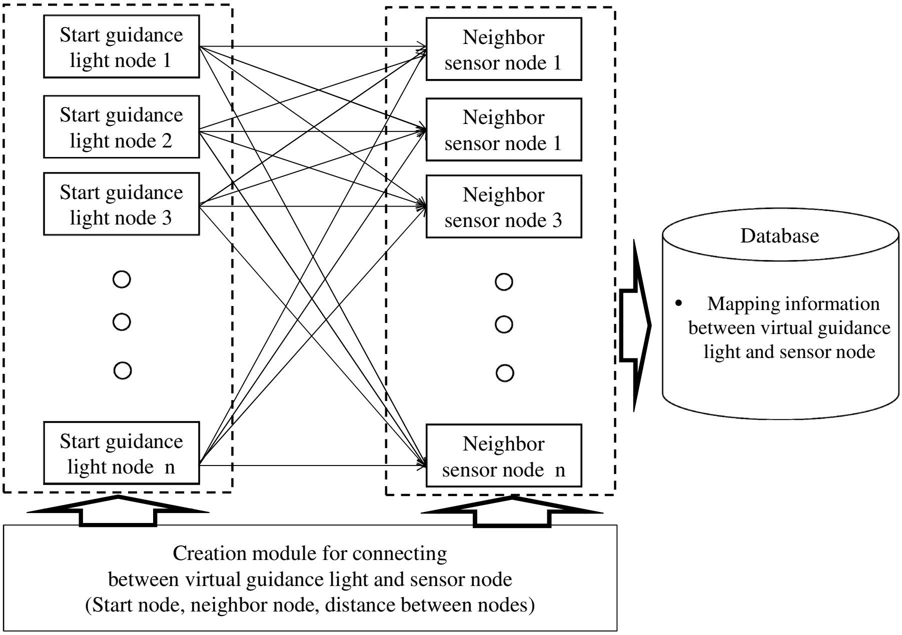

Start guidance light node 1
Start guidance light node 2
Start guidance light node 3
Start guidance light node n
Neighbor sensor node 1
Neighbor sensor node 1
Neighbor sensor node 3
Neighbor sensor node n
Database
• Mapping information between virtual guidance light and sensor node
Creation module for connecting between virtual guidance light and sensor node (Start node, neighbor node, distance between nodes)

Fig. 5. Architecture for connecting between virtual guidance light and sensor nodes

3 Conclusions

The evacuee guidance simulator evaluates to evacuate people from the disaster area or the outbreak of fire with rapidity and safety for the virtual condition that a disaster or fire occurs in a large scale building. This paper proposes architecture for creating the virtual objects such as sensors and guidance lights of this simulator. This simulator is expected to minimize loss of life in case of an emergency in a large scale building.

Acknowledgments. This research was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2011(Grants No. 0045590) and the Technology Innovation Program funded by the Ministry of Knowledge Economy of Korea (grant number: 10042421). This work was supported by the Industrial Strategic Technology Development Program funded by the Ministry of Knowledge Economy (MKE, Korea). [10038653, Development of Semantic based Open USN Service Platform]. (No. 2011-0015009). Corresponding author; DoHyeun Kim (e-mail: kimdh@jejunu.ac.kr.).

References

1. Yuan, W., Hai, T.K.: A novel algorithm of simulating multi-velocity evacuation based on cellular automata modeling and tenability condition. Physica A 379, 250–262 (2007)
2. Shi, P., Zlatanova, S.: Evacuation Route Calculation of Inner Buildings. In: Geoinformation for Disaster Management, pp. 1143–1161. Springer, Heidelberg (2005)

3. Gillieron, P., Merminod, B.: Personal navigation system for indoor applications. In: 11th IAIN World Congress (2003)
4. Gwynne, S., Galea, E.R., Lawrence, P.J., Filippidis, L.: Modeling Occupant Integration with Fire Conditions Using the Building EXODUS Evacuation Model. Fire Safety Journal 36(4), 327–357 (2001)
5. Kim, H.S., Kang, J.E., Jung, S.H.: A Study on the Guidance System for Fire Escape using WSN. Journal of Korean Institute of Information Scientists and Engineers 1(1), 58–61 (2010)
6. Tavares, R.M.: Evacuation Processes Versus Evacuation Models. Fire Technology 45, 419–430 (2009)
7. Shi, P., Zlatanova, S.: Evacuation Route Calculation of Inner Buildings. In: Geoinformation for Disaster Management, pp. 1143–1161. Springer, Heidelberg (2005)
8. Gillieron, P., Merminod, B.: Personal navigation system for indoor applications. In: 11th IAIN World Congress (2003)

Design of Parallel Pipelined Algorithm for Field Arithmetic Architecture Based on Cellular Array

Kee-Won Kim[1] and Jun-Cheol Jeon[2,*]

[1] Department of Software Science at Dankook University, Yongin, Korea
nirkim@gmail.com
[2] Department of Computer Engineering at Kumoh National Institute of Technology, Gumi, Korea
jcjeon@kumoh.ac.kr

Abstract. In this study, we present an efficient finite field arithmetic algorithm for multiplication which is a core algorithm for division and exponentiation operations. In order to obtain a dedicated pipelined algorithm, we adopt Montgomery algorithm and cellular systolic array. First of all, we select an effective Montgomery factor for the design of our parallel algorithm, then we induce an efficient multiplication algorithm from the typical binary MM algorithm using the factor. In this paper, we show the detail derivation process in order to obtain the recursive equations for pipelined computation.

Keywords: Finite field, Montgomery algorithm, Cellular systolic array, Cryptography.

1 Introduction

Arithmetic operations in the finite field GF(2^m) have recently been applied in a variety of fields, including cryptography and error-correcting codes [1]. Plus, a number of modern public-key cryptography systems and schemes, for example, Diffie–Hellman key pre-distribution, the Elgamal cryptosystem, and Elliptic Curve Cryptosystem, require the operations of division, exponentiation, and inversion, which are normally implemented using an AB or AB^2 multiplier [2]. However, a fast multiplication architecture with low complexity is still needed to design dedicated high-speed circuits.

One of most interesting and useful advances in this realm has been the Montgomery multiplication(MM) algorithm, introduced by Montgomery [3] for fast modular integer multiplication. The multiplication was successfully adapted to finite field GF(2^m) by Koc and Acar [4]. In [5], MM is implemented using systolic arrays for all-one polynomials and trinomials. Recently, in [6], they have considered concurrent error detection for MM over binary field. Three different multipliers, namely the bit-serial, digit-serial, and bit-parallel multipliers, have been considered and the concurrent error detection scheme has been derived and implemented for each of them.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 920–925, 2013.

Recently, Huang et al. [7] proposed the semi-systolic polynomial basis multiplier over GF(2^m) to reduce both space and time complexities. Also they proposed the semi-systolic polynomial basis multipliers with concurrent error detection and correction capability. Kim et al.[8] proposed much faster multiplier than the architecture proposed in [7]. They proposed a two-fold architecture so that two different architectures are operated at the same time.

In this paper, we induce an efficient multiplication algorithm for reduction of hardware complexity of typical architectures. The proposed algorithm enables multiplication to operate in pipelined computation so that two different operands can be computed in the same hardware architecture.

2 Montgomery Multiplication on GF(2^m)

GF(2^m) is a kind of finite field [9] that contains 2^m different elements. This finite field is an extension of GF(2) and any $A \in$ GF(2^m) can be represented as a polynomial of degree $m-1$ over GF(2), such as

$$A = a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + \cdots + a_1 x + a_0,$$

where $a_i \in \{0,1\}$, $0 \leq i \leq m-1$.

Let x be a root of the polynomial, then the irreducible polynomial G is represented as a following equation.

$$G(x) = g_m x^m + g_{m-1}x^{m-1} + \cdots + g_1 x + g_0, \quad (1)$$

where $g_i \in$ GF(2), $0 \leq i \leq m-1$.

Let α and β be two elements of $GF(2^m)$, then we define $\gamma = \alpha \cdot \beta \bmod G$, where G denotes $G(x)$. Also, let A and B be two Montgomery residues, then they are defined as

$$A = \alpha \cdot R \bmod G = a_0 + a_1 x + \cdots + a_{m-2}x^{m-2} + a_{m-1}x^{m-1} \text{ and} \quad (2)$$

$$B = \beta \cdot R \bmod G = b_0 + b_1 x + \cdots + b_{m-2}x^{m-2} + b_{m-1}x^{m-1}, \quad (3)$$

where a Montgomery factor, R and an irreducible polynomial, G are relatively prime, and gcd(R,G)=1. Then, the Montgomery Multiplication algorithm over $GF(2^m)$ can be formulated as [2]

$$P = A \cdot B \cdot R^{-1} \bmod G, \quad (4)$$

where R^{-1} is the inverse of R modulo G, and $R \cdot R^{-1} + G \cdot G' = 1$.

Then, (4) can be expressed as the following by the definition of the Montgomery residue as shown in (2) and (3).

$$P = (\alpha \cdot R) \cdot (\beta \cdot R) \cdot R^{-1} \bmod G = \gamma \cdot R \bmod G. \quad (5)$$

It means that P is the Montgomery residue of γ. This makes it possible to convert the operands to Montgomery residues once at the beginning, and then, do several consecutive multiplications/squarings, and convert the final result to the original representation. The final conversion is a multiplication by R^{-1}, i.e., $\gamma = P \cdot R^{-1}$ mod G. The polynomial R plays an important role in the complexity of the algorithm as we need to do modulo R multiplication and a final division by R.

