Ying Cai
Thomas Magedanz
Minglu Li
Jinchun Xia
Carlo Giannelli (Eds.)

LNICST
48

Mobile Wireless Middleware, Operating Systems, and Applications

Third International Conference, Mobilware 2010
Chicago, IL, USA, June/July 2010
Revised Selected Papers

ICST

Springer

Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering 48

Ying Cai Thomas Magedanz Minglu Li
Jinchun Xia Carlo Giannelli (Eds.)

Mobile Wireless Middleware, Operating Systems, and Applications

Third International Conference, Mobilware 2010
Chicago, IL, USA, June 30 - July 2, 2010
Revised Selected Papers

Springer

Volume Editors

Ying Cai
Iowa State University, Department of Computer Science
Ames, IA 50011, USA
E-mail: yingcai@cs.iastate.edu

Thomas Magedanz
Technische Universität Berlin
Institute for Telecommunications Systems
10587 Berlin, Germany
E-mail: thomas.magedanz@tu-berlin.de

Minglu Li
Department of Computer Science and Engineering
Shanghai JiaoTong University
Shanghai, China
E-mail: li-ml@cs.sjtu.edu.cn

Jinchun Xia
San Jose State University, Department of Computer Engineering
San Jose, CA, 95192, USA
E-mail: xiajinchun@gmail.com

Carlo Giannelli
Università degli Studi di Bologna
Facoltà de Ingegneria DEIS
40136 Bologna, Italy
E-mail: carlo.giannelli@unibo.it

Library of Congress Control Number: Applied for

CR Subject Classification (1998): C.2, H.4, D.2, H.5, H.3, I.2

ISSN 1867-8211
ISBN-10 3-642-17757-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-17757-6 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper

06/3180 5 4 3 2 1 0

Preface

The advances in wireless communication technologies and the proliferation of mobile devices have enabled the realization of intelligent environments for people to communicate with each other, interact with information-processing devices, and receive a wide range of mobile wireless services through various types of networks and systems everywhere, anytime. A key enabler of this pervasive and ubiquitous connectivity environments is the advancement of software technology in various communication sectors, ranging from communication middleware and operating systems to networking protocols and applications. The international conference series on **Mobile Wireless Middleware, Operating Systems, and Applications (MOBILWARE)** is dedicated to address emerging topics and challenges in various mobile wireless software-related areas. The scope of the conference includes the design, implementation, deployment, and evaluation of middleware, operating systems, and applications for computing and communications in mobile wireless systems.

MOBILWARE 2010 was the third edition of this conference, which was made possible thanks to the sponsorship of ICST and Create-Net and most importantly the hard work of the TPC and reviewers.

Similar to the last successful editions, we had 35 submissions from 23 different countries this year, reflecting the international interest for the conference topics. After a thorough review process, we finalized an excellent technical program including 18 regular papers and 4 short papers. These papers were grouped into six technical sessions on:

- Location, and tracking supports and services
- Human — computer interface for mobile devices
- Mobility management and hand–off management
- Novel applications and communication protocols for wireless networks
- Mobile intelligent middleware
- Short papers

We want to express our sincere gratitude to all the authors who submitted their papers to this conference and to all the reviewers whose diligent work was crucial for the finalization of this high-quality final technical program.

The MOBILWARE 2010 technical program also included two excellent keynote speeches, namely, “Computing Challenges in Intelligent Transportation Systems” given by Ouri Wolfson from the Department of Computer Science at the University of Illinois at Chicago, and “Event Processing on Mobile Phones: Mobile 3.0?” presented by Archan Misra, Senior Scientist, Telcordia Technologies. We thank the keynote speakers for contributing to the quality and the success of this event.

In addition to the main conference, Mobilware 2010 featured two workshops:

- Workshop on Mobile and Location-Based Business Applications (MAPPS 2010)
- Workshop on Mobile Multimedia Networking (IWMMN 2010)

We express our deepest thanks to the workshop organizers.

Finally, we would like to thank Paolo Bellavista and Carl Chang, the General Co-chairs, for their constant motivation and support, as well as Gergely Nagy, the conference Organization Chair, and Carlo Giannelli, the Publication and Web Chair, for helping in all the organizational matters. In addition, we would like to thank the whole ICST team for their constant support in making this event happen.

Ying Cai
Thomas Magedanz
Minglu Li

Mobilware 2010 Organization

General Co-chairs

Paolo Bellavista	University of Bologna, Italy
Carl K. Chang	Iowa State University, USA

Steering Committee

Imrich Chlamtac (Chair)	Create-Net, Italy
Paolo Bellavista	University of Bologna, Italy
Carl K. Chang	Iowa State University, USA

Technical Program Committee Co-chairs

Ying Cai	Iowa State University, USA
Thomas Magedanz	Fraunhofer FOKUS, Germany
Minglu Li	Shanghai Jiao Tong University, China

Conference Organization Chair

Gergely Nagy	ICST

Conference Publicity Co-chairs

Europe: Andrej Krenker	Sintesio, Slovenia
North America: Jatinder Pal Singh	Stanford University, USA
North America: Janise McNair	University of Florida, USA
South America: C. Esteve Rothenberg	CpQD, Brazil
Middle East: Ghadaa Alaa	Information Technology Institute, Egypt
Asia: Michael Chen	Institute for Information Industry, Taiwan

Publication and Web Chair

Carlo Giannelli	University of Bologna, Italy

Workshop/Tutorial Chair

Jinchun Xia	San Jose State University, USA

Local Chair

Jennifer Juehring	Iowa State University, USA
Laurel Tweed	Iowa State University, USA

Technical Program Committee

J.M. Bonnin	Institut TELECOM/TELECOM Bretagne, France
C. Borcea	NJIT University, USA
J.J. Bu	Zhejiang University, P.R. China
R.H. Campbell	UIUC, USA
J. Cao	Polytechnic University, Hong Kong
Z. Chen	Florida International University, USA
L. Cheng	Lehigh University Bethlehem, USA
A. Chibani	Paris 12 University, France
I. Demeure	ENST, Telecom ParisTech, France
A. Fasbender	Ericsson GmbH, Germany
X. Fu	University of Massachusetts Lowell, USA
A. Gavras	Eurescom, Germany
M. Gerla	University of California at Los Angeles, USA
R. Glitho	Concordia University Montreal, Canada
J. Govil	CISCO, USA
M. Gunes	FU Berlin, Germany
C. Hesselman	Novay, The Netherlands
R. Hill	Indiana University, USA
S. Holtel	Vodafone Group R&D, Germany
F. Hong	Ocean University of China, P.R. China
Chih-Lin Hu	National Central University, Taiwan
L. Iftode	Rutgers University, USA
A. Jaokar	Futuretext, UK
W. Jiang	Huazhong University of Science and Technology, P.R. China
C. Julien	University of Texas at Austin, USA
T.G. Kanter	Mid Sweden University, Sweden
P.J. Keleher	University of Maryland, USA
W.S. Ku	Auburn University, USA
Y. Lan	Northeastern University, P.R. China
P. Langendoerfer	IHP, Germany
X. (Cindy) Li	University of North Carolina at Pembroke, USA
X. Li	Oklahoma State University, USA
D. Maggiorani	University of Milan, Italy
R. Minerva	Telecom Italia, Italy
P. Nixon	University College Dublin, Ireland
J.K. Nurminen	Nokia, Finland
H. Ohsaki	Osaka University, Japan
G. Ormazabal	Verizon Laboratories, USA
J. Payton	University of North Carolina at Charlotte, USA

S. Pandey	CISCO Systems Inc., USA
F. Ren	Tsinghua University, P.R. China
K. (Quinn) Ren	Illinois Institute of Technology, USA
G.C. Roman	Washington University St. Louis, USA
R. Schwaiger	Deutsche Telekom Labs, Germany
J. Song	KAIST, Korea
L. Sun	Chinese Academy of Sciences, P.R. China
W. Sun	Fudan University, P. R. China
J. Taheri	University of Sydney, Australia
M. Tao	Shanghai Jiao Tong University, P.R. China
X. Tao	Nanjing University, P.R. China
A. Tripath	University of Minnesota, USA
N. Ventura	University Cape Town, South Africa
Xiaodong Wang	National University of Defense Technology, P.R. China
Xinbing Wang	Shanghai Jiao Tong University, P.R. China
B. Xu	Tsinghua University, P.R. China
G. Xue	Shanghai Jiao Tong University, P.R. China
W. Xue	Tsinghua University, P.R. China
M.A. Youssef	Nile University, Egypt
F. Zambonelli	University of Modena and Reggio Emilia, Italy
L. Zhang	Indiana University South Bend, USA
Q. Zheng	Xian Jiaotong University, P. R. China
Y. Zhu	Shanghai Jiao Tong University, P.R. China

MAPPS 2010 Workshop Organization

Workshop Chairs

Claudia Linnhoff-Popien	Ludwig Maximilians University Munich, Germany
Peter Ruppel	Ludwig Maximilians University Munich, Germany
Stephan Verclas	T-Systems International GmbH, Germany

Technical Program Committee

Alapan Arnab	T-Systems, South Africa
Eduard Babulak	University of the South Pacific, Fiji
Lothar Borrmann	Siemens AG, Germany
Nico Deblauwe	IWT-agency for Innovation, Belgium
Jerry Zeyu Gao	San Jose State University, USA
Mario Jaritz	T-Mobile, Germany
Mikkel Baun Kjærgaard	Aarhus University, Denmark
Axel Küpper	Technical University Berlin, Germany
Peter Reichl	FTW Vienna, Austria
Gregor Schiele	University of Mannheim, Germany
Roland Schwaiger	Deutsche Telekom, Germany
Thomas Strang	German Aerospace Center, Germany
Sean Taylor	Redwood Technologies Ltd., UK

IWMMN 2010 Workshop Organization

General Chairs

Honggang Wang	University of Massachusetts, Dartmouth, USA
Jinchun Xia	San Jose State University, USA

Program Chairs

Wei Wang	South Dakota State University, USA
Shaoen Wu	School of Computing, University of Southern Mississippi, USA
Hong Liu	University of Massachusetts, Dartmouth, USA

Technical Program Committee

Sunho Lim	Texas Tech University, USA
Yijuan Lu	Texas State University-San Marcos, USA
Ju Liu	Shandong University, China
Naoki Wakamiya	Osaka University, Japan
Ioannis Andreopoulos	University College London, UK
Tigang Jiang	University of Electronic Science and Technology of China
Paolo Bellavista	Università degli Studi di Bologna, Italy
Rui L.A. Aguia	Universidade de Aveiro, Portugal
Suhair H. Amer	Southeast Missouri State University, USA
Harry Skianis	University of Aegean, Greece
Amer Dawoud	University of Southern Mississippi, USA
Cheng Luo	Coppin State University, USA
Seema Verma	Banasthali University, India

Table of Contents

Session 1: Novel Applications and Communication Protocols for Wireless Networks (Chair: Paolo Bellavista)

Session 2: Mobility Management and Handoff Management (Chair: Thomas Magedanz)

Session 3: Human-Computer Interface for Mobile Devices (Chair: Honggang Wang)

Session 4: Mobile Intelligent Middleware (Chair: Ying Cai)

Session 5: Location-Aware and Context-Aware Networking and Computing (Chair: Chris Thompson)

Session 6: Short Papers (Chair: Paolo Bellavista)

International Workshop on Mobile and Location-Based Business Applications (MAPPS 2010)

International Workshop on Mobile Multimedia Networking (IWMMN 2010)

Session 1: Novel Applications and Communication Protocols for Wireless Networks (Chair: Paolo Bellavista)

A Secure Mobile OTP Token

Fred Cheng

International Technological University and FPC Consultancy,
Los Altos Hills, California USA
fredtcheng@yahoo.com

Abstract. Implementing a mobile One-time Password (OTP) Token on a cellular phone is a hot topic since the past few years. The proposed solutions had made certain improvements on network security. But none of them can fully prevent the OTP seed (K) tracing from MIMT OTP code interception or Shoulder-surfing security attacks while also meet the following criteria – fully compliant with existing authentication systems, inter-operable with other token and easy to deploy or support. This paper presents a cipher called Rubbing Encryption Algorithm (REAL) and the implementation of a Mobile OTP Token using this algorithm. The newly designed REAL Mobile OTP Token addresses and improves the aforementioned issues successfully.

Keywords: One-time Password, OTP Token, Authentication, Encryption Algorithm, MITM Attack, Shoulder-surfing Attack, Security Attack.

1 Introduction

One-time Password (OTP) Token can automatically generate a series of dynamic password. It has gained a leading position in the Two-factor Authentication (2FA) system for better network security. As the cellular phone became popular in the past few years, many solutions were proposed to embed the OTP Token inside such mobile device [1][2][3]. But they encountered certain deficiencies such as the mobile token can not fully resist OTP seed (K) tracing by Man-in-the-middle (MITM) OTP code interception [23] and Shoulder-surfing [24] security attack. Other issues include poor interoperability and compliance with existing authentication systems, plus higher deployment and support cost. In particular, many proposals store the OTP generation seed and personal secrecies inside the cellular phone. It compromises network security when the phone is lost or stolen [4].

To address the aforementioned issues, we introduce a Rubbing Encryption Algorithm (REAL) in this paper. A user does not need to memorize and enter the key when decrypts a ciphertext by REAL. This special feature allows REAL to use a very long and complex key for encryption. So REAL can securely encrypt a short word length OTP code with very high security level. That is why the locally stored REAL OTP codes can retain its security even if the phone is lost or stolen. A user can lay the hardware token over the REAL ciphertext image on a cellular phone's screen to electronically "rub" (decrypt) the OTP code. This is why the cipher gets its name – "Rubbing Encryption ALgorithm" (REAL).

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 3–16, 2010.

We also present the design of a REAL Mobile OTP Token. This token is compliant to the OTP token proposed by the Initiative for Open AuTHentication (OATH) [6]. A cellular phone is used as the OTP generating platform. Such token is fully interoperable with the existing authentication system. A low cost plastic card is used as the REAL key (code pointer) bearing hardware token. OTP codes are pre-generated using the OATH's event-based OTP code algorithm [5]. The codes are encrypted by a specific REAL key assigned to a user. The confidential REAL encrypted codes are stored in the Data File. The OTP generation program and each user's Data File are provisioned and downloaded through the Internet. After installation, the user activates the OTP program. User lays her hardware token over the REAL ciphertext image on the phone's screen to obtain the OTP code. The user then enters this OTP code into the login window to complete the 2FA process.

Each REAL hardware token carries one key on each side of the token. Three versions of the REAL Mobile OTP Token are implemented. The basic version works with codes from just one OTP generating seed. It provides the basic secure OTP code. An improved version works with OTP codes from two different OTP generating seeds. The token matches the two code generation seeds with the authentication server automatically. So a valid OTP code will not be mistakenly rejected. This token can resist attack on OTP seed (K) tracing by MITM data interception attack. The third token is similar to the second version but the REAL key is dynamically decoupled from the code pointer's physical locations on the hardware token. This version can resist both the Shoulder-surfing and the seed tracing from MITM interception attacks.

We organize this paper as follows. In section 2, we briefly provide the background and related work. The Rubbing Encryption Algorithm (REAL) is introduced in Section 3. The implementation of the REAL Mobile OTP Token is discussed in Section 4. We then review the design goals, analyze the security attacks and other security concerns in Section 5. We further conclude this paper and possible improvement work in Section 6.

2 Background and Related Work

Several implementations have emerged as the key methods to use a cellular phone in remote network authentication. One such solution focuses on using cellular phone as a standalone OTP token [1][7][8]. The mobile device is used as a computational platform to generate OTP code. These tokens usually do not have any capability to resist the OTP seed (K) tracing by MITM interception and Shoulder-surfing attacks. It stores the secret seed (K) and counter value in the phone. So the network security may be comprised if the phone is lost or stolen as the secrecies can be exposed.

An alternative proposal focuses on using the cellular network as a secure out of band channel to transmit or receive the OTP code to or from the authentication server [2][9][10]. The OTP code is transmitted as an image data or through the Short Message Service (SMS) in text form. This new channel effectively prevents the traditional MITM attack. But Shoulder-surfing attack is still an issue. Cellular QoS (quality of service) will affect the reliability of OTP generation when using cellular SMS. SMS is a best effort delivery service. Cellular service providers cannot guarantee a real-time or in-time delivery. Moreover, when a user is out of the cellular service coverage area, such as in a basement of the building or in the rural area, using SMS sometimes is not even possible. Besides, new software and hardware are needed to allow the authentication server to interface to SMS system. All these increase the total system cost and complicate the server management task.

The third and most recent approach involves using Subscriber Identity Module (SIM) on a cell phone and other newly proposed protocols such as the Liberty Alliance Federation Standard [9] or The Free Auth Project [10] to perform the authentication procedure [3] [13][14][15][16][17][18]. In this scheme, the mobile device also carries part of the authentication secret information. The authentication is carried out directly through the phone, cellular network to the remote server. Again, QoS of authentication are limited by cellular service coverage area. In particular, using different authentication protocols and OTP algorithms usually leads to a new authentication system. New software and hardware are required at server to implement such scheme as the proposed solutions are not fully compliant with the existing authentication servers. Though this approach may prevent the traditional MITM attack, the Shoulder-surfing attack may still be an issue.

Overall, a standalone cellular phone-based Mobile OTP Token has its merits. It can be easily implemented to have full compliance with the existing infrastructure. No additional cost and server work are required for such token. Its usage is also not limited by the cellular service's coverage. But we need to resolve the security issues associated with this approach. This is exactly what the REAL Mobile OTP Token sets out to do.

3 The Cipher System

The Rubbing Encryption Algorithm (REAL) operates through multiple steps to ensure the desired data security. REAL's general theory and its operation procedures are presented in the following subsections.

3.1 REAL General Theory

Using Shannon entropy theory, the uncertainty H(X) of a numeric image X, which consists of a series of T characters selected from Y different symbols, can be found from equation (1) [19].

$$H(X) = -\sum_{i=1}^{T} P_i (\mathrm{Log}_2 P_i) , \tag{1}$$

where P_i is the occurrence possibility of a symbol Y_i.

When each symbol has an equal chance of occurrence and X selects equal number of symbol to form the T characters, we then have an equiprobable numeric image X. H(X) reaches its maximum value when above condition is met [20]. Since X selects equal number (n) of Y different symbols to form the T characters, T = nY, where n is a positive integer. Each symbol's occurrence possibility P_i becomes

$$P_i = n/T = 1/Y . \tag{2}$$

Substituting P_i into equation (1), H(X) becomes

$$H(X) = T(\text{Log}_2\ Y)/Y. \tag{3}$$

Similarly, given a symbol S displayed in the same equiprobable numeric image X, its uncertainty H(S) can be found as follows.

$$H(S) = T(\text{Log}_2\ Y)/\ Y^2. \tag{4}$$

The relationship on image and symbol's uncertainties to the image size (T) and symbol's variety (Y) can be found from equation (3) and (4). When using more variety of symbols (Y), both the image and symbol's uncertainties decrease.. Symbol's uncertain reduces even at a faster rate when its variety increases. Both the numeric image and symbol's uncertainties increase when image size (T) increases.

We can obtain a better security to a given plaintext by forming a higher uncertainty numeric image as the ciphertext. Maintaining higher uncertainty for the symbols used in the original plaintext is also an important step. Optimizing the image size (T) and symbol variety (Y) according to equation (3) and (4) is the key task to maintain the desirable high security for both the image X and symbol S.

In encrypting, REAL places the original plaintext symbols S as part of the characters of image X. This image X will be displayed on a screen for viewing. Such image X is called a REAL Image. A proper REAL encryption ensures that S' uncertainty always stay not less than other symbols that are not shown in the plaintext. REAL Image size T will be chosen to have a high uncertainty level and fit the display screen size at the same time. The higher Shannon uncertainty values both the REAL Image and its symbol have, the securer such REAL Image is.

The REAL Image can be easily expanded into a multiple dimensional spatial form factor. By maintaining an equiprobable image in each dimension, the multiple dimensional REAL Image will retain its peak uncertainty. It will be a very secure spatial REAL Image. The multiple dimensional REAL Image feature enables a new spatial encryption method to protect the desired security.

3.2 REAL Encryption Procedures

Fig. 1 shows the general REAL encryption procedure. For ease of discussion, a REAL Image X of size T and the set of symbol with numerals of 0 to 9 are chosen to illustrate the proposed algorithm.

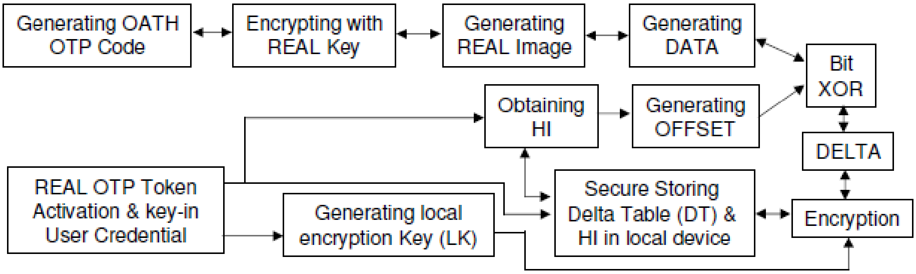

Generating OATH OTP Code
Encrypting with REAL Key
Generating REAL Image
Generating DATA
Bit XOR
Obtaining HI
Generating OFFSET
DELTA
REAL OTP Token Activation & key-in User Credential
Generating local encryption Key (LK)
Secure Storing Delta Table (DT) & HI in local device
Encryption

Fig. 1. REAL cipher and Mobile OTP Token operation procedure

Key Generation. REAL derives its encryption key from the specific spatial data on REAL Image. Since REAL can encrypt and display ciphertext data in multiple dimensional form factor, its encryption key can be of the same multiple dimensions as well. The REAL key is embedded on its hardware token.

To generate a key, we first choose a desired REAL Image form factor that fits well with the given display screen size and an easy reading symbol font size (as shown in Fig. 3.a.). The encryption key is the character locations that display the plaintext symbols in a REAL Image. Given a two dimensional REAL Image with a plastic card (REAL hardware token), we can randomly place pointers around the card's periphery as REAL key. For D characters of plaintext, we use "D+1" number of pointers as REAL key. The extra one pointer is used as the indicator to choose either front side or back side of the token during the REAL operation. Locations of the pointer can be denoted as W_i where i = 0 to D. Fig. 2. shows an example of such a REAL hardware token with seven code pointers for a 6-digit OTP code. In this example, we have two different sets of key on each side of the token.

We can find the total possible number (N) of key or total number of different hardware token from the equation

$$N = C(T, (D+1)) \ . \tag{5}$$

Given a REAL Image size of 40 (T) and OTP code size of 6 (D), we can have 18.6 million different tokens or keys (on each side of token). Each token can only decrypt its own encrypted REAL Image (REAL ciphertext).

Generating REAL Image. REAL places the plaintext's symbols into the corresponding W_i locations in REAL Image according to each symbol's occurring sequence. We use the following example to illustrate REAL Image's generation.

Assuming D = 6 and plaintext code = 807235, we then have $D_5 = 8$, $D_4 = 0$, $D_3 = 7$, $D_2 = 2$, $D_1 = 3$ and $D_0 = 5$ as the plaintext symbols. Row I and Row II of Table 1 show how the plaintext symbols can be randomly assigned to each W_i location. The pseudo code to generate such REAL Image is shown as follows.

```
REAL_Image_GEN(K, i, T)
0  OTP(i) = Truncate(HMAC-SHA-1(K, i));
1  OTP(i) = Concatenate (D_5|D_4| D_3|D_2| D_1|D_0);
2  Sequentially placing D_5 through D_0 into the
   corresponding $W_i$ locations of the REAL Image;
3  Fill in an odd random number (3 in our example) in $W_6$
   to indicate using key on the front side of the token;
4  Fill the rest of REAL Image elements with randomly
   chosen symbols so that the Image reaches
   equiprobable state,      //This is REAL_Image(i);
5  DATA(i) = Concatenate(elements of REAL_Image(i));
6  End of program.
```

In this pseudo code, K is a 160-bit randomly chosen OTP seed, i is the event counter value, T is the REAL Image size and REAL_Image(i) is the encrypted OTP code generated from OATH OTP formula [5] as shown in step 0. DATA(i) is the concatenation of all the elements of the REAL_Image(i).

Table 1. Generating a REAL Image

	W_6	W_5	W_4	W_3	W_2	W_1	W_0
I	D_6	D_5	D_4	D_3	D_2	D_1	D_0
II	3	8	0	7	2	3	5

Offset Generation. To further protect the plaintext data, REAL does not store the encrypted DATA(i) directly. A random value (Offset) is used to generate a logic operation difference (Delta) between DATA(i) and Offset (i). Delta then indirectly represents its corresponding REAL ciphertext. By safely guarding the random Offset, Delta is very secure and so is the REAL Image. Offset (i) is generated by a one-way-hashing operation from the (i-1)th index value. This hashed index value is called HI. Offset(i) generation procedure is shown in the pseudo code below.

```
Offset_GEN
0  i = 0,        //initialize program loop counter;
1  HI(0)= TRUNCATE(HMAC-SHA-1(K_1, 1)), //K_1 is a 160
   bit random number;
2  Bit_159 to Bit_0 = HMAC-SHA-1(HI(i), 1);
3  Bit_319 to Bit_160 = HMAC-SHA-1(HI(i), 19);
4  Offset(i) = Concatenate(Bit_319 to Bit_0);
5  i = i + 1;
6  HI(i) = TRUNCATE(HMAC-SHA-1(HI(i-1), 1));
7  If i > Max_Count, go to Step 8,    //Max_Count =
   total counts of DATA(i);
8  End of program.
```

Delta Table and Secure Storage. Delta(i) is the Bit-Exclusive-OR (B-XOR) difference between the related DATA(i) and Offset(i). Their relationship can be described as

$$\text{Delta(i)} = \text{B-XOR(DATA(i), Offset(i))} . \tag{6}$$

Delta Table (DT) is the compilation of the entire Delta(i) in a special relationship to the corresponding HI(i). That is, Delta(i) is not stored according to the original sequence of DATA(i). Delta(i) is rearranged to follow the value significance of its related HI(i). The new tabulated Delta(i) form the DT. DT and the last HI data will then be further encrypted and securely stored in the designated device.

3.3 REAL Decryption Procedures

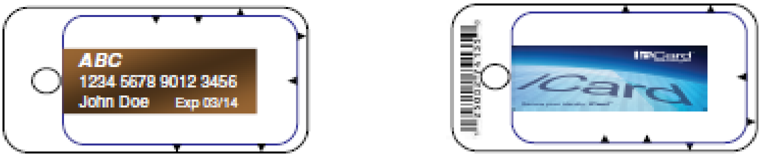
To decrypt a REAL ciphertext, we mainly follow the procedures of section 3.2 but in a reversed order (Fig. 1). Obtaining the last HI(i-1) and Delta(i) value is the first step. A new HI(i) value is generated using the procedure shown in step 8 of Offset_GEN listed in Section 3.2. Once HI(i) is available, Delta(i) can be found by sorting through Delta Table (DT). Following the same step 2 through 4 procedures shown in Offset_GEN, Offset(i) can be generated from HI(i). DATA(i) can then be obtained through the Bit-Exclusive-OR operation of Delta(i) and Offset(i). Subsequently, REAL Image(i) can be reconfigured from DATA(i). By overlaying the unique REAL hardware token on top of the REAL Image(i), the plaintext data will be (rubbed out and) indicated by the pointers on the token (as shown in Fig. 3b).

4 Design the Secure REAL Mobile OTP Token

Our design goals and considerations are as follows. The REAL Mobile OTP token will work as a standalone token. It will be fully compliant to existing OATH server, inter-operable with other OATH tokens and easy to deploy and low support cost. The network security will not be compromised even if the phone is lost or stolen. Moreover, the token will have the capability to resist security attacks such as OTP seed tracing by MITM code interception or Shoulder-surfing.

4.1 REAL Mobile OTP Hardware Token

We choose a plastic card with the size of 1" x 2" as hardware token (as shown in Fig. 2). The material of our token conforms to credit card's ISO/IEC FDIS 7810 standard [21]. The token can be easily put on key chain or kept in a wallet. The REAL encryption key is embedded on the token by the code pointer's locations. The code pointer is the solid black triangle printed on the token periphery. The token making process is very similar to a regular credit card. So the cost of this hardware token is comparable to a credit card.



a. REAL Mobile OTP Token (Front) b. REAL Mobile OTP Token (Back)

Fig. 2. a. Token's periphery is transparent. Overlaid symbols can be clearly read through the token. b. Barcode serial number is printed on the back side to correlate the token with the specific REAL encryption keys. One token carries two set of keys with one each on the front and back side. REAL Image will indicate which set of key to be used.

4.2 REAL Mobile OTP Client Software

REAL Mobile OTP Token has two major parts of software. They are Program File and Data File. The Program File contains a set of executable programs to generate the REAL Image and other housekeeping tasks.

All the pre-generated OTP codes are encrypted by REAL and stored as Delta Table (DT) inside the confidential Data File. Data File consists of DT and last HI value. The entire Data File are generated right after a user securely logins and activates the REAL Mobile OTP Token provisioning work. The confidential Data File is encrypted by server using a key (LK) generated from user's credential information (UC) as shown in the following pseudo code.

$$LK = \text{HMAC-SHA-1}(\text{HMAC-SHA-1}(UC, UC), UC)\,, \tag{7}$$

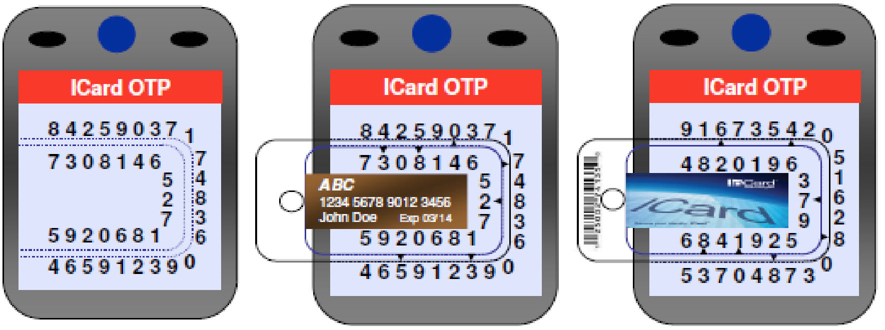
where UC is the user credential information such as the userID, password or PIN.