3 Proposed Algorithm

Based on the property of parallel architecture, we choose the Montgomery factor, $R = x^{\lfloor m/2 \rfloor}$. Then, the Montgomery multiplication over GF(2^m) can be formulated as

$$P = A \cdot B \cdot x^{-\lfloor m/2 \rfloor} \bmod G\,. \tag{6}$$

We know that x is a root of $G(\omega)$ given by (1), i.e., $G(x)=0$ and $g_m = g_0 = 1$ over all irreducible polynomials. Thus, (1) can be rewritten as the followings.

$$g_m x^m + g_{m-1}x^{m-1} + g_{m-2}x^{m-2} + \cdots + g_1 x + g_0 = 0\,. \tag{7}$$

$$x^m \bmod G = g_{m-1}x^{m-1} + g_{m-2}x^{m-2} + \cdots + g_1 x + 1\,. \tag{8}$$

Also, (7) multiplied by x^{-1} is computed as (9).

$$x^{-1} \bmod G = x^{m-1} + g_{m-1}x^{m-2} + \cdots + g_2 x + g_1\,. \tag{9}$$

Meanwhile, (10) is represented by substituting (3) in place of B on (6).

$$\begin{aligned}
P &= [b_0 A + b_1 Ax + \cdots + b_{m-2}Ax^{m-2} + b_{m-1}Ax^{m-1}]x^{-\lfloor m/2 \rfloor} \bmod G \\
&= [b_0 Ax^{-\lfloor m/2 \rfloor} + b_1 Ax^{-\lfloor m/2 \rfloor+1} + \cdots + b_{\lfloor m/2 \rfloor-2}Ax^{-2} + b_{\lfloor m/2 \rfloor-1}Ax^{-1} + \\
&\quad + b_{\lfloor m/2 \rfloor}A + b_{\lfloor m/2 \rfloor+1}Ax + \cdots + b_{m-2}Ax^{m-\lfloor m/2 \rfloor-2} + b_{m-1}Ax^{m-\lfloor m/2 \rfloor-1}] \bmod G \\
&= [b_{\lfloor m/2 \rfloor-1}Ax^{-1} + b_{\lfloor m/2 \rfloor-2}Ax^{-2} + \cdots + b_1 Ax^{-\lfloor m/2 \rfloor+1} + b_0 Ax^{-\lfloor m/2 \rfloor} + \\
&\quad + b_{\lfloor m/2 \rfloor}A + b_{\lfloor m/2 \rfloor+1}Ax + \cdots + b_{m-2}Ax^{\lceil m/2 \rceil-2} + b_{m-1}Ax^{\lceil m/2 \rceil-1}] \bmod G.
\end{aligned} \tag{10}$$

Now, it expresses that P can be divided into two parts. One is based on the negative powers of x and the other is based on the positive powers of x. (10) can be denoted by

$$P = C + D\,, \tag{11}$$

where

$$C = [b_{\lfloor m/2 \rfloor-1}Ax^{-1} + b_{\lfloor m/2 \rfloor-2}Ax^{-2} + \cdots + b_1 Ax^{-\lfloor m/2 \rfloor+1} + b_0 Ax^{-\lfloor m/2 \rfloor}] \bmod G \quad \text{and} \tag{12}$$

$$D = [b_{\lfloor m/2 \rfloor}A + b_{\lfloor m/2 \rfloor+1}Ax + \cdots + b_{m-2}Ax^{\lceil m/2 \rceil-2} + b_{m-1}Ax^{\lceil m/2 \rceil-1}] \bmod G. \tag{13}$$

Meanwhile, let $\overline{A}^{(i)}$ and $A^{(i)}$ be Ax^{-i} modG and Ax^{i} modG, respectively. Then the equations can be expressed as

$$\overline{A}^{(i)} = \sum_{j=0}^{m-1} \overline{a}_j^{(i)} x^j = \overline{a}_0^{(i)} + \overline{a}_1^{(i)} x + \cdots + \overline{a}_{m-2}^{(i)} x^{m-2} + \overline{a}_{m-1}^{(i)} x^{m-1} \text{ and} \tag{14}$$

$$A^{(i)} = \sum_{j=0}^{m-1} a_j^{(i)} x^j = a_0^{(i)} + a_1^{(i)} x + \cdots + a_{m-2}^{(i)} x^{m-2} + a_{m-1}^{(i)} x^{m-1}, \tag{15}$$

where $\overline{A}^{(0)} = A^{(0)} = A$.

By using (8) and (9), (14) and (15) are rewritten as

$$\begin{aligned}
\overline{A}^{(i)} &= x^{-1}\overline{A}^{(i-1)} \bmod G \\
&= x^{-1}(\overline{a}_0^{(i-1)} + \overline{a}_1^{(i-1)} x + \cdots + \overline{a}_{m-2}^{(i-1)} x^{m-2} + \overline{a}_{m-1}^{(i-1)} x^{m-1}) \bmod G \\
&= (\overline{a}_0^{(i-1)} x^{-1} + \overline{a}_1^{(i-1)} + \overline{a}_2^{(i-1)} x + \cdots + \overline{a}_{m-2}^{(i-1)} x^{m-3} + \overline{a}_{m-1}^{(i-1)} x^{m-2}) \bmod G \\
&= \overline{a}_0^{(i-1)}(g_1 + g_2 x + \cdots + g_{m-1} x^{m-2} + x^{m-1}) + \overline{a}_1^{(i-1)} + \overline{a}_2^{(i-1)} x + \cdots \\
&\quad + \overline{a}_{m-2}^{(i-1)} x^{m-3} + \overline{a}_{m-1}^{(i-1)} x^{m-2} \\
&= (\overline{a}_1^{(i-1)} + \overline{a}_0^{(i-1)} g_1) + (\overline{a}_2^{(i-1)} + \overline{a}_0^{(i-1)} g_2) x + (\overline{a}_3^{(i-1)} + \overline{a}_0^{(i-1)} g_3) x^2 + \cdots \\
&\quad + (\overline{a}_{m-2}^{(i-1)} + \overline{a}_0^{(i-1)} g_{m-2}) x^{m-3} + (\overline{a}_{m-1}^{(i-1)} + \overline{a}_0^{(i-1)} g_{m-1}) x^{m-2} + \overline{a}_0^{(i-1)} x^{m-1}
\end{aligned} \quad \text{and} \tag{16}$$

$$\begin{aligned}
A^{(i)} &= x A^{(i-1)} \bmod G \\
&= x(a_0^{(i-1)} + a_1^{(i-1)} x + \cdots + a_{m-2}^{(i-1)} x^{m-2} + a_{m-1}^{(i-1)} x^{m-1}) \bmod G \\
&= (a_0^{(i-1)} x + a_1^{(i-1)} x^2 + \cdots + a_{m-2}^{(i-1)} x^{m-1} + a_{m-1}^{(i-1)} x^{m}) \bmod G \\
&= a_0^{(i-1)} x + a_1^{(i-1)} x^2 + \cdots + a_{m-2}^{(i-1)} x^{m-1} \\
&\quad + a_{m-1}^{(i-1)}(1 + g_1 x + \cdots + g_{m-2} x^{m-2} + g_{m-1} x^{m-1}) \\
&= a_{m-1}^{(i-1)} + (a_0^{(i-1)} + a_{m-1}^{(i-1)} g_1) x + (a_1^{(i-1)} + a_{m-1}^{(i-1)} g_2) x^2 + \cdots \\
&\quad + (a_{m-3}^{(i-1)} + a_{m-1}^{(i-1)} g_{m-2}) x^{m-2} + (a_{m-2}^{(i-1)} + a_{m-1}^{(i-1)} g_{m-1}) x^{m-1},
\end{aligned} \tag{17}$$

where

$$\overline{a}_j^{(i)} = \overline{a}_{j+1}^{(i-1)} + \overline{a}_0^{(i-1)} g_{j+1},\ 0 \le j \le m-1 \text{ when } \overline{a}_m^{(i-1)} = 0 \text{ and } g_m = 1, \tag{18}$$

$$a_j^{(i)} = a_{j-1}^{(i-1)} + a_{m-1}^{(i-1)} g_j,\ 0 \le j \le m-1 \text{ when } a_{-1}^{(i-1)} = 0 \text{ and } g_0 = 1. \tag{19}$$

Also, using the formulae of $\overline{A}^{(i)}$ and $A^{(i)}$, the terms C and D in (11) are represented by the following equations.

$$\begin{aligned}
C &= [b_0 A x^{-\lfloor m/2 \rfloor} + b_1 A x^{-\lfloor m/2 \rfloor+1} + \cdots + b_{\lfloor m/2 \rfloor-2} A x^{-2} + b_{\lfloor m/2 \rfloor-1} A x^{-1}] \bmod G \\
&= z\overline{A}^{(0)} + b_{\lfloor m/2 \rfloor-1}\overline{A}^{(1)} + b_{\lfloor m/2 \rfloor-2}\overline{A}^{(2)} + \cdots + +b_1\overline{A}^{(\lfloor m/2 \rfloor-1)} + b_0\overline{A}^{(\lfloor m/2 \rfloor)}
\end{aligned} \quad \text{and} \tag{20}$$

$$D = [b_{\lfloor m/2 \rfloor} A + b_{\lfloor m/2 \rfloor+1} Ax + \cdots + b_{m-2} Ax^{\lceil m/2 \rceil-2} + b_{m-1} Ax^{\lceil m/2 \rceil-1}] \bmod G \qquad (21)$$
$$= b_{\lfloor m/2 \rfloor} A^{(0)} + b_{\lfloor m/2 \rfloor+1} A^{(1)} + \cdots + b_{m-2} A^{(\lceil m/2 \rceil-2)} + b_{m-1} A^{(\lceil m/2 \rceil-1)},$$

where $z = 0$.

The coefficients of C and D are produced by summing the corresponding coefficients of each term in (20) and (21), respectively. It means that c_j and d_j for $0 \le j \le m$-1 are represented as

$$c_j = z\overline{a}_j^{(0)} + b_{\lfloor m/2 \rfloor-1}\overline{a}_j^{(1)} + b_{\lfloor m/2 \rfloor-2}\overline{a}_j^{(2)} + \cdots + +b_1\overline{a}_j^{(\lfloor m/2 \rfloor-1)} + b_0\overline{a}_j^{(\lfloor m/2 \rfloor)} \text{ and} \qquad (22)$$

$$d_j = b_{\lfloor m/2 \rfloor} a_j^{(0)} + b_{\lfloor m/2 \rfloor+1} a_j^{(1)} + \cdots + b_{m-2} a_j^{(\lceil m/2 \rceil-2)} + b_{m-1} a_j^{(\lceil m/2 \rceil-1)}. \qquad (23)$$

Now, we obtain the following recurrence equations from (22) and (23).

$$c_j^{(i)} = \begin{cases} c_j^{(i-1)} + z\overline{a}_j^{(i-1)} & , i = 1 \\ c_j^{(i-1)} + b_{\lfloor m/2 \rfloor-i+1}\overline{a}_j^{(i-1)} & , 1 < i \le \lfloor m/2 \rfloor + 1, \end{cases} \qquad (24)$$

where $c_j^{(0)} = 0$ for $0 \le j \le m$-1 and $z = 0$, and

$$d_j^{(i)} = d_j^{(i-1)} + b_{\lfloor m/2 \rfloor+i-1} a_j^{(i-1)} \text{ , } 1 \le i \le \lceil m/2 \rceil \qquad (25)$$

where $d_j^{(0)} = 0$ for $0 \le j \le m$-1.

Finally, the mentioned equations, (18), (19), (24) and (25) can be generalized for the computation of our parallel architectures, where <*p*> denotes *p* modulo *m*.

$$c_{<m-j>}^{(i)} = \begin{cases} c_{<m-j>}^{(i-1)} + z\overline{a}_{<m-j>}^{(i-1)} & , i = 1 \\ c_{<m-j>}^{(i-1)} + b_{\lfloor m/2 \rfloor+1-i}\overline{a}_{<m-j>}^{(i-1)} & , 2 \le i \le \lfloor m/2 \rfloor + 1, \end{cases} \qquad (26)$$

$$\overline{a}_{m-1-j}^{(i)} = \overline{a}_{m-j}^{(i-1)} + \overline{a}_0^{(i-1)} g_{m-j} \text{ , } 1 \le i \le \lfloor m/2 \rfloor, \qquad (27)$$

where $0 \le j \le m-1$, $z = 0$, $\overline{a}_m^{(i-1)} = 0$, $g_m = 1$.

$$d_{<j-1>}^{(i)} = d_{<j-1>}^{(i-1)} + b_{\lfloor m/2 \rfloor-1+i} a_{<j-1>}^{(i-1)} \text{ , } 1 \le i \le \lceil m/2 \rceil, \qquad (28)$$

$$a_j^{(i)} = a_{j-1}^{(i-1)} + a_{m-1}^{(i-1)} g_j, \quad 1 \le i \le \lceil m/2 \rceil - 1, \qquad (29)$$

where $0 \le j \le m-1$, $a_{-1}^{(i-1)} = 0$, $g_0 = 1$.