The encrypted Data File and Program File are zipped together after provisioning and can be downloaded into a user's cellular phone through a secured Internet connection. After the auto-installation, both the Program File and the encrypted confidential Data File are securely ported to the cellular phone.

4.3 Using REAL Mobile OTP Token

Once the Program File and Data File are properly installed, a user can activate the token through cell phone's screen or menu. After keying in the user credential information (UC), the phone will display a REAL Image as shown in Fig. 3.a. The user overlays and properly aligns her hardware token (always using front side first) on the screen to decrypt (rub) the OTP code as shown in Fig. 3.b. The first pointer points to a symbol N_{1f} with a value of 3. Since N_{1f} is an odd number, it means that we should use key on the front side to rub OTP code. If N_{1f} is an even number, we should use key on the back side.

OTP code rubbing sequence starts from the most top left symbol on outer ring and begins from the second pointer (pointing to a symbol N_2) location. N_i is $D_{(7-i)}$ of the OTP code. Following this method, we can find the OTP code as 807235. In Fig. 3.c, N_{1f} is 8. Using the back side code pointers, we find the OTP code to be 478818.



a. REAL Image

b. Rubbing (Decrypting) OTP Code (using front key)

c. Rubbing OTP Code (using back side key)

Fig. 3. a. A REAL Image contains an REAL encrypted OTP code. Phone's screen size, token's physical dimension, optimal symbol font size for ease of reading (rubbing) and the desirable card open space for server's Logo lead us to have a 40 symbols REAL Image. b. N_{1f} read from REAL hardware token's front side is an odd number 3. Front side of token is used to rub (decrypt) the OTP Code. The OTP code is 807235. c. N_{1f} read from front side of REAL hardware token is an even number 8. Back side of the token should be used to rub (decrypt) the OTP Code. The OTP code is 478818.

4.4 Token That Resists Certain Security Attacks

Man-in-the-middle (MITM) can be in the network to intercept a user's static password for next round use. This is the so called "Replay" attack. The OTP's dynamic password has successfully thwarted such attack [22]. REAL Mobile OTP Token (MR1 Token) also has the same capability to prevent such MITM replay attack.

Man-in-the-middle data intercept attack [23]. MITM can obtain series of OTP codes by continuingly intercepting a user's login password. Even with a dynamic password, a hacker can try to trace the seed (K) through the pseudo random sequence the captured passwords show. This is a direct attack on an OTP algorithm. If the malicious person has enough pseudo random number data base and computation power, a typical OTP token can not resist such attack effectively.

An improved REAL Mobile OTP Token can provide extra protection when such event happens. We name this version MR2 Token. Each user will be assigned with two sets of OTP generating seed (K_A and K_B). The user will use one of the two OTP tokens each time but in a mixed random order. The OTP codes generated by token A will be encrypted by the REAL key on front side of token. Token B will use the REAL key on the back side to encrypt and decrypt its OTP codes. So even though the user carries only one REAL Mobile OTP hardware token, he actually has two tokens with him all the time.

The first symbol (N_{1f}) pointed by the first pointer on the front side is still used as the indicator to select front and back side's key. During the REAL Image generation procedure, an odd number of N_{1f} is used when an OTP code is from token A and an even N_{1f} is used with an OTP code from token B. The authentication server will know the sequence of which token is used as the provisioning is provided by server itself. A user can operate as if she has only one token. The two pseudo random OTP codes are used in a mixed random order. So the intercepted OTP codes can no longer provide a meaningful pseudo random sequence. It makes random number seed tracing very difficult. This attack is then prevented.

Shoulder-surfing attack [24]. Shoulder-surfing attack happens when a malicious person secretly observes the action and screen while a user is using her OTP token. The malicious person may then have the REAL Image and code pointer information to retrace the secrecy or OTP code. He may not have the OTP code as the code usually displays in other non-numeric symbols during the login process.

To fend off such attack, we employ a random offset to decouple the direct relationship among pointer locations and OTP code symbols in REAL Image. Token of such feature is called MR3 Token. The first symbol (N_{1f}) pointed by the first pointer on the front side is still used as the indicator to select front and back side's key. But both N_{1f} and N_{1b} will be used as an adder to each symbol's numeric value pointed by the rest of the six code pointers. The ten's digit will be dropped if the added value is greater than or equal to 10. The general equation is as follows.

$$D_{if} = (\text{Value of } N_{1f} + \text{Value of } N_{(7-i)\,f}) \bmod 10\ , \tag{8}$$

$$D_{ib} = (\text{Value of } N_{1b} + \text{Value of } N_{(7-i)\,b}) \bmod 10\ , \tag{9}$$

where D_{if} is the ith digit of OTP code and $N_{(7-i)\ f}$ is the symbol pointed by its corresponding (7-i)th pointer when using REAL key on front side. When making the REAL Image, each $N_{(7-i)\ f}$ value should be adjusted according to equation (8).

In Fig. 3.b, the N_{1f} is equal to 3, an odd number. We use the key on front side of hardware token to rub the OTP code. The second pointer indicates $N_{2f} = 8$. From equation (8), we find $D_{5f} = 1$. Following this new decryption procedure, we have the full OTP code as 130568.

The REAL Image in Fig. 3.c shows $N_{1f} = 8$, an even number. We use REAL key on the back side to decrypt the OTP code. This code is actually generated from Token B. The first pointer on the back side shows a number $N_{1b} = 6$. We find $N_{2b} = 4$. So $D_{5b} = 0$. Following the procedure, we can obtain the full OTP Code as 034474.

Both N_{1f} and N_{1b} are randomly chosen and the modular operation also adds additional randomness to the codes. The codes obtained from Fig. 3.b and Fig 3.c show no physical direct relationship with the original code pointer locations. It then retains the security and resists the attack from Shoulder-surfing.

5 Analysis

5.1 Design Goals Review

For a 5,000 pre-generated OTP codes stored in a REAL Delta Table, the entire code size is about 200 KBytes. It is sufficient for a consecutive 2.7 years use with an average daily usage of 5 OTP codes. The entire software code size of REAL Mobile OTP Token is less than 3 Mbytes. Both software program deployment and technical support can be easily done through Internet. There is no need to change or add any hardware or software on the existing OATH authentication server. The hardware token can be made, delivery and activated following the same logistic like credit card. It helps to achieve easy deployment and low cost operation.

The token uses the same OATH OTP algorithm to generate the OTP codes. It ensures that the OTP codes are 100% compliant to any OATH authentication server. As long as both tokens use the same key (K) and counter value (C), REAL Mobile OTP Token can maintain full interoperability with other tokens. The token can directly replace any existing or expired OATH OTP token.

5.2 OTP Code Security and Integrity

The plaintext OTP code is generated by the same OATH algorithm. The REAL decrypted OTP code is as secure as the one generated by an OATH compatible token. A detailed OATH OTP code security analysis can be found in Appendix A of RFC4226 [5] for further reference.

REAL encrypted OTP codes are encrypted again by Hashed Index (HI) value when they are placed into Delta Table (DT). DT and HI value are further encrypted using a key (LK) generated by user credential data. It prevents these data from being tampered. They provide the desired level of integrity and security protection.

5.3 Real Image Security Level

REAL Image (RI) contains the OTP code. Its security level affects how secure the OTP code is protected. Given an image size of 40 (T) and a 6-digit (D) OTP code, equation (5) shows a total number of 37.2 million different code pointer patterns (REAL key) can be embedded on both sides of the hardware token. Without the aid of the hardware token and a known OTP code, to guess the correct key from a REAL Image, the possibility (P_1) is 1 out of 37.2 million. Even with a known REAL Image alone, equation (5) also shows that to directly guess the correct 6-digit OTP code the possibility (P_1) is 1 out of 3.8 million.

On the other hand, a traditional OATH token display the full OTP code on the screen without any encryption. The chance (P_1) to obtain the correct code from the image on the LCD screen will be 1 out of 1. If just considering the displayed image alone, REAL OTP Token's security level is much stronger than a traditional OATH token. In other word, REAL securely encrypts the 6-digit OATH OTP code inside the REAL Image. So REAL does not degrade the OATH OTP code security level.

5.4 Security Attacks

Man-in-the-middle (MITM) Replay Attack is a decades old problem. A dynamic password from an OTP token such as our MR1 token can successfully thwart such attack [22]. The MR2 is a two OTP tokens in one physical token form. It can send a stream of OTP codes generated from two different OTP tokens in a mixed random order. The intercepted OTP codes by MITM are just a randomly mixed number series. The original pseudo random sequence of an OTP generating algorithm is broken and difficult to trace. So MR2 token can resist the OTP seed tracing by MITM interception attack. The MR3 token randomly decouples the REAL key from its hardware token code pointer's physical locations. It effectively breaks the key and pointers' physical linking information that a malicious person tries to get. The MR3 token thus successfully resist the attack from Shoulder-surfing or Seed tracing by MITM code interception.

5.5 Other Security Concerns

Cellular phone is lost or stolen. Cellular phone is small and prone to get lost or stolen. A malicious person has to crack the user's credential first before activating the REAL OTP function on a cellular phone. Even if the cellular phone is activated, trying to correctly guess the OTP code with a known REAL Image but without the specific hardware token, the possibility (P_1) is 1 out of 3.8 million. It is much tougher than a brute force guess the 6-digit OTP code (1 out of 1 million).

Hardware token is lost, stolen or secretly copied. A regular software or hardware token is a standalone OTP code generator. The loss of such token means losing all the future OTP codes. REAL hardware token does not contain any electronic to generate OTP code. The token will only work with a specific cell phone that has the user Data File. So losing a REAL hardware token or the pointer pattern being photo copied, the security will not be compromised. Having hardware token alone, the possibility (P_2) to correctly guess the OTP code is like to guess a correct REAL Image. The possibility (P_2) is 1 out of 10^{40}. It is 10^{34} times tougher than a brute force guess of the 6-digit OTP code.

Confidential File is secretly copied or stolen. A malicious person may copy the user specific confidential Data File without user's knowledge. The person can then set up a cell phone that imitates the user's REAL Mobile OTP token environment. It can generate correct REAL Image independently. But the intruder will face the difficulty of guessing the correct OTP code without the hardware token even if user credential information is correctly guessed. Intruder will then meet the difficulty of probability P_1 (1 out of 3.8 million) that we discussed previously.

Limitations. Security attack technique advances daily. Trojan and MITM attack though not found in the cellular phone today, they have successfully infiltrated the PCs world and created huge damages. The existing 2FA system alone can not solve this issue [25]. Layered security solutions and discipline are needed. REAL Mobile OTP Token will become part of the security solutions for this new challenge.

6 Conclusion and Future Work

Using cellular phone as a One-time Password (OTP) token to generate dynamic session password is becoming popular recently. But it has met certain difficulties. Some of the solutions can not fully prevent the OTP seed tracing by MITM code interception or Shoulder-surfing security attacks. Other have issues regarding poor compliance and interoperability with existing authentication infrastructure, geographical limitation due to poor or no cellular service, plus high deployment and support cost. In particular, many of the implementations may comprise the security when phone is lost, stolen or data file is secretly copied.

A Rubbing Encryption Algorithm (REAL) is used as the base cipher for a new Mobile OTP Token. REAL decryption does not require entering encryption key on local computing device. It allows REAL to use a long and complex key to encrypt a short word length plaintext such as the OTP codes. This feature ensures high level of security on REAL ciphertext (REAL Image). The ciphertext can be securely stored in a cellular phone even if the device gets lost or stolen. A user can use an inexpensive non-electronic plastic hardware token to electronically "rub" (decrypt) the OTP code from the phone screen's REAL Image. We implement and analyze an OATH Compliant Mobile OTP Token using REAL. This token is in compliant and interoperable with existing authentication infrastructure. Token's deployment and support is easy and with low cost. The token also has capability to resist the MITM data interception to trace OTP generating seed. Furthermore, it can resist the Shoulder-surfing attack as well.

REAL can be used in multiple dimension form factor also. It has potential to be used in other applications with new implementations [26]. REAL can also be further enhanced on its security strength. These are the future work for us.

Acknowledgments. We will like to thank Innovative Card Technologies, Inc. for their diligent efforts on marketing the OATH Compliant REAL Mobile OTP Token. Their feedback provided many important inputs on algorithm and token design fine tuning, enhancement and direction for future improvement.

References

1. RSA.: RSA SecureID, Software Authenticator, `http://www.rsa.com/node.aspx?id=1313`
2. Mizuno, S., Yamada, K., Takahashi, K.: Authentication Using Multiple Communication Channels. In: DIM 2005, November 11 (2005)
3. Kostiainen, K., Ekberg, J.E., Asokan, N.: On-board Credentials with Open Provisioning. In: ASIACCS 2009 (March 2009)
4. Wikipedia.: Two-factor Authentication – Challenges, `http://en.wikipidia.org/wiki/two-factor_authentication`
5. M'Raihi, D., Bellare, M., Hoornaert, F., Naccache, D. Ranen, O.: HOTP: An HMAC-Based One-time Password Algorithm, The Internet Society, Network Working Group. RFC4226 (December 2005)
6. Initiative for Open AuTHentication.: Oath Vision, `http://www.openauthentication.org/about`
7. Verisign.: Authentication for Business Partners and the Mobile Workforce, `http://www.verisign.com/authentication/enterprise-authentication/enterprise-otp/`
8. Deepnet Security: MobileID - A Mbile, To-way and To-factor Athentication, `http://www.deepnetsecurity.com/products2/MobileID.asp`
9. Aloul, F., Zahidi, S., El-Hajj, W.: Two Factor Authentication Using Mobile Phones. In: 2009 IEEE/ACS International Conference on Computer Systems and Applications (2009)
10. Liao, K., Sung, M., Lee, W., Lin, T.: A One-Time Password Scheme with QR-Code Based on Mobile Phone, doi: 10.1109/NCM.2009.324
11. Liberty Aliance Project: Liberty Alliance, `http://www.projectliberty.org/`
12. FreeAuthProject.: The FreeAuth Project, `http://www.freeauth.org/site`
13. Abe, T., Itosh, H., Takahashi, K.: Implementing Identity Provider on Mobile Phone. In: DIM 2007, November 2 (2007)
14. Haverinen, H., Asokan, N., Maattanen, T.: Authentication and Key Generation for Mobile IP Using GSM Authentication and Roaming. In: ICC 2001 (2001)
15. Hallsteinsen, S., Jorstad, I., Thanh, D.: Using the Mobile Phone as s Security Token for Unified Authentication. In: ICSNC 2007. IEEE Computer Society, Los Alamitos (2007)
16. Thanh, D., Jonvik, T., Feng, B., Thuan, D., Jorstad, I.: Simple Strong Authentication for Internet Applications Using Mobile Phones. IEEE GLOBECOM (2008)
17. Wangensteen, A., Lunde, L., Jorstad, I., Thanh, D.: A Generic Authentication System Based on SIM. In: The International Conference on Internet Surveillance and Protection, ICISP 2006 (2006)
18. Thanh, D., Jonvik, T., Thuan, D., Jorstad, I.: Enhancing Internet Service Security Using GSM SIM Authentication. In: IEEE GLOBECOM (2006)
19. Stinson, D.: Cryptography – Theory and Practice, pp. 44–67. CRC Press, Boca Raton (1995)
20. Shastri, A., Govil, R.: Optimal Discrete Entropy. Applied Mathematics E-Notes 1, 73–76 (2001)
21. International Organization for Standardization.: ISO/IEC 7810:2003. November 17 (2009)
22. Lamport, L.: Password Authentication with Insecure Communication. Communications of the ACM 24(11), 770–772 (1981)

23. Wikipedia.: Man-in-the-middle Attack (April 15, 2010), `http://en.wikipedia.org/wiki/Man_in_the_middle_attack`
24. Wikipedia.: Shoulder Surfing (Computer Security) (April 15, 2010), `http://en.wikipedia.org/wiki/Shoulder_surfing_computer_security`
25. Schneier, B.: The Failure of Two-factor Authentication. Communications of the ACM (April 2005)
26. Cheng, F.: A Novel Rubbing Encryption Algorithm and The Implementation of the Web-based One-time Password Token. In: COMPSAC 2010, July 19 (2010)

ISI and ICI Suppression for Mobile OFDM System by Using a Hybrid 2-Layer Diversity Receiver

Jing Gao and Tomohisa Wada

Graduate School of Engineering and Science, University of the Ryukyus,
1 Senbaru, Nishihara, Okinawa, 903-0213, Japan
gaojing722@yahoo.com, wada@ie.u-ryukyu.ac.jp

Abstract. An OFDM system is very sensitive to orthogonality relation. For a mobile wireless system, it is impossible to avoid Doppler-induced inter carrier interference (ICI). Moreover, while beyond guard interval delayed signal exists in channel, the delay-induced ICI and inter symbol interference (ISI) will occur. Although a conventional carrier diversity (CD) receiver can render the OFDM system less sensitive to the white-noise-likely ICI, but it requires significant channel knowledge. On the other hand, a pre-FFT adaptive array (AA) receiver can improve the instantaneous signal to interference-and-noise ratio (SINR) at the input of FFT. In this paper, a hybrid AA/CD two layers receiver is investigated not only on a tradeoff between high performance and low complexity, but also into a method for both ISI and ICI suppression. Simulation results show that a suitable combination of pre-FFT AA and post-FFT CD can provide good performance by comparison with conventional OFDM receiver in mobile wireless channel, especially, while ISI and ICI occur at the same time.

Keywords: beyond GI delay, Doppler, adaptive array, carrier diversity.

1 Introduction

An orthogonal frequency division multiplexing (OFDM) is adopted as a modulation method for wireless communication system because it is robust to frequency selective fading due to the using of guard interval (GI) [1]. However, while the beyond GI delayed signal exists, not only inter-symbol interference (ISI) but also inter-carrier interference (delay-ICI, namely) occurs. In mobile application, a Doppler spread results in inter-carrier interference (Doppler-ICI, to distinguish from delay-ICI) [2]. Since these effects degrade the OFDM signal, it is a severe challenge to increase the system performance and the accuracy of channel estimation. As well known, an OFDM system is very sensitive to the quality of channel estimation, and apart from the FFT, which is the most complex unit of the receiver [3]. In [4]-[6], a post-FFT carrier diversity (CD) combiner and a post-FFT adaptive array (AA) for interference and noise suppression have been proposed. Although the proposed post-FFT CD and AA type combiners can optimize signal-to-interference-and-noise (SINR), it is costly to implement such a multi-antenna-multi-FFT receiver. In [7], a pre-FFT adaptive array (AA) was proposed for suppressing the beyond GI delayed signal based on the maximized SINR and the minimum mean square error (MMSE) criteria in time domain, and authors gave the optimum array weights. Otherwise, only one-FFT-one-branch was considered as the receiver in [7].

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 17–28, 2010.

In this paper, a hybrid time-domain AA and frequency-domain CD two-layer multi-antenna receiver is proposed for a tradeoff between high-performance and low-complexity. Benefiting by the AA layer, the hybrid receiver can halve the number of CD branches, and improve the channel estimation quality through the depressing of maximum excess delay (τ_{max}) [9]. The proposed AA/CD receiver is studied based on the AA criteria of maxi-ratio combining (MRC) and MMSE [8], while the CD combiner exploits MRC and equal gain combining (EGC) schemes. Therefore, total four approaches of the hybrid receiver are studied by focusing on vehicle mobile multi-path application.

This paper is organized as follows. Section II introduces the proposed hybrid AA/CD receiver. Section III discusses the approach of AA/CD combination for ISI and ICI suppression. Section IV presents the performance of the proposed receiver as compared to a conventional CD receiver by computer simulation. Conclusion is given in section V.

2 Hybrid AA/CD Receiver

The proposed hybrid AA/CD receiver is structured on two layers of time-domain AA and frequency-domain CD with channel estimations as shown in Fig. 1. There are *B* (>*1*) sets of AA which consist of a number of *A* (>*1*) antennas. The *B* sets of AA should be located far each other so that uncorrelated CD can be achieved. Inversely, the elements among each AA set must be configured close enough, hence strong correlation AA combining can be easily provided. Here, we consider an OFDM system using an inverse fast Fourier transform (IFFT) of length *N* with subcarrier spacing f_0. By defining the sampling duration as ($T_c=1/Nf_0$) and the length of guard interval (GI) as $G=N/8$ samples, then the length of resulting baseband OFDM symbol is ($N_s=G+N$) or ($T_s=T_g+T$) in time. The GI is a copy from the last part of the effective symbol. Throughout this paper, the added GI is referred to as "*h-GI,*" the original part is distinguished as "*t-GI.*"

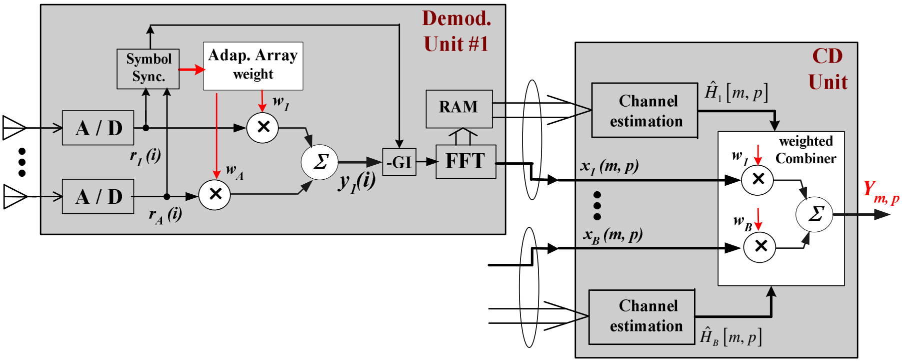


Fig. 1. Diagram block of the hybrid adaptive array and carrier diversity two layers receiver

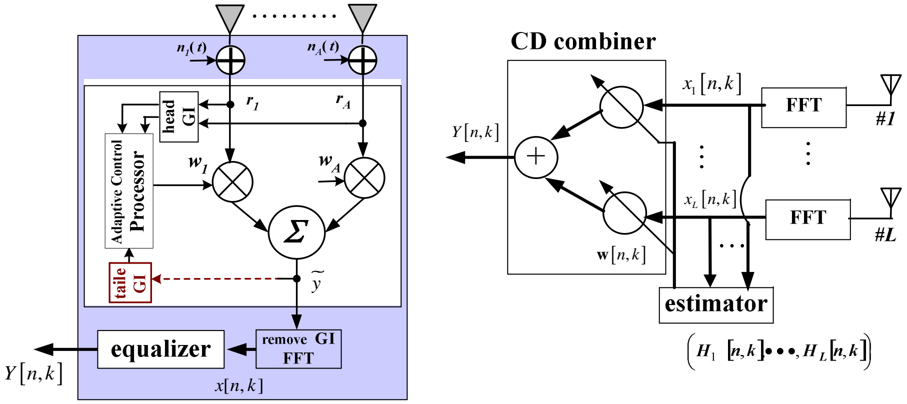
Assuming that at least one beyond GI delayed signal exits in the multipath channel excepting the synchronizing desired one. The receiver input signal of i^{th} sampling during the m^{th} OFDM symbol at a^{th} element can be written as

$$r_a(m,i) = \sum_{n=-\infty}^{+\infty} \sum_{k=0}^{N-1} d_{n,k} e^{j2\pi k f_0 \{(m-n)T_s + iT_c\}} h_a(m,i,n,k) + n_a(i) . \tag{1}$$

where $d_{n,k}$ is the data symbol modulating the k^{th} tone during the n^{th} OFDM symbol. $h_a(m,i,n,k)$ is the multipath channel impulse response (CIR) by taking the transmission filter into account. $n_a(i)$ is additive white Gaussian noise (AWGN).

2.1 Time-Domain Adaptive Array (AA)

A multi-antenna adaptive array (AA) can improve the SINR by the performing of spatial filtering to desired/undesired path-signal in time domain. Moreover, in an OFDM system, this can increase the accuracy of the channel estimation after FFT, so a more effective channel equalization (EQ) is available. In addition, since each OFDM symbol only needs one weighted vector $\{w_i(t)\}$, the AA is an attractive solution due to low computation complexity. However, a high correlation between the antenna signals is preferred. Fig. 2 (a) shows the AA structure.



(a) Adaptive Array Receiver **(b) Carrier Diversity Receiver**

Fig. 2. Two types of OFDM receiver

Two AA algorithms are used, the one is maximizing ratio combining (MRC) algorithm, which exploits the *h-GI* and *t-GI* of the same OFDM symbol [8] [10]. The other one is the sample matrix inversion (SMI), which is based on MMSE criteria. By defining the weight vector of the *A* elements array as

$$\mathbf{w} = [w_1, w_2, ..., w_A]^T . \tag{2}$$

and the (A x N_s) received signal vector is expressed as

$$\mathbf{r}(i) = \left[r_1(i), r_2(i), \ldots, r_A(i)\right]^T, \quad (-G \le i \le N). \tag{3}$$

The combining output of AA can be written as

$$y(i) = \mathbf{w}^H \mathbf{r}. \tag{4}$$

where above and follow, the superscripts T , H and * denote transposing, conjugate-transposing and conjugating operator respectively. Then, weighting vector of MRC can be derived as

$$\mathbf{w}_{MRC} = E\left[\mathbf{r}_h(i)\, y_t^*(i)\right]. \tag{5}$$

where *E[-]* stands for expectation function, and ***r**$_h$**(i)*** denotes *h-GI* of received signal, $y_t(i)$ is the *t-GI* of array output. For SMI algorithm, by defining the (G x G) received signal auto correlation matrix as

$$\mathbf{R}_{rr} = E\left[\mathbf{r}_h \mathbf{r}_h^H\right]. \tag{6}$$

then the weighting vector of SMI is given as

$$\mathbf{w}_{SMI} = \mathbf{R}_{rr}^{-1} E\left[\mathbf{r}_h(i)\, y_t^*(i)\right]. \tag{7}$$

The inversion of autocorrelation matrix $\mathbf{R}_{rr}$ performs null-steering to the interference. As this, the SMI scheme is more effective than MRC on undesired signal suppressing.

2.2 Frequency-Domain Carrier Diversity (CD)

A CD scheme utilizes a few branches of independent OFDM signal for subcarrier-by-subcarrier diversity combining in frequency domain. It can modify the SNR on subcarrier base after FFT. However, it requires accurate channel estimation for high performance. As shown in Fig. 2 (b), L branches of independent post-FFT signal are combined. The subcarrier signal from the l^{th} branch at the p^{th} tone of the m^{th} OFDM symbol can be written as

$$x_l(m,p) = d_{m,p} H_l(m,p) + n_l(m,p). \tag{8}$$

where $n_l(m, p)$ is additive white Gaussian noise (AWGN) from the l^{th} branch. $H_l(m,p)$ is the channel transfer function (CTF), which is independent for different branch. $d_{m,p}$ is the transmitted complex signal modulating the p^{th} tone of m^{th} symbol. The derived MRC and EGC weight of the l^{th} branch can be written as

$$\text{MRC:} \quad w_l(m,p) = \frac{H_l^*(m,p)}{\sum_{l=1}^{L} \left|H_l(m,p)\right|^2}. \tag{9}$$

$$\text{SMI:} \quad w_l(m,p) = \frac{H_l^*(m,p)}{\left|H_l(m,p)\right| \sum_{l=1}^{L} \left|H_l(m,p)\right|}. \tag{10}$$

The data $d_{m,p}$ can be estimated as $\widehat{Y}_{m,p}$ by the CD combiner

$$\widehat{Y}_{m,p} = \sum_{l=1}^{L} w_l\left(m,p\right) x_l\left(m,p\right). \tag{11}$$

It is worth noting that, the MRC diversity combining are weighted corresponding to the instantaneous signal power of each branch, while the EGC selects equal gain factors.

2.3 Channel Estimation

For scattered pilots based 2-dimesion (2D) channel estimation, the CTF at the pilot $d_{m,p}$ can be obtained as

$$H_{m,p} = \frac{x\left(m,p\right)}{d_{m,p}}. \tag{12}$$

Since the performance of OFDM receiver is very sensitive to channel estimation, in this paper, to evaluate performance, the same estimation method is used in both the proposed hybrid and conventional receiver. 2-dimesion (2D) channel estimation based on the scattered pilots (SPs, box with P) is separated into symbol and subcarrier direction estimation as shown in Fig. 3. At first, the CTF of the tone with black circle is estimated using 2-tap linear interpolation in symbol direction. Then, in subcarrier direction, after down/up sampling by factor of 3, a 36-tap Window-sinc-filter is used to estimate the CTF at the position with grey circle.

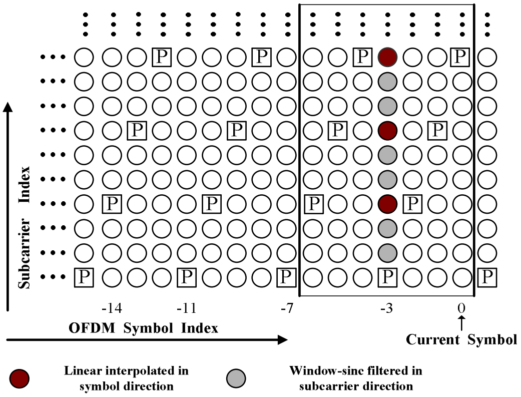

P
Subcarrier Index
-14
-11
-7
-3
0
OFDM Symbol Index
Current Symbol
Linear interpolated in symbol direction
Window-sinc filtered in subcarrier direction

Fig. 3. The scattered pilot pattern and the interpolation zone

3 The Approach of AA/CD Combination for ISI/ICI Suppression

In section 2, the two AA schemes (MRC and SMI) and the two CD combiners (MRC and EGC) were described. Our purpose here is to find the high performance hybrid joint of the AA and CD even in hard conditions of fading channel.