Based on the proposed algorithms in (26) thru (29), the hardware architecture can be efficiently composed compared to the architecture in [8]. As you see the equations, equation pair (26) and (28) are induced very similarly. (27) and (29) are also induced as an identical type. It means two operations can be computed with the same structure. Thus the algorithms can reduce almost a half of hardware complexity compared to the architecture proposed in [8].

4 Conclusion

In this paper, we propose a parallel pipelined algorithm for Montgomery multiplication over finite fields. We induced an efficient algorithm which is highly suitable for the design of pipelined structures. Our algorithm enabled the computation to share the hardware architecture so that we expect that it reduce not only time complexity but also hardware complexity compared to the recent study. In addition, we expect that our algorithm can be efficiently used for various applications including crypto coprocessor design, which demand high-speed computation, for security purposes.

Acknowledgements. This paper was supported by Research Fund, Kumoh National Institute of Technology.

References

1. Hariri, A., Reyhani-Masoleh, A.: Concurrent Error Detection in Montgomery Multiplication over Binary Extension Fields. IEEE Trans. Computers 60(9), 1341–1353 (2011)
2. Jeon, J.C., Yoo, K.Y.: Montgomery exponent architecture based on programmable cellular automata. Mathematics and Computers in Simulation 79, 1189–1196 (2008)
3. Montgomery, P.: Modular Multiplication without Trial Division. Mathematics of Computation 44(170), 519–521 (1985)
4. Koc, C., Acar, T.: Montgomery Multiplication in GF(2^k). Designs, Codes and Cryptography 14(1), 57–69 (1998)
5. Lee, C.Y., Horng, J.S., Jou, I.C., Lu, E.H.: Low-Complexity Bit-Parallel Systolic Montgomery Multipliers for Special Classes of GF(2^m). IEEE Transactions on Computers 54(9), 1061–1070 (2005)
6. Hariri, A., Reyhani-Masoleh, A.: Concurrent Error Detection in Montgomery Multiplication over Binary Extension Fields. IEEE Trans. Computers 60(9), 1341–1353 (2011)
7. Huang, W.T., Chang, C.H., Chiou, C.W., Chou, F.H.: Concurrent error detection and correction in a polynomial basis multiplier over GF(2^m). IET Information Security 4(3), 111–124 (2010)
8. Kim, K.W., Jeon, J.C.: Finite Field Arithmetic Architecture Based on Cellular Array. Int'l Journal of Cyber-Security and Digital Forensics 1(2), 122–129 (2012)
9. Lidl, R., Niederreiter, H.: Introduction to Finite Fields and Their Applications. Cambridge Univ. Press (1986)

Follower Classification through Social Network Analysis in Twitter

Jae-Wook Seol*, Kwang-Yong Jeong, and Kyung-Soon Lee**

Dept. of Computer Science & Engineering, CAIIT, Chonbuk National University
{wodnr754,kyjeong0520}@naver.com, selfsolee@chonbuk.ac.kr

Abstract. Through 'Twitter', one of the Social Network Service, people can have relationships by using 'Follow', a function of Twitter. Every user has different purposes, so there are various 'Followers', These Followers follow somebody in favor of them or just to support them without reasons or to criticize or watch one's behavior or tweet(one's comments). In this paper, a Model is suggested that why they follow certain users by using network relations between followers. User's influential supporters and influential non-supporters are extracted and then supporters, neutrals, and non-supporters are classified by follower's retweet information, profile and recent tweet sentiment analysis. In order to verify this suggestion's validity, random 30,000 users who follow one of the 5 politicians are extracted to experiment. After the experiment, I got to know that supports from influential support-followers and influential non-support-followers and non-support-followers classification was effective.

Keywords: Twitter, Follower, follow network, user behavior, SNS.

1 Introduction

Twitter, as one of the representative SNS, enables people to express their own opinions or information. Experiments are actively progressing because users not only can exchange one's information that does not belong to the users but also the information is a lot faster than those in internet cafes or blogs[1,2,3,4].

As to findings related to analyzing user's acts, there are a finding[5] that analyzed user's acts through clickstream analysis, user's attribute classification[6] based on user's communication and acts and last users classification[7] in online political debating sites.

Target users have various followers. These followers follow them in favor of, to support with reasons or to watch their acts and comments with negative perspective. Therefore, it is informative to show that followers follow target users with what kind of purposes.

Through experimenting Tweet follower's acts, I used these 4 aspects. 1) Socially influential users have followers who support them very actively. These followers are

[*] Co-equally contributed.

[**] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 926–931, 2013.

seen as same group as those influential users. Also, there are followers who do not support the users. They are seen as opposite group as those influential users. 2) They tend to retweet to express empathies and deliver their opinions on their timeline tweets. So followers how have many retweets are quite influential. 3) The more followers, the more users who read the tweets. So there are many retweets and when an incident is mentioned, the ripple effect is big. 4) Profiles usually tend to show Tweeter user's interests. Profiles enable people to notice user's personalities.

This paper extracts influential followers through the number of followers and tweets that retweeted a lot mentioned by target users and suggest the method how to classify supporters, neutrals and non-supporters through analyzing follower's retweet information, profiles and recent tweet's feelings

In order to verify these suggested methods, up to 30,000 users profiles are randomly collected among target user's followers. Collected users are arranged in order of high number of followers and up to 600 recent tweets were collected from upper 1000 users who have the most followers. Experimental targets are 400 followers of each target user. After the experiment, we can know that classifying influential support followers, influential non-support followers and non-support followers is effective.

2 Follower Classification through Social Network Analysis Model

The order to classify the model classification, we extract target influential support followers and target influential non-support followers. By using them, we classify the followers of the target user into support and non-support. Here, followers that were not classified are classified by profiles, retweet information, recent tweet information.

2.1 Influential Support Followers and Influential Non-support Followers Extraction

Influential support followers tweet a lot about target users and these tweets are retweeted a lot and have many followers. Here, the definition of tweet is tweets that mentioned target users written by the users themselves. Not the tweets retweeted by other users. Here, influential support followers are written as ***LPFollower***(Latent Positive Follower). As an opposite concept, influential non-support followers are written as ***LNFollower***(Latent Negative Follower).

Next is the previous step that extract influential followers. As to random 30,000 followers of the target users, they are arranged in order of high number of followers. As to these arranged followers who are upper 1,000, we measure follower's influence(***Uinfluence***) by using the average number of retweets(***AvgRT***) that mentioned target users and tweet's rate(***Umention***) that mentioned target users.

Equation that gets the degree of user's influence is calculated as follows:

$$UInfluence(u) = AvgRT(u,t) \times UMention(u,t) \tag{1}$$

Where the degree of influence is equation(2) times equation(3) which is average value of retweets that mentioned target user among the tweets that follower u wrote and multiple value that the rate of tweet that mentioned target among tweets that users u wrote.

The equation of the average number of retweets which is the ones of follower's tweet for target user is calculated as follows:

$$AvgRT(u,t) = \frac{1}{N}\sum_{i=1}^{N} RT\ (u, tTweet_i) \qquad (2)$$

Where *t* is a target users, *u* is one of followers to the target user, *N* is the number of tweets mentioned the target among u's tweets and *tTweet* is a tweet that mentioned among follower u's tweets. If retweeted a lot, riffle effect becomes big.

Among target user's follower u's all tweets, the equation of tweet's rate that mentioned target user is calculated as follows:

$$Umention(u,t) = \frac{Tweet(u,tTweet)}{Tweet(u)} \qquad (3)$$

Where *Tweet(u)* is the total number of tweets by a follower u, *Tweet(u, tTweet)* is the number of the mentioned tweets that user u mention the target user in his/her tweets. High rate that mentioned target users means a big interest in target users.

Through the experiment, the extent of influence above 0.1 followers is observed as high influence. In order to classify extracted followers through the degree of influence into LPFollower and LNFollower, Retweet distribution was used.

Retweet distribution(***RTdtb***) is when follower A of target user writes tweets that mentioned target user, those interested in the target user retweet the tweet. The higher rate the target users' followers retweet, the more similar characteristic or more support the target users.

The retweet distribution equation that gets how many followers retweet is calculated as follows:

$$RTdtb(u,t) = \frac{RT(u,tTweet,tFollower)}{RT(u,tTweet)} \qquad (1)$$

Where $RT(u, tTweet)$ shows the number of retweets that mentioned target among follower u's tweets. *RT(u, tTweet, tFollower)* is the retweeted number of target user's followers among *RT(u, tTweet).*

Table 1. Target user's the number of LPFs and the number of LNFs

	Moon	**Park**	**Jeong**	**You**	**Ahn**
The number of LPFs	24	22	17	5	12
The number of LNFs	3	10	1	2	6

Table 1 shows the number of LPF and LNF of 5 target users.

2.2 Support Follower Classification by Using the User's Profiles, Retweet Information, Supporting Words

1. As retweet represents empathy, if LPFollowers extracted by 2.1 retweets target users, they classify as support followers, if LNFollwers retweets target users, they classify as non-support followers.

2. Users have a tendency to show interest, preference, hobbies on their profile. So in the profile, if the target users are mentioned, they are classified as support followers. Twitter users show who they like. If the name that they support and supporting words("Support", "Contribution", "Win", "Respect", "Cheering", "Hope") are shown, they are classified as support followers.

2.3 Support Follower and Non-support Follower Classification through Using Sentiment Analysis about the User's Recent Tweets

In order to know target user's recent feelings, we analyse their current 600 tweets. In the Tweet, RT is different from retweet so you can add some opinions in front of the tweet. If there is an opinion in front of RT and if it is positive, +1, if it is negative, -1 is given. Tweets that do not apply RT get +1 or -1 whether they support or not. The sum of tweet from -2 to 2 is neutral, over 3 is support, under -3 is non-support. The dictionary of politic collected key words(Saenuri party, Minju-united party, Moon Jae-In, Park Gun-Hye, politic from September 1 to 21 in 2012). All the words(about 150,000) that include window size 3 are arranged in order of high frequency. Upper 20,000 words were extracted.

3 Experiment Evaluation

3.1 Tweet Experiment Data

To verify validity of suggested method, the date from Aug 1 to 30 in 2012 and 5 users(Moon Jae-In, Park Gun-Hye, You Si-Min, Jeong Bong-Ju, Ahn Chul-Su) are selected and among the 30,000 random followers of target users, basic information is collected by using twitter API[8] and twitter4j[9]. Collected followers are arranged in order of the number of followers and upper 1,000 followers' 600 tweets were collected. Experimental objects are 400 followers of each target users.<Table 2>

Table 2. Tweet Experiment data

	The total number of followers	The number of followers in order of the high rank among extracted random up to 30,000	The number of tweets of selected followers
Moon	261,105	1,000	418,626
Park	223,070	1,000	381,954
Jeong	394,317	1,000	324,315
You	510,351	1,000	352,516
Ahn	98,566	1,000	177,023
Total	1,487,409	5,000	1,901,002

3.2 Evaluation

The result that each 400 followers of 5 target users are classified into support, neutral and non-support is same as Table 3. In order to build correct set, two people built their own correct set and compared if it is correct or not. The evaluation was evaluated by Precision, Recall, F1 score and Accuracy. Above 0.65 showed the greatest performance when retweet distribution above 0.65, 0.75 and 0.85 are evaluated by 5-fold cross validation. Table 3 shows the result of 400 followers when retweet distribution is above 0.65.