3.1 Using the AA for ISI Suppression

In a multipath channel, once the beyond GI delayed signal exists, the ISI and delay-induced ICI will occur at the same time. After FFT-demodulating such an antenna signal, the modified data symbol in synchronized m^{th} OFDM block can be written as

$$x_a(m,p) = H(m,p)\,d_{m,p} + \sum_{\substack{k=0\\ k\neq p}}^{N-1} H(m,k)\,d_{m,k} + \sum_{k=0}^{N-1} e^{j2\pi k f_0 T_s} H(m-1,k)\,d_{m-1,k}\,. \tag{13}$$

where the 1^{st} right term is the desired contribution of data $d_{m,p}$ in the m^{th} block transmitted over multipath. The rest two right terms are the contribution due to beyond GI delayed path, and the 2^{nd} right term denotes the delay-ICI that rose from the destroyed orthogonality among tones of the m^{th} block, the 3^{rd} right term includes the ISI and delay-ICI components caused by the tones in the *(m-1)*th block. The power of the delay-ICI and ISI is proportional to the power of the beyond-GI path signal in a positive factor that is less than one [11]. For combating with this, using AA to depress the delayed path signal in time-domain is an effective method.

An 8K-point-FFT OFDM signal with GI length of (*T/8*) is used for computation testing the two AA schemes (SMI and MRC). The channel with 5-path delayed signal (arrived at 10^0/0dB, 70^0/3dB, 170^0/1dB, 270^0/2dB, 350^0/4dB) is divided to two cases. The one is a normal short-delay channel (notated as "S_CH"), all of the five paths signal in which are transmitted within GI duration (at 350^0/4dB with $\tau = 48T_g/128$). The second one is a long-delay channel (notated as "L_CH"), in which one beyond-GI delayed signal exists (at 350^0/4dB with $\tau_{max} = 129T_g/128$).

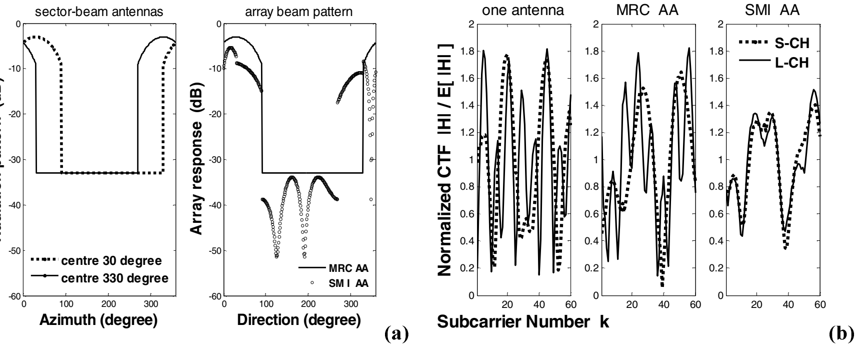


Fig. 4. (a) The radiation pattern of two 120-degree sector-beam antennas and the beam patterns formed by MRC/SMI adaptive array (AA); (b) the varying of channel transfer function (CTF) versus subcarrier index in short-delay channel ("S-CH") and long-delay channel ("L-CH")

Fig. 4 (a) shows the radio patterns (RP) of the two 120-degree sector-beam antennas and the array beam-pattern (BP) of them under 'L-CH" condition. The desired signal is at 10^0. Differing from the MRC, the SMI scheme AA gives the lower relative side-lobe-level and deepest nulls toward undesired path. Fig. 4 (b) shows the CTF varying (|*H*| / *E*[|*H*|]) over continual 60 subcarriers by using different AA methods. Clearly, the SMI AA can greatly improve the dispersion in subcarrier direction by comparison with the MRC scheme, especially, while large long delayed signal exits (referring to condition “L-CH”). This means that, by using the SMI scheme, not only the ISI and delay-ICI can be depressed, but also the more accurate CTF estimation in subcarrier direction can be achieved.

3.2 Using the CD Combining for Doppler-ICI Suppression

As shown above, the AA used as spatial filter can suppress the delay-ICI and ISI by steering the nulls toward the large delayed path signals. However, for mobile channel, the AA receiver cannot render the OFDM signal less sensitive to Doppler shift. In this section, the frequency domain CD combiner is investigated on Doppler-ICI suppression. With assumption of that only Doppler effects are considered, the FFT demodulated OFDM signal on the p^{th} subcarrier of m^{th} symbol can be written as

$$\begin{aligned} x_l(m,p) &= d(m,p)\frac{\sin \pi f_{Dl}T}{N\sin\left(\pi f_{Dl}T/N\right)} e^{j\pi f_{Dl}T\frac{N-1}{N}} + \sum_{k=0,k\neq p}^{N-1} d(m,k)\, I^l_{k-p} \\ &= d(m,p) I_{0l} + I_l(m,p)\,. \end{aligned} \tag{14}$$

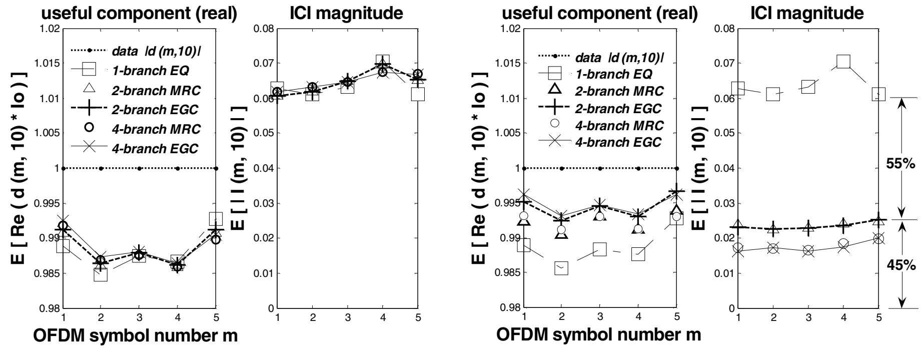
where, l denotes the CD branch number, I^l_{k-p} are the ICI contribution coefficients from the k^{th} subcarrier [12]. The term I_{0l} impaired the desired data $d(m,p)$ by an amplitude reduction and phase shift due to the Doppler frequency f_{Dl}. The component $I_l(m,p)$ is ICI, which smears the useful component $d(m,p)I_{0l}$ with white-likely noise [13-14].

At scattered pilot (SP) positions, by using the CTF estimated from (14) and (12), the CD weighting determined by (8) and (9) can be rewritten as $w_l=H_l^*/\alpha_l$ where α_l is a real coefficient. Then, the CD combining output is obtained from equation (11) as

$$\begin{aligned} Y_{m,p} &= \sum_l \frac{1}{\alpha_l} H_l^*(m,p)\, x_l(m,p) \\ &= d(m,p)\sum_l \frac{1}{\alpha_l}\left(\left|I_{0l}\right|^2 + \frac{I_l^*(m,p)}{d^*(m,p)} I_{0l}\right) + \sum_l \frac{1}{\alpha_l}\left(\frac{\left|I_l(m,p)\right|^2}{d^*(m,p)} + I_{0l}^* I_l(m,p)\right) \\ &= d(m,p) I_0 + I(m,p)\,. \end{aligned} \tag{15}$$

where the term $d(m,p)I_0$ and $I(m,p)$ are the combined useful component and ICI component, respectively. In mobile application, since the Doppler coefficient f_{Dl}T of l^{th} branch can be positive or negative according to the relationship between the directions of vehicle motion and signal arriving, therefore the CD performance from which Doppler branches will be different.

Assuming here, the continual five symbols of OFDM signal modulated by 64-point FFT are received from four Doppler branches with random coefficients (f_{D1}T, f_{D2}T, f_{D3}T, f_{D4}T). The time-variant phase is assumed to be ($2\pi f_D T_s/10$) from one OFDM symbol to the next. In Fig. 5, the two branches (f_{D1}T, f_{D3}T) CD (2-branch MRC/EGC) and the four branches CD (4-branch MRC/ EGC) combiners are compared with one branch ($f_{D1}T$) equalizing (1-branch EQ) on the term of useful component real part and ICI magnitude. In Fig. 5 (a), the CD is combined with positive Doppler coefficient {$f_{Dl}T$} branches. It means that only the branch signals arriving from the forward



(a): $0 < (f_{D1}T,\ f_{D2}T,\ f_{D3}T,\ f_{D4}T) < 0.1$ (b): $-0.1 < (f_{D1}T,\ f_{D2}T,\ f_{D3}T,\ f_{D4}T) < 0.1$

Fig. 5. Expected average of the useful component real part Re(*d(*m,10*)*Io*) and the ICI component magnitude |*I(*m,10*)*| at subcarrier position *d(*m,10*)* after CD combining. The transmitted complex data sequences by the continual five OFDM symbols are randomly set as {|*d(*m,k*)*|<$2^{0.5}$*; m*=1,2,...,5; *k*=0,1,...,63} excepting *d(*m,10*)=1+0j*. The two positions *d(*1,10*)* and *d(*5,10*)* are the pilots. The CTFs of other three data positions ($H_{2,10}$, $H_{3,10}$, $H_{4,10}$) are estimated by symbol direction linear-interpolation as shown in Fig. 3. The CD combining "2/4-branch MRC/EGC" is performed over (a) positive random Doppler coefficient branches; (b) positive/or negative random Doppler coefficient branches.

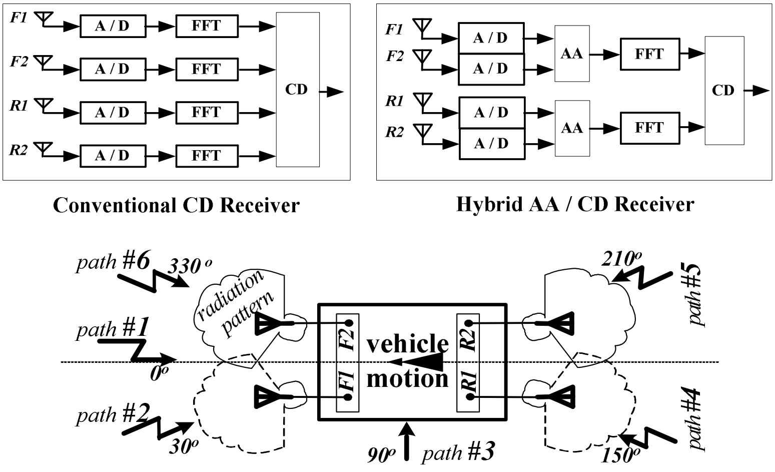
direction of vehicle moving are combined on subcarrier basis. In Fig. 5 (b), the CD is performed over the received signals with forward or rear arriving directions.

It is obvious that, the CD combining over only positive Doppler branches (Fig. 5 a) shows little ICI-suppression as good as the 1-branch EQ, while the combining over positive/negative Doppler branches (Fig. 5 b) can depress the ICI effectively (by 55% relative to 1-branch EQ). In addition, for ICI-suppression, the MRC and EGC carrier diversity are about the same, the 4-branch CD just gives a little improvement by comparison with the 2-branch CD. The rest combining error is mainly due to the error of linear interpolation CTF estimation in symbol index.

4 Computation Simulation Results

In this section, the bit error rate (BER) performance of the proposed hybrid AA/CD receiver is shown as compared with a conventional post-FFT CD receiver without error correction.

The upside of Fig. 6 shows the block diagrams of the two receiver models: the conventional CD and the hybrid AA/CD receiver. They are based on four antennas F_1, F_2, R_1 and R_2. The spacing between F_1 and F_2, R_1 and R_2 is half of the carry wavelength, while the pair of {F_1, F_2} is far from the {R_1, R_2}. The conventional CD receiver performs the MRC criteria (convention cd-MRC). For the hybrid receiver, the two AA weighting sets are determined using the MRC or SMI scheme, and the post-FFT CD performs the MRC or EGC combining. In this paper, following notations are used to denote the hybrid configurations: "hybrid aa-SMI/cd-EGC", "hybrid aa-SMI/cd-MRC", "hybrid aa-MRC/cd-EGC" and "hybrid aa-MRC/cd-MRC".



Conventional CD Receiver

Hybrid AA / CD Receiver

Vehicular Aerial Deployment and RP

Fig. 6. The simulated receiver models and the vehicular antenna radiation patterns (RP)

Since car antenna is typically mounted on windshield glass, its radiation pattern (RP) is directionally constrained due to metal of car body. As illustrated in the south of Fig. 6, the distorted RP of the two front antennas {F_1, F_2} concentrates on the forward direction, while the two rear ones {R_1, R_2} focuses to the rear direction of vehicle moving. Here, their centre directions are assumed to be (30°, 330°) and (150°, 210°), respectively, and their half-power-beam-width is the same 120-degree (see Fig. 4 a). When AA is applied, the antenna set with similar RP is chosen.

4.1 System Parameters and Channel Profile

Table 1 shows the simulation system parameters. Mode3 of the ISDB-T standard with 64QAM digital modulation is used. Table 2 summaries three channel models with typical six path delayed signals (TU6) for the simulation. The D/U is the desired signal (path#1) to delayed signal power ratio. In channel I an II, all of the signals arrive within GI duration. In channel III, one beyond GI delayed signal (#6) exists.

Table 1. System parameters

Carrier frequency	f_c	563.143MHz (*UHF-28ch*)
Subcarrier spacing	F_0	0.992 kHz
Number of carriers	N	8192
Number of effective carriers	N_e	5617
Effective symbol duration	T_e	1008 us
Guard interval duration	T_g	(1/8)T_e
Digital modulation		64QAM

Table 2. Simulation channel

Path	D/U	AOA	Delay time		
	(dB)	(degree)	Channel-I	Channel-II	Channel-III
#1	0	10	0.01*(Tg/8)		
#2	3	90	3.0*(Tg/8)		
#3	5	170	6.0*(Tg/8)		
#4	1.5	190	0.5*(Tg/8)		
#5	2	270	1.0*(Tg/8)		
#6	4	350	3.0*(Tg/8)	5.5*(Tg/8)	9.0*(Tg/8)

4.2 Simulation Results and Discussion

Fig. 7 shows the BER performance for mobile application in channel I under SNR=20dB, SNR=25dB and SNR=35dB. The “aa-MRC/cd-MRC” approach of hybrid receiver is shown as compared to the conventional CD receiver. Since the correlation between antenna signals is low (due to the distorted RP as shown in Fig.6), the performance of AA is weakened but the CD is enhanced, therefore the conventional cd-MRC shows a better performance on noise and Doppler-ICI suppressing through the two more CD branches as shown in Fig. 7.

Channel II and III are set as a low AWGN channel with SNR=35dB. Figs. 8 and 9 show the BER versus maximum Doppler shifts for the four approaches of hybrid AA/CD receiver in channel II and III, respectively.

In Fig. 8, when channel is fast fading ($f_{Doppler} \geqq 40H_Z$), the four types of hybrid AA/CD receiver show the similar robustness to Doppler shifts. Otherwise, when the channel is slow fading, the aa-SMI/cd-EGC and the aa-SMI/cd-MRC methods can improve the performance of hybrid receiver significantly. This is because the SMI AA performs both the beam forming and null steering to suppress undesired signals. The conventional cd-MRC receiver shows a little more robustness to Doppler than the hybrid one due to ICI-suppression benefiting by the two more CD branches.

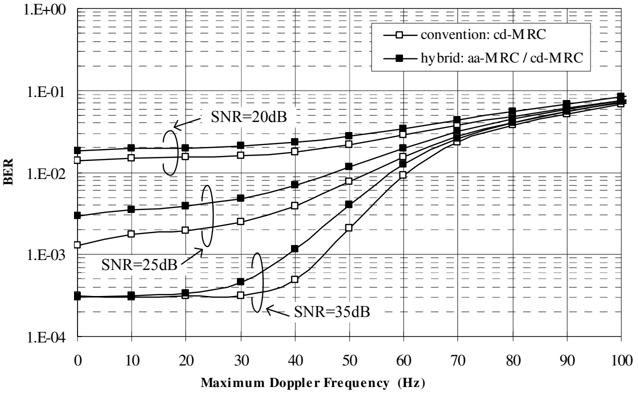


Fig. 7. BER versus maximum Doppler shift in Channel-I

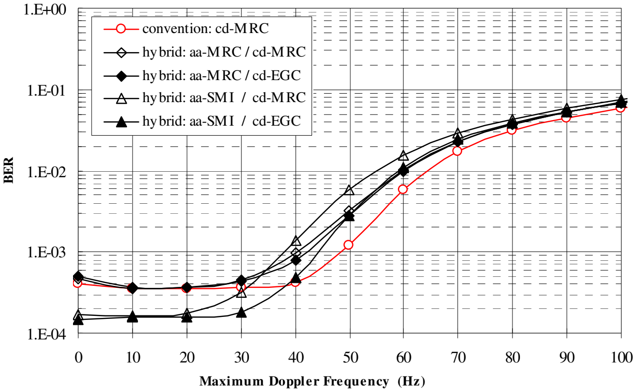


Fig. 8. BER versus maximum Doppler shift in Channel-II for SNR=35dB

In Fig. 9, since a beyond GI delayed signal exists in channel III, a significant degradation of the BER performance occurred for the conventional CD receiver and the hybrid AA/CD receiver with MRC AA scheme. This degradation is caused by ISI and delay-ICI. However, the aa-SMI/cd-EGC and aa-SMI/cd-MRC methods of the hybrid receiver show good performance. This is because the beyond GI delayed signal is suppressed effectively by the SMI AA.

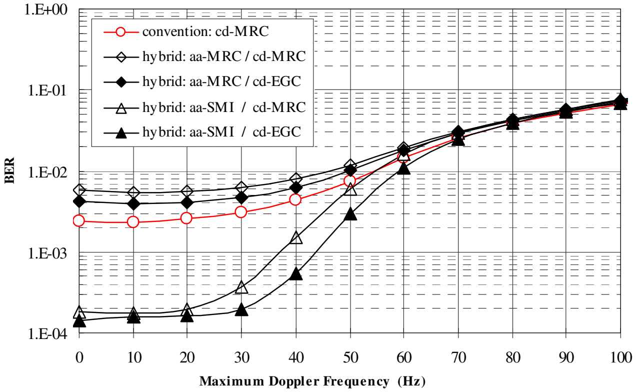


Fig. 9. BER versus maximum Doppler shift in Channel-III with one beyond GI delayed signal for SNR=35dB

5 Conclusion

It is a severe challenge to implement a conventional antenna-branch based post-FFT subcarrier diversity (CD) receiver in reasons of the system cost and the achievable accuracy of channel estimation while the ISI and ICI occur. On the other hand, a pre-FFT adaptive array (AA) OFDM receiver can depress the ISI and long-delay-induced ICI in time domain, and reduce the number of the required FFT and estimators.

In this paper, a hybrid AA/CD two layers receiver, which can halve the number of CD branches by comparison with the conventional post-FFT CD receiver, is proposed and analyzed. For a good tradeoff between high performance and low complexity, joint the AA scheme of MRC (/or SMI) and the CD combiners of MRC (/or EGC), total four approaches of the hybrid receiver are studied for ISI and ICI suppression by computer simulation. Although the conventional CD receiver suffers large and beyond-GI delayed multi-path condition, a hybrid AA/CD receiver with SMI AA scheme shows a little performance degradation. This approach can effectively increase the accuracy of CTF estimating in subcarrier axis through suppressing the large delayed path signals, thereby increasing the CD performance in mobile multi-path channel by comparison with other receivers, especially at relatively low $f_D T$.

References

1. Bingham, J.A.C.: Mlticarrier modulation for data transmission: An idea whose time has come. IEEE Commun. Mag. 28, 5–14 (1950)
2. Clerk Maxwell, J.: The Doppler spread effect. IEEE Treaties on Electricity and Magnetism 2, 68–73 (1992)
3. Speth, M., Fechtel, S., Fock, G., Meyr, H.: Broadband transmission using OFDM: system Performance and receiver Complexity. The work was supported by the deutsche forschungsgemein-schaft under contract No. Me 651/14-1 (in Germany)
4. Li, Y.G., Cimini Jr., L.J., Sollenberger, N.R.: Robust channel estimation for OFDM systems with rapid dipersive fading channels. IEEE Transactions on communications 46, 902–915 (1998)
5. Li, Y.G., Sollenberger, N.R.: Adaptive antenna arrays for OFDM system with cochannel interference. IEEE Trans. Commun. 47, 217–229 (1999)
6. Rashid, F., Manikas, A.: Diversity reception for OFDM systems using antenna arrays. IEEE Trans. Commun. 0-7803-9410-0/06 (2006)
7. Budsabathon, M., Hara, Y., Hara, S.: Optimum beamforming for Pre-FFT OFDM adaptive antenna array. IEEE Trans. On Vehicular Technology 53, 945–955 (2004)
8. Sathish, Chandran (Ed.).: Adaptive Antenna Arrays (Trends and Applications). Constrained Adaptive Filters, pp. 42–62. Springer, Heidelberg (2004)
9. Sklar, B.: Digital Communications, 2nd edn., pp. 958–974. 15-3, 4. PH PTR (2001)
10. Hori, S., Kikuma, N., Wada, T., Fujimoto, M.: Experimental study on array beamforming utilizing the guard interval in OFDM. In: International Symposium on Antennas and propagation ISAP 2005, Korea, pp. 257–260 (August 2005)
11. Speth, M., Fechtel, S.A.: Optimum Receiver Design for Wireless Broad-Band Systems Using OFDM—Part I. IEEE Trans. 47(11) (November 1999)
12. Armstrong, J.: Analysis of New and Existing Methods of Reducing Intercarrier Interference Due to Carrier Frequency Offset in OFDM. IEEE Trans. on Communications 47(3) (March 1999)
13. Paul, H.: A Technique for Orthogonal Frequency Division Multiplexing frequency Offset Correction. IEEE Trans. on Com. 42(10), 2908 (1994)
14. Pollet, T., van Bladel, M., Moeneclaey, M.: BER Sensitivity of OFDM system to Carrier Frequency Offset and Wiener Phase Noise. IEEE Trans. on Communications 43(2/3/4) (February/March/April 1995)

Using Smartphones to Detect Car Accidents and Provide Situational Awareness to Emergency Responders

Chris Thompson, Jules White, Brian Dougherty,
Adam Albright, and Douglas C. Schmidt

Institute for Software Integrated Systems,
Vanderbilt University, Nashville, TN USA
{cm.thompson,jules.white,brian.p.dougherty,
adam.albright,d.schmidt}@vanderbilt.edu

Abstract. Accident detection systems help reduce fatalities stemming from car accidents by decreasing the response time of emergency responders. Smartphones and their onboard sensors (such as GPS receivers and accelerometers) are promising platforms for constructing such systems. This paper provides three contributions to the study of using smartphone-based accident detection systems. First, we describe solutions to key issues associated with detecting traffic accidents, such as preventing false positives by utilizing mobile context information and polling onboard sensors to detect large accelerations. Second, we present the architecture of our prototype smartphone-based accident detection system and empirically analyze its ability to resist false positives as well as its capabilities for accident reconstruction. Third, we discuss how smartphone-based accident detection can reduce overall traffic congestion and increase the preparedness of emergency responders.

1 Introduction

Emerging trends and challenges. Car accidents are a leading cause of death [2]. Automated car accident detection can save lives by decreasing the time required for information to reach emergency responders [6,5,7]. Conventional vehicular sensor systems for accident detection, such as OnStar, notify emergency responders immediately by utilizing built-in cellular radios and detect car accidents with in-vehicle sensors, such as accelerometers and airbag deployment monitors. Figure 1 shows how traditional accident detection systems operate.

Car accident detection and highway congestion control is an emerging application for wireless mobile sensor networks. Recent advances in smartphone technologies are making it possible to detect car accidents in a more portable and cost effective manner than conventional in-vehicle solutions. Rapid accident detection and response can save lives and reduce congestion by alerting motorists as soon as possible, giving them time to reroute. Recent smartphones, such as the HTC Nexus One (an Android-based device), have significantly increased computational abilities compared to previous devices. For example, the Nexus One has a 1Ghz processor and 512MB of RAM compared to the older Palm Treo's 312Mhz processor and 64MB of RAM. The pervasiveness of smartphones also means that the infrastructure required to establish such a wireless mobile

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 29–42, 2010.

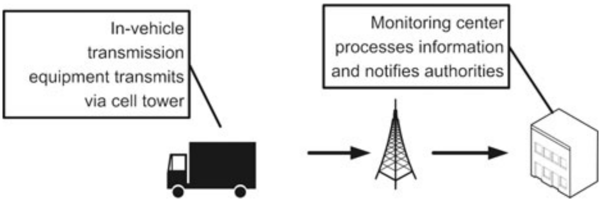

In-vehicle transmission equipment transmits via cell tower
Monitoring center processes information and notifies authorities

Fig. 1. A Traditional Accident Detection System

sensor network is already in place and available after installing appropriate application software.

Smartphone manufacturers also have begun including a plethora of sensors that enable devices to detect the context in which they are being used. For example, the HTC Dream (also an Android-based device), possesses a compass, accelerometer, and GPS receiver allowing application developers to determine the geographic position, heading, and movement of the user. The processing power, popularity, and relatively low cost [12] (compared to other traffic monitoring techniques) make smartphones an appealing platform to construct a wireless mobile sensor network that detects car accidents.

Smartphone-based accident detection applications provide several advantages relative to conventional in-vehicle accident detection systems, *e.g.*, they are vehicle-independent, increasingly pervasive, and provide rich data for accident analysis, including pictures and videos. Building a smartphone-based wireless mobile sensor network for accident detection system is hard, however, because phones can be dropped (and generate false positives) and the phone is not directly connected to the vehicle. In contrast, conventional in-vehicle accident detection systems rarely incur false positives because they rely on sensors, such as accelerometers and airbag sensors, that directly detect damage to the vehicle.

Solution approach → Use onboard sensors and physical context information to detect car accidents. This paper shows how smartphones in a wireless mobile sensor network can capture the streams of data provided by their accelerometers, compasses, and GPS sensors to provide a portable "black box" that detects traffic accidents and records data related to accident events, such as the G-forces (accelerations) experienced by the driver. We also present an architecture for detecting car accidents based on WreckWatch, which is a mobile client/server application we developed to automatically detect car accidents. Figure 2 shows how sensors built into a smartphone detect a major acceleration event indicative of an accident and utilize the built-in 3G data connection to transmit that information to a central server. That server then processes the information and notifies the authorities as well as any emergency contacts.

WreckWatch provides functionality similar to an accident/event data recorder by recording the path, speed, and forces of acceleration on a vehicle leading up to and during an accident [4]. It can also notify emergency responders of accidents, aggregate images and video uploaded by bystanders at the scene of an accident, and send prerecorded text and/or audio messages to emergency contacts. We built WreckWatch using Google Android on the client and Java/MySQL with Jetty and the Spring Framework on the server. The WreckWatch server utilizes custom XML and JSON to communicate with the client applications and the clients use standard HTTP `post` operations to

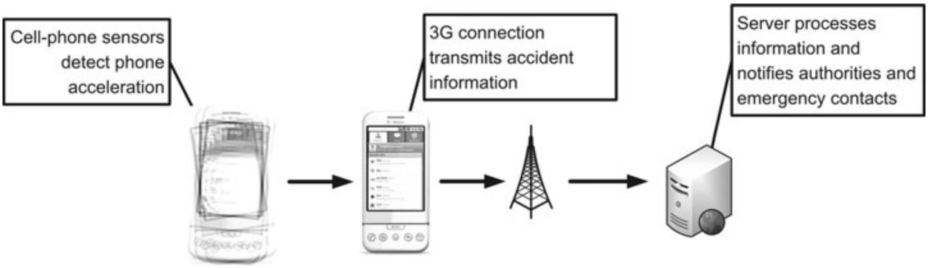

Cell-phone sensors detect phone acceleration
3G connection transmits accident information
Server processes information and notifies authorities and emergency contacts

Fig. 2. Smartphone-Based Accident Detection System

transmit information to the server. WreckWatch also uses a digital PBX running Asterisk to communicate with first responders and emergency contacts.

Paper organization. The remainder of this paper is organized as follows: Section 2 describes the challenges associated with using smartphones to detect traffic accidents; Section 3 describes how WreckWatch overcomes these challenges; Section 4 empirically evaluates WreckWatch's ability to prevent false positives and accident reconstruction capabilities; Section 4 compares our work on smartphone-based accident detection systems with related work; and Section 5 presents concluding remarks.

2 Challenges Associated with Automatically Detecting Car Accidents

This section describes the challenges associated with detecting car accidents via software running on smartphones. A key challenge of developing software to detect collisions is the lack of integration between the smartphone and the vehicle. In contrast, conventional in-vehicle car accident detection systems rely on internal sensors (*e.g.*, airbag deployment sensors) and can assume that any instance of high acceleration/-deceleration is caused by a collision. These assumptions must be rethought by smartphone applications seeking to replace or augment the functionality of conventional in-vehicle systems.

2.1 Challenge 1: Detecting Accident without Electronic Control Unit Interaction

Conventional in-vehicle accident detection systems rely on sensor networks throughout the car and direct interaction with the vehicle's electronic control units (ECUs). These sensors detect acceleration/deceleration, airbag deployment, and vehicular rollover [3,14]. Metrics from these sensors aid in generating a detailed accident profile, such as locating where the vehicle was struck, number of times it was hit, severity of the collision, and airbag deployment.

Smartphone-based accident detection applications must provide similar information. Without direct access to ECUs, however, it is harder to collect information about the vehicle. Although many cars have accident/event data recorders (ADRs/EDRs), it is unrealistic to expect drivers to connect their smartphones to these ADRs/EDRs every time they got in the car, which would require a standardized interface (physical and software) to ensure compatibility. Moreover, while many new cars have some form of ADR/EDR, any smartphone application that required interaction with an onboard computer would be useless in cars that lacked one. It is therefore necessary to collect the same or similar information utilizing only the sensors present on the smartphone device.

Section 3.2 explains how WreckWatch addresses this challenge by using the sensors in the Android platform to detect accelerations/decelerations experienced by car occupants and Section 4 analyzes device sensor data captured by WreckWatch.

2.2 Challenge 2: Preventing False Positives

Vehicle-based accident detection systems monitor a network of sensors to determine if an accident has occurred. Instances of high acceleration/deceleration are due to a large change in velocity over a very short period of time. These speeds are hard to attain if a vehicle is not controlled by a human driver, which simplifies accident detection since we can assume any instance of high acceleration constitutes a collision involving human drivers. Since smartphones are portable, however, it is not as hard to attain such speeds. For instance, it is not hard to drop a phone from six feet in the air, but dropping a vehicle from that height would require significantly more effort.

Since a smartphone-based accident detection application contacts emergency responders—and may dispatch police/rescue teams—it is essential to identify and suppress false positives. Due to smartphone mobility it is hard to programmatically differentiate between an actual car accident versus a dropped purse or a fall on a hard surface. The inability to accurately identify and ignore false positives could render smartphone-based accident detection applications useless by wasting emergency responder resources responding to incident reports that were not car accidents.

Section 3.2 explains how WreckWatch addresses this challenge by using device usage context, such as speed, to filter out potential false positives and Section 4.2 provides empirical results evaluating WreckWatch's ability to suppress false positives.

3 Solution Approach

This section describes the client/server architecture of WreckWatch and outlines the solutions to the challenges presenting in Section 2.

3.1 The WreckWatch Client/Server Architecture

WreckWatch is separated into two main components—the WreckWatch server and the WreckWatch client—that are shown in Figure 3 and described below.

The WreckWatch client acts as a mobile sensor, relays accident information to the server, and provides an interface for third-party observers to contribute information to the accident report. For example, Figure 4 shows how images of an accident can be uploaded to the WreckWatch server. Emergency responders can access the uploaded images via mobile devices en route or a standard web browser at an emergency response center. The WreckWatch client provides mapping functionality through Google Maps on the device to ensure that emergency responders can continuously receive information

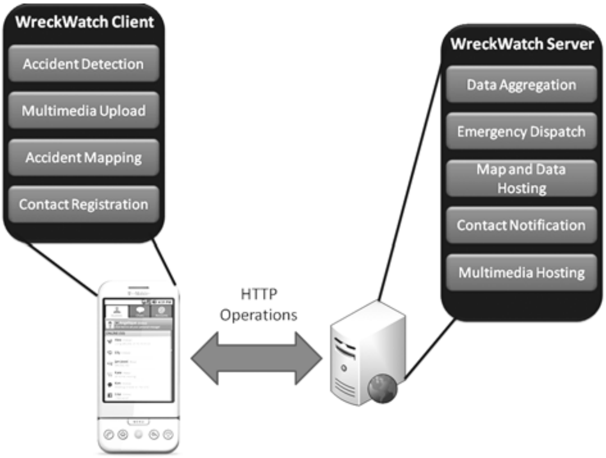


Fig. 3. WreckWatch Architecture Diagram

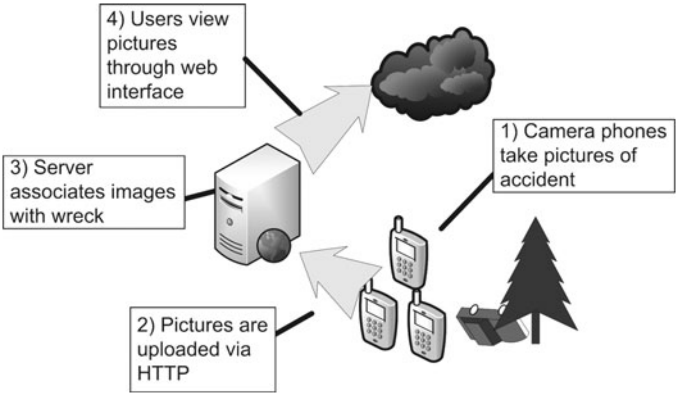

4) Users view pictures through web interface
1) Camera phones take pictures of accident
3) Server associates images with wreck
2) Pictures are uploaded via HTTP

Fig. 4. Accident Image Upload

about an accident to prepare them for whatever they encounter at the accident site. This map also allows other motorists to intelligently route themselves around an accident, thereby reducing congestion.

The WreckWatch Android client is written in Java based on Android 1.5 with Google APIs. It consists of several Android application *activities*[1] for mapping, testing, and image upload. Background services detect accidents by polling smartphone system sensors, such as the GPS receiver and accelerometers. The polling rate is configurable at compile-time to meet user needs and to provide the appropriate power consumption characteristics. The WreckWatch client can gather data from phone databases (such as

[1] Activities are basic building block components for Android applications and can be thought of as a "screen" or "view" that provide a single, focused thing a user can do.

an address book) to designate emergency contacts. Communication to the server from the Android client uses standard HTTP `post` operations.

The WreckWatch server provides data aggregation and a communication conduit to emergency responders, family, and friends. It allows clients to submit accident characteristics (such as acceleration, route, and speed) and presents several interfaces, such as a Google Map and XML/JSON web services, for accessing this information. As accident information becomes available, the WreckWatch server posts location, route and severity information to a Google Map to aid emergency responders, as well as other drivers attempting to navigate the roads near the accident. This map is available over HTTP through a standard web browser and is built with AJAX and HTML, as shown in Figure 5.

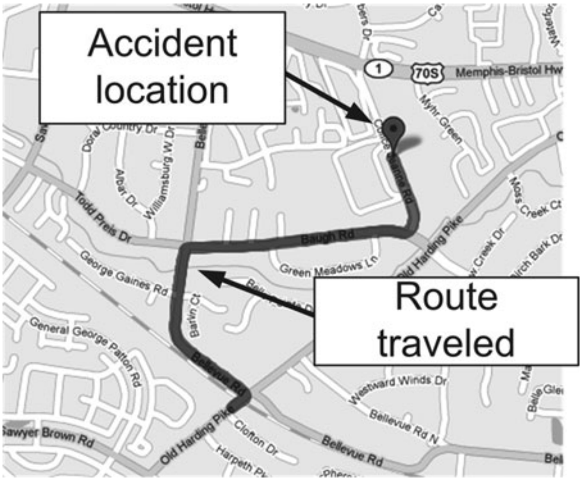


Fig. 5. WreckWatch Accident Map

The WreckWatch server uses digital PBX functionality to make/receive phone calls and provision phone lines dynamically. It can therefore interact with emergency responders via traditional circuit-switched networks and create accident information hotlines in response to serious accidents via an Asterisk-based digital PBX running Linux. The server can also be configured with emergency contacts to notify via text and/or audio messages in the event of an accident. This data is configured at some time prior to a collision event so the server need not interact with the client to notify family or friends.

The WreckWatch server is a web-based service based entirely on freely-available APIs and open-source software. It is written in Java and built using Jetty atop the Spring Framework. It utilizes a MySQL database to store accident information and image meta-information. The server communicates with the clients via a RESTful architecture over HTTP using custom XML (for the Android application) and JSON (for the web-based application).

All communication between the clients and the server is initiated by clients. The server's operations (such as accident information upload) are performed by individual handlers that can be configured at runtime and are specified by parameters in an HTTP request. This architecture enables the addition of new operations and functionality without any software modifications or the need to recompile. All configuration is handled by an XML file that is parsed during server startup.

The PBX is built on Asterisk and connects to the server through a Java API. The Android client and web client pull information from the server and can be configured based on user needs. Due to the loose coupling and use of open standards between clients and server, additional clients for other platforms (such as other smartphones or desktop applications) can be implemented without the need to update the server. The WreckWatch server architecture also supports a heterogeneous group of clients, while providing appropriate qualities of service to each device.

3.2 WreckWatch Solution Implementations

The remainder of this section outlines how WreckWatch addresses the challenges presented in Section 2.

Utilization of Onboard Accelerometers to Detect Collisions. The challenge presented in Section 2.1 explains why it is hard to detect car accidents without ECU interaction. To address that challenge, WreckWatch uses Android's onboard sensors to detect the forces and accelerations associated with a car accident, as shown in Figure 6. The Android platform provides an orientation sensor comprised of three independent accelerometers that allow WreckWatch to detect car accidents in the same manner as vehicle ECUs.

In the event of an accident, the smartphone will experience the same forces and accelerations experienced by the occupants of the vehicle. Moreover, if the smartphone remains stationary relative to the vehicle during the collision, it is possible to use the data gathered from the smartphone to recreate and model the forces it experienced. In this case, the smartphone can provide data much like that gathered by vehicular ECUs.

Smartphones are often carried in some form of pocket [10] attached to a person. In these cases, the smartphone would experience the same forces as vehicle occupants, and

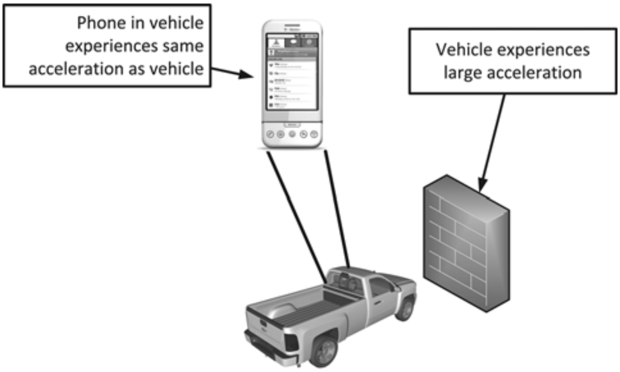

Phone in vehicle experiences same acceleration as vehicle
Vehicle experiences large acceleration

Fig. 6. Device Sensors Provide Acceleration Information

could thus provide more information than in-vehicle systems by recording the forces experienced by occupants rather than just the vehicle itself. When this directionality and movement is combined with speed and location information from the GPS receiver, it is possible to fully reconstruct the accident, including any secondary impacts.

Using Context Information to Eliminate False-Positives. Section 2.2 describes the potential for false positives, which is a key concern with applications that automatically dispatch police or rescue. To address that challenge, WreckWatch employs the following sensor-based and context filters:

- **In order to prevent excessive power consumption and WreckWatch is only enabled when plugged in** GPS receivers consume a substantial amount of power and sampling them at the rate necessary to accurately determine speed would make WreckWatch unusable because it would limit the lifetime of the device to several hours. However, users are able to plug smartphones into cigarette lights in vehicles to provide them with power. Requiring users to plug the smartphone in helps to establish context, which will eliminate false positives, and also mitigates the power consumption of the GPS receiver. However, it is also possible to plug a smartphone in to a wall socket in a home which necessitates additional filters.
- **Speed filter determines whether users are in vehicles.** WreckWatch uses the smartphone's GPS to determine device and (consequently) vehicle speed. However, it only begins recording accelerometer information and looking for potential accidents above 15mph. This filter helps eliminate any acceleration events due to significant accidental smartphone drops that might occur outside a vehicle as well as reducing battery drain. After WreckWatch determines that users are in vehicles, it maintains that as their context until the device is unplugged, which prevents WreckWatch system from shutting off at stop lights. This speed threshold can be adjusted at compile time to prevent overloading operators and falsely alerting family of an accident.
- **Acceleration filter prevents drops and sudden stops from triggering accident notifications.** Filtering alone does not eliminate all false positives, such as a drop inside the vehicle or a sudden stop. To address these issues, therefore, WreckWatch ignores any acceleration events below 4G's. This value is designed to detect even minor accidents but filter out a drop or sudden stop and was chosen based on the empirical analysis presented in Section 4. This threshold is significantly lower than the acceleration required to deploy airbags because of physical environment of the smartphone.

  Accelerometers attached to the vehicle are what trigger airbag deployment. These accelerometers are physically mounted to the chassis of the car meaning that their motion will directly mirror that of the vehicle and will experience every force the vehicle experiences. Smartphones, however, are likely to be held in a pocket or in a cup holder. Car safety systems are designed to reduce the force on the occupants of the car during an accident and because of this, the forces experienced by the phone will be significantly less than the forces experienced by the accelerometers in the car. These systems accomplish this reduction in force by increasing the time over which the change in velocity occurs. The net change in speed is the same, but the acceleration is less because it occurs over a longer period of time. Therefore in order to detect car accidents, the detection threshold must be significantly lower than that required to deploy the airbag. In contrast, the peak accelerations experienced inside of a football helmet during play are approximately 29.2 G's [13]. This value represents the maximum value experienced by a player and would be significantly larger than many minor collisions.

4 Empirical Results

This section describes empirical results of tests performed on the WreckWatch application described in Section 3. These results demonstrate WreckWatch's ability to prevent false positives and gather information to reconstruct an accident accurately.

4.1 Overview of the Experimentation Platform

All experiments were performed on a Google ION device running the vendor image of Android 1.5 on a 525 Mhz processor with 288 MB of RAM. The device was factory reset before loading WreckWatch and no additional third-party applications were installed. WreckWatch recorded acceleration on three axes at the highest possible rate and wrote these values to a CSV file on the SD card in the device. This data was then downloaded to a Windows desktop computer for analysis in Excel.

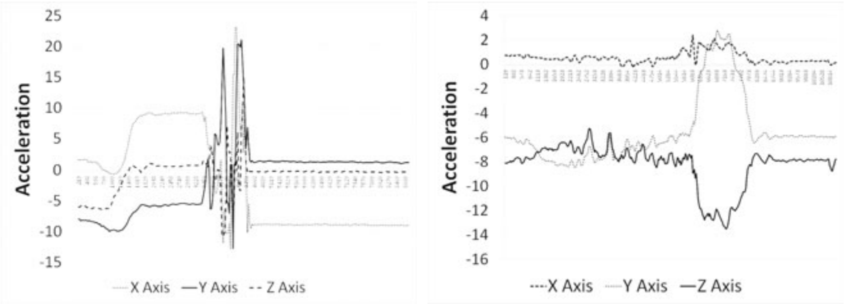
In all graphs, positive z-axis values indicate positive acceleration in the direction from the battery cover toward the screen. Likewise, positive y-axis values indicate positive acceleration in the direction from the USB connector toward the smartphone speaker. Finally, positive x-axis values indicate positive acceleration from left to right when looking at the device with the USB connector closest the observer.

4.2 Evaluating Possibility of False Positives

As described in Section 2.2, avoiding false positives is a key challenge when detecting car accidents with smartphones. To analyze the potential for false positives, we conducted two experiments designed to simulate events that generate accelerations whose values could potentially be interpreted as car accidents. For the first test, the Android device was dropped from ear height in the driver's seat of a car. The device bounced off the seat and wedged between the seat and center console. Figure 7a shows the acceleration on each axis during the collision with the floor.

Using 9.8 m/s as an approximate value for Earth's gravity, the device experienced approximately 2G's in each direction with nearly 3G's on the x-axis before coming to rest. The required acceleration to trigger airbag deployment is 60G's [8,1]. In addition to being ∼30 times smaller than required to deploy an airbag, this value is well below the 4G's used as a filter. It is therefore unlikely a smartphone could be dropped in a manner that would exceed 4G's. This data supports the use of a filter as presented in Section 3.2 to prevent false positives.

Another potential scenario that could potentially generate a false positive is a sudden stop. This test was performed in a vehicle by reaching a speed of approximately 25



(a) Acceleration During a Fall

(b) Acceleration During a Sudden Stop

Fig. 7. Acceleration During Falls and Sudden Stops

mph and engaging in a sudden stop. The test results are approximate as the exact speed was unknown and braking pressure was not exact. Figure 7b shows the acceleration experienced on each axis during the stop. As described in Section 3.2, because the smartphone remained stationary relative to the vehicle, it experienced the same forces as the vehicle. In this instance, the acceleration experienced by the smartphone was actually less than that experienced during the fall.

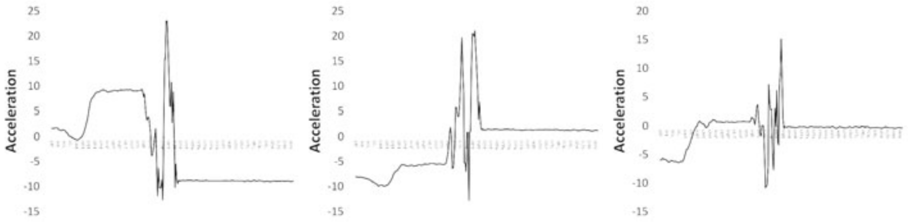
This result is attributed to the fact that although the stop was sudden and forceful, the car (and consequently the smartphone) came to a rest over a period of time that was longer than during the drop test. In other words, the change in velocity was greater but the actual acceleration was less because the change occurred over a longer period of time. Based on this data, it is unlikely for the smartphone to experience 4G's of acceleration simply due to a sudden stop.

4.3 Evaluating Accident Reconstruction Capabilities

WreckWatch can reconstruct an accident based solely on the data gathered from the smartphone. Due to the smartphone's presence in the vehicle during an accident, the smartphone will usually experience the same forces at the same time as the occupants and the vehicle itself. For example, ∼40% of cell phones are carried in some form of pocket [10], in which case the device will experience the same forces experienced by the person wearing the pocket.

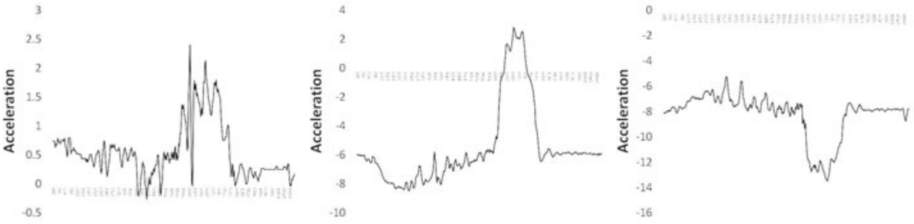
If the smartphone experiences the same forces as the occupants of the vehicle, we can identify what happened during the accident and reconstruct it. To demonstrate this approach, we next analyze the two experiments conducted in Section 4.2.

The graph in Figure 8a shows it is possible to determine that the smartphone was initially experiencing zero acceleration along the x-axis indicating that the x-axis was perpendicular to the ground. This orientation is consistent with holding the smartphone to the ear. While falling, the smartphone tilted such the left edge of the smartphone (relative to the screen with the screen facing away from the ground) was the closest edge to the sky and then flipped again such that the left edge was closest to the ground. When Figures 8a, 8b, and 8c are combined it is clear that the bottom of the smartphone made contact first, followed by the left edge, and finally the back of the device.



(a) X-Axis Acceleration (b) Y-Axis Acceleration (c) Z-Axis Acceleration

Fig. 8. Acceleration During a Sudden Stop



(a) X-Axis Acceleration (b) Y-Axis Acceleration (c) Z-Axis Acceleration

Fig. 9. Acceleration While Dropped in a Car

The acceleration experienced during the sudden stop was actually less than that experienced during the fall. Given what is known about the event, it is therefore possible to identify the orientation of the smartphone during the event. By examining the graphs in Figure 8 it is possible to determine that the smartphone was resting at an angle such that the top of the smartphone was higher than the bottom of the smartphone. The decrease in acceleration along the z-axis is indicative of the force induced on the device by the seat as the car came to a rest. Graphs of other sudden stop events also have a similar appearance so long as the device remained stationary relative to the car.

These reconstruction capabilities give accident investigators the ability to identify what was experienced by the occupants of the vehicle and provide them with information that an ADR/EDR simply cannot provide. This information can also be combined with that present in the ADR/EDR to better understand the entire accident rather than simply the forces experienced by the vehicle itself. WreckWatch gives investigators the capability to analyze a real-world accident in a manner similar to the way they would a controlled collision involving crash-test dummies. Although WreckWatch cannot provide investigators with all impact information (*e.g.*, the forces experienced at the ribs [9] or the pressure on the face [11]), it can provide them with specific information about the overall force on the body and how effectively the restraints protected the passenger.

5 Concluding Remarks

Although conventional in-vehicle accident detection systems provide emergency responders with crucial information at the earliest possible time, adoption of these systems is limited by their non-portability and cost. Smartphones present a promising platform on which to construct an accident detection system. Significant challenges, however, are associated with developing an accident detection application for smartphone-based sensor networks. This paper described how the WreckWatch smartphone application accurately detected traffic accidents by combining (1) contextual information to determine when a user is in a vehicle with (2) high G-force filters that helped to suppress false positives, such as a dropped phone or sudden stop that may occur while a vehicle is in motion.

In developing and evaluating WreckWatch, we learned the following lessons:

• **In the event of an extreme accident the phone may be destroyed preventing it from contacting emergency responders.** As with equipment embedded in the vehicle, which is how systems like OnStar function, there is a chance that the phone would become damaged during an accident and be unable to transmit accident information. Without providing redundant or ruggedized equipment, which would significantly increase cost and reduce usability, there is little that can be done to prevent the destruction of communication equipment. This is a weakness of such a system however the severity of such an accident would likely draw enough attention from witnesses that WreckWatch's notification would be superfluous.

• **Accidents exert extreme forces on a phone that are unlikely to occur when dropping it.** The forces experienced during a car collision are extreme and highly unlikely to occur in any other event other than a high-speed collision. These events are therefore easier to identify and categorize accordingly. Moreover, by combining the accident detection process with contextual information to determine when the user is in a vehicle, false positives are less likely.

• **Smartphones can surpass the functionality of conventional in-vehicle accident detection systems.** Modern smartphone platforms possessing a GPS receiver and accelerometers can be utilized to detect car accidents and represent a portable alternative to conventional in-vehicle systems, such OnStar. Moreover, smartphone-based applications can surpass the functionality of conventional systems by leveraging the other device features and network functionality, such as contact management and Internet access, which allows accident victims to alert emergency personnel, family, and friends immediately and automatically.

• **Collision events can be modeled based on data collected from a smartphone.** If the smartphone remains stationary relative to the vehicle during the collision, the smartphone will experience the same forces as the vehicle, which allows reconstruction of the accident based on the data gathered from the smartphone. This data allows accident investigators to determine not only what happened during an accident, but also provides them with insight into the forces experienced by the occupants. In this case, a smartphone-based accident detection system provides more information than a system like OnStar that only collects information about the vehicle itself. This data could then be used to analyze the effectiveness of the safety features of the vehicle, such as seat belts.

• **It may not be possible to detect all accidents with smartphones.** Due to the filters utilized to prevent false positives, it may be possible to experience a low speed "fender-bender" without the application detecting it. More work is needed to enhance the filtering mechanisms to account for these types of collisions. In particular, WreckWatch's filtering algorithm could be enhanced to determine whether the user is in a vehicle or not utilizing history information. For example, users often travel similar routes to work and WreckWatch could learn where stops or reductions in speed are common by analysis of trends (*e.g.* if a person usually travels through an area at 40mph but occasionally slows to a stop indicating a potential traffic jam). Likewise, WreckWatch could use known intersections to identify potential stops and anticipate them or download traffic information to predict the location of traffic jams resulting from long-duration reductions in speed.

• **In-vehicle Bluetooth radios connecting the phone and vehicle increase the potential for smartphone-based accident detection systems.** Although WreckWatch does not rely on any interaction with the vehicle, direct interaction with the ADR/EDR would increase the accuracy and information available to smartphone-based accident detection systems, such as whether brakes were applied and at what pressure, whether the occupants were wearing seat belts, whether cruise control was on, whether head lamps were on, etc [4]. Many vehicles already possess a hardware connection to the ECU for problem diagnosis. This connection could be used to attach a Bluetooth transmitter that would establish a wireless connection to WreckWatch when the vehicle was started. Minor modifications to WreckWatch would be needed to record and process the additional sensor information from the vehicle.

• **Integrating accident detection systems with Intelligent Transportation Systems (ITS) can help city planners and motorists combine accident data with other roadway information.** City planners and transportation departments currently use ITS to identify road problems and hazardous conditions. Many cities offer services (such as a 511 telephone number) to allow motorists to access information regarding congestion and accidents on major roadways. Integrating WreckWatch with ITS implementations would reduce the latency between an accident event and the availability of the information. This integration could also help city planners create a database of accident locations that could be cross-referenced with hazardous road conditions. Since WreckWatch uses open standards an application that already performs many ITS services could be configured to download accident information using XML over HTTP. This information could then be incorporated into reports generated by the ITS and processed accordingly.

The WreckWatch application is open-source and can be downloaded from `vuphone.googlecode.com`. Also available from this repository are smartphone applications for social networking, campus dining, and social events.

References

1. National Highway Traffic Safety Administration. Federal Motor Vehicle Safety Standards: Occupant Crash Protection - Supplemental Notice of Proposed Rulemaking (1999)
2. National Highway Transportation Safety Administration. 2007 Traffic Safety Annual Assessment - Highlights (2008)
3. Alsliety, M.: How does SDR fit the telematics model?
4. Askland, A.: Double Edged Sword That Is the Event Data Recorder. The Temp. J. Sci. Tech. & Envtl. L. 25(1) (2006)

5. Champion, H.R., Augenstein, J., Blatt, A.J., Cushing, B., Digges, K., Siegel, J.H., Flanigan, M.C.: Automatic crash notification and the URGENCY algorithm: Its history, value, and use. Advanced Emergency Nursing Journal 26(2), 143 (2004)
6. Drabek, T.E.: Managing the emergency response. Public Administration Review 45, 85–92 (1985)
7. Evanco, W.: The Impact of Rapid Incident Detection on Freeway Accident Fatalities. Mitretek Systems, Inc. WN96W0000071 (1996)
8. Fildes, B., Newstead, S., Barnes, J.S., Morris, A.P.: Airbag effectiveness in real world crashes (2001)
9. Groesch, L., Netzer, G., Kassing, L.: Dummy for car crash testing, US Patent 4701132 (October 20, 1987)
10. Ichikawa, F., Chipchase, J., Grignani, R.: Where's the phone? a study of mobile phone location in public spaces. In: Proc. IEE Mobility Conference 2005, Citeseer (2005)
11. Mellander, H., Nilsson, S., Warner, C.Y., Wille, M.G., Koch, M.: Load-sensing faceform for crash dummy instrumentation, US Patent 4691556 (September 8, 1987)
12. Mohan, P., Padmanabhan, V.N., Ramjee, R.: Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In: Proceedings of the 6th ACM conference on Embedded network sensor systems, pp. 323–336. ACM, New York (2008)
13. Naunheim, R.S., Standeven, J., Richter, C., Lewis, L.M.: Comparison of impact data in hockey, football, and soccer. The Journal of Trauma 48(5), 938 (2000)
14. Verma, M., Lange, R., McGarry, D.: A Study Of US Crash Statistics From Automated Crash Notification Data (2007)

Session 2: Mobility Management and Handoff Management (Chair: Thomas Magedanz)

XtreemOS-MD: Grid Computing from Mobile Devices

Alvaro Martínez[1], Santiago Prieto[1], Noé Gallego[1], Ramon Nou[2],
Jacobo Giralt[2], and Toni Cortes[2]

[1] Telefónica I+D, Spain
{amr,spm,e.xtreemos-ngm}@tid.es
[2] Barcelona Supercomputing Center, Spain
{ramon.nou,jacobo.giralt,toni.cortes}@bsc.es

Abstract. Grid and Cloud computing are well known topics, and currently one of the focus of many research and commercial projects. Nevertheless, the transparent access to Grid facilities from mobile devices (like PDAs or smartphones) is normally out of the scope of those initiatives, taking into account the intrinsic limitations of such devices: screen size and resolution, storage capacity, computational power, etc.

To provide an integrated solution for mobile access to Grid computing, aspects like Virtual Organizations (VOs) support, graphical job management, flexible authentication and access to the file system user's volume in the Grid from a mobile device should be covered. That is the purpose of XtreemOS-MD, the mobile flavour of XtreemOS - a Linux-based operating system to support VOs for Grids -, the transparent access to Grid facilities from mobile devices.

Keywords: grid, cloud computing, mobile device.

1 Introduction

The Grid [10], as a distributed computational system that allows sharing non-centralized resources to solve a single problem requiring a very significant computational power or storage capacity, is a relatively old and well known concept. Grid systems have been used specially to solve complex scientific and technical problems [17]. Middleware solutions like the Globus toolkit [9], the most extended solution, or others like Condor [26], Legion [15] or UNICORE [7], have classically been used to implement the Grid architectures. But those solutions are designed as a "sum of services" infrastructure, where tools are developed independently [12][13][14], and the lack of common programming interfaces or unified models of component interaction can impact negatively in the user-friendliness. The final consequence is that a certain level of experience and technology knowledge is required to effectively use the Grid. Precisely, XtreemOS [29] was designed to become a new Grid operating system allowing transparent and native access to the Grid environment, and then simplifying in big manner the use of Grid systems.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 45–58, 2010.

A more recent concept is the cloud computing, referring to a dynamically scalable style of computing, where virtualized resources are provided as a massively scalable service over the Internet [25]. Those systems can be implemented using a centralized infrastructure of servers, but also Grid platforms could be used to provide the virtualized resources forming the cloud. Commercial solutions like Amazon EC2 [1] or Google AppEngine [16] are already exploiting this idea, and other big players are currently launching their own "cloud" solutions, like Microsoft with the Windows Azure [19] platform.

But these distributed systems, Grids and Clouds, are not only useful for executing complex tasks, but also they are useful for simpler tasks, but still too complex for certain limited computational-power devices, like PDAs or smartphones. Despite the fact of their intrinsic computing limitations, the integration of an increasing number of mobile consumer devices into the Grid has been considered as potentially beneficial to the computational Grid [24]. Moreover, we believe that it opens the way for new application scenarios. For instance, a mobile device multimedia player application may use the Grid to transparently transcode any video file so that it is adapted to its media capabilities, or the Grid might be used to quickly calculate the next move of your opponent in a mobile device chess game application or, simply, the user's device can take benefit of an "unlimited" storage capability in the Grid when recording or playing multimedia content. But most importantly, all those scenarios can be implemented in a native and transparent way, so that the user does not even know that there is a Grid infrastructure behind.

1.1 State of the Art

The inclusion of mobile devices into Grid computing platforms has followed two different research lines: the first one adds mobile devices as simple Grid clients and the second tries to include them as Grid resources, extending the capabilities of the Grid thanks to the near-unlimited number of resources shared. The first approach has been followed by some Grid frameworks [20], first by just providing simple and very limited web user interfaces to mobile devices with Web proxies [8] and, one step further, by developing specific mobile device applications downloadable from Grid portals [22], either the applications being written in Java or using other development frameworks [4]. The second approach has being tackled by several initiatives, usually trying to create Grids by the simple combination of mobile devices as computing resources, essentially imitating the common Grid services and scenarios. Several architectures have been proposed, like a Proxy-Based clustered architecture [24], or more ambitious ones under the general term of "Mobile Grid Computing" [28] or even others proposing a "beyond the device as portal" model [3]. However, as far as we know, actual implementations of these approaches for mobile phones are almost inexistent. Moreover, even if this latter approach could gain acceptance with the increase of performance and storage capacity experimented in the smartphones world, issues like low battery power, lack of security and confidentiality, etc. make us think in a different kind of resource sharing, focused on network and input/output devices (GPS, camera, microphone,etc.) and data sharing, rather than computing power.