Table 3. The classified result of 400 followers

	Moon	Park	You	Jeong	Ahn	Average
precision	81.9%	84.1%	88.2%	78.8%	73.2%	**81.2%**
Recall	63.4%	68.6%	72.3%	72.8%	64.2%	**68.2%**
F1 score	72.4%	75.6%	78.9%	75.7%	68.4%	**74.2%**
Accuracy	79.6%	74.5%	86.7%	78.7%	75.0%	**78.9%**

3.3 Discussion

Table 4 shows the effectivess of support follower classification when using LPFollowers. Previous experiment classified followers that retweeted target user's tweets as supper followers.

Table 4. The Result of Support Classification though LPFollower

	The number of followers that actually support	The number of followers that retweeted LPFollwer's tweets	The number of followers that retweeted target user's tweets
Moon	149	95	53
Park	144	168	85
You	48	28	17
Jeong	192	173	71
Ahn	199	111	41

Table 5. The evaluation of support classification through profile analysis

	The number of followers that mentioned target users on the profile	The number of support followers that classified through profile analysis
Moon	7	0
Park	8	0
You	13	2
Jeong	2	1
Ahn	5	2

Table 5 shows that the number of followers that mentioned target users on the profiles.

4 Conclusion and Future Work

This paper suggested the model that classify target user's followers into support and non-support by using LPFollower and LNFollower. Followers that were not classified through this model are classified by profile, retweet information, recent tweet sentiment analysis. We now know that support, non-support extract was effective through LPFollower and LNFollower about 5 target users.

In the Future work, we are planning to classify the ways followers tweet.

Acknowledgements. This work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(MEST)"(2012R1A1A2044811).

This research was financially supported by the Ministry of Education, Science Technology (MEST) and Korea Industrial Technology Foundation (KOTEF) though the Human Resource Training Project for Regional Innovation. And it was also supported by the second stage of Brain Korea 21.

References

1. Xu, Z., Zhang, Y., Wu, Y., Yang, Q.: Modeling User Posting Behavior on Social Media. In: 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 545–554 (2012)
2. Kwak, H., Lee, C., Park, H., Moon, S.: What is Twitter, a Social Network or a News Media? In: 19th International Conference on World Wide Web, pp. 591–600 (2010)
3. Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., Zheng, Z.: Time is of the Essence: Improving Recency Ranking Using Twitter Data. In: 19th International Conference on World Wide Web, pp. 331–340 (2010)
4. Sankaranarayanany, J., Samety, H., Teitlery, B.E., Liebermany, M.D., Sperlingz, J.: TwitterStand: News in Tweets. In: 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 42–51 (2009)
5. Rao, D., Yarowsky, D., Shreevats, A., Gupta, M.: Classifying Latent User Attributes in Twitter. In: 2nd International Workshop on Search and Mining User-Generated Contents, pp. 37–44 (2010)
6. Benevenuto, F., Rodrigues, T., Cha, M., Almeida, V.: Characterizing User Behavior in Online Social Networks. In: 9th ACM SIGCOMM Conference on Internet Measurement Conference, pp. 49–62 (2009)
7. Malouf, R., Mullen, T.: Taking sides: user classification for informal online political discourse. Internet Research 18(2), 177–190 (2008)
8. Twitter Developers, `https://dev.twitter.com/`
9. Twitter4j, `http://twitter4j.org/`

An Initial Quantization Parameter Decision Method Based on Frame Complexity with Multiple Objectives of GOP for Rate Control of H.264

Yalin Wu and Sun-Woo Ko*

Department of Smart Information Systems, Jeonju University, Jeonju, Jeonbuk, Korea
{wuyalin,godfriend}@hanmail.net

Abstract. In this paper, we proposed an initial quantization parameter (*QP*) decision method based on frame complexity with multiple objectives of *GOP* (Group of Pictures) for H.264 rate control algorithm. Primarily, we choose four video sequences with different characteristics to constitute a sample space and find their optimal *initial QP*s which can guarantee to generate video sequences with consistent quality by minimizing the variation of *QP*s in a *GOP*, while ensuring the minimizing actual encoding bit rate closer to the target bit rate in various bit rates. And then we calculate the spatial characteristic of tested video sequences using the proportion of the number of complex MBs in the first I-frame. The optimal *initial QP*s are represented in a two-dimensional matrix form arranged in spatial characteristic and target bit rate according to proposed selection method of optimal *initial QP*. When any video sequence is given under any target bit rate, its spatial characteristic is calculated and mapped to one of four samples through the proposed mapping method. Finally, its optimal *initial QP* is determined by picking an element of matrix according to the mapped spatial characteristic and given target bit rate. Simulation results show that the proposed method achieves more obvious consistency in objective *PSNR*s and has secured encoding bit rate than noted algorithms.

Keywords: Initialization quantization parameter, Rate control, H.264, GOP.

1 Introduction

Rate control (*RC*) involves adjusting encoding parameters in order to achieve a target bit rate. The most obvious parameter to be adjusted is the quantization parameter (*QP*) since increasing *QP* reduces coded bit rate and vice versa. And the initialization of rate control is a very important section in the rate control strategy. It includes selecting an *initial QP* for the first instantaneous decoding refresh (*IDR*) picture in a video sequence. However, unfortunately, there are few works about how to decide the *initial QP* value of video sequences. In JVT-G012 [1], the *initial QP* is decided by the number of bits per pixel (*BPP*) which is determined according to bit rate, frame rate and frame resolution. However, this method does not take into account features and

* Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 932–938, 2013.

complexities of video sequences. Therefore, video sequences reconstructed after the method JVT-G012 [1] can be in low quality or the quality of reconstructed video sequences can be changed extremely. In order to make up this problem, Wu [2] propose to use the characteristics of video sequences as well as *BPP* to determine the *initial QP*. However, their methods do not consider reconstructed video sequence quality balance and the provided parameters cannot be applied any video sequence.

In order to solve the existing problems, we propose a novel method to decide an *initial QP* in any bit rate. Primarily, in order to realize the goal of proposed algorithm, we screen out four video sequences that are representative samples. And then we search their optimal *initial QP*s which are used to generate their reconstructed video sequences with high quality and consistent quality in a *GOP* at target bit rates in the range of 0.4 to 2.0 Mbps. And then we calculate the spatial characteristic of tested video sequences using the proportion of the number of complex *MB*s in the first I-frame. Afterwards, we use these optimal *initial QP*s to build a two-dimensional matrix according to spatial characteristic and bit rate of four sample video sequences. Moreover, we propose a method to map spatial characteristic of tested video sequences by spatial characteristic of four sample video sequences. For any video sequence, we can choose its *initial QP* by simply picking an element of the lookup table in a form of two-dimensional matrix according to its mapped spatial characteristic and given target bit rate.

2 Proposed Novel Method for Initial Quantization Parameter Determination

2.1 Spatial Characteristic of Tested Video Sequences

Primarily, the Bus, Flower, Waterfall and Foreman video sequences can play roles as representative samples, since they show relatively uncorrelated characteristics according to [2] [3] [6].

In this paper, the *initial QP* of *RC* is calculated according to the complexity of video sequences and given target bit rate. Due to the target bit rate is given, we only need to provide the computing method of spatial complexity of sample video sequences.

In H.264, the smallest encoding domain is *MB*. Since the variance of a *MB* corresponds to total energy of the *AC* coefficients of the *MB*, it can be used to measure the spatial complexity of the *MB*. Thus, we use the variance to identify the high and low complexity of the *MB* [5].

$$MB_{\text{variance}} = \frac{1}{256}(\sum_{i=0}^{15}\sum_{j=0}^{15}[Y(i,j)]^{2} - \frac{1}{256}\left[\sum_{i=0}^{15}\sum_{j=0}^{15}Y(i,j)\right]^{2}). \tag{1}$$

where the MB_{variance} is the variance of *MB* and $Y(i,j)$ is the luminance value of the pixel at (*i*, *j*). The complexity of an MB can be classified as high or low according to its variance as follow:

$$complexity = \begin{cases} high, \; MB_{\text{variance}} > T \\ low, \; MB_{\text{variance}} \leq T \end{cases}. \tag{2}$$

where T is threshold that is set to 92735 which is recommended number in [5]. Base on this idea, we can calculate the spatial complexity of the first I-frame according to the proportion of the number of complex *MB*s in the first I-frame as follow:

$$Frame_{Complex} = MB_{Complex} \big/ MB_{Frame} \times 100\%. \tag{3}$$

where the $Frame_{Complex}$ is the proportion of the number of complex MBs in the first I-frame. We can use this value to quantize the spatial complexity. The $MB_{Complex}$ is the number of the complex MBs of I-frame. The MB_{Frame} is the number of the MBs of I-frame.

2.2 Proposed Method for Initializing Quantization Parameters

On the basis of RC algorithm of H.264, the *initial QP* and target bit rate bear direct relevance for performance of encoding. The strategy of this proposed method is that the reconstructed video sequences have consistent and superior quality and the actual bit rate is the least and closer to target bit rate in various tested target bit rates by the optimal *initial QP*. In order to realize this algorithm, primarily, we obtain average *PSNR* and bit rate of front 60 frames of testing sample video sequences and the differences of *QP*s in a *GOP* of 52 *initial QP*s of given the target bit rate. The *PSNR*, bit rate and differences of QPs represent picture quality, amount of data and stationary quality of a *GOP*. Therefore, we can calculate the optimal *initial QP*s of given the target bit rate of all sample videos according to pick up the specific *initial QP* that can be used to generate the maximum *PSNR* and minimums of bit rate and differences of *QP*s. However, the *PSNR*, bit rate and difference of *QP* are not same magnitude. As a result, *PSNR*, bit rate and difference of *QP* should be respectively normalized. The process of normalization is expressed as follows:

$$NPSNR_{initial_{Qp}} = \frac{PSNR_{initial_{Qp}} - \underset{initial_{Qp}=0,\cdots,51}{MIN}(PSNR_{initial_{Qp}})}{\underset{initial_{Qp}=0,\cdots,51}{MAX}(PSNR_{initial_{Qp}}) - \underset{initial_{Qp}=0,\cdots,51}{MIN}(PSNR_{initial_{Qp}})}, Initial_{Qp} = 0,\cdots,51. \tag{4}$$

$$NBIT_{initial_{Qp}} = \frac{BIT_{initial_{Qp}} - \underset{initial_{Qp}=0,\cdots,51}{MIN}(BIT_{initial_{Qp}})}{\underset{initial_{Qp}=0,\cdots,51}{MAX}(BIT_{initial_{Qp}}) - \underset{initial_{Qp}=0,\cdots,51}{MIN}(BIT_{initial_{Qp}})}, Initial_{Qp} = 0,\cdots,51. \tag{5}$$

$$NDQp_{initial_{Qp}} = \frac{DQp_{initial_{Qp}} - \underset{initial_{Qp}=0,\cdots,51}{MIN}(DQp_{initial_{Qp}})}{\underset{initial_{Qp}=0,\cdots,51}{MAX}(DQp_{initial_{Qp}}) - \underset{initial_{Qp}=0,\cdots,51}{MIN}(DQp_{initial_{Qp}})}, Initial_{Qp} = 0,\cdots,51. \tag{6}$$