By contrast, our approach, which will be detailed below, has allowed us to implement a Grid layer providing: native access to distributed Grid file systems, seamless job execution and management in the Grid from the mobile, and a complete and secure integration with Virtual Organizations (VOs). In addition, sharing of specific resources of a mobile device is also possible. Moreover, the solution has been implemented and tested with real mobile Linux-based devices.

1.2 Transparent Mobile Grid Computing

Precisely the main focus of this paper is about the possibility of transparently and natively using a Grid infrastructure from a mobile device, to benefit from Grid facilities but hiding its complexity to the users. We believe that these goal can only be totally achieved by integrating a Grid middleware into the mobile device operating system and by avoiding portal or proxy-based only models that basically limit the potential of Grid computing for mobile devices. In addition, the sharing of mobile device resources to the Grid may be easily achieved thanks to the possibility of having native access to device resources. To this purpose, we will present XtreemOS-MD, the version for mobile devices of the XtreemOS project, a Linux-based operating system designed to work with Grids in a transparent way. XtreemOS-MD tries to cover the two research lines mentioned: the universal access to the Grid services and also the sharing of resources provided by the mobile devices, like their network capabilities, GPS and the content stored locally in the mobile device (photos, videos, etc.)

2 XtreemOS-MD General Architecture

As commented in the introduction, the approach selected by the XtreemOS project, and followed as well by its mobile version -XtreemOS-MD, is based on providing a Grid native support inside the operating system.

XtreemOS was born to provide the needed abstraction from the hardware and software and to secure the resources shared between different users, let's say what a traditional operating system provides, but focused on Grids in this case. The design of XtreemOS [6] has taken into account some particular characteristics of Grid systems and has faced challenges like scalability, transparency, interoperability, security, etc. The overall architecture of XtreemOS, from the Linux kernel to the application level, has been divided in two different levels: the Foundation layer (F-layer) providing VO support and the Grid services layer (G-layer), supported by the previous one and providing a set of services that can be used by the application layers.

The mobile device version of XtreemOS follows the same general architecture and it is adapted to the mobile device intrinsic limitations. The main targets for XtreemOS-MD have been mobile devices like PDAs and smartphones. The particularities of these mobile devices revealed the need of a specific version of the software, where aspects like mobility, limited computing and storage capabilities, and battery lifetime have been considered. The XtreemOS-MD architecture addresses those important limitations and adapts the XtreemOS general requirements and design principles to the mobile devices selected. For this reason, specific modules for providing context information and also for managing the resources shared by the mobile devices have been added to the architecture. This modules are deeply explained in section 4.

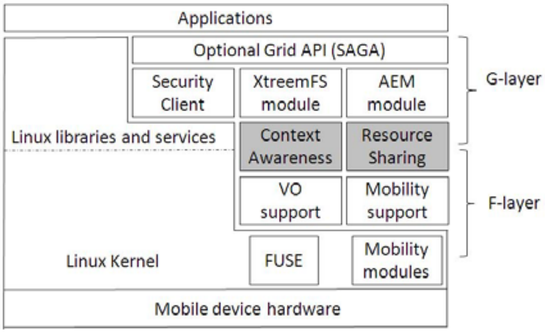


Fig. 1. XtreemOS-MD general architecture

Another important aspect when thinking on mobile devices is the fact that the users are in general not computer-experts. This means that the ease of installation and use should be seriously taken into account, and the provision of graphical interfaces for installation, configuration and use of the mobile device software should be a priority, as it is for XtreemOS-MD. Other factors like the compatibility with native applications and the security model of each device have been considered when deciding the roadmap for mobile devices supported. In a first phase, XtreemOS-MD software is developed for Nokia N8x0 family of Internet tablets (based on the Maemo Linux OS [18]) and for the NeoFreeRunner smartphone (based on the SHR (Stable Hybrid Release) Linux OS from Openmoko [23]). We should highlight that XtreemOS-MD is mounted over a Linux operating system, thus potentially any mobile platforms based on Linux could support XtreemOS-MD software with minor adaptations.

The general architecture of XtreemOS-MD is depicted in Figure 1 where it is shown the different modules belonging to each layer that will be discussed in the following sections. The blocks in grey are precisely the most significant difference between XtreemOS and XtreemOS-MD architectures and will be further analyzed along the article.

3 The Foundation Layer

In the Grid computing world, an important concept is the Virtual Organization (VO), which refers to a set of users and real organizations that collectively provide resources they want to exploit for a common goal [5]. XtreemOS and XtreemOS-MD provide a full support for VO management [21], allowing not only to join an existing VO, but even to administrate them from a graphical interface specially adapted for limited resolution screens, like the ones provided typically by mobile devices.

The F-layer comprises the components that modify the Linux OS itself, to make it aware of virtual organizations in the Grid and to provide mobility for the Grid access. The VO support module provides several VO-related facilities to Grid users like management of user's credentials, dynamic management of UID/GID, authentication and policy-based authorization and session management.

Regarding mobility, it introduces a series of difficulties for keeping sessions, loosing connectivity, etc. Mobility in XtreemOS-MD is provided by a mobility module at kernel level plus a mobility support module included in the F-layer. The solution adopted is based on an adaptation for ARM (Advanced RISC machine) architectures of the USAGI MIPv6 implementation [27], that allows users to stay connected to the Grid while they are moving, maintaining the same IP address and with only minimum delays when handing off between access networks.

The main differences that can be found in XtreemOS-MD regarding the XtreemOS architecture are the following modules:

- **Context awareness module:** That provides to higher levels information about the context of the user's terminal, including information like battery level, geographical position, current operation mode and especially the important information about network connectivity, one of the main issues when targeting mobile environments. The intrinsic mobility of mobile devices can derive in a frequent change on the connectivity conditions, moving from one network to another, changing from WiFi to 3G or even loosing temporarily the connection. XtreemOS-MD pays a special attention to this issue and thanks to this module, higher layers may define context-based rules to determine when the mobile device should be "moved" to offline mode (disconnected from the Grid). For example, a user could define a rule to move to offline mode when connected using a non-free 3G connection, or when the remaining battery falls below an specified threshold.
- **Resource sharing module:** Taking into account the limitations of mobile devices, other Grid initiatives have left them apart. Nevertheless, one of the strengths of XtreemOS is the existence of different flavours for PCs, clusters and mobile devices. It's clear that the limitations are there, and even if the computational and storage power of those devices are quickly being increased, for the moment they are not the most suitable options to run jobs or to store data. Additional problems like the lack of reliability and confidentiality appear, as the mobile devices can be manipulated and are quite accessible. Anyway, even having in mind those issues, XtreemOS-MD wants to become not only a mobile Grid client , but also a way to make available some resources of the mobile devices. Computing and storage are not the mobile devices' strengths, but we can think in the possibility of sharing resources like the camera, the GPS, or the network connectivity, and some use cases appear where those resources are needed. For example, a user could share its plane-rate 3G connection to their colleagues in a meeting, connected from their mobile devices to the local Grid via Wifi, but without a free external access to Internet.

  The resource sharing module included in the architecture is in charge of publishing in the Grid the resources offered by each mobile device and also provides the means for accessing them from a Grid client, also covering the privacy issues associated (and, for example, requesting the users authorization before giving access to their resources shared).

4 The Services Layer

Located on top of the Foundation layer, the services layer or G-layer of XtreemOS-MD is in charge of providing access to the Grid services from mobile devices. The three main services offered in the mobile device version are provided by the security client module, which enable users to authenticate against the Grid, the job management module, used for executing and managing jobs in remote machines in the Grid, and the remote file system module, which allows a transparent access to remote data.

4.1 Security Client Module

This module is implemented as a CDA (Credential Distribution Authority) client in charge of the negotiation and obtainment of user's credentials from the XtreemOS CDA server. Once the user's credentials are obtained they are passed to the VO support module in the F-layer and a new session is started in the VO. One of the issues associated with authentication in mobile devices is related to the usual lack of appropriate keyboards, and impossibility of using external authentication devices, like biometric systems (fingerprint, retina recognition, etc.). In fact, many laptops currently provide an integrated fingerprint reader, but none of the mobile devices considered including such a device.

In order to reduce those limitations, XtreemOS-MD offers the possibility of using a proxy for obtaining the credentials needed in the authentication process. Currently XtreemOS-MD does not use SIM card based authentication because it would limit the solution only to devices with a SIM card reader. The proxy (CDAProxy) is the module added to the XtreemOS architecture to communicate the local CDA client running on the mobile device and the remote CDA server, and which could also serve local credentials stored in the proxy. The connection between the client and proxy is secured by SSL. The main benefit is that the proxy may be executed out of the mobile device, in a more powerful user's PC for

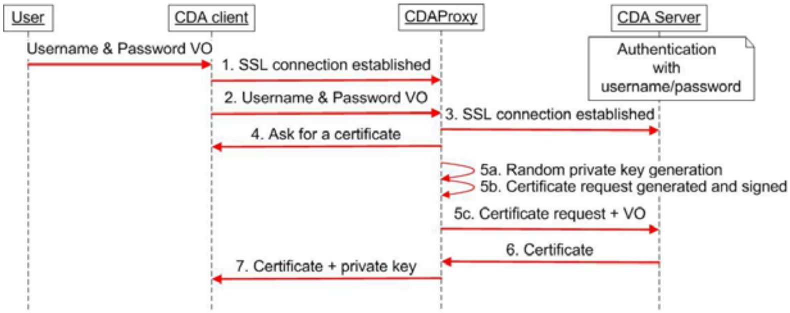

User
CDA client
CDAProxy
CDA Server
Authentication with username/password
Username & Password VO
1. SSL connection established
2. Username & Password VO
3. SSL connection established
4. Ask for a certificate
5a. Random private key generation
5b. Certificate request generated and signed
5c. Certificate request + VO
6. Certificate
7. Certificate + private key

Fig. 2. A CDAProxy can be used by XtreemOS-MD in order to reduce the time needed for certificate generation

instance. That reduces the time for generating the certificate, a heavy computational task, and offers alternative ways of authentication (biometric systems, etc). This delegated authentication process is shown in Figure 2. After the initial connections establishment between the client, the proxy and the server (steps 1 to 3), the client asks for a certificate (step 4). Then, the proxy issues the private key and also generates and signs the corresponding certificate request (steps 5a and 5b). Finally, the certificate request is sent to the CDA server (step 5c), which will answer back with the certificate (steps 6 and 7). The certificate generation is just needed from time to time, depending on the concrete policies related to certificate validity and expiration. It would be possible then to use a proxy running in a nearby machine to perform the certificate generation, which would not be anymore needed once stored the credentials in the mobile device. After credential expiration, the mobile device could connect again to any CDAProxy to renew the credentials.

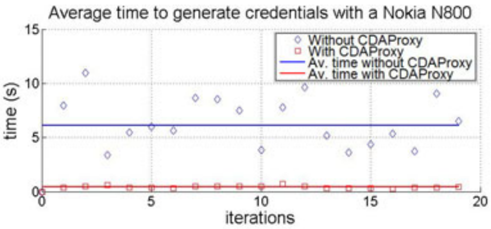


Fig. 3. Certificate generation time with and without CDAProxy

Due to the fact that the proxy would be normally executed in a more powerful equipment than the mobile device, the time for generating the key would be considerably lower than the required time to generate the same key in the mobile device. Several tests have been performed comparing the average time for key generation in a Nokia N800 mobile device and a mid-range PC. Results are shown in Figure 3, where it is noticed that the average time for certificate generation is around 15 times lower (0,41 seconds against 6,1 seconds) when using the CDAProxy instead of the security client in the Nokia N800 device. Also, it's quite remarkable the high variance of the times for credential generation in the Nokia device.

4.2 File System Service (XtreemFS Service)

Transparently integrating XtreemFS, mobile device users will access, from their usual file system clients, not only local files, but also their files stored in the Grid. Taking into account the typical storage limitations of the mobile devices, the XtreemOS-MD users will benefit from a great storage capacity increase, as they will access their remote volume in the Grid (whatever the size of this volume is) from their mobile device.

Before we present how this integration is achieved, and how mobile device limitations have been taken into account, let's first see a very brief summary of how XtreemFS works. XtreemFS is mainly composed by three components MRC (Metadata Replica Catalogue), OSDs (Object Storage Devices), and Clients. The MRC is in charge of maintaining all metadata information about file such as access rights, dates, replica locations, etc. The OSDs are the resources where data is actually stored. A file may be in a single OSD or in several if the file is stripped. OSDs also take care of coordinating stripped files and replicated files. Finally, the client is a user library built using FUSE [11], a file system in user space, which is located below the F-layer as a kernel-level module. Whenever a client wants to access a file, it firsts contacts the MRC to get the information about the OSDs storing the file and a credential to prove it has access to it. With his credential, the client can access the file in the OSDs with no further connection with the MRC (unless the credential expires, in which case a new credential will be requested for this file). The current version of XtreemOS, only allows mobile devices to act as clients (although the possibility of sharing files from the device is currently under development). Nevertheless, we had to design the file system in order to make sure that it was compliant with the mobile devices peculiarities. XtreemOS-MD offers then a transparent access to the remote user's volume based on XtreemFS.

The first important issue in a mobile device is that a stable connection can't be assumed. In XtreemFS, OSDs, do not keep state about the clients (the little state that is actually stored is just for some performance optimizations). A client just needs the credential to be able to access a file from the OSDs, and there is no need for the OSD to recognize the client as a previous one. In the same line, avoiding any kind of persistent link between the file system and the client reduces the battery wasted as only the needed data are sent through the network device. In addition, this mechanism allows the mobile device to go to standby (or even disconnect) between two accesses (i.e. reads) to the same file. Finally, manually mounting the volume is not even needed, as XtreemOS-MD provides an auto-mounting mechanism, which will be also in charge of requesting the user's credential when the volume is going to be used and is still not mounted. Once authenticated the first time, the credentials are stored and the user's password will not be requested again during the current session.

4.3 Execution of Jobs in the Grid Service

The job management service delivers to the user the control to launch, pause, resume, stop or check if a job is running on the Grid. This service is covered by the AEM (Application Execution Management) module which not only manages jobs but also acts as a gateway for the user to the resources associated to them. This component was designed with mobile devices in mind, so main issues were resolved at design level while only minor ones were exposed later when the actual implementation on device platforms took place. AEM is composed of several services running on an XOSD (XtreemOS Daemon) providing a single access point to the system. AEM architecture is distributed, in every node of XtreemOS we will find an XOSD running with several different services depending on the nature of the node. Mainly there are core and resource node types; in a core node we have Job, Execution and Resource management services. On the other hand, a resource node should have Execution and Resource management. However, other services can be run in order to distribute functionality over the Grid. Mobile devices benefit from the non-persistent connection between calls, so the client can change its IP address and continue interacting with the system using his credentials. Client identification and authentication is only based on the certificate presented on each request. Also AEM supports that the clients can change their entry point to the whole Grid, represented by an XOSD. This feature benefits mobile devices in the way that they can simply access the closest node removing the service discovery overhead by leaving it to the grid. Any XOSD is an entry point to the whole AEM. Requests are internally redirected to the correct XOSD and service which is located using the distributed overlay. For mobile devices usage, AEM has a C interface (C-XATI), which is generated by DIXI as well, in addition to the Java one. With this C interface there is no difference between the accessing flavours of AEM. A client in an XtreemOS node can submit a Job, and see its status in the mobile device, set a user monitoring event in the job and receive the feedback in the mobile. C-XATI takes care of some issues affecting mobile devices and other limited devices; external libraries not available in the mobile device and connections. AEM also works as a proxy to register a mobile component (GPS, camera) with the resource discovery service to share it with the Grid or Cloud. The whole monitoring infrastructure of XtreemOS is exposed to the mobile devices through XATI (C-XATI). Every functionality present in the current version of the software has been directly ported to the mobile devices flavour of XtreemOS. Nevertheless, we are currently exploring some specific adaptations, for the case of connection loss from the client side. Particularly, a user might request a call-back on a monitoring event, like a job changing its state, but its connection may be closed before the call-back is sent. In that scenario, AEM would offer the client a way to request pending call-backs whenever it detects the reconnection. This functionality is not only useful for mobile devices, but also for laptops or even desktop clients, although is more of a requisite on them because of the frequency this connection losses could happen.

5 Performance of XtreemOS-MD Services

One remarkable aspect of XtreemOS-MD services is their performance. Regarding the file system and job execution services, we have designed some scenarios to compare the performance of those services when using a mobile device versus the performance of an XtreemOS client running in a regular PC. We have repeated this tests directly on a mobile device, a Nokia N800, to show the benefits of performing those operations in the Grid instead of doing them directly on the device itself, which is less powerful in terms of computation. The results are reflected in Figure 4, where it can be appreciated than the performance of XtreemOS-MD running in the Nokia N800 is similar to the one achieved with the XtreemOS client running in a PC (a Pentium 4). But also it's remarkable the performance improvement obtained when using XtreemOS-MD while compared to the direct execution of the operations in the mobile device. The four different tests reflected in Figure 4, where the values shown in axis Y are the average from a series of different realizations, correspond to:

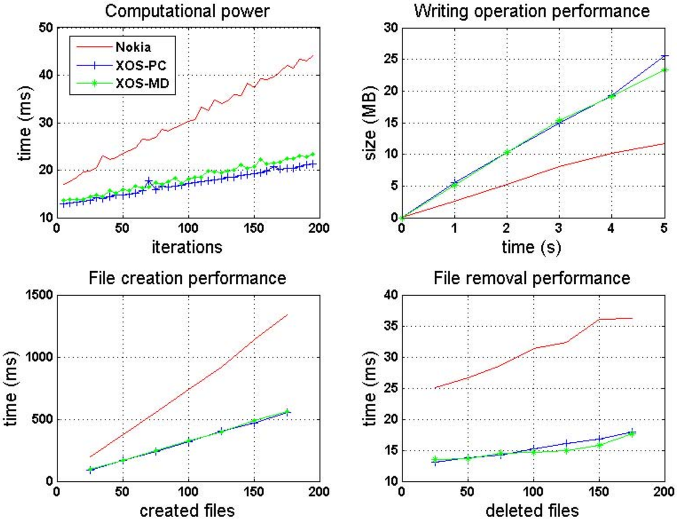


Fig. 4. Comparison of computational power (job execution) and performance of some file operations when using directly the mobile device (Nokia N800), XtreemOS client running in a PC and XtreemOS-MD running in the Nokia N800

- Computational power: a simple script including a loop incrementing a variable that is passed a number of iterations.
- File creation performance: a script that creates a number of empty files in the file system (XtreemFS or local file system in the Nokia realization).
- File removal performance: a script that removes a number of empty files in the file system (XtreemFS or local file system in the Nokia realization).
- Writing operation performance: a script that writes some data in a file of the file system (local in the Nokia or in the XtreemFS when using XtreemOS and XtreemOS-MD) for a defined time. The axis Y represents the average size of the file generated during the time represented in axis X.

These results show that XtreemOS-MD services have a similar performance than XtreemOS ones. That means that the time for completing any operation on the Grid does not depend on the client used (PC with XtreemOS or mobile device with XtreemOS-MD). The reason behind is that the communications and protocols designed to interact with the Grid are light enough to be implemented even in limited mobile devices. The use of native code in the mobile devices against the Java code used by the implementation of the services for the XtreemOS PC version is one of the reasons permitting a quite similar performance of XtreemOS-MD services running on a mobile device and XtreemOS services running on a more powerful PC.

Those tests are simple tests for performance evaluation, but thinking on a real use case, we have designed a different scenario, related with video conversion in the Grid. To test this use case we have launched several jobs of video conversion in the Grid, using the job execution service for launching the jobs, and also directly in a Nokia N800 mobile device. For the Grid-aware tests, the input video file is read from the user's grid volume, then it is converted to a different video quality and finally the output video file is stored in the same user's volume in the grid. In both cases, mobile and PC, the Grid testbed used was the same. The result, obviously dependent of the file size, has shown to be completely independent of the client used, be it either a PC or mobile device, as stated in Figure 5 where the red line shows the average time for the video conversion job when launching the job from a PC running XtreemOS and the green line corresponds to the average time when launching the job from a mobile device running XtreemOS-MD.

But the most remarkable result is the comparison with the time spent by the Nokia N800 in the video conversion process, especially with the increase of the file size. For a video file of just around 30 MB, the time spent by the Nokia is around 20 minutes, 10 times more than the Grid. And it's not just a matter of time, but also a problem of battery consumption. Some measurements relative to the battery consumed by the Nokia N800 during the conversion process have been taken and the consumption for a file of 30 MB is around 4% of the total capacity (around 10 times higher than the battery power that would have been consumed during a similar period of inactivity).

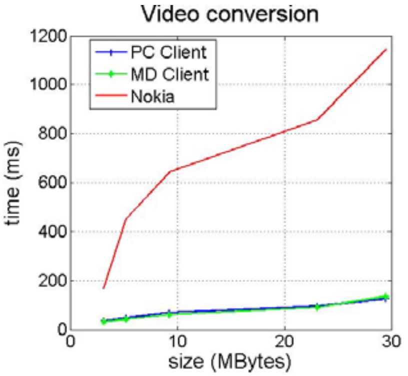


Fig. 5. Video conversion performance in the Grid (PC and mobile device client) and in a Nokia N800

6 XtreemOS-MD Applications

One of the strenghts of XtreemOS-MD is the possibility of using all the existing classical non-Grid applications, and even modified them slightly to add Grid features to them. For example, it's much more simple to extend an existing instant messaging client (like Pidgin for example) to incorporate the support for writing logs, etc. in the XtreemFS or to allow chatting with anyone connected to your same VO than writing from scratch a new instant messaging Grid-aware application as would be necessary if the Grid client would be based in a simple web portal solution. Apart from those existing or modified applications, some additional applications have been developed to be released together with the XtreemOS releases, offering some interesting specific functionalities.

6.1 The JobMA Application

The classical Linux command-line interface is not the most appropriate one when targeting mobile phone users. In order to simplify the process for managing jobs in the Grid, XtreemOS-MD provides a graphical application called JobMA (Job Management Application). This application provides an intuitive access to XtreemOS job management facilities like create and launch jobs, stop, resume, or cancel running jobs, and also job monitoring. Advanced jobs, specified by JSDL (Job Submission Description Language) file [2], can be loaded and executed, but also basic jobs can be defined with a specific GUI form.

6.2 The Grid-Player Application

Nowadays, a video player is one of the features with more demand in a mobile device. Nevertheless, the video capabilities and the low screen resolutions of those devices highly limit their possibilities. This opens the door to a new multimedia application, that we called the Grid-Player, consisting on a video player capable of translating the original format and resolution of a video when not supported in the mobile device (for example, when codec is not available, etc.). The process for video transcoding is really heavy from a computational point of view, so that the remote processing in the Grid instead of directly on the local device is an important strength of the application; and this is not just a question of saving time, but also saving battery power, one of the main limitations of current mobile devices. The architecture of this Grid-Player application is shown in Figure 6, where it's reflected that the Multimedia Player application running in the mobile device will connect to the Grid in order to transcode one video stored in the user's file system (or accessible via HTTP), storing again in the user's volume the resulting video, optimized for the mobile device that requested the conversion (codec, resolution, etc.)

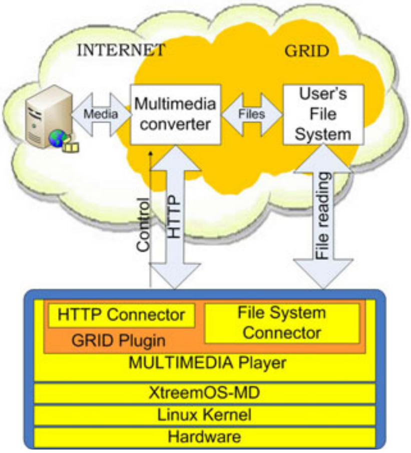


Fig. 6. Grid-Player application architecture

A Grid-Player first prototype, based on the JobMA application and making use of open source video-conversion software (*ffmpeg* library) running on the Grid, has already been demonstrated with a very good acceptance. In fact, this prototype has been used to execute the tests which results are shown in Figure 5. This kind of application could serve as a motivation to use the Grid (and concretely XtreemOS-MD), even not having the users a real knowledge on the infrastructure behind. For them, it will be just an application for playing local or remote videos stored in whatever format, resolution or video codec used.

7 Conclusions

Along this article we have presented XtreemOS-MD as a new way to enter in the Mobile Grid Computing arena. The transparent Grid access from mobile devices will offer the possibility of using a wide range of applications from a quite limited device like a PDA or a smartphone. Thinking on the increasing importance of mobility, XtreemOS-MD could be very interesting as a new way to use remote desktop applications, like typical office packages for instance, or to improve mobile device applications, including multimedia or games, by using Grid capabilities directly from the mobile device. Finally, taking into account the number of mobile devices around the world, and the XtreemOS-MD ease of use, it could serve as a catalyst for popularizing the Grid, extending their use to non computer savvy users. It should be noticed that XtreemOS-MD first prototype has been released before summer 2009. This 1.0 release includes the access to the main XtreemOS services from mobile devices like the Nokia N8x0 PDA and the NeoFreeRunner smartphone.

Acknowlegments

This work was partially supported by the EU IST program as part of the XtreemOS project (contract FP6-033576), by the Spanish Ministry of Science and Technology under the TIN2007-60625 grant, and by the Catalan Government under the 2009-SGR-980 grant.

References

1. Amazon Web Services LLC. Amazon Elastic Compute Cloud (Amazon EC2), http://aws.amazon.com/ec2
2. Anjomshoaa, A., Drescher, M., Fellows, D., Ly, A., McGough, S., Pulsipher, D., Savva, A.: Job Submission Description Language (JSDL) Specification (November 2005), http://www.gridforum.org/documents/GFD.56.pdf
3. Clarke, B., Humphrey, M.: Beyond the "Device as Portal": Meeting the Requirements of Wireless and Mobile Devices in the Legion Grid Computing System. In: Proceedings of the 16th International Parallel and Distributed Processing Symposium table of contents. IEEE Computer Society, Washington (2000)
4. Chu, D.C., Humphrey, M.: Mobile OGSI.NET: Grid Computing on Mobile Devices. In: Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing (GRID 2004), November 08-08, pp. 182–191 (2004)
5. Coppola, M., Jégou, Y., Matthews, B., Morin, C., Prieto, L.P., Sanchez, O.D., Yang, E.Y., Yu, H.: Virtual organization support within a grid-wide operating system. IEEE Internet Computing 12(2) (2008)
6. Cortes, T., et al.: XtreemOS: a Vision for a Grid Operating System, White paper (May 2008)
7. Erwin, D.W., Snelling, D.F.: UNICORE: A Grid computing environment. In: Sakellariou, R., Keane, J.A., Gurd, J.R., Freeman, L. (eds.) Euro-Par 2001. LNCS, vol. 2150, p. 825. Springer, Heidelberg (2001)
8. González-Castaño, F.J., Vales-Alonso, J., Livny, M., Costa-Montenegro, E., Anido-Rifón, L.: Condor grid computing from mobile handheld devices. ACM SIGMOBILE Mobile Computing and Communications Review 6(2), 18–27 (2002)
9. 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)

10. Foster, I., Kesselman, C.: The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
11. FUSE: Filesystem in Userspace, http://fuse.sourceforge.net/
12. Globus Alliance. The Globus Toolkit 4.1.3 Developer's Guide, http://www.globus.org/toolkit/docs/development/4.1.3/developer/index.html
13. Condor Team. Condor Manual. University of Wisconsin-Madison, http://www.cs.wisc.edu/condor/manual/v7.5/
14. Grimshaw, A.S., Humphrey, M.A., Natrajan, A.: A philosophical and technical comparison of Legion and Globus. IBM Journal of Research and Development 48(2), 233–254 (2004)
15. Grimshaw, A.S., Wulf, W.A., the Legion Team: The Legion vision of a worldwide virtual computer. Communications of the ACM 40(1), 39–45 (1997)
16. Google Inc.: Google Application Engine, http://code.google.com/intl/appengine/appengine/
17. Gridipedia web site. The history of Grid, http://www.gridipedia.eu/historyofgrid.html
18. Nokia. Maemo operating system, http://nokia.maemo.com
19. Microsoft Corporation. Windows Azure Platform, http://www.microsoft.com/azure/default.mspx
20. Millard, D., Woukeu, A., Tao, F.B., Davis, H.: Experiences with Writing Grid Clients for Mobile devices (2005)
21. Morin, C.: XtreemOS: A grid operating system making your computer ready for participating in virtual organizations. In: Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC 2007), pp. 393–402 (May 2007)
22. Novotny, J., Russell, M., Wehrens, O.: GridSphere: a portal framework for building collaborations. Journal of Concurrency and Computation: Practice and Experience 16(5), 503–513 (2004)
23. OpenMoko. SHR (Stable Hybrid Release) distribution, http://wiki.openmoko.org/wiki/SHR
24. Phan, T., Huang, L., Dulan, C.: Challenge: integrating mobile wireless devices into the computational grid. In: Proceedings of the 8th Annual International Conference on Mobile Computing and Networking, Atlanta, Georgia, USA, September 23-28 (2002)
25. SYS-CON Media Inc. Twenty-One Experts Define Cloud Computing (2008), http://cloudcomputing.sys-con.com/node/612375/print
26. Thain, D., Tannenbaum, T., Livny, M.: Condor and the Grid in Grid Computing: Making the Global Infrastructure a Reality. In: Berman, F., Fox, G., Hey, T. (eds.), Wiley, Chichester (2002)
27. USAGI Project. Linux IPv6 Development Project, http://www.linux-ipv6.org/
28. Wesner, S., Jähnert, J.M., Toro, M.A.: Mobile Collaborative Business Grids - A short overview of the Akogrimo Project
29. XtreemOS project. Building and Promoting a Linux-based Operating System to Support Virtual Organizations for Next Generation Grids, http://www.xtreemos.org/

The Proxy-Based Mobile Grid

Azade Khalaj, Hanan Lutfiyya, and Mark Perry

Computer Science Department
The University of Western Ontario, London ON N6A 3K7, Canada
{akhalaj,hanan,markp}@csd.uwo.ca

Abstract. The increase in the popularity of small digital mobile devices also implies an increased demand in applications. The limited computing capabilities on the mobile devices and the unreliability of wireless links are barriers to the smooth access of mobile devices to the Grid applications and resources. In this paper a proxy-based approach is presented that is able to support various kinds of applications to be used by mobile devices by providing specific-purpose services on the proxy. The implemented prototype that includes some of the realized proxy services and the example client application for a mobile device show the viability of the proposed approach.

Keywords: Mobile Grid, Grid Computing, Mobility, Proxy, Proxy-based Architecture.

1 Introduction

In recent years we have seen a proliferation of mobile consumer electronic devices e.g., smartphones, PDAs and tablet PC. With this proliferation there is an increased demand for the following: (i) The ability to run resource-intense applications such as video playing or editing. However, the devices have limited compute power due to size and weight constraints. This suggests a need to offload computation from mobile devices to servers with sufficient computational power; (ii) Access to peripheral devices such as printers; (iii) Access to remote services that rely on information from multiple sources. For example, urban planners are proposing to allow city residents to provide information in real-time about specific events e.g., car sensors may provide information about traffic patterns. In other words there is a desire to access through a mobile device a "hardware and software infrastructure that provides dependable, consistent, pervasive and inexpensive access to high-end computational capabilities and peripheral devices" [24]. We will refer to this infrastructure as the Mobile Grid.

The challenges in providing seamless and transparent access to the mobile grid include the following: (i) Most mobile devices have limited computing resources; (ii) Devices are mobile and often connect to the Grid through wireless connections which are not as reliable or have the same bandwidth as wired connections; (iii) Battery power is limited and this may cause frequent disconnection of mobile devices.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 59–69, 2010.

The literature shows two main approaches to creating and accessing a Mobile Grid infrastructure. In the first approach researchers extend the available tools and middleware [1,20,29,9] for the Grid computing. A software is installed on the mobile device. The software delegates tasks to components added to the Grid middleware on the wired part of the Grid. The second approach, using a proxy-based architecture, introduces a proxy component (several example of this approach are referenced in section 5). In this approach, the application on the mobile device delegates the task of communication with services on the Grid to a proxy machine which is assumed to be a highly capable node on the wired network. The components on the Grid see the proxy as a Grid component and applications on the mobile device interface with proxy and communicate with it as a bridge to the Grid. In this approach, the Grid middleware remains unchanged and the resource constrained mobile devices are not exposed to the complexity of the Grid, since dealing with these complexities is shifted to the proxy.

The limitation of the first approach is that it does not adequately address issues related to accessing peripheral devices or remote services since the Grid middleware typically focusses on executing submitted tasks. Using a proxy eliminates the need to alter the currently deployed Grid middleware. A proxy is more flexible and thus facilitates access to peripheral devices or remote services. The use of proxies is challenging in itself.

The goal of our work is to develop a middleware that facilitates the use of Grid resources for mobile devcies through a proxy. Grdi resources include application services or hardware. However, the development of a middleware requires that we identify the types of services that a proxy should provide. These services enable proxies to support different applications with different requirements.

These services can be implemented as independent services on the proxy that are to be called Proxy Services and are deployed on the proxy when a client application has a request to use it. When there is no client using a Proxy Service, the Proxy Service can be terminated on the proxy machine to free resources for other processes. In this way, dynamic proxies will be created that have a changing set of Proxy Services. Beside the identified possible Proxy Services, the process of finding a proxy (by a mobile device) and two application scenarios as usecases of our proxy-based Mobile Grid infrastructure are described in this paper.

The remainder of the paper is organised as follows. In section 2 two usecases are explained that in them applications on the mobile devices use our infrastructure to call some Grid services. Section 3 describes the role of proxy in our proposed Mobile Grid infrastructure. A simple prototype of our proposed infrastructure and an example application which is used to test the prototype are presented in section 4. Related research done in this area and the conclusion and future work are presented in sections 5 and 6 respectively.

2 Usecases

To understand the requirements needed to be provided by an infrastructure that uses proxies, two possible application scenarios are presented in this section.

These applications communicate different kinds of data with services and components on the Grid. The proxy machine is the actual client of services on the Grid, so the Grid should be available only on the proxy. It is enough to have a light-weight application on the mobile device to communicate to the proxy. The proxy can buffer the results received from Grid resources in the case of a disconnection between the mobile device and the wired network and deliver the result to the mobile devices whenever it reconnects to the wired network.

2.1 The e-Health Application Scenario

Imagine a situation where a paramedic is treating the patient at the scene of an accident or at the patient's home. The paramedic has an application on his/her handheld device that is able to use teh proxy provided in our Mobile Grid middleware as a client. We assume that the application has triggered the process of finding an appropriate proxy after being launched by the paramedic on the cell phone (the details of the process of finding a proxy are described in section 3.2). Upon completion of this process a proxy is assigned to the application that has the requested Proxy Services of the client application deployed on.

The paramedic gathers the needed information about the health status of the patient using the medical equipment available in the ambulance. If an "ECG Signal Aalyzer" service is needed the paramedics uses the client application on the handheld device to discover the ECG signal analyzer service on the Grid. The paramedic might need to talk to a consultant at the affiliated medical center through an audio/video connection. This connection can be established between the paramedic's cell phone and the medical help center. Since this connection passes through the proxy, a specific Proxy Service on the proxy can be used to control the QoS level for the audio/video stream based on the condition of the wireless link and the hardware specification of the device. If it is necessary the "Hospital Finder" service can be invoked by the client application in the same way as invoking the "ECG Signal Analyzer" and an appropriate hospital is selected taking into account several factors such as the distance, the traffic condition of route to the hospital and the availability of required facilities at the hospital.

Since all this communication passes through the proxy, the proxy can provide a Proxy Service which is responsible for handling disconnections. The Disconnection Handler Proxy Service is responsible for buffering messages sent by the Grid services, e.g. "the ECG Signal Analyzer", to the client application at the proxy in the case of a disconnection happening between the mobile device and the network. After the reconnection, the Disconnection Handler Proxy Service will send the buffered messages to the client application on the handheld device.

2.2 Finding the Closest Printer

A university provides its visitors with access to some of the campus printers by providing a "Printer Finder" service. The "Printer Finder" service is implemented to find an appropriate printer for the user. An appropriate printer is the one that the visitor has permission to use and is located in the proximity of the user. A small client application for handheld devices is provided by the university that is designed to call the provided web service. Visitors are able to call the "Printer Finder" service at anytime and anywhere to find the closest accessible printer. The client application for the handheld device is designed to use our Mobile Grid infrastructure to call the "Printer Finder" service provided by the university. If a person visiting the campus needs to find a printer, he/she launchs the client application. The client application first starts the process of finding a proxy and after that it submits a request to the proxy that includes the address of requested service and its current GPS position. Consequently, the proxy contacts the proxy finder service and relay the request of mobile device to it. After receiving the result from the printer finder web service, the proxy sends back the result which includes information about the found printer to the client application. The client application is supported by the disconnection handler mechanism provided in our infrastructure. If during a call to the "Proxy Finder" service a disconnection appears, the Mobile Grid infrastructure is able to buffer the result for the client application and deliver the result after a reconnection to the client.

3 Our Proposed Approach

In our proxy-based Mobile Grid infrastructure proxies play the role of gateway for mobile nodes to the Grid. The proxy enables mobile devices to be part of the Grid as a resource provider or resource consumer by doing some tasks on behalf of the mobile device. A proxy is machine on the wired network that has access to the services provided by the Grid. A proxy can provide several services for the proxy based architecture. These services are named Proxy Services and can be deployed or un-deployed on the proxy dynamically based on the requirements of the client applications on the mobile device associated to the proxy. The proxy can execute other programs beside Proxy Services providing for the Mobile Grid. Some of the possible Proxy Services are introduced later in this section. A client application on the mobile device should know the Proxy Services that it needs for its operation. The client asks its proxy to deploy the required Proxy Services and after the deployment of Proxy Services the client application can use Proxy Services to communicate with the resources on the Grid. When the client application is terminated and there are no more client applications using a Proxy Service, the proxy can un-deploy the Proxy Service to free the resources assigned to that Proxy Service. In this way dynamic proxies are created; Proxies provide a dynamic set of Proxy Services that changes according to the requirements of the client applications.

In our approach the application on the mobile device needs to know the address of machines which are currently proxies. The addresses and other information about these nodes are kept in several Proxy Finder Servers. The client application should have the address of at list one Proxy Finder Server and it is the responsibility of the Proxy Finder Server to find the appropriate proxy for a client application. Proxy Finder Servers are also responsible for handling issues related to the mobility of mobile devices and failure at proxies.

PFSs and proxies can be owned by the Grid resource provider that intends to provide its mobile clients with access to its Grid resource. Another possible owner for PFSs and proxies can be a ”Mobile Grid Provider”. Grid resource providers subscribe to this ”Mobile Grid Provider” company to make accessible their recources by mobile clients. In both cases clients can receive the address of PFSs from the provider of their intended Grid resource.

3.1 Proxy Services

Proxies provide services for mobile devices that enable the application on mobile devices to use the resources available on the Grid smoothly even in the presence of disconnections. In this section we describe a set of Proxy Services.

- **Relay -** A role of the proxy can be the relaying of data between an application on a mobile device and the Grid resource. In this case the application on mobile device which is using a service on the Grid submits its request to the proxy. The proxy relays this request to the service and after that relays the results received from the recourse to the client on the mobile device. The Relay Proxy Service is used in the case of occurrence of a disconnection or changing of the proxy.
- **Downscale -** For some applications that have multimedia streaming or the image transmission, the proxy can downscale the traffic with a suitable transmission bit rate to the mobile device according to mobile devices physical specification or the transient condition of wireless communication links. By shifting the task of downscaling to the proxy, services provided originally for powerful desktop machines connected to high bandwidth wired links can be used by mobile devices without any change. One example use of this Proxy Service was shown in the first usecase (section 2.1).
- **Grid Service Discoverer -** The proxy can find the Grid services needed by the client application on the mobile device. The Grid Service Discoverer Proxy Service receives the information about the required service from the client application and tries to find the service by searching its service repository. This Proxy Service can be used in the healthcare domain usecase presented in section 2.1 to find a “Hospital Finder” or “ECG Siganl Analyzer” services.
- **Disconnection Handler -** Failure might happen during a session for several reasons, such as a disconnection between the mobile device and the network or the low battery power of the mobile device. The Disconnection Handler Proxy Service can buffer the results received from the invoked service on the proxy and deliver the result to the client application after the reconnection. In both usecases a Disconnection Handler Proxy Service can be used to buffer the results received from the Grid services on the proxy.
- **Task offloader -** Because of the limited resources on a mobile device an application on the mobile device might want to offload parts of its tasks to the proxy. The proxy processes these tasks locally or it may submit them to a machine with available resources on the Grid.

- **Checkpointing -** The mobile device can send the checkpoints of its running applications to the proxy. This is often necessary since checkpoints may need a good deal of storage which may not be available at the mobile device. The Checkpointing Proxy Service is responsible to manage checkpoints received from client applications.
- **Movement tracker -** This Proxy Service tracks mobile device and can be queried at any time by other Proxy Services to retrieve the current location of the mobile device.
- **Masquerade -** A proxy can create a cluster of mobile devices which volunteer to provide resources, e.g. CPU cycle or storage space for the Grid applications. The proxy can creates a bridge for the cluster. This allows Grid users which are willing to use these available resources see an incorporated resource presented by the proxy [14,19,15]. The Masquerade Proxy Service is responsible for providing a set of services to create an incorporated resource. This set of services can include:
  - Task assignment to mobile devices based on their available shared resources.
  - Task replication to achieve higher reliability.
  - Load balancing among mobile devices to avoid overloading a device while there are other devices with available resources.
  - Failure prediction; The proxy monitors the residual battery power of devices by querying devices or trace their movement to predict a possible failure in near future.
  - Task migration; By predicting a failure the proxy can migrate the assigned task and its associated data from the device close to failure, to another mobile device in the cluster.

3.2 Components Interaction

The detailed description of operation and the interaction between different components of the system are presented in the following subsections.

Proxy Registration at Proxy Finder Servers. The proxy registers at several Proxy Finder Servers in its proximity by providing the information about itself. The information includes the IP address, GPS position and its available Proxy Services. By registering at several Proxy Finder Servers, the resources offered by a proxy can be more widely accessible. Ideally the proxy registers itself in Proxy Finder Servers in its proximity. In this case each Proxy Finder Server has a list of proxies close to it. The information about proxies saved at Proxy Finder Server might be updated by proxies. For example if a proxy decides to dedicate a smaller share of its processing power to the Mobile Grid it informs the Proxy Finder Server that it is registered with.

Find a Proxy. The client application should know the address of at least one or more Proxy Finder Server beforehand. Thus, the mobile device can send a message to one of the Proxy Finder Servers and request it to find an appropriate proxy. The mobile device should specify what kind of Proxy Services it expects the proxy to provide. The Proxy Finder Server selects a proxy from the list of registered proxies based on a set of input parameters. The first parameter is the geographical location of the proxy; a proxy is chosen which is closest to the current location of the mobile device. The second parameter is the list of available Proxy Services that a proxy is able to provide. By knowing the current location of mobile device and the list of required Proxy Services from the client application on the mobile device, the Proxy Finder Server can select one (or several) proxy (or proxies) and sends a request message on behalf of the mobile device to the chosen proxy (or proxies).

The decision to accept the client is made by the proxy, i.e. the proxy accepts a client's request if it has enough resources to provide services for a new client. To accept a new client the proxy might need to deploy a new Proxy Service or need to download the Proxy Service code and install it. The proxy can accept a new client based of on its available resources and its current load. If the current load on the proxy is too heavy to accept a new client the proxy can decide to reject the request.

If the proxy decides to accept the request it informs the Proxy Finder Server about the acceptance and deploys required Proxy Services (if not deployed yet). Upon receiving the first accept message from a proxy, the Proxy Finder Server sends the address of the proxy to the client application on the mobile device. Afterward the client application can start to use the Proxy Services on the proxy to communicate with resources on the Grid.

Change the Proxy of a Mobile Device. This architecture also supports the change of the associated proxy of a mobile device under some circumstances. The associated proxy to a mobile device may change if the device moves to a location far from the proxy, or the proxy decides to reduce its load in the case of overloading.

The decision of transferring the session of a mobile device to another proxy is made by the proxy. Since the proxy might have other programs beside the Proxy Services it is hosting, it monitors its resources and when it is becoming overloaded or needs more resources for its other programs, the proxy may decide to transfer sessions related to a client to another proxy.

As mentioned before, a proxy can transfer the sessions of a client to another proxy if the mobile device moves to a location far from the proxy. To know if the mobile device is too far from the proxy, the proxy needs to trace the mobile device. The mobile device periodically sends its current location to the proxy. If the proxy notices that a mobile device has moved to a location far from the proxy, it may decide to transfer the session of that device to another proxy closer to the mobile device. It is the responsibility of proxy to find a new appropriate proxy closer to current location of mobile device and transfer its session to the new proxy. The process of transferring the session is transparent to the mobile device, i.e. mobile device sees a continuous service during the transferring process.

Handling the Proxy Failure. To be aware of a failure at a proxy, it is required that the availability of the proxy be checked regularly. Performing this check by mobile devices associated with the proxy is not a good option, since frequent disconnections may happen between mobile devices and the proxy because of the movement of mobile device or the changing condition of the wireless media. An alternative solution is to assign the task of checking the availability of proxies to the Proxy Finder Server. Whenever a proxy accepts a new request from a Proxy Finder Server the proxy periodically sends a keep alive message to the Proxy Finder Server. On the other hand, if the Proxy Finder Server does not receive a keep alive message after a period of time it assumes that the proxy has failed. In this case the Proxy Finder Server starts the process of finding a new proxy for mobile devices associated with the failed proxy.

4 Prototype

To test the viability of the proposed approach a prototype is implemented. The prototype includes the basic parts of the infrastructure. The Proxy Finder Server is implemented to be able to register the proxies. The functionality of finding an appropriate proxy for requests sent by mobile devices is implemented for the Proxy Finder Server as well. Currently, two proxy services are implemented: Relay and Disconnection Handler. As an example client application, the second scenario described in section 2 is implemented. For the first version of the prototype, GPS coordinates are not used for calculating the distance between components. Instead, a proxy is selected randomly.

The mobile device used in our experiments is an HTC Magic smartphone and Android 1.6 is used to implement the client application on the smartphone. Our experiments includes one machine as the Proxy Finder Server and four machines as proxies. The Proxy Finder Server functionality, the proxy functionality (which includes the registration process) and Proxy Services are implemented as web services. The Apache Tomcat 6.0 is used as our application server on the Proxy Finder Server and proxy machines.

5 Related Work

In the proxy-based approach to creating the Mobile Grid the responsibility of supporting the mobile devices is shifted to the proxy and there is no need to change the deployed Grid middleware. There are several papers published with focus on the proxy-based Mobile Grid architecture that proxy is employed to provide a specific service for specific applications. In some papers the proxy is used as a bridge to the Grid [26,5,3,2,27,12]. The proxy plays the role of a client for the Grid resources on behalf of the mobile device. A light-weight application is installed on the mobile device to be able to communicate with the proxy and there is no need to have the Grid middleware on the mobile device. In several research the proxy is used to offload tasks from the resource constrained mobile devices to more powerful resources on the Grid [16,31,4]. The earlier research on the mobile Grid is mostly focused on making it possible to integrate mobile devices as resource providers to the Grid through a proxy. Resources can be services available on mobile devices or the processing capabilities of the mobile device. [18] and [11] are examples of this role of proxy. The cluster-based design is used in several research [14,19,23,25,7,22,15,13]. A cluster of mobile devices is created by a proxy node with the aim of hiding the heterogeneity and dynamic nature of the mobile wireless environment from the clients on the wired network. In addition, the proxy-based architecture is used to support multimedia applications on the mobile devices [28,30,17] and for peer-to-peer resource sharing [8,6]. There are also some papers on providing secure communication for mobile devices using a proxy [10,12,6] and also control the level of Quality of Service (QoS) [11,21,12] by monitoring done at the proxy.

In almost all of this work a specific type of applications is targeted and the proxy is designed to support the requirements of that application. In our infrastructure several kinds of application can be supported each one with a related Proxy Service on the proxy. Even in the case of emergence of new applications with new requirements an appropriate Proxy Service to handle those new requirements can be designed and added to the proxies. Another advantage of our approach to the previous work is that the proxies in our infrastructure change based on the demands of the clients. A proxy service is deployed on the proxy if there is a client requesting that service. If a proxy service in not in use by any client that service can be un-deployed to preserve the resources on the proxy.

6 Conclusion and Future Work

The increasing popularity of small digital mobile devices leads to a higher demand to enabling these devices to support various kinds of applications as ones which are available for desktop computers. An obstacle to meeting this demand is limited hardware capabilities of mobile devices. The characteristics of wireless communication also add challenges in using the available Grid resources by the mobile devices. The proxy-based Mobile Grid can support the mobile devices by offloading the compute-intensive tasks from resource restricted mobile devices to the more capable machines on the Grid and also by handling the disconnections appear because of the unreliability of wireless links. In this paper we explained what services should be available at the proxy to meet the requirements of various kinds of applications on a mobile device. An initial version of a prototype and a simple example application are implemented to show the the viability of the proposed approach.

Currently, it is assumed that the set of proxy services available on the proxy machine is fixed. Though, to preserve resources on the proxy, the proxy services available on the proxy can change dynamically. Implementing the dynamic proxies are postponed to the future work. Using the GPS coordinates to calculate the distance between components in the system is planed to be done in the future. Implementation of other proxy services mentioned in section 3.1 and example scenarios that use these proxy proxies are also postponed to the future. To find the overhead added to comunications by using our proposed infrastructure and also performance measurement, several experiments should be disigned and executed. Designing and executing the performance measurement experiments are planned to be done as well for future work.

References

1. Coronato, A., De Pietro, G.: MiPeG: A middleware infrastructure for pervasive grids. In: Future Generation Computer Systems, pp. 17–29 (2008)
2. Khatua, S., Dasgupta, S., Mukherjee, N.: Pervasive Access to the Data Grid. In: The International Conference on Grid Computing and Applications (2006)
3. Singh, A., Trivedi, A., Ramamritham, K., et al.: PTC: Proxies that Transcode and Cache in Heterogeneous Web Client Environments. In: World Wide Web, pp. 7–28 (2004)
4. Rossetto, A.G.M., Borges, V.C.M., Silva, A.P.C., et al.: SuMMIT - A Framework for Coorcinating Applications Execution in Mobile Grid Environment. In: The 8th IEEE/ACM International Conference on Grid Computing, pp. 129–136 (2007)
5. Sajjad, A., Jameel, H., Kalim, U., et al.: AutoMAGI - an Autonomic middleware for enabling Mobile Access to Grid Infrastructure. In: Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services, pp. 73–73 (2005)
6. Harrison, A., Kelley, I., Mieilica, E., et al.: Mobile Peer-To-Grid Architecture for Paramedical Emergency Operations. In: Proceeding of Challenges and Opportunities of HealthGrids, pp. 283–294 (2006)
7. Huang, C., Zhu, Z., Wu, Y., et al.: Power-Aware Hierarchical Scheduling with Respect to Resource Intermittence in Wireless Grids. In: Proceeding of Machine Learning and Cybernetics, pp. 693–698 (2006)
8. Yang, C., Chen, C., Chen, H., et al.: A Peer-to-Peer File Resource Sharing System for Mobile Devices. In: The 3rd International Conference on Grid and Pervasive Computing Workshops, pp. 275–280 (2008)
9. Coulson, G., Grace, P., Blair, G., et al.: A middleware approach for pervasive grid environments. In: UK e-Science Programme Workshop on Ubiquitous Computing and e-Research (2005)
10. Cho, H., Lee, B., Kim, M., et al.: A Secure Mobile Healthcare System based on Surrogate Host. In: The Sixth IEEE International Conference on Computer and Information Technology (2006)
11. Ohta, K., Yoshikawa, T., Nakagawa, T., et al.: Design and implementation of mobile grid middleware for handsets. In: 11th International Conference on Parallel and Distributed Systems, vol. 2, pp. 679–683 (2005)
12. Racz, P., Burgos, J.E., Inacio, N., et al.: Mobility and QoS Support for a Commercial Mobile Grid in Akogrimo. In: 16th IST, Mobile and Wireless Communications Summit, pp. 1–5 (2007)
13. Choi, S.K., Cho, I.S., Chung, K.S., et al.: Group-based Resource Selection Algorithm Supporting Fault-Tolerance in Mobile Grid. In: Third International Conference on Semantics, Knowledge and Grid, pp. 426–429 (2007)
14. Isaiadis, S., Getov, V., Kelly, I., et al.: Dynamic Service-based Integration of Mobile Clusters in Grids. In: Grid Computing: Achievements and Prospects, pp. 159–171 (2008)
15. Katsaros, K., Polyzos, G.C.: Optimizing operation of a hierarchical campus-wide mobile grid. In: The 18th IEEE Personal In-door and Mobile Radio Communications conference, PIMRC (2007)
16. Trung, T.M., Moon, Y., Youn, C., et al.: A gateway replication scheme for improving the reliability of mobile-to-grid services. In: IEEE International Conference on e-Business Engineering, pp. 456–463 (2005)

17. Mohapatra, S., Cornea, R., Dutt, N., et al.: Integrated power management for video streaming to mobile handheld devices. In: The Eleventh ACM International Conference on Multimedia, pp. 582–591 (2003)
18. Hampshire, A.: Extending the open grid services architecture to intermittently available wireless networks. In: UK eScience All Hands (2004)
19. Phan, T., Huang, L., Dulan, C.: Challenge: Integrating Mobile Wireless Devices Into the Computational Grid. In: The 8th ACM International Conference on Mobile Computing and Networking (2002)
20. Chu, D., Humphrey, M.: Mobile OGSI.NET: Grid Computing on Mobile Devices. In: Fifth IEEE/ACM International Workshop on Grid Computing (GRID 2004), pp. 182–191 (2004)
21. Hwang, J., Aravamudham, P.: Middleware Services for P2P Computing in Wireless Grid Networks. IEEE Internet Computing 8(4), 40–46 (2004)
22. Kim, I.K., Jang, S.H., Lee, J.S.: Adaptive Distance Filter-based Traffic Reduction for Mobile Grid. In: 27th International Conference on Distributed Computing Systems Workshops, pp. 8–8 (2007)
23. Katsaros, K., Polyzos, G.C.: Evaluation of scheduling policies in a Mobile Grid architecture. In: International Symposium on Performance Evaluation of Computer and Telecommunication Systems (2008)
24. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco (1998)
25. Park, S., Ko, Y., Kim, J.: Disconnected Operation Service in Mobile Grid Computing. In: Orlowska, M.E., Weerawarana, S., Papazoglou, M.P., Yang, J. (eds.) ICSOC 2003. LNCS, vol. 2910, pp. 499–513. Springer, Heidelberg (2003)
26. Koufi, V., Vassilacopoulos, G.: HDGPortal: A Grid portal application for pervasive access to process-based healthcare systems. In: Second International Conference on Pervasive Computing Technologies for Healthcare, pp. 121–126 (2008)
27. Messig, M., Goscinski, A.: Autonomic system management in mobile grid environments. In: The fifth Australasian symposium on ACSW frontiers, pp. 49–58 (2007)
28. Huang, Y., Mohapatra, S., Venkatasubramanian, N.: An energy- efficient middleware for supporting multimedia services in mobile grid environments. In: International Conference on Information Technology: Coding and Computing, vol. 2, pp. 220–225 (2005)
29. Rao, S., Kiran.Kasula, V.D., Bano, S.: Management Study of Layered Architecture to Incorporate Mobile Devices and Grid Computing. In: Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics, pp. 123–127 (2008)
30. Huang, Y., Venkatasubramanian, N., Wang, Z.: MAPGrid: A New Architecture for Empowering Mobile Data Placement in Grid Environments. In: Seventh IEEE International Symposium on Cluster Computing and the Grid 2007, pp. 725–730 (2007)
31. Guan, T., Zaluska, E., De Roure, D.: Extending Pervasive Devices with the Semantic Grid: A Service Infrastructure Approach. In: The Sixth IEEE International Conference on Computer and Information Technology, pp. 113–113 (2006)

Access Network Discovery and Selection in the Future Broadband Wireless Environment

Marius Corici, Jens Fiedler, Thomas Magedanz, and Dragos Vingarzan

Fraunhofer FOKUS Institute
Kaiserin Augusta Allee 31, 13589, Berlin, Germany
{marius-iulian.corici,jens.fiedler,thomas.magedanz,
dragos.vingarzan}@fokus.fraunhofer.de

Abstract. The Future Broadband Wireless environment is characterized by the co-existence of a multitude of wireless networks e.g. LTE, UMTS, WiMAX, WiFi etc. In order to be able to offer the best connectivity, according to the requirements of the user and to the preferences of the operator, a novel functionality was introduced in the network and in the mobile devices for access network discovery and selection. This paper introduces this functionality as standardized in the 3GPP Evolved Packet Core (EPC), highlighting its main concepts and technical scenarios. Further a set of novel optimizations are evaluated, followed by the description of the Fraunhofer FOKUS OpenEPC implementation.

Keywords: Access Network Discovery and Selection, Wireless Broadband Networks, 3GPP Evolved Packet Core, heterogeneous wireless access.

1 Introduction

With the deployment of multiple access networks of various technologies, the mobile communication system is evolving towards a dense wireless environment where multiple access networks overlap and complement each other in terms of bandwidth, transmission delay and operational costs.

On the other side, the mobile users demand an increasing and diverse amount of resources for their mobile applications (like video, conferencing, streaming) in addition to the classic voice communication. To fulfill the requirements of this evolution, in which the telecommunication environment is transformed into a data dominant one, the network operators have to deploy a novel, more cost efficient network infrastructure.

The current access network selection mechanism presumes that the user of the mobile device decides independently the access through which it communicates. However, the mobile device is not aware of the momentary context in the network, so this mechanism does not offer any guarantee that the selected access is able to sustain the communication at a satisfying operational cost. Also it relies on multiple scans of the wireless environment, power consuming operation performed by the mobile device for the discovery of the available accesses. As it is foreseen that a multitude of accesses will be available at the same location delivering wireless broadband services

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 70–83, 2010.

to a multitude of devices, a novel mechanism for network discovery and selection has to be deployed, considering the information available in the core network e.g. the momentary preferences of the operator, the momentary operational costs of the accesses etc.

In order to solve this issue, the NGMN Forum [1] indicated that the novel wireless network environment will include a mechanism to provide services cost-efficiently by balancing the performance requested by the mobile users and the resources available in the different access networks.

To fulfill the requirements of the operators for the heterogeneous wireless broadband environment, the 3rd Generation Partnership Project (3GPP) initiated the standardization process for the Evolved Packet Core (EPC) as an all-IP based multi-access core network which integrates both 3GPP i.e. LTE, LTE-A, UMTS, GSM and non-3GPP i.e. cdma2000, WiFi, WiMAX wireless technologies. In the EPC, the balancing mechanism recommended by the NGMN Forum is separated into intra-3GPP and with non-3GPP accesses load balancing.

From the perspective of the operator, the 3GPP accesses are seen as a single integrated access network, which implies that the discovery and selection procedures are supported by the inter- and intra-access technology mobility management. The non-3GPP accesses are seen as an extension of the resources offered by the 3GPP accesses, which makes the discovery and selection functionality a generic enabler loosely coupled with the existing architecture. In this case, the functionality limits its goal to offering operator policies to the mobile device that optimize its handover decision.

This paper presents the access network discovery and selection functionality in the 3GPP EPC. The challenges cover several possible enhancements for the ANDSF, like the rapid handovers, caused by Femto-Cells, or the missing links between the ANDSF and service platforms (e.g. for location based services) and the ANDSF and subscriber profiles for access-network pre-selection or dynamic discovery of access network in the area of a mobile terminal. Furthermore its novel concepts and possible solutions are presented according to the standardization direction in the 3GPP and to the NGMN Forum recommendations. The Fraunhofer FOKUS OpenEPC proof-of concept implementation of the functionality is described together with the already implemented enhancements.

The remainder of this paper is organized as follows: Section II provides an analysis of the current status for the access network discovery and selection in 3GPP. In section III a set of novel concepts which enhanced the functionality are presented followed in Section IV by the proof-of concept implementation of the FOKUS OpenEPC platform. A set of conclusions are provided in Section V.