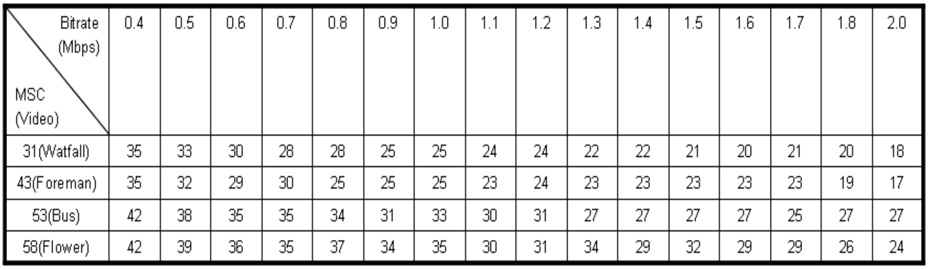
where $NPSNR_{initialQp}$, $NBIT_{initialQp}$ and $NDQp_{initial_{Qp}}$ are normalized *PSNR*, bit rate and differences of *QP*. $PSNR_{initialQp}$, $BIT_{initialQp}$ and $DQp_{initial_{Qp}}$ are the values that are before normalized. Afterwards, the selection algorithm of the optimal *initial QP* is designed as follow:

$$BestInitial_{Qp} = \mathop{\arg\min}_{initial_{Qp}=0,\cdots,51} (1/NPSNR_{initial_{Qp}} + NBIT_{initial_{Qp}} + NDQP_{initial_{Qp}}). \tag{7}$$

where the $BestInitial_{Qp}$ is the Optimal *initial QP* of given target bit rate of sample videos.

In this paper, we propose a matrix that contains 4 groups of the optimal *initial QP*s for 4 different type video sequences at the target bit rate ranged from 0.4 Mbps to 2.0 Mbps. For any video sequence, we can choose its *initial QP* by simply picking an element of the lookup table in Table 1 according to its mapped spatial complexity and given target bit rate.

Table 1. Lookup table determining optimal *Initial QP* of a video sequence according to its mapped spatial complexity and target bit rate (MSC:Mapped Spatial Complexity)



Bitrate (Mbps) MSC (Video)	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	2.0
31(Watfall)	35	33	30	28	28	25	25	24	24	22	22	21	20	21	20	18
43(Foreman)	35	32	29	30	25	25	25	23	24	23	23	23	23	23	19	17
53(Bus)	42	38	35	35	34	31	33	30	31	27	27	27	27	25	27	27
58(Flower)	42	39	36	35	37	34	35	30	31	34	29	32	29	29	26	24

2.3 The Method of Spatial Complexity Mapping and Proposed Method for Initialization

Based on a principle that is the similar performance of encoding can be obtained for video sequences that own similar complexity, we suggest a spatial complexity, which is computed by the proportion of the number of complex *MB*s in the first I-Frame, mapping method based on the Scalar Quantization. This mapping method maps proportion of any video sequence into the sample proportion space which consists of a finite set of proportion of sample video sequences. When any video sequence is given, its proportion is calculated. And then it is mapped by choosing the nearest matching proportion from a set of sample proportion space as follows:

$$Diff_i = |SampRate_i - TestRate|, \text{ for } i = 1, 2, \ldots, N. \tag{8}$$

$$MappedRate = SR_{Min(Diff_i)}. \tag{9}$$

where $Diff_i$ are the difference of $SampRate_i$, i^{th} sample proportion of N sample proportions (PS_i) and $TestRate$, the proportion of a test video sequence. We can get a proportion mapped into the test video sequence, $MappedRate$ using (9), where $SR_{Min(Diffi)}$ means the sample proportion, $SampRate_i$ that is the closest to $TestRate$.

3 Experimental Testing Results

The JM10.2 [6] that is a standard coding software tool of the H.264 is used to verify the proposed method. And Bus, Flower, Waterfall, Foreman, Mobile, Paris, Bridge-far and Silent video sequences with horizontal and vertical resolutions of 352 and 228 pixels, respectively, are used to compare the proposed method with the other two methods [1] and [2]. From experiments, four video sequences of Bus, Flower, Waterfall and Foreman are chosen to generate a sample space and calculate a set of sample spatial complexity space. And four video sequences of Mobile, Bridge-far, Paris and Silent are used test video sequences to check the generalization characteristic of the proposed method. Through the spatial complexity mapping method, Mobile is mapped to Bus, Bridge-far is mapped to waterfall, and Paris and Silent are mapped to Foreman.

The B-picture is not included due to the use of the H.264 baseline profile, and 15 pictures are configured as one *GOP* in which each video applies 60 pictures. The same number of slices is used for each picture and is determined by 18 along the vertical direction. The range of test target bit rates (units: Mbps) are from 0.4 to 2.0. And then we calculate the *initial QP* according to mapped spatial complexity and given target bit rate using Table 1.

$$PSNR = 10\log_{10}\left(2^n - 1\right)^2 \Big/ MSE. \tag{10}$$

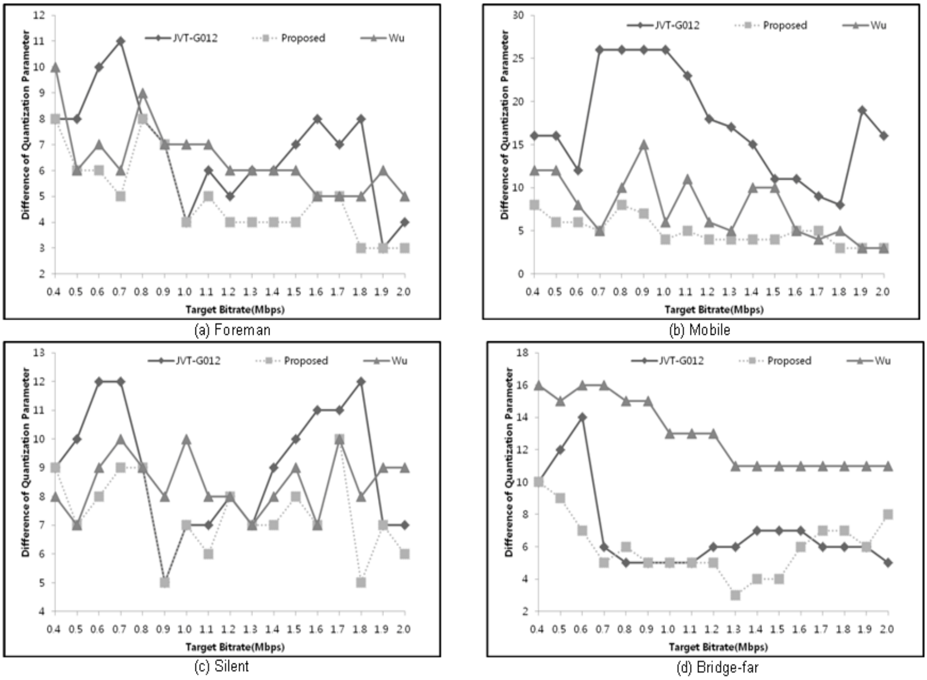
$$MSE = \sum_{x=0}^{X-1}\sum_{Y=0}^{Y-1}\left(\hat{p}(x,y) - p(x,y)\right)^2. \tag{11}$$

$$\Delta R = Rt - Rb. \tag{12}$$

where Rt is the actual bit rate of the proposed method, JVT-G012 [1] and the method of Wu [2], and Rb is the actual bit rate of JVT-G012 [1]. We average the $PSNR$ and the ΔR values obtained at various target bit rates for each of the test video sequences and list the averaged results in Table 2.

Table 2. Comparison of average $PSNR$ and ΔR

Video Squence	Average $PSNR\,(db)$		
	JVT-012	Wu	Proposed
Bus	33.11	33.12	33.17
Flower	32.14	32.17	32.27
Foreman	40.10	40.09	40.20
Waterfall	38.50	38.33	38.59
Mobile	31.01	31.15	31.12
Silent	41.17	41.29	41.39
Paris	37.60	37.72	37.98
Bridge-far	40.09	39.97	40.10



(a) Foreman

(b) Mobile

(c) Silent

(d) Bridge-far

Fig. 1. Result of difference of quantization parameter

From the average *PSNR* results in Table 2, we can see that the proposed method can achieve the better *PSNR* performance than the other two methods in all situations. And from the average ΔR result, we can see that the proposed method can achieve better bit rate performance than others in all situations. The average actual bit rate of the proposed method is the least and closer to the target bit rate.

Furthermore, the test video sequences can also achieve the best performance by means of optimal *initial QP*s that are chosen from the proposed 2-D lookup table by mapped spatial complexity and bit rate shown in Table 1. As a result, we can find out that the proposed method is more effective than others and it also has the generalization characteristic.

In particular, the prominent point of the proposed method is that it generates a reconstructed video sequence without extreme changes of quality in *GOP*. We use the difference of quantization parameter of images of *GOP* to express balance property of quality of reconstructed videos. The smaller value of difference of quantization parameter can indicate that the reconstructed video sequence quality is more consistent in *GOP* (Group of picture). Conversely, the reconstructed video sequence quality is not steady. In Fig. 1, we can see that the proposed method achieve the smaller difference of *QP*s than JVT-G012 [1] and Wu [2] in the most case.

4 Conclusions

In this paper, we proposed an initial quantization parameter (*QP*) decision method based on frame complexity with multiple objectives of *GOP* (Group of Pictures) for H.264 rate control algorithm. We choose four video sequences with different characteristics and find their optimal *initial QP*s as the target bit rate is from 0.4 Mbps to 2.0 Mbps. And then we calculate the spatial characteristic of tested video sequences using the proportion of the number of complex *MB*s in the first I-frame. The optimal *initial QP*s are represented in a two-dimensional matrix form arranged in spatial characteristic and target bit rate. Moreover, we propose a method of mapping spatial characteristic for any video sequences. And using this proposed matrix, we can decide initial quantization parameters for any test video sequences according to mapped spatial complexity and given target bit rate. In the experiment, four sample video sequences and four test video sequences are used. Experimental results demonstrate that the proposed method can achieve better *PSNR* performance within the bit rate constraint and more stable quality of reconstructed video sequences than the other two rate control initialization algorithms [2] and [4]. Our main contribution is that we propose a novel method of rate control initialization based on image quality balance of *GOP*.

Acknowledgements. This research is partially supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2012.