2 Access Network Discovery and Selection in 3GPP

Recognizing the need for the integration into a converged wireless environment of 3GPP and non-3GPP access technologies, 3GPP initiated the standardization of the Evolved Packet Core [2], [3] as an all-IP architecture which is able to support access control, subscription based resource reservations, security and seamless mobility between the different access networks.

As depicted in Fig. 1, the functionality is supported by a set of gateways which enable the exchange of data traffic with the mobile devices, named User Endpoints (UEs). The gateways are managed by a central policy based control entity – the Policy and Charging Rules Function (PCRF) which makes the subscription based decisions on the access control and on the resources to be reserved for each data flow of the UE. Completing the architecture, a Subscription Profile Repository (SPR) maintains the information related to the user profile. As in the current stage of standardization the SPR is not clearly defined as functionality, during this paper, it will be associated with the Home Subscriber Service (HSS) – the user profile repository imported from the IP Multimedia Subsystem (IMS) present also in EPC for storage of authentication and authorization of the UE and of its location information in the 3GPP wireless environment.

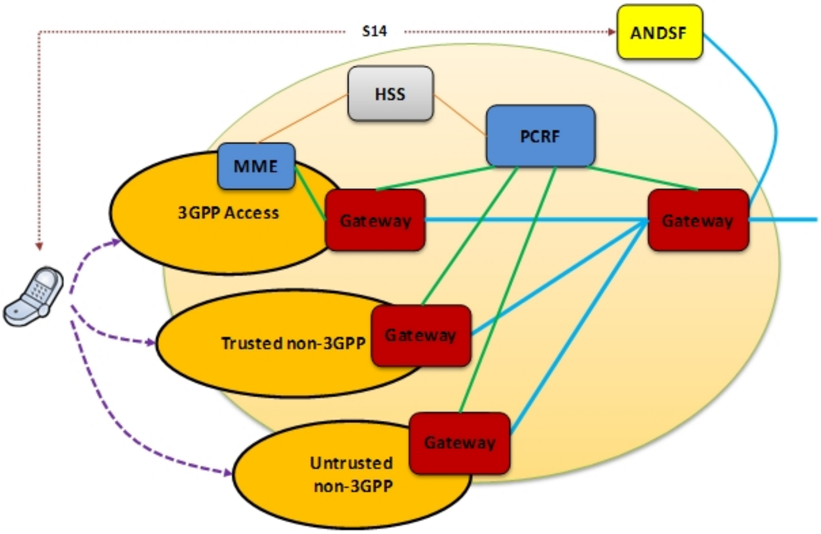


Fig. 1. EPC Simplified Architecture

For the mobility management inside the LTE and between the 3GPP access technologies, EPC contains a Mobility Management Entity (MME) which maintains local to the specified accesses the network discovery and selection. Due to the control of 3GPP on the standardization of the access technologies, the MME is able to select the target access cell and to prepare and command a handover of the UE.

For the interconnection with or between the other non-3GPP access technologies (e.g. WiMAX, WiFi etc.), a network discovery and selection functionality is introduced on top of the EPC architecture as an enabler. It presumes that two functional entities are deployed, one in the network – denominated as Access Network Discovery and Selection Function [4] and one in the UE – denominated in this document as Client Mobility Manager (CliMM).

The UE is able to receive discovery information and selection policies from the ANDSF using a logical interface. On this interface, OMA Device Management (DM) [5] protocol is deployed, which supports dynamic updating mechanisms, but it is not suitable for real-time communication. The OMA DM protocol was initially standardized as PULL mode protocol. At specific moments, the UE updates the device management information. As this did not provide enough dynamicity to the information refreshing, an out of band mechanism for triggering the fetching of data from the network was developed as a rudimentary PUSH mode. Currently it relies on OMA DM specific mechanism which use a trigger based on exterior services like SMS [6] which may not be suitable considering the new all-IP communication environment [4].

A Management Object (MO) [7] for the network discovery and selection functionality was specified by 3GPP. The MO describes the information exchanged between the UE and the ANDSF.

The UE transmits to the ANDSF the momentary location as geo-location or as information on the accesses to which it is currently connected to. For example, for a UE connected to an LTE access network, the location information may contain the operator domain, the Tracking Area Code and the cell identification.

The ANDSF responds to the UE with a set of policies separated for different physical areas which contain information on the operators deploying access networks, on the access networks (e.g. for WiFi the ESSID and the BSSID), a time interval when they are available and a prioritization between them. This information enables the UE to select the target access network restricted to a specific location and time interval and ordered by the operator preference.

The ANDSF maintains a Coverage Map database, which contains static information on the accesses available at specific locations. For example, a query to this repository with a specific location – geo-location or 3GPP cell-ID – will return a set of operators deploying WiMAX and WiFi accesses in the area and the Access Point information – NAP_ID, ESSID etc.

Although this solution provides the operators with a minimal mechanism for access network discovery and selection control, the information transmitted from the ANDSF to the UE does not state any information on the availability of the resources that are required for a seamless communication.

Also, the information is static; the UE does not have any guarantee that the access networks received are available in the area. For example, a WiFi access network may be available in the vicinity of the UE, but because of various external factors (interference, environmental conditions, operational failure etc.) it may not sustain the communication.

Due to the minimal coupling to the core network of the operator (the communication is executed on an application level protocol and the functionality does not interact with any other functionality deployed in the network), the access network discovery and selection can be deployed as a stand-alone 3rd party enabler which enables the mobile device to select the most appropriate access network and the services to be adapted to its environment.

3 Beyond 3GPP ANDSF

Although the functional goals of the ANDSF are already set, due to the incipient phase of standardization, several optimizations can be brought to the access network discovery and selection functionality. In this paper a set of these optimizations are analyzed and their suitability and possible integration into the 3GPP architecture are evaluated.

3.1 Subscription Profile Based ANDSF

As described in the previous section, currently the ANDSF is connected only to the mobile device, as the recipient of the access network discovery and selection decisions. In this context, the ANDSF has no interaction with the subscription profile which is already present in the network core for the access control and resource reservation procedures.

A first optimization proposed by this paper is to introduce a trigger into the ANDSF with the subscriber profile as a decision parameter. Using this enhancement, when a request for access discovery and selection policies is received, the ANDSF is able to select from the access networks which are located in the vicinity of the mobile device only those to which the UE can and is allowed to connect to.

For example, if a request is received from a mobile device which is not able to connect to WiMAX access networks because it does not include the required device interface or it is not allowed from the agreement with the network operator, then these accesses can be safely removed from the further ANDSF decision and thus from the information transmitted to the mobile device.

Also if the subscription profile is modified, due to an administrative change, then a notification may be transmitted to the ANDSF which in turn generates new policies and pushes them to the mobile device.

This optimization allows the ANDSF to reduce its processing for individual mobile devices to the set of access networks to which they can connect to and also the information exchanged over the network.

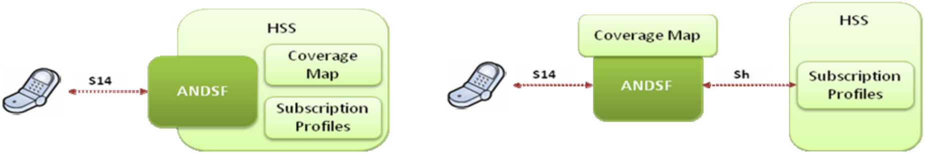


Fig. 2. Subscription Profile Architectures - ANDSF integrated into HSS (*left*) and ANDSF interfaced with HSS (*right*)

As depicted in Fig. 2, from an architectural perspective, this optimization can be implemented by using two mechanisms:

- ANDSF is integrated into the subscription repository
- A new interface between the ANDSF and the subscription repository

ANDSF already stores the Coverage Map containing the access networks which are available at specific locations. From this perspective, the repository can be integrated with the subscription profile repository, which using our convention is the HSS. In this case, the ANDSF becomes a communication entity of the HSS which is able to respond to the requests of the mobile device.

The ANDSF-HSS integration allows faster responses to the UEs as there is no other communication necessary. However, in this case the HSS has to be extended with more complex decisions related to the UEs by matching the information of the Coverage Map with the one of the subscription profile and the ability to push information to the mobile devices.

Also the direct connection between the operator storage and the mobile device is not advisable due to security. Various attacks can be performed on the storage itself if the UEs can transmit queries directly to the HSS.

In order to circumvent these disadvantages, the same functionality can be obtained by maintaining the ANDSF as a separate entity in the core network and introducing a new interface between it and the HSS. A similar interface is already standardized: the Sh interface between the IMS Application Servers (ASs) and the HSS [11]. The Sh interface allows for the same functionality as the interface here proposed: subscription profile fetching upon request and its modification notifications. In order not to modify the HSS side of the communication and to introduce only the ANDSF endpoint, it is beneficial if the same reference point would be deployed.

Compared with the integration of the ANDSF in the HSS, the novel interface has the advantage of isolating the database from the direct communication with the mobile device. Also it allows that the access discovery and selection functionality to be maintained as a separate enabler, who may be further integrated with other functionality of the core network to provide other optimizations.

3.2 Dynamic Discovery

The ANDSF maintains static information on the access networks available in specific areas. Currently, it is introduced through administrative means and due to the external factors (e.g. the control on the availability and mobility of a WiFi access point, the weather conditions etc.) their availability in the exact location of the UE may not be determined.

In this case, even if the UE receives the discovery and selection criteria for a specific area, it has first to discover that the accesses are available. This presumes that a scan of the wireless environment is to be performed. Only after determining the presence of the access network, it is possible for it to select the one which offers the best service continuity.

So, a handover is executed in two steps, one in which the access network is discovered and one in which it is selected as target access network, which increases the delay of the handover procedures and the power consumption of the device.

In order to reduce this delay and to make the ANDSF decisions more accurate and also to be able to introduce dynamically information in the Coverage Map, a novel Dynamic Discovery Enabler independent of the selection one may be considered. It presumes two different operations:

- Dynamic information is introduced in the Coverage Map
- The dynamic discovery information is transmitted to the UE.

In order to introduce dynamic information in the Coverage Map, the UEs should be able to scan the wireless environment upon request and based on the criteria received from the ANDSF. For example, if the ANDSF requires receiving information on the

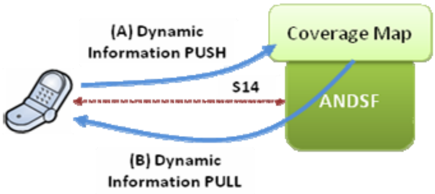


Fig. 3. Dynamic Discovery Procedures: (A) PUSH of information from the UE to the Coverage Map, (B) PULL of information to the UE through the ANDSF decisions

WiFi access networks available at a specific location, it transmits to the UE a scan request for the WiFi frequencies. The UE powers-up the WiFi device interface and scans the environment. The list of the discovered access networks together with the momentary location of the UE are transmitted to the ANDSF which in its turn introduces the information in the Coverage Map.

This information allows the ANDSF to have a degree of certitude on the momentary availability of the access networks at the specific locations. Using this procedure the ANDSF is enabled to make decisions having different levels of probability that the various access networks are available. The level of probability should be transmitted also to the mobile devices requesting discovery and selection information. When the probability that one access network is available in a specific location, it enables the UE to directly select the access network, without passing through the discovery phase.

Using the Dynamic Discovery Enabler, the ANDSF does not require anymore that the Coverage Map information is introduced through administrative means. It can receive information directly from the various UEs and to compute a momentary level of probability that the access network is available for other or the same mobile devices.

From this perspective, the ANDSF can be deployed as a third party enabler, completely decoupled from an operator network. It can maintain in its Coverage Map information on all the access networks available at specific locations, from more than one operator. The service provided by ANDSF to the mobile devices consists of offering dynamic discovery information, independent of operators, which may be beneficial to the UEs especially in roaming cases in which the UE has to choose a different operator than the one to which it has an agreement to in order to sustain its communication.

3.3 Location Enabler

ANDSF maintains the location of the UE and the access network to which it is currently connected to as main parameter of the discovery and selection decision. Several ambient aware applications may use this information in order to transmit context aware information to the mobile devices.

Currently in the EPC, there is no interface to transmit ambient information from the core network to the service platforms, thus the services have to use other exterior means to determine the location of the mobile devices. These mechanisms presume that an application on the mobile device determines and transmits the location directly to the service platform. The two operations, the one executed for the service platform and the one executed for the access network discovery and selection enabler are similar.

In order to reduce this redundancy, the ANDSF can expose the location information to the service platforms through a Location Enabler (Fig. 4). Using this information, the applications use not only the service profile of the user, but also its momentary vicinity. The usage of the ANDSF as a central location enabler for the various platforms reduces the communication over the wireless link – as only once the location of the mobile device is determined, compared to the state of the art solution in which all the services determine independently the location.

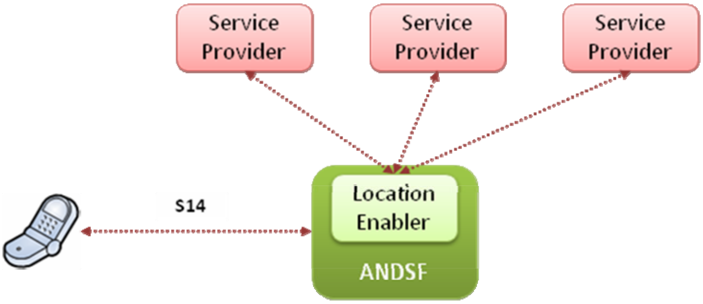


Fig. 4. Novel Location Enabler Functionality for the ANDSF

In order to be able to deploy the location enabler, a novel interface has to be considered between the ANDSF and the service providers. Currently only one similar interface is considered in the EPC – the Rx interface which allows the services to notify the PCRF on the resources that have to be reserved for specific data flows [10]. It also allows the PCRF to notify the services on specific events related to the data flows e.g. loss of communication etc. The same high level interface may be deployed between the ANDSF and the Service Providers. As the location enabler has a different functionality then the PCRF, a further evaluation of the operations and of the information exchanged is to be further considered.

3.4 Femto Cells Discovery and Selection

With the foreseen deployment of 3GPP accesses femto-cells the wireless environment is highly modified [9]. A femto is located at the user premises, has only a small coverage and is able to offer the same amount of resources as an operator controlled base station.

From the network discovery and selection perspective, they are introduced in the network as on top extensions of the existing infrastructure having the same advantages and limitations as the non-3GPP accesses: increased throughput at specific locations, parallel access to the existing wide coverage networks, reduced assurance of availability and limited coverage which translates into fast loss of signal during mobility. Also only a limited number of subscribers are allowed to use a specific femto cell, which if maintained as part of the 3GPP mobility management would make the MME decisions more complicated (e.g. the list of accepted users to the femto and the information on which femto cells the user is allowed to use).

Because of this, the ANDSF is a better candidate for Femto Cell Discovery and Selection. As seen for the non-3GPP accesses, there is no need for a complete mobility management scheme for the wireless accesses which come as extensions to the wide are wireless infrastructure and the ANDSF already provides the functionality for integrating non-3GPP hotspots and enterprise accesses.

For this only minor changes are to be introduced in the architecture. The femtocells are considered as a novel type of access networks with similar characteristics as the WiFi access networks. The discovery and selection information send by the ANDSF to the UE will include this new category.

From the perspective of the handover procedures, they can be executed as in the current state of the art for the handovers with non-3GPP accesses ensuring the same seamless quality.

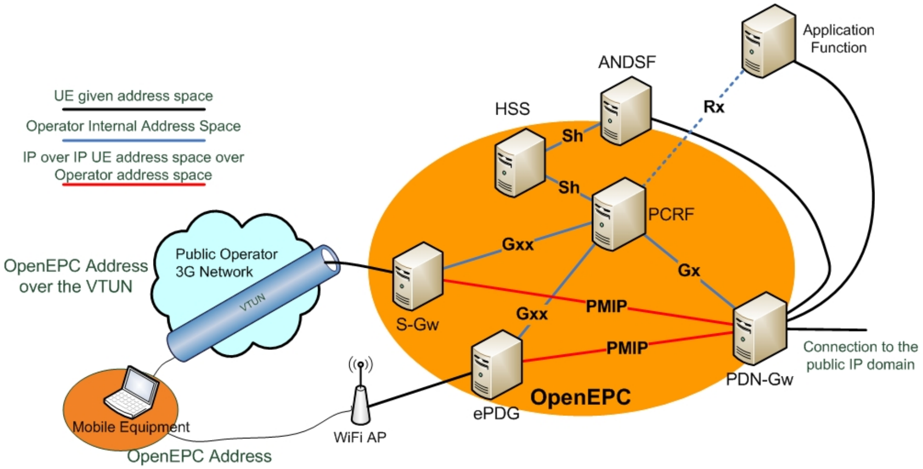

UE given address space
Operator Internal Address Space
IP over IP UE address space over Operator address space
OpenEPC Address over the VTUN
Public Operator 3G Network
VTUN
Mobile Equipment
OpenEPC Address
WiFi AP
HSS
ANDSF
Application Function
Sh
Sh
Rx
PCRF
Gxx
Gx
S-Gw
Gxx
PMIP
PMIP
ePDG
OpenEPC
PDN-Gw
Connection to the public IP domain

Fig. 5. OpenEPC Testbed

4 OpenEPC ANDSF Realizations

Fraunhofer FOKUS OpenEPC [8] platform implements a set of components according to the 3GPP EPC specification. By its highly modularized structure, it enables easy integration of access technologies and wireless broadband applications. Also, fast innovation in the challenging areas of mobile broadband networks research like new approaches to mobility, QoS, security and optimizations of the architecture are addressed.

As depicted in Fig. 5 currently it is able to provide subscription based access control, resource reservation and seamless mobility between UMTS and WiFi using as support IPv4, IPv6 or a mixture of the both. Other access technologies will be shortly integrated i.e. LTE, femto and fixed access.

OpenEPC implements the 3GPP standard for the access network discovery and selection functionality several own additions, including the connection with the subscription profile described in Subsection 3. As depicted in Fig. 6, it contains a CliMM in the UE, a separate ANDSF entity in the network which is connected to the HSS representing the repository maintaining the user profiles.

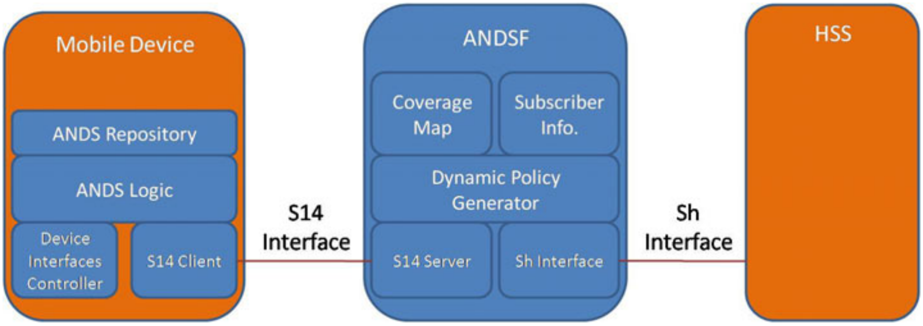


Fig. 6. OpenEPC Access Network Discovery and Selection Realization

The CliMM integrates a client for the interface to the ANDSF named S14 in the 3GPP specifications. Through this, it is able to communicate with the ANDSF in either PUSH or PULL mode. The functionality in the mobile device is separated into ANDS Logic which makes the discovery and selection decisions and an ANDS Repository storing the policies received from the network. The result of the decisions taken by the ANDS Logic is enforced on a Device Interface Controller which enables attachment and detachment from specific access networks and the forwarding of the data traffic accordingly.

The simple structure of CliMM and its complete independence on the applications enables its easy deployment on various mobile platforms. Currently Linux and Windows operating systems are sustained, shortly followed by Android, Symbian and other.

The OpenEPC ANDSF maintains Coverage Map and subscriber information repositories. The Coverage Map is indexed by areas, enabling a fast determination of the access networks that may be located in the vicinity of the mobile device. ANDSF maintains subscription information for the mobile devices which are registered through the S14 server. This way, there is no need for further queries on the profile, enabling a fast response customized on the capabilities of the mobile device and on its access rights e.g. the access networks to which a mobile device is capable and allowed to connect to.

The interface introduced in Subsection 3.1 between the ANDSF and the HSS was implemented, based on the FOKUS OpenIMS Core Sh interface. It allows the ANDSF to fetch subscription information for newly attached UEs and to receive notifications in case it is modified.

A central Dynamic Policy Generator makes decisions on the information which is to be then transmitted to the mobile device in either PUSH or PULL mode over the S14 interface. Compared to the specifications, the decisions are made not only based on the location of the mobile device, but also dynamically customized based on the subscription profile. Through this mechanism, the redundancy of the discovery and selection information is reduced, thus reducing the processing power consumed by the mobile devices. For example, if a mobile device is not allowed to connect to WiMAX in a specific area or it is not capable of connected to LTE accesses, the correspondent information is eliminated from the ANDSF decision and thus from the one transmitted to the mobile device.

For demonstrating the proof of concepts of the access network discovery and selection, the OpenEPC discovery and selection functionality is capable not only on the transmission of indications to the UE on the different access networks which can be considered in case it decides a handover, but it is also capable of transmitting handover commands. This enables a fast adaptation of the UE to the modifications from the subscription profile than in the real deployment scenarios. For example, if the subscription profile is modified and the UE is not allowed anymore to connect to the WiMAX access to which it is already connected to, then a handover command is transmitted from the ANDSF on which the UE makes the decision to select another access. Because of this feature, the OpenEPC ANDSF is able to present in real-time its integration with the subscription profile information compared to the real deployment scenarios in which the UE executes the handover procedures only when they are triggered by other mobile device internal mechanisms like loss of signal in a specific access network.

4.1 OpenEPC ANDSF - Evaluation Scenario

For showing the capabilities of the OpenEPC in the area of access network discovery and selection multiple testing scenarios had been implemented, from which the following was selected as the most representative.

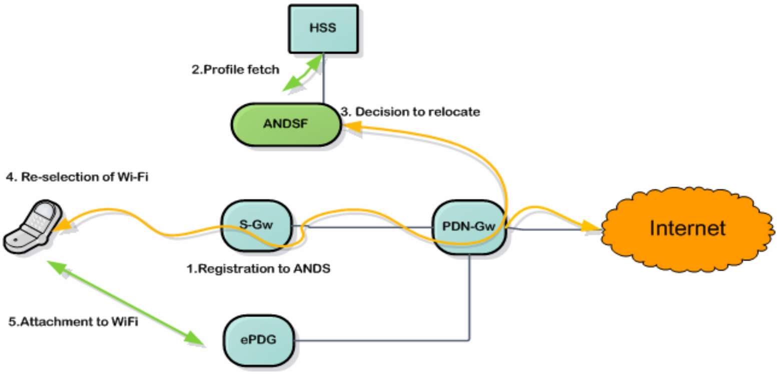


Fig. 7. Subscription Based Access Network Discovery and Selection

It is presented here as validation proof of the ANDSF functionality of the OpenEPC. As the goal of the OpenEPC is to provide a platform open for innovation and as the access network discovery and selection functionality is not very time constraint in the 3GPP EPC, the experimental measurements do not truthfully provide a view on the efficiency of the functionality, thus they were not included in this paper.

The OpenEPC operator can restrict the access of a UE to specific access networks by modifying the subscription profile. For brevity, the scenario selected here presents an initial connectivity case, although it was tested also during the active connectivity of the mobile device. As exemplified in Fig. 7, the mobile device initially attaches to the UMTS network.

After the initial attachment, it connects over the S14 interface to the ANDSF and requests the default discovery and selection policies for a newly attached UE. The ANDSF fetches the subscription profile from the HSS, which was modified during the inactive time of the UE, as to restrict it to connect to the UMTS networks. Then, the ANDSF makes a policy decision in order to find another suitable access network to which the UE can handover to. It finds that a WiFi access is available in the vicinity of the mobile device and it sends a response to the UE containing a policy which indicates that a handover to the WiFi access is required. As to shorten the duration of the demonstration of the scenario, the policy also indicates that an immediate handover is required. The UE executes the handover to the WiFi access network.

The modification of the subscription profile can happen also during the service of the mobile device. In this case, the ANDSF receives a notification from the HSS containing the modifications of the subscription profile and upon this trigger it makes the decision that new access discovery and selection policies have to be transmitted to the UE. The ANDSF alerts the UE, which triggers an immediate policy fetching from the ANDSF to the UE. Using the new policies, the UE executes the handover procedures as in the previous initial connection case.

The access network discovery and selection functionality may be triggered also by the loss of signal to the access network to which the UE is connected to. In this case, the UE either uses the policies of handover as they were previously received from the ANDSF or requires new policies.

It is to be noted that in OpenEPC the policy transmission to the mobile device can be either synchronous with the handover trigger or asynchronous based on the location change of the UE. In the second case, when the network or the UE notices a change in location new policies are either pushed or pulled to the UE.

This scenario uses the non-standard interface between the HSS and the ANDSF as described in the previous sections. It is used to bring to the ANDSF from the subscription profile the information on which access networks the UE is momentarily allowed to connect to.

As the communication for access network discovery and selection was done as a background communication to the active applications of the mobile device and as the resources consumed by this communication can be easily supplied by the current and future access networks to all the mobile devices, no impediment was seen on the communication of the mobile device due to this novel enabler.

This scenario was tested with and without having a real-time video application established between the UE and a server providing the service. Due to the underlying mobility protocol deployed – Proxy Mobile IP, the same IP address was allocated to the two device interfaces of the UE which made the service to be seamless to the user. This scenario opens the possibility for the operators to deploy seamless services across the multiple access technologies without having to extend the mobility related functionality in the mobile devices and without any modification of the service platforms. Using the minimal CliMM application on the mobile device, the access network discovery and selection functionality can be easily deployed to real operator scenarios.

5 Summary and Conclusions

This paper presented the access network discovery and selection functionality in the 3GPP EPC including the general procedures and protocols already defined between the network-located function and the mobile device. From the perspective of 3GPP, the functionality is seen as an over the top enabler. The overall system can function without its deployment, but it provides its benefits when it is present. This allows a fast integration into the existing architecture, without requiring the modification of the other components.

Maintaining this general scope, a set of issues and potential improvements were further analyzed on how it can be extended without modifying the already existing functionality. Several options on how they can be integrated in the architecture were presented according to the directions of 3GPP standardization. These possibilities open new areas of research and optimization which may be further considered according to the different real operator deployments.

Furthermore, this paper presented the realization of the new concepts in the Fraunhofer FOKUS OpenEPC platform and the validation scenarios which were developed.

As proven by this article, the EPC access network discovery and selection functionality, by its clear separation from the other functions of the network core, can be further developed as a dynamic discovery or location enabler. Also its functionality may be extended with a correlation with the subscription profiles maintained by the operator and with femto-cell integration through which a better service is provided to the mobile devices. A further evaluation of these concepts and afferent architectures is required on a per-deployment basis in order to obtain an efficient operation of the EPC.

References

1. NGMN Forum, NGMN White Paper on NGMN beyond HSPA & EVDO, http://www.ngmn.org
2. 3GPP TS 23.401 General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (December 2009), http://www.3gpp.org version 9.3.0
3. 3GPP TS 23.402 Architecture enhancements for non-3GPP accesses (December 2009), http://www.3gpp.org version 9.3.0
4. 3GPP TS 24.302 Access to the Evolved Packet Core (EPC) via non-3GPP access networks version 9.1.1 (December 2009), http://www.3gpp.org
5. Open Mobile Alliance, OMA Device Management Protocol, version 1.2.1(June 2008)
6. Open Mobile Alliance, OMA Device Management Notification Initiated Session version 1.2.1 (July 2008), http://www.openmobilealliance.org
7. 3GPP TS 24.312 Access Network Discovery and Selection Function (ANDSF) Management Object (MO) version 9.0.0 (December 2009), http://www.3gpp.org
8. Fraunhofer FOKUS OpenEPC, http://www.openepc.net

9. 3GPP TS 22.220 Service requirements for Home NodeBs and Home eNodeBs version 10.1.0 (December 2009), http://www.3gpp.org
10. 3GPP TS 23.203 Policy and Charging Control Architecture version 9.1.0 (December 2009), http://www.3gpp.org
11. 3GPP TS 29.328 IP Multimedia Subsystem (IMS) Sh interface; Signalling flows and message contents version 9.0.0 (December 2009), http://www.3gpp.org
12. 3GPP TS 23.008 Organization of Subscriber Data version 9.1.0 (December 2009), http://www.3gpp.org

Internet Connectivity Sharing in Multi-path Spontaneous Networks: Comparing and Integrating Network- and Application-Layer Approaches

Paolo Bellavista and Carlo Giannelli

Dip. Elettronica Informatica e Sistemistica, University of Bologna
Viale Risorgimento 2, 40136 Bologna, Italy
{paolo.bellavista,carlo.giannelli}@unibo.it

Abstract. Spontaneous networking, where wireless mobile nodes opportunistically exploit multi-hop ad-hoc paths toward peers to share content and available resources in an impromptu way, has recently received growing interest from both industry and academia. In this paper, we specifically focus on the notable case of sharing connectivity to the traditional Internet, with the general goal of an overall better exploitation of connectivity resources, often underutilized as the population of wireless devices grows, as well as their local computing/memory/bandwidth resources. In particular, here we show how our novel middleware, called RAMP, can exploit both network- and application-layer solutions to dynamically manage mission-oriented paths toward peers offering Internet connectivity. Thanks to our middleware-level cross-layer approach, RAMP can dynamically select and combine different solutions for multi-hop multi-path ad-hoc path formation and can take proper management decisions based on run-time context. The reported results demonstrate the suitability of dynamically integrating network- and application-layer approaches to achieve the best overhead/performance tradeoff depending on specific application requirements.

Keywords: Internet Connectivity, Spontaneous and Collaborative Networks, Middleware, Heterogeneous Wireless Networks, Multi-hop Multi-path Connectivity.

1 Introduction

In the last couple of years spontaneous networking has received growing and growing attention for its promising aspects of better exploitation of available wireless connectivity, resource connectivity sharing, and immediate connectivity offer in regions with difficult coverage [1, 2]. Notwithstanding first interesting research results have been achieved [3-6], several technical challenges are still open, such as the concurrent and effective exploitation of heterogeneous wireless technologies (multi-hop paths made up by multiple heterogeneous links), of multiple wireless technologies/cards at the same node (different heterogeneous paths traversing a single node), and of the combination of single-hop infrastructure-based links and ad-hoc ones.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 84–99, 2010.