References

1. Li, Z., Pan, F., Lim, K.P., Feng, G.: Lin, X., et al.: Adaptive Basic Unit Layer Rate Control for JVT. In: Doc. JVT-G012, Thailand (March 2003)
2. Wu, W., Kim, H.K.: A Novel Rate Control Initialization Algorithm for H.264. IEEE Transactions on Consumer Electronics 55(2), 665–669 (2009)
3. YUV Video Sequences, `http://trace.eas.asu.edu/yuv/index.html`
4. Kown, S.K., Punchihewa, A., Bailey, D.G., Kim, S.W., Lee, J.: Adaptive Simplification of Prediction Modes for H.264 Intra-Picture Coding. IEEE Transactions on Broadcasting 58(1), 125–129 (2012)
5. Huang, Y.H., Ou, T.S., Chen, H.H.: Fast Decision of Block Size, Prediction Mode, and Intra Block for H.264 Intra Prediction. IEEE Transactions on Circuits and Systems for Video Technology 20(8), 1122–1132 (2010)
6. JM Reference Software Version 10.2, `http://iphome.hhi.de/suehring/download`

The Development of Privacy Telephone Sets in Encryption System against Eavesdropping

Seok-Pil Lee[1] and Eui-seok Nahm[2,*]

[1] Dept. of Digital Media Technology, Sangmyung University, Korea
esprit@smu.ac.kr

[2] Dept. of Ubiquitous Information and Technology, Far East University, Korea
nahmes@kdu.ac.kr

Abstract. We developed an encryption system using an AES encoding algorithm for the Internet telephone system security to prevent eavesdropping. It works through tapping by encoding/decoding the output data of the Internet telephone on the sending and receiving sides when the caller or callee is making an internet phone call. The developed encryption system has the merit of no deterioration of voice data not related to the encoding process. The privacy telephone set against eavesdropping was designed to prevent eavesdropping on internet telephones during communication and involved encoding/decoding the output data at the transmitter and receiver of internet telephones.

Keywords: Internet Telephone, Tapping, AES, Eavesdropping, Encryption System.

1 Introduction

The security of information is an issue owing to the wide range of information sharing through the Internet. Action to protect privacy and secret information is taken constantly [1, 2]. Specifically, the Internet telephone information can be completely exposed therefore can be monitored or tapped without special equipment, as it uses IP packet data. In the case of a PSTN (Public Switched Telephone Network), poaching is possible through only a physical approach. In contrast, in the case of the Internet telephone, even a distant attacker can easily distort the signaling message (SIP, Session Initiation Protocol, or H.323) and can even eavesdrop on the voice packets [2, 3, 4].

To protect from such an attack and to use internet telephone service safely, a security system should assure user certification, message certification, and voice data confidentiality [5-6]. This paper considered a digest based user certification defined by SIP standard to ensure the confidentiality of the voice data, and TLS (Transport Layer Security) among the hops for SIP messages when they are sent and received. Therefore, both ends of the phone would be supplied with confidentiality, integrity, and user certification of SIP messages when S/MIME is applied.

This paper has dealt particularly with measures against confidentiality infringement attacks.

[*] Corresponding author.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 939–944, 2013.

2 The Proposed System

The proposed encryption system is equipped at both the transmitter and receiver of an internet telephone. The encryption system does not have a self-IP address, which prevents outside from eavesdropping. Instead, it uses the IP address and the MAC address of the Internet telephone when carrying out communication. The encryption system uses key-exchange, utilizing the telephone MAC data and voice RTP (Real-time Transport Protocol) packets during internet telephone usage when the other IP cannot be exposed due to a firewall. Moreover, there is a gateway/firewall when each of the Internet telephone service users enrolls in a different service networks. Each of the encryption systems is provided with the other's public key and encodes with each session key used in cipher communication, utilizing the other public key and informing the other as well. Although the public key may be exposed to the outside, only the encryption system possesses the very private key. Therefore, the session key used in cipher communication cannot be exposed to a third party.

The proposed private telephone system shall be installed to the receiver and transmitter on the internet phone line so as to ensure a private telephone. In addition, the system should be developed without having its own IP in order to avoid exposure to eavesdropping. The private telephone set has two RJ-45 ports, one of which is connected to the internet phone and the other which is connected to an external network.

The system makes encryptions of the packets from the internet phone and transmits them to the other port while the private telephone set installed onto the receiver makes decryptions of the encrypted packet so as to deliver them to the internet phone of the receiver.

2.1 The Private Telephone Switch

The private telephone set has an installed switch that decides the private telephone function. Basically, two Ethernet ports of the set operate in bridge mode and execute encryption by only filtering RTP packets. The mode switch of the private telephone set decides the treatment process of the filtered RTP packets within the equipment.

As illustrated in Figure 1, the packets are encrypted in the On mode of the private telephone switch and the packets are bypassed by following the process in the Off mode. Figure 2 shows the flowchart of the private telephone mode switch.

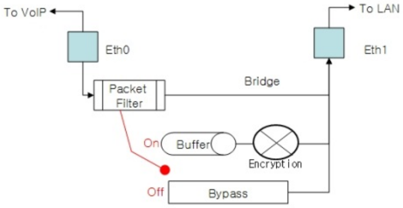


Fig. 1. Process map of private telephone mode

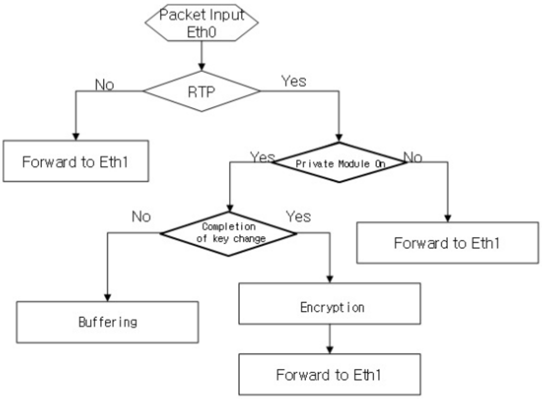


Fig. 2. The flowchart of the private telephone mode switch

2.2 Trusted Computing

The unique information of each private telephone set is stored by using the protected storage function which is one of features of trusted computing. The keys and data are protected by the RTS (Root of Trust for Storage) which accredits a small volatile storage and has recurred keys for encryption. The keys are managed by the KCM (Key Cache Manager).

2.3 Encryption Module

The encryption module shall be referred to in Figure 3. The major parts are the CPU, memory, the Trusted Platform Module, the RJ45 interface, and the LED screen. Storage is developed by using flash memory and application programs including embedded Linux. Also basic system information is stored in the flash memory.

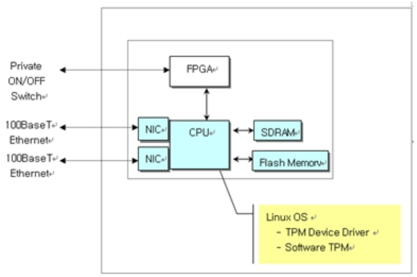


Fig. 3. Encryption module

3 Simulation

The internet telephone backbone system is as shown in Figure 4. The SIP server, which is in charge of call setup and the Internet telephone SBC (Session Board Controller), is responsible for the service and voice security of the Internet telephone using a private IP. These are the direct factors that are relevant to the encryption system operation. Of these factors, the SBC processing voice media RTP packet participates in session set-up of the RTP packets and routing. Furthermore, the RTP transmission is executed via the following two modes regardless of whether SBC is used.

- Direct Mode: The RTP packet connects each telephone directly.
- Redirect Mode: RTP packet transfer via SBC.

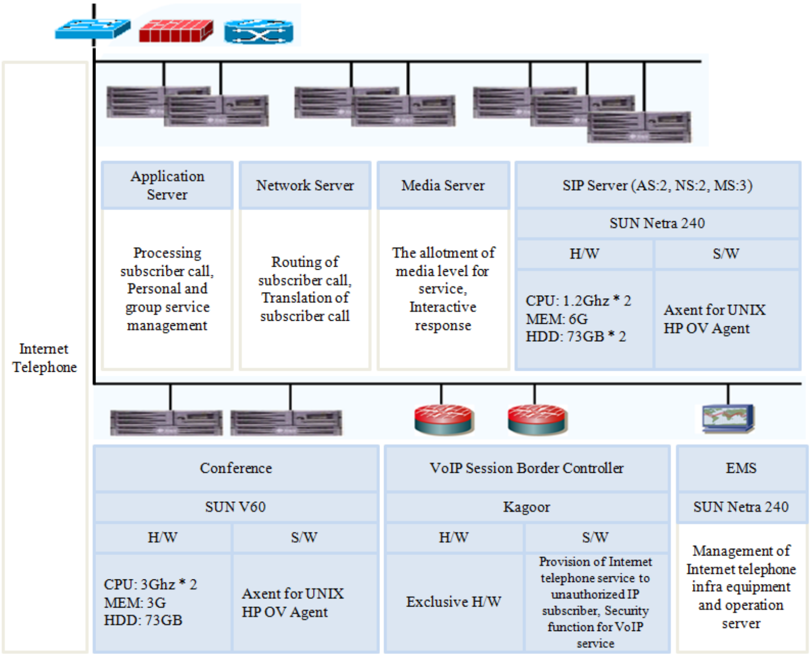

Internet Telephone
Application Server
Processing subscriber call, Personal and group service management
Network Server
Routing of subscriber call, Translation of subscriber call
Media Server
The allotment of media level for service, Interactive response
SIP Server (AS:2, NS:2, MS:3)
SUN Netra 240
H/W
S/W
CPU: 1.2Ghz * 2
MEM: 6G
HDD: 73GB * 2
Axent for UNIX
HP OV Agent
Conference
SUN V60
H/W
S/W
CPU: 3Ghz * 2
MEM: 3G
HDD: 73GB
Axent for UNIX
HP OV Agent
VoIP Session Border Controller
Kagoor
H/W
S/W
Exclusive H/W
Provision of Internet telephone service to unauthorized IP subscriber, Security function for VoIP service
EMS
SUN Netra 240
Management of Internet telephone infra equipment and operation server

Fig. 4. Internet telephone backbone constitution

In the Direct Mode, the RTP packet source/destination is always appointed to the opposite telephone IP. In contrast, in the Redirect Mode, the RTP packet source/destination is set to SBC IP 203.245.210.225.

Basic processing methods for RTP are Off-Net calls and On-Net calls.

In an Off-Net call (redirect mode), the route is IP Phone ↔ SBC ↔ Trunk Gateway ↔ PSTN.

In an On –Net call (direct mode), the route is IP Phone ↔ IP Phone.

When a private network is involved, sending and receiving is done via SBC (direct mode). Only in the circumstance of telephones that use many private IPs in only one officially approved IP and among telephones using a registered IP, RTP is transferred directly (direct mode).

This condition is applied only when the telephone is enrolled as a user of SBC equipment. If it is enrolled in another type of SBC, the work is done via SBC. For the service influence test of the private telephone set, the test is conducted using SAGE equipment. The differences in SIP messages were analyzed for each mode. Both cases were performed on the same LAN using a private IP.

The voice quality of the proposed system is equal to that of existing internet telephones and the transfer delay is minimized. Whether or not the proposed system affects the Internet telephone voice quality was verified through a two-sample T-test. The details of this are given below:

Table 1 shows the average value of the transfer delay (X_T) of 10 packets in two tests.

Table 1. The measured value of the transfer delay

No	With Cipher equipment	Without an encryption system
Test 1	1.094266667	0.16075
Test 2	1.09716667	0.160416667
Average	1.095716667	0.160583333

Without the encryption system, the average value of the transfer delay (X_T) took 0.16 msec, but with encryption system, it took about 1.09 msec. Processing delay of 0.07 msec is measured to process encryption and decryption algorithm, this value can be said to be less effective for this service, considering that it is 0.21 % of the entire propagation delay of 150 msec.

The test is conducted to measure the phone quality including the voice quality and the voice delivery delay time in cases of connection to general phones without the private telephone set. The same test is conducted in the cases of phones with the set. As a result, the test outcomes are as shown in Table 2.

Table 2. Test outcome on phone quality

Without the encryption module

MOS		Delay (RTD, ms) D1	Noise(dB)		Gain	
Near G1.1	Far G1.2		Near	Far	Near	Far
4.485	4.49	135.5	12	19.25	-7.9	1
4.49	4.49	138.1	13	21	-7	1
4.47	4.49	128	11	17	-8	1
6E-05	0	21.218	0.286	1.929	0.125	0
0.02	0	10.1	2	4	1	0

Table 2. (*continued*)

With the encryption module

MOS		Delay (RTD, ms) D1	Noise(dB)		Gain	
Near G1.1	Far G1.2		Near	Far	Near	Far
4.481	4.486	132.32	12.62	15.75	-7	-1
4.49	4.49	138	14	16	-7	-1
4.45	4.47	127.9	12	15	-7	-1
0	0	24.34	0.55	0.21	0	0
0.04	0.02	10.1	2	1	0	0

Since the installation of the private telephone set should not be exposed to external intruders, the set should communicate by using the IP of phones without its own IP. To verify the qualification, analysis on floating ARP packets in the network is made. The outcome of the analysis shows that phones only respond to the ARPs and IPs while the MAC address of the private telephone set was never exposed externally.

4 Conclusion

In this study, an encryption system was proposed to guarantee the voice confidentiality when using an internet telephone. A128-bit AES (Advanced Encryption Standard) method was used to reduce the voice delay. During the operation of an encryption system, in the key-exchange process of the first stage, the RSA (Rivest Shamir Adleman) algorithm, an asymmetric encryption method, was used. The key-exchange process was carried out using RTP packets produced in the telephone, including the key in the payload part, and communicated with the opposite encryption system to avoid any exposure of the encryption system to the outside.

References

1. Butcher, D., Xiangyang, L., Guo, G.J.: Security Challenge and Defense in VoIP Infrastructures. IEEE Trans. on Systems, Man, and Cybernetics 37(6), 1152–1162 (2007)
2. Edelson, E.: Voice over IP: Security pitfalls. Network Security (2), 4–7 (2005)
3. Hunter, P.: VoIP the latest security concern: DoS attack the greatest threat. Network Security (11), 5–7 (2002)
4. Shan, L., Jiang, N.: Research on Security Mechanisms of SIP-Based VoIP System. In: Proc. of Int. Conf. on Hybrid Intelligent Systems, vol. 2, pp. 408–410 (2009)
5. Yoon, S., Jeong, J., Jeong, H.: A study on the tightening the security of the key management protocol for VoIP. In: Proc. of New Trends in Information Science and Service Science, p. 638 (2010)
6. Abdelnur, H., Cridlig, V., State, R., Festor, O.: VoIP security assessment: Method and tools. In: Proc. of IEEE Workshop on VoIP Management and Security, p. 29 (2006)

New ID-Based Proxy Signature Scheme with Message Recovery

Eun-Jun Yoon[1,*], YongSoo Choi[2], and Cheonshik Kim[3,*]

[1] Department of Cyber Security, Kyungil University, Republic of Korea
ejyoon@kiu.ac.kr
[2] BK21 Ubiquitous Information Security, Korea University, Republic of Korea
ciechoi@korea.ac.kr
[3] Department of Computer Science, Sejong University, Republic of Korea
mipsan@paran.com

Abstract. In 2012, Singh-Verma proposed an ID-based proxy signature scheme with message recovery. Unfortunately, by giving two concrete attacks, Tian et al. showed that Singh-Verma's scheme is not secure. This paper proposes an improvement of Singh-Verma's scheme to eliminate the security problems.

Keywords: Proxy signature, Cryptanalysis, ID-based cryptography, Mobile.

1 Introduction

An ID-based message recovery signature scheme is a kind of useful lightweight signature, in which the message itself is not required to be transmitted together with a signature [1-3]. A Proxy signature scheme allows an original signer to delegate a proxy signer to sign messages on its behalf, which has found numerous practical applications such as grid computing and mobile agent systems [4,5]. In 2012 combining the advantages of ID-based message recovery signatures and proxy signature, Singh-Verma [4] proposed the first ID-based proxy signature scheme with message recovery. They proved its security in the random oracle model and believed that it can be used in wireless e-commerce, mobile agent systems and mobile communication. Unfortunately, by giving two concrete attacks, Tian et al. [5] showed that Singh-Verma's scheme is not secure. This paper proposes an improvement of Singh-Verma's scheme to eliminate the security problems.

2 Review of Singh-Verma's Signature Scheme

This section reviews the Singh-Verma's ID-based proxy signature scheme with message recovery [4]. Throughout the paper, notations are employed in Table 1.

[*] Corresponding authors.

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 945–951, 2013.

Table 1. Notation used in scheme

$a\|\|b$	a concatenation operation of two strings a and b.
$\oplus$	a bit-wise exclusive-or computation in the binary system.
$[x]_{10}$	the decimal representation of $x \in \{0,1\}^*$.
$[y]_2$	the binary representation of $y \in Z$.
$_{l1}\|\beta\|$	the first l_1 bits of β from the left side.
$\|\beta\|_{l_2}$	the first l_2 bits of β from the right side.
G_1, G_2	two cyclic groups of the same order q, where $\|q\| = l_1 + l_2$.
H_0, H_1, H_2	three cryptographic hash functions $\{0,1\}^* \to G_1^*$, $\{0,1\}^* \times G_2 \to Z_q$, $G_2 \to Z_q^*$.
F_1, F_2	two cryptographic hash functions $\{0,1\}^{l_2} \to \{0,1\}^{l_1}$, $\{0,1\}^{l_1} \to \{0,1\}^{l_2}$.

There are eight phases in Singh-Verma's scheme: (1) Setup, (2) Extract, (3) Delegate, (4) DVerify, (5) PKGen, (6) PSign, (7) Verify, and (8) ID phases.

(1) Setup: Given a security parameter λ, the PKG(Private Key Generator) does the following steps:

1. Choose a random generator P of G_1 and the master secret key $s \in Z_q^*$.
2. Set $P_{pub} = sP$ as his/her public key.
3. Publish the public parameters $PP = (G_1, G_2, e, P, P_{pub}, H_0, H_1, H_2, F_1, F_2, l_1, l_2)$.

(2) Extract: On input the master secret key s and a user's identity $ID_i \in \{0,1\}^*$, the PKG computes the user's private key $d_i = sH_0(ID_i)$ and sets its public key as $q_i = H_0(ID_i)$.

(3) Delegate: The original signer ID_A does the following steps:

1. Take as input his/her private key d_A and a delegation warrant m_w.
2. Pick a random value $k_A \in Z_q^*$.
3. Compute $r_A = e(P,P)^{k_A}$, $h_A = H_1(m_w, r_A)$ and $S = h_A \cdot d_A + k_A P$.
4. Output the delegation $W_{A \to B} = (m_w, r_A, S)$.

(4) DVerify: Upon receiving $W_{A \to B} = (m_w, r_A, S)$, the proxy signer ID_B does the following steps:

1. Compute $q_A = H_0(ID_A)$, $h_A = H_1(m_w, r_A)$.
2. Check if $e(S, P) = r_A \cdot e(q_A, P_{pub})^{h_A}$.
3. If so, ID_B accepts the delegation; otherwise, he/she requests a valid one from ID_A or terminates the protocol.

(5) PKGen: After accepting $W_{A \to B}$, ID_B sets $d_p = S + h_A \cdot d_B$ as his/her proxy signing key.

(6) PSign: Given a message $m \in \{0,1\}^*$ which conforms to the warrant m_w, the proxy signer ID_B with the proxy signing key d_p does the following steps:

1. Select a random value $k_B \in Z_q^*$ and set $r_B = e(P,P)^{k_B}$.
2. Set $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
3. Compute $v = r_A \cdot r_B$ and $V_B = H_2(v) + \alpha$
4. Compute $U = k_B P + d_p$.
5. Output the proxy signature $\delta = (m_w, r_A, V_B, U)$.

(7) Verify: On input a proxy signature $\delta = (m_w, r_A, V_B, U)$, a verifier does the following steps:

1. Set $h_A = H_1(m_w, r_A)$ and $\alpha = V_B - H_2(e(U,P)e(q_A + q_B, P_{pub})^{-h_A})$.
2. Compute $\beta = [\alpha]_2$ and $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$.
3. Accept the proxy signature δ if m conforms to m_w and ${}_{l1}|\beta| = F_1(m)$.

The correctness of the scheme is justified as follows:

$$\begin{aligned} e(U,P)e(q_A + q_B, \quad P_{pub})^{-h_A} &= e(k_B P + d_p, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(k_B P + h_A \cdot d_B + S, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(k_B P + h_A \cdot d_B + h_A \cdot d_A + k_A P, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e((k_B + k_A)P, P)e(h_A \cdot (d_B + d_A), P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e((k_A + k_B)P, P) \\ &= r_A \cdot r_B \\ &= v \end{aligned} \tag{1}$$

Hence, we can obtain $V_B - H_2(e(U,P)e(q_A + q_B, P_{pub})^{-h_A}) = VB - H_2(v) = \alpha$. Since $\beta = F_1(m)||(F_2(F_1(m)) \oplus m) = [\alpha]_2$, we can obtain $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$. Finally, the integrity of m is justified by ${}_{l1}|\beta| = F_1(m)$.

(8) ID: On input a valid proxy signature $\delta = (m_w, r_A, V_B, U)$ the proxy signer's identity ID_B can be revealed by m_w.

3 Cryptanalysis on Singh-Verma's Signature Scheme

Tian et al. [5] demonstrated that Singh and Verma's ID-based message recovery proxy signature scheme is insecure to two forgery attacks as follows.

(1) Forgery Attack 1: Assume that an adversary A has obtained a valid proxy signature $\delta = (m_w, r_A, V_B, U)$ on message m. To produce a valid proxy signature δ' on a new message m', A does the following steps:

1. Pick a random value $t \in Z_q^*$.
2. Compute $U' = U + tP$ and $v' = e(U,P)e(q_A + q_B, P_{pub})^{-h_A} \cdot e(P,P)^t = v \cdot e(P,P)^t$.
3. Set $\beta' = F_1(m')||(F_2(F_1(m')) \oplus m')$ and $\alpha' = [\beta']_{10}$.
4. Compute $V'_B = H_2(v') + \alpha'$.
5. Output the proxy signature $\delta' = (m_w, r_A, V'_B, U')$.

We can easily see that $\delta' = (m_w, r_A, V'_B, U')$ is a valid proxy signature on the message m' as follows:

$$\begin{aligned} e(U', P)e(q_A + q_B, \quad P_{pub})^{h_A} &= e(U + tP, P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(tP, P)e(U,P)e(q_A + q_B, P_{pub})^{-h_A} \\ &= e(tP, P) \cdot v \\ &= v' \end{aligned} \tag{2}$$

Therefore, Singh-Verma's ID-based proxy signature scheme with message recovery is not secure to the above forgery attack 1.

(2) Forgery Attack 2: Assume that A is an adversary who aims to forge a proxy signature δ on any message m, but he/she has not yet obtained a valid proxy signature. Then A does the following steps:

1. Produce a delegation warrant m_w such that m conforms to it.
2. Select two random values $r_A, U \in G_1$, and set $h_A = H_1(m_w, r_A)$ and $v = e(U, P)e(q_A + q_B, P_{pub})^{-h_A}$.
3. Compute $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
4. Compute $V_B = H_2(v) + \alpha$
5. Output the proxy signature $\delta = (m_w, r_A, V_B, U)$.

We can easily see that $\delta = (m_w, r_A, V_B, U)$ is a valid proxy signature on the message m. Since $v = e(U, P)e(q_A + q_B, P_{pub})^{-h_A}$, we can obtain $V_B - H_2(e(U, P)e(q_A + q_B, P_{pub})^{-h_A}) = V_B - H_2(v) = \alpha$. As $\beta = F_1(m)||(F_2(F_1(m)) \oplus m) = [\alpha]_2$, hence we can obtain $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$. Finally, we can find out that ${}_{l1}|\beta| = F_1(m)$. Consequently, $\delta = (m_w, r_A, V_B, U)$ is indeed a valid proxy signature on m. Therefore, Singh-Verma's ID-based proxy signature scheme with message recovery is not secure to the above forgery attack 2.

4 Proposed Signature Scheme

This section proposes an improved Singh-Verma's ID-based proxy signature scheme with message recovery. The proposed scheme also consists on eight phases: (1) Setup, (2) Extract, (3) Delegate, (4) DVerify, (5) PKGen, (6) PSign, (7) Verify, and (8) ID phases. The proposed scheme works as follows.

(1) Setup: Given a security parameter λ, the PKG does the following steps:

1. Choose a random generator P of G_1 and the master secret key $s \in Z_q^*$.
2. Set $P_{pub} = sP$ as his/her public key.
3. Publish the public parameters $P = (G_1, G_2, e, P, P_{pub}, H_0, H_1, H_2, F_1, F_2, l_1, l_2)$.

(2) Extract: On input the master secret key s and a user's identity $ID_i \in \{0,1\}^*$, the PKG computes the user's private key $d_i = sH_0(ID_i)$ and sets its public key as $q_i = H_0(ID_i)$.

(3) Delegate: The original signer ID_A does the following steps:

1. Take as input his/her private key d_A and a delegation warrant m_w.
2. Pick a random value $k_A \in Z_q^*$.
3. Compute $r_A = k_A P$.
4. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.
5. Compute $S = k_A H_w + d_A$.
6. Output the delegation $W_{A \to B} = (m_w, r_A, S)$.

(4) DVerify: Upon receiving $W_{A \to B} = (m_w, r_A, S)$, the proxy signer ID_B does the following steps:

1. Compute $q_A = H_0(ID_A)$.
2. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.

3. Check if $e(S,P) = e(r_A, H_w) \cdot e(q_A, P_{pub})$.
4. If so, ID_B accepts the delegation; otherwise, he/she requests a valid one from ID_A or terminates the protocol.

(5) PKGen: After accepting $W_{A\to B}$, ID_B sets $d_p = S + d_B$ as its proxy signing key.

(6) PSign: Given a message $m \in \{0,1\}^*$ which conforms to the warrant m_w, the proxy signer ID_B with the proxy signing key d_p does the following steps:

1. Select a random value $k_B \in Z_q^*$ and set $r_B = k_B P$.
2. Set $\beta = F_1(m)||(F_2(F_1(m)) \oplus m)$ and $\alpha = [\beta]_{10}$.
3. Compute $H_m = H_0(ID_B, \alpha, r_B) \in G_1^*$.
4. Compute $U = k_B H_m + d_p$.
5. Output the proxy signature $\delta = (m_w, r_A, r_B, U)$.

(7) Verify: On input a proxy signature $\delta = (m_w, r_A, r_B, U)$, a verifier does the following steps:

1. Compute $H_w = H_0(ID_A, m_w, r_A) \in G_1^*$.
2. Compute $\beta = [\alpha]_2$ and $m = F_2({}_{l1}|\beta|) \oplus |\beta|_{l_2}$.
3. Compute $H_m = H_0(ID_B, \alpha, r_B) \in G_1^*$.
4. Accept the proxy signature δ if m conforms to m_w and

$$H_2(e(U,P)e(q_A + q_B, P_{pub})^{-1}) \equiv H_2(e(H_m, r_B)e(H_w, r_A)) \tag{3}$$

The correctness of the scheme is justified as follows:

$$\begin{aligned} e(U,P)e(q_A + q_B, \quad P_{pub})^{-1} &= e(k_B H_m + d_p, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + d_B + S, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + d_B + k_A H_w + d_A, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + k_A H_w, P)e(d_B + d_A, P)e(q_A + q_B, P_{pub})^{-1} \\ &= e(k_B H_m + k_A H_w, P) \\ &= e(k_B H_m, P)e(k_A H_w, P) \\ &= e(H_m, k_B P)e(H_w, k_A P) \\ &= e(H_m, r_B)e(H_w, r_A) \end{aligned} \tag{4}$$

(8) ID: On input a valid proxy signature $\delta = (m_w, r_A, r_B, U)$ the proxy signer's identity ID_B can be revealed by m_w.

5 Security Analysis

This section demonstrates a concrete security proof of our proposed scheme.

(1) Unforgeability [6]: Only a designated proxy signer can create a valid proxy signature for the original signer. In other words, nobody can forge a valid proxy signature without the delegation of the original signer. It means that any entity (other than the real proxy signer ID_B), including the original signer ID_A, cannot generate a valid proxy signature. Only an authorized proxy signer ID_B can create a valid proxy signature δ. If any attacker wants to forge a proxy signature, he/she must be authorized by the original signer signing on a warrant m_w and use the proxy signer's proxy secret key $d_p = S + d_B$ to compute δ. However, this is impossible since the identity of the attacker was not in m_w signed by the original signer. Not to mention, the attacker does not know $d_p = S + d_B$. Under this situation, even if the attacker want to (1) fake the proxy signer key as $d_{p'}$, (2) change value $U = k_B H_m + d_p$ to U', or (3) randomly select $k'_B \in Z_q^*$, trying to counterfeit the proxy signature, his/her attempt deems to fail without knowing the proxy secret key $d_p = S + d_B$. Therefore, the proposed scheme provides the unforgeability property.

(2) Verifiability [6]: After checking and verifying the proxy signature, a verifier can be convinced that the received message is signed by the proxy signer authorized by the original signer. In the proposed Verify phase, after checking and verifying the proxy signature δ, the verifier can calculate to check whether the verification equation $H_2(e(U,P)e(q_A + q_B, P_{pub})^{-1})? = H_2(e(H_m, r_B)e(H_w, r_A))$ holds. If it does, the verifier can be convinced that the received message is signed by one of the proxy signer members authorized by the original signer because $U = k_B H_m + d_p$ and $e(H_m, r_B)e(H_w, r_A)$ are used in the verification equation. Therefore, the proposed scheme provides the verifiability property.

6 Conclusion

In 2012, combining the advantages of ID-based message recovery signatures and proxy signatures, Singh-Verma proposed an ID-based proxy signature scheme with message recovery that can be used in wireless e-commerce, mobile agent systems and mobile communication. Unfortunately, Tian et al. showed that Singh-Verma's scheme is not secure against two forgery attacks. For this reason, Singh-Verma's scheme is insecure for practical application. This paper proposed an improvement of Singh-Verma's scheme to eliminate the security problems. The proposed scheme also requires smaller bandwidth in contrast to previous ID-based proxy signature schemes. Hence the proposed scheme can be a good alternative for certificate based proxy signatures used for mobile agent.

Acknowledgements. This work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2010-0010106) and partially supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2012- H0301-12-2004) supervised by the NIPA(National IT Industry Promotion Agency).

References

1. Mambo, M., Usuda, K., Okamoto, E.: Proxy signature for delegating signing operation. In: Proc. 3rd ACM Conference on CCS, pp. 48–57 (1996)
2. Zhang, F., Susilo, W., Mu, Y.: Identity-based partial message recovery signatures (or how to shorten ID-based signatures). In: Proc. 9th Conference on FC, pp. 45–56 (2005)

3. Tso, R., Gu, C., Okamoto, T., Okamoto, E.: Efficient ID-Based Digital Signatures with Message Recovery. In: Proc. 6th International Conference on CANS, pp. 47–59 (2007)
4. Singh, H., Verma, G.K.: ID-based proxy signature scheme with message recovery. Journal of Systems and Software 85, 209–214 (2012)
5. Tian, M., Huang, L., Yang, W.: Cryptanalysis of an ID-based proxy signature scheme with message recovery. Applied Mathematics & Information Sciences 6(3), 47–59 (2012)
6. Chou, J.: A Novel Anonymous Proxy Signature Scheme. Advances in Multimedia 427961, 1–10 (2012)

Erratum: Scheduling Optimization of the RFID Tagged Explosive Storage Based on Genetic Algorithm

Xiaoling Wu[1], Huawei Fu[1,2], Xiaomin He[2], Guangcong Liu[2], Jianjun Li[1],
Hainan Chen[1,2], Qianqiu Wang[1], and Qing He[1]

[1] Guangzhou Institute of Advanced Technology, Chinese Academy of Sciences
[2] Guangdong University of Technology, Guangzhou 510006
xl.wu@giat.ac.cn

J.J. Park et al. (Eds.): GPC 2013, LNCS 7861, pp. 358–366, 2013.

DOI 10.1007/978-3-642-38027-3_112

The authors of the paper "Scheduling Optimization of the RFID Tagged Explosive Storage Based on Genetic Algorithm" (Xiaoling Wu, Huawei Fu, Xiaomin He, Guangcong Liu, Jianjun Li, Hainan Chen, Qianqiu Wang, and Qing He), DOI: 10.1007/978-3-642-38027-3_38, appearing on pages 358-366 of this publication, have decided to retract the paper, because there are serious flaws in both the data and some steps of the algorithm. This makes the data unreliable.

The original online version for this chapter can be found at
http://dx.doi.org/10.1007/978-3-642-38027-3_38

Author Index