Anyway, it starts to be widely recognized not only the relevant potential of spontaneous networks for better exploitation of resources in collaborative smart environments of the future, but also that the complexity of spontaneous network management makes it inadequate to handle it directly in the supported collaborative applications. We claim the need for novel middleware capable of simplifying the development of applications on top of spontaneous networks, by properly and effectively managing the complexity associated with multi-hop multi-path heterogeneous connectivity, with no need of complete, global, and strictly updated knowledge about the dynamic topology and characteristics of the exploited paths. To this purpose, we have developed an innovative middleware, called Real Ad-hoc Multi-hop Peer-to-peer (RAMP), which transparently manages the technical challenges related to i) global decisions based on limited local visibility, ii) erratic behavior of mobile peers sharing resources in an impromptu way, and iii) IP addressing in spontaneous networks [6].

In this paper, we specifically focus on a notable case of collaborative resource sharing, i.e., sharing bandwidth and connectivity toward the traditional Internet. The rationale is that many portable devices are nowadays equipped with multiple wireless interfaces and flat-rate/large-bandwidth subscription for Internet connectivity: their potentially available bandwidth is often underutilized, while it could be shared with other peers in current vicinity, thus better exploiting the growing availability of computing/memory/bandwidth resources at portable wireless terminals. In particular, this paper shows how RAMP can exploit both network- and application-layer approaches to dynamically handle mission-oriented, multi-hop, heterogeneous, and sporadic paths toward peers that offer a portion of their connectivity bandwidth to the Internet. We claim that the creation and management of these spontaneous intermittent paths at the network layer (L3 approach) can achieve good performance and limited overhead in the case of relatively stable and short paths, but at the expense of minor flexibility. Instead, application-layer solutions (L7 approach) can achieve relevantly better flexibility, e.g., by enabling the exploitation of different multi-hop heterogeneous paths traversing the same node, at the expense of a relatively greater overhead. Our solution guideline is that in spontaneous networks it is suitable to take path management decisions (either L3 or L7 or a combination of them) only at runtime and based on currently applicable context.

According to this principle, we have extended the RAMP prototype to support Internet connectivity sharing in spontaneous networks. In particular, three middleware components have been added: *InternetClient*, active on nodes requesting Internet connectivity to their peers, *InternetService*, running at Border Nodes (BNs, i.e., nodes directly connected to the traditional Internet and offering part of their underutilized connectivity), and *Layer3Manager*, active on peers that allow RAMP to modify local routing rules working at the operating system level. The RAMP extension originally presented in this paper exploits L3 and L7 approaches to support three different modes for Internet connectivity sharing in spontaneous networks: i) a low-overhead L3 Single-Path (L3SP) solution, ii) a highly flexible L7 Multi-Path (L7MP) one, and iii) a hybrid L3L7-Combo Multi Path (L3L7CMP) one. Potentially available paths are created and selected based on runtime context by comparing end-to-end path performance, estimated dynamically in a very lightweight way. In addition, RAMP enables even the same application instance to exploit different approaches/paths simultaneously (for different connection requests); in other words, approach/path management is performed dynamically with per-connection granularity.

The RAMP prototype is available for download as a useful tool for the community of researchers in the field and can be easily deployed over real environments with standard wireless cards and execution platforms. The reported results demonstrate the suitability of dynamically integrating network- and application-layer approaches to achieve the proper overhead/performance tradeoff at runtime. In particular, RAMP has demonstrated to be able to effectively exploit dynamically available BNs depending on their provided bandwidth, estimated in a very lightweight way at service provisioning time. Moreover, the additional overhead imposed by L7 has demonstrated to be limited, largely counterbalanced by increased connectivity reliability and throughput thanks to the simultaneous exploitation of multiple paths.

The rest of the paper is organized as follows. Section 2 summarizes the pros and cons associated with network/application-layer approaches to Internet connectivity sharing in spontaneous networks, while Section 3 details the different modes that RAMP enables. Section 4 goes into the technical details of the RAMP architecture and of some notable implementation insights. Experimental results demonstrating the suitability of combining network/application-layer approaches in RAMP are in Section 5, while related work, conclusive remarks and on-going research end the paper.

2 Internet Connectivity Sharing in Multi-hop Multi-path Spontaneous Networks

To better point out the challenging environments targeted by RAMP and to highlight the differences between network- and application-layer Internet connectivity sharing, let us rapidly sketch a practical example of multi-path spontaneous network. Consider the realistic case of a group of students in a lecture hall carrying on mobile clients equipped with multiple heterogeneous interfaces (see Figure 1), e.g., laptops with IEEE 802.11 and Bluetooth, cell phones with UMTS and Bluetooth, and smart phones with UMTS, IEEE 802.11, and Bluetooth. Some of the nodes (the BNs) get direct connectivity to the Internet, by taking advantage of their flat-rate UMTS subscription or by connecting to a free-of-charge IEEE 802.11 access point of the university campus, e.g., NodeA and NodeD. BNs can share connectivity via subgroups created in an impromptu way, by exploiting their local wireless interfaces to get and offer single-hop connectivity, even participating to multiple subnets simultaneously.

The wide variety of exploited interfaces strongly pushes for the adoption of standard IP as the common layer (as a useful and practical simplifying assumption). In addition, this choice enables solutions that can be practically deployed over already existing networks, thus promoting easy deployability and potentially rapid market penetration. In such a scenario, the sharing of Internet connectivity can be realized either through more traditional L3 path formation solutions or by supporting inter-node packet dispatching at the application layer (L7 store/carry/forward techniques).

In general, as a preliminary and introductory overview, L3 solutions aim at configuring peer nodes with proper settings for their default gateways, by using the traditional mechanisms designed and implemented for the wired Internet, to create multi-hop paths toward BNs. Once an L3 multi-hop path has been configured, nodes residing on that path can get Internet connectivity. For instance, in Figure 1, once NodeG, NodeF, and NodeB have set their default gateway, NodeG can use the

NodeF-NodeB-NodeA L3 path. Note that each single node can exploit only one L3 path, even if alternative multiple paths are potentially available, e.g., NodeF-NodeE-NodeD.

On the other hand, L7 approaches can enable Internet connectivity by dispatching packets among cooperative nodes at a higher layer of solution, without interacting with operating system-level routing rules. In this case, with packet routing performed at the application layer, packet delivery does not depend on default gateway configuration and each node can use the L7 multi-hop path currently deemed as the most suitable, e.g., because it provides largest bandwidth (less loaded path) or requires lowest power consumption (local exploitation of Bluetooth interface). For instance, NodeF can access the Internet via the NodeB-NodeA path, while nodeG via the NodeF-NodeE-NodeD one. In addition, the same node can exploit different L7 paths for different connections simultaneously, e.g., in order to maximize the total throughput. For instance, NodeF can exploit the NodeB-NodeA path to download a given Web page and the NodeE-NodeD one to connect to an FTP server. In other words, sharing Internet connectivity at L7 enables the simultaneous adoption of different overlay networks, e.g., different nodes in the same subnet may access the Internet in a different way. Furthermore, the same node can exploit different overlay networks at the same time. Moreover, the adoption of an L7 solution does not prevent from the capability of exploiting facilities available at L3: it is possible to exploit both L3 and L7 solutions simultaneously, e.g., the same NodeF accesses the Internet via NodeA by exploiting an L3 path and via NodeD by adopting an L7 path.

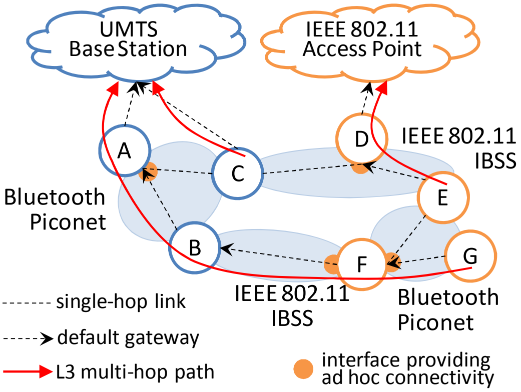


Fig. 1. Multi-path spontaneous network scenario

Considered the above scenario, we claim the suitability of adopting the following solution guidelines to share Internet connectivity in multi-hop multi-path heterogeneous spontaneous networks:

1) **supporting both L3 and L7 approaches.** Since they provide different technical pros and cons, as better detailed in the following section, a middleware support should enable both and dynamically select the most suitable one depending on runtime context (characteristics of the deployment environment, application

requirements, most suitable overhead/performance tradeoff, …). In general, L3 approaches tend to impose little routing overhead after the first initialization phase, but require careful management because path formation at a node may impact on the possible selections at other nodes. L7 approaches, instead, generally impose higher overhead, but enable multi-path Internet access;

2) **context-aware estimation of available paths.** The availability of multiple BNs can improve performance, but calls for suitable metrics to dynamically evaluate which is the most suitable path to be enabled for a given client node and for a given application connection. Metrics should provide quantitative estimations of path quality, at the same time with minimum impact on overhead, e.g., by avoiding frequent dissemination of monitoring information between nodes;
3) **differentiated metrics at session initialization and at service provisioning time.** The path evaluation process should be different at service initialization and provisioning time. In the former case it is appropriate to exploit rather static context data, easily retrievable before actual connections are established, e.g., estimated bandwidth based on path hops number, thus providing a coarse-grained but lightweight estimation of available paths. In the latter case it is adequate to adopt finer-grained metrics, by exploiting the visibility at zero cost of the actual path performance that nodes are currently experiencing at runtime.

3 RAMP for Internet Connectivity Sharing

According to the above solution guidelines, we have extended our RAMP middleware to support Internet connectivity in spontaneous networks exploiting both L3 paths, managed by exploiting routing configuration mechanisms and tools at the operating system level, and L7 paths, managed by exploiting application-layer middleware components to dispatch packets to collaborative nodes. As already stated, three middleware components have been added: *InternetClient*, active on client nodes requesting Internet connectivity, *InternetService*, running at BNs, and *Layer3Manager*, active on peers participating to L3 paths. To take advantage of proper dynamic selection based on currently applicable context, RAMP enables three different modes for Internet connectivity sharing: L3SP, L7MP, and L3L7CMP, as detailed in the following.

3.1 Multiple and Combined Layer Modes in RAMP

L3SP is based on the dynamic configuration of standard routing rules at the operating system level on intermediate nodes in order to create the needed L3 path from the client to a suitable BN currently offering Internet connectivity. In this case, the RAMP middleware transparently works to create the L3 path by modifying the default gateway configuration on any node along the path. For instance, in Figure 2, to access the Internet via BN_1, NodeC, NodeY, and NodeX must specify respectively NodeY, NodeX, and BN_1 as their default gateway. To this purpose, any node along the path has to collaborate to packet forwarding and to offer the possibility of modifying its local routing rules dynamically. By delving into finer details, the RAMP client exploits the InternetClient component to send an L3 path configuration request to the nodes along the path (dynamically determined in an innovative and effective way by

RAMP [6]), while Layer3Manager components on any intermediate node modify local routing rules. Once every node has enforced the required routing rule modifications, any application at the client node can access the Internet directly via the L3 multi-hop path towards BN, with no additional need for runtime support by InternetClient or Layer3Manager, thus imposing minimum overhead. However, all applications at a node can exploit only one path to the Internet, even in the case of multiple BN availability. Let us note that, due to the relatively high costs of path formation via routing configuration, L3 paths should be created only when necessary (reactive approach in response to client connectivity requests, no proactivity) and by adopting a lightweight coordinated view among neighbors to avoid clashing routing requests.

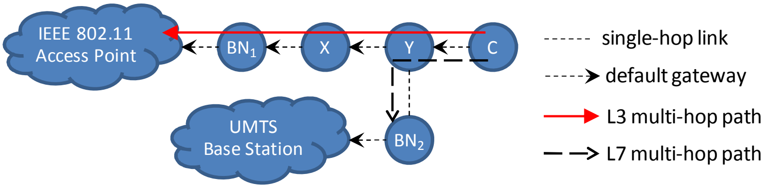


Fig. 2. L3 and L7 paths

L7MP is based on the exploitation of RAMP-supported L7 paths, with no need of any modification of underlying routing rules because packet forwarding is performed hop-by-hop at the application level. As a consequence, applications can exploit multiple BNs even simultaneously (additional implementation details about the RAMP support for L7 paths are in the following section). The L7MP mode adopts a double-proxy architecture: an InternetClient proxy on any node requesting Internet connectivity and an InternetService proxy at any BNs sharing Internet connectivity. InternetClients are in charge of receiving local application requests and of dispatching them to one of the previously discovered InternetServices. The selection of the most suitable InternetService is performed with per-connection granularity, based on currently applicable context (see the following). The InternetService receiving the client request performs the actual connection with the Internet end-point, waits for a response, and finally forwards the response to the origin InternetClient. Finally, InternetClient transparently forwards the response to the local application client. For instance, in the case of HTTP applications, InternetClient acts as a proxy receiving/sending HTTP requests/responses from/to the local Web browser, while InternetService contacts the remote Web server, performing the actual HTTP interaction on behalf of the browser.

Let us note that L7MP allows the simultaneous exploitation of multiple paths and multiple BNs by the same client node. On the one hand, this permits to potentially increase the overall achievable throughput, which is particularly important in the case of scenarios with single-hop links with limited bandwidth (quite common in spontaneous networking). On the other hand, this helps in achieving greater reliability because, in the case of disruption of a single path, on-going connections may benefit from being rapidly switched to other available paths. However, L7MP tends to impose an additional overhead due to application-layer routing, e.g., for packet data encapsulation into application-level RAMP packets.

L3L7CMP combines the exploitation of both L3 and L7 approaches. In this case, InternetClient is aware of the (possible) availability of one L3 path and multiple L7 paths; based on dynamically gathered context, it dynamically selects which is the most suitable choice for the specific connection request. When choosing an L7 path, InternetClient uses the double-proxy architecture presented for L7MP; instead, in the case of L3 path, it adopts a single-proxy approach and directly contacts the requested Internet end-point via the selected BN.

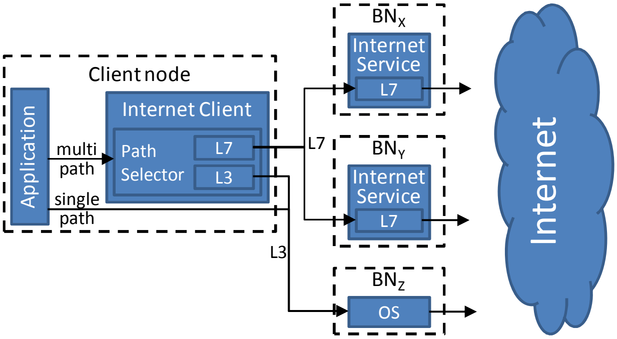


Fig. 3. Internet requests adopting proposed approaches

We have decided to support the different L3SP, L7MP, and L3L7CMP modes in RAMP because we claim that the most appropriate solution could be chosen only at runtime depending on application requirements and applicable context. There is not a single mode always preferable to the others: in fact, the three modes permit to achieve different tradeoffs in terms of path (re-)configuration costs, multi-path support, and communication capabilities/overhead, as better detailed in the following.

On the one hand, L3SP may be expensive in terms of management overhead because it requires i) routing rule modifications on intermediate nodes and ii) solving possible conflicts between routing requirements of different nodes. For instance, in Figure 2, if the previous default gateway of NodeY was BN_2, NodeC request could disrupt the active Internet connection due to default gateway change. In addition, an L3 path can be modified depending on successive requests of nodes partially sharing some peers: if, after a while, NodeY selects BN_2 as default gateway again, NodeC path to the Internet is modified even if NodeC has not asked for any change. Moreover, if nodes are highly mobile, L3 paths are ineffective because intermediary nodes can abruptly leave the network rather frequently, thus requiring repeated L3 configuration processes. Instead, L7MP does not require any path pre-configuration because the path to be exploited is specified anytime InternetClient sends a packet to an InternetService at a BN: packets encapsulating application requests are managed exactly as any RAMP packet. Moreover, RAMP supports advanced store&forward routing, permitting to correctly dispatch packets even in case of intermittent connectivity (additional details on the RAMP Web site [6]).

On the other hand, once a L3 path has been correctly configured, L3SP imposes minimum overhead at service provisioning time because it exploits operating system-level forwarding. Any Internet connection automatically exploits the L3 multi-hop path, despite the adopted application-layer protocol and with no need of any further InternetService/Client intervention; in other words, RAMP clients can access the Internet as if they were BNs. On the opposite, L7MP suffers from the additional overhead imposed by the exploitation of the double-proxy architecture and by the request/response encapsulation into RAMP packets. L3L7CMP can partially reduce overhead, since it can sometimes adopt a single-proxy approach avoiding data encapsulation; however, InternetClient currently supports only HTTP (and Pseudo-HTTP), as better detailed in the following; therefore, for instance, L7MP and L3L7CMP cannot provide access to an RTP-based stream server.

In short, RAMP users can exploit the desired mode and switch among modes dynamically. The L3SP mode well suits the case of relatively stable topologies and is the only one to use to enable application-level protocols not currently supported by InternetClient/Service. L7MP is more suitable for highly dynamic scenarios where there is the need to support increased connectivity reliability (multi-path plus store&forward RAMP features). L3L7CMP is a compromise between the two since it couples both L3 and L7 approaches, with slightly lower overhead than L7MP, but limited store&forward capabilities.

3.2 *PathLength* and *PathThroughput* Metrics

RAMP adopts the *PathLength* and *PathThroughput* metrics to evaluate which is the most suitable BN to create an L3 path to and, in the case of multi-path, to which BN an application request should be forwarded to (either via L3 or L7 paths). These metrics associate potentially available paths with quantitative weights in the [0, 1] range (greater the weight, more suitable the path), adopting a lightweight end-to-end perspective. At session initialization, InternetClient discovers the available InternetServices and assigns them weights according to *PathLength* (basically greatest weights to shortest paths):

$$w_i = \frac{1-(path_i Length / averageLength / \# paths)}{\# paths - 1}$$

where $\mathtt{path_iLength}$ is the number of $\mathtt{path_i}$ hops and `averageLength` is the average length of the `#paths` available paths. Shortest path priority pushes traversing traffic to a limited set of nodes. In addition, while we are aware that path throughput depends on a wide set of parameters, such as adopted wireless technologies and traffic load, based on our previous work we believe that path length can also provide a rough estimation of the maximum bandwidth achievable, useful to quickly take an initial configuration decision in a very lightweight way [7].

L3SP exploits these weights to evaluate the most suitable BN: InternetClients running in the same neighborhood have a homogeneous vision of available paths, thus supporting the formation of a path evaluated as suitable for the whole locality (see weights in Figure 4). In fact, *PathLength* tends to partition the network in different parts; in this way, it is scarcely probable that neighbors try to activate conflicting L3 paths. In the current RAMP implementation, in the case of multiple paths with the same weight, the user has to explicitly select the preferred one. Path reconfiguration is triggered only in case of connectivity disruption, to prevent from disturbing working connections of other nodes. L7MP and L3L7CMP exploit weights to decide how to proportionally partition the request load among the available paths: for instance,

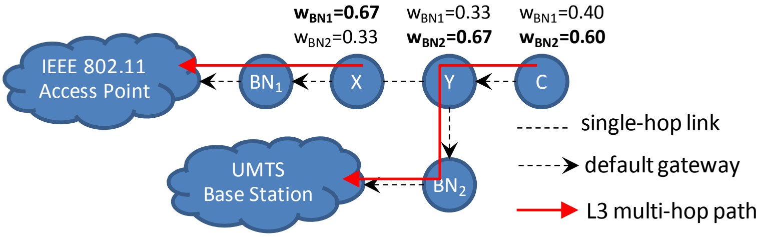


Fig. 4. PathLength metric application (selected BN in bold style)

considering NodeC in Figure 4, InternetClient computes $w_{BN1} = 0.4$ and $w_{BN2} = 0.6$; thus, every 5 application requests, it exploits BN_1 two times and BN_2 three times.

In addition, in L7MP and L3L7CMP modes, InternetClient monitors end-to-end throughput at service provisioning time and evaluates path quality via the *PathThroughput* metric. In particular, InternetClient keeps track of:

$$\frac{requestPayload + responsePayload}{elapsedTime}$$

values with per-connection granularity. Thus, it can achieve an approximated but lightweight estimation of path performance, with no additional communication overhead. Based on these values, InternetClient periodically (whenever it has gathered 20 throughput values for one of its paths) reassigns weights to paths by adopting the following *PathThroughput* metric:

$$w_i = path_i Throughput \,/\, averageThroughput \,/\, \#\, paths$$

where `path_iThroughput` is the throughput of path `i` in the last time window and `averageThroughput` the average throughput of the `#paths` available paths. For instance, if BN_1 and BN_2 offer 25 and 10KB/s throughput respectively, w_{BN1}/w_{BN2} is equal to 0.71/0.29; therefore, 71% of the connections will exploit the former path, 29% the latter. L7MP and L3L7CMP do not partition the network topology as L3SP does: any node can exploit all the visible BNs depending on its locally perceived connectivity quality. For instance, if BN_2 becomes overloaded, NodeY assigns a higher weight to (and thus exploits more frequently) BN_1, by leaving almost all BN_2 bandwidth to NodeC, which could evaluate BN_2 as the most suitable. Let us note that, as better detailed in Section 5, the overall achieved throughput may depend also on factors not strictly related to spontaneous network path performance, e.g., payload size and HTTP server load, and *PathThroughput* can achieve good performance estimation anyway, by adopting a very lightweight and completely distributed approach.

4 RAMP Internet Connectivity Service: Architecture and Implementation Insights

RAMP is designed according to a 2-layer architecture, with a higher Service Layer and a lower Core Layer (Figure 5-left). The former supports peer-to-peer service provisioning via registration, advertising, and discovery; the latter provides communication abstractions for end-to-end unicast and broadcast. In particular, Service Manager allows the registration and advertising of local applications (registerLocalService), while Discovery supports available remote services (findRemoteServices), by allowing the identification of the path toward the targeted node and the retrieval of its capabilities. E2EComm offers multi-hop unicast and TTL-bound broadcast primitives (receive, sendUnicast, and sendBroadcast); Dispatcher interacts with single-hop neighbors (e.g., via UDP/TCP depending on what dynamically specified) to collaboratively route packets; Heartbeater works to keep track of single-hop neighbors by periodically inquiring the available subnets via UDP broadcast.

In addition, RAMP exploits the mechanisms developed within the Multi-hop Multi-path Heterogeneous Connectivity (MMHC) project, already presented elsewhere [7], for the dynamic setting of ad-hoc subnets (layer-2 link creation and layer-3 network configuration). MMHC provides the best multi-hop Internet connectivity via proper local configuration by exploiting innovative context indicators (e.g., probability of joint peer mobility) to maximize connectivity reliability, throughput, and availability. In particular, RAMP takes advantage of MMHC to create/manage heterogeneous single-hop links and to identify nodes. Each subnet is created in a distributed way, without the need of global scope visibility, and by considering any available interface (e.g., IEEE 802.11 and Bluetooth); nodes assign IP addresses to the subnets they have created without any need of coordination [7]. The former MMHC rerouting mechanisms manage multi-hop paths only based on modifications of default gateway configuration, e.g., to use the wireless interface with minimum energy consumption or maximum throughput. Instead, RAMP more flexibly dispatches packets at the application layer by exploiting every available single-hop link enabled by the underlying MMHC, with the valuable additional advantage of simultaneous exploitation of any available path, despite operating system-level configuration of routing tables.

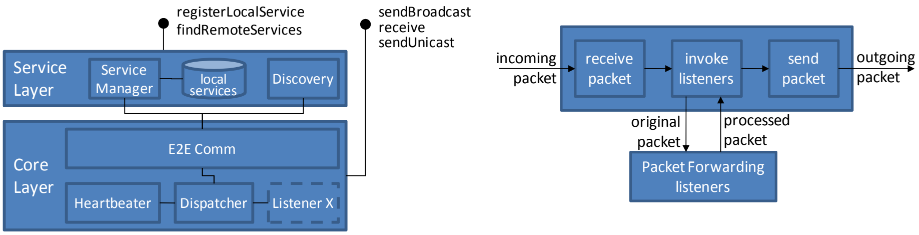


Fig. 5. RAMP architecture (left) and activity flow in RAMP Dispatcher (right)

To the purpose of enabling the flexible introduction of any application-layer operation on RAMP transmitted packets at runtime, we have implemented the Dispatcher according to a listener-based architecture, which permits to efficiently and easily monitor and/or modify exchanged packets at any traversing node (Figure 5-right). The addPacketForwardingListener Dispatcher method permits to add and register listeners to monitor incoming packets. In this way developers can implement and deploy additional listener-based components to support novel features, by extending RAMP capabilities without any modification of the basic Dispatcher. A detailed and general description of the RAMP middleware is out of scope here (please see the RAMP Web site [6]); in the following we focus on the crucial and original technical aspects of the RAMP support for Internet connectivity sharing.

BNs willing to share Internet connectivity activate InternetService and register it via `registerLocalService`; clients requiring Internet connectivity activate InternetClient, which exploits `findRemoteServices` to discover nodes offering InternetService. Service discovery is based on a TTL-bound broadcast research (default TTL=5), with neighbors that reply if they offer the required service; the reply message is used both to identify the service node (see below) and to invoke the service. This mechanism is suitable for medium-size networks, which is usually the case for the targeted spontaneous scenarios. In the rare case of very large-scale spontaneous networks (e.g., with hundreds of nodes and 20/30 hops diameter), it is possible to smoothly adopt more sophisticated discovery algorithms that cache information on intermediary nodes, analogously to AODV; anyway, this kind of wide-scale networks are not the primary RAMP target. L7 paths can be used via the `sendUnicast` primitive, which identifies the destination via the `dest` parameter, i.e., the ordered set of intermediary nodes composing the multi-hop path between sender and receiver. In fact, RAMP identifies a remote node via the IP addresses of the intermediary nodes in the path to that node. For instance, in Figure 6 NodeA identifies NodeB via the [2, 4, 6] sequence while NodeB identifies NodeA as [5, 3, 1] (sequences differ depending on path directions because different wireless ingress interfaces are exploited in the two ways).

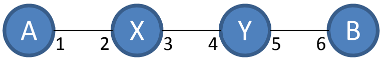


Fig. 6. Node identification depending on traversed interfaces

L3 path creation exploits the Layer3Manager component, registered as `PacketForwardingListener` to the local Dispatcher of every node along the client-to-BN path. A node requires to create an L3 path by sending (via InternetClient) a unicast Layer3Request packet to InternetService at the selected BN. On intermediate nodes, whenever Layer3Manager recognizes a traversing Layer3Request packet, it exploits `dest` and `currentHop` header fields to properly modify operating system-level routing rules. In particular, it sets as local default gateway the host in `dest` with position `currentHop` (the first host has index 0). For instance, in the path from NodeA to NodeB in Figure 6, when the Layer3Request packet reaches NodeX/NodeY, `currentHop` has value 1/2 and thus Layer3Manager sets NodeY/NodeB as default gateway. InternetService on the last node (e.g., NodeB) sends an ack to InternetClient to notify that the L3 path is ready, while its Layer3Manager component does not change its default gateway. Once the L3 path is ready, applications at the client node can adopt the L3SP mode to get Internet connectivity (no proxy).

On Linux nodes Layer3Manager exploits Linux `route` command to set the default gateway and `iptables` command to enable NAT traversal. For instance, on NodeX Layer3Manager executes:

```
route add default gw IP4 interf
iptables -t nat -A POSTROUTING -s IP1 -j MASQUERADE
```

where `IP1`/`4` is the IP address of the egress/ingress interface on NodeA/Y and `inter` is the name (e.g., eth0 or wlan0) of the NodeX interface with IP address `IP3`. In addition, at activation time, Layer3Manager temporarily enables operating system packet forwarding with the command

```
sysctl -w net.ipv4.ip_forward=1
```

In addition, InternetClient supports both L7MP and L3L7CMP acting as proxy and waiting at a well known port for local application requests, including the remote end-point (IP address, TCP/UDP, and port number) and the payload to send. At each request, InternetClient, based on current weights, selects one of the available paths, either L3 or L7: in the former case (only L3L7CMP) InternetClient contacts the end-point directly (single proxy); in the latter case it exploits `sendUnicast` and `receive` to send application requests and receive responses from the selected InternetService (double proxy).

InternetClient/Service can manage HTTP requests/responses. On the client-side, InternetClient parses requests to find the end-point IP address and port in the `Host` HTTP header. On the server-side, InternetService interacts with Web servers, receives their responses, and dispatches them to the client. In this way it is possible to transparently surf the Web in a spontaneous network by simply setting the local InternetClient as the Web browser proxy. Note that, given the instability of spontaneous networks, InternetClient uses `Connection:close` header, thus imposing to open new connections for each request/response pair. In addition, InternetClient/Service support a pseudo-HTTP format to allow also non-Web-based applications to exploit RAMP-based Internet connectivity: on the client-side, applications simply have to specify the standard `Host` and `Content-Length` headers plus our `Layer4Protocol` header with either `TCP` or `UDP` value, followed by an empty line and the payload, in either text or binary format (Figure 7); on the server-side, applications have only to indicate the desired `Content-Length` header and payload.

As summarizing implementation considerations, L3SP is certainly the simplest one: by directly exploiting L3-based connectivity, applications can access the Internet despite the adopted communication paradigm. Instead, L7MP and L3L7CMP currently support HTTP (and pseudo-HTTP), thus being easily applicable only to service components following the request/response communication paradigm. To support other application-level protocols, e.g., RTP for audio/video streams, there is the need to specifically add novel features to the RAMP InternetClient/Service. However, L3SP has a strong dependence on the underlying operating system, requiring modifying Layer3Manager to support different operating systems. For instance, the Layer3Manager version in the current RAMP prototype works only on Linux platforms. In addition, to enable packet forwarding and routing rule modifications, Layer3Manager requires running with administrator privileges (e.g., Linux superuser). This requirement can be viewed as a significant limitation. On the opposite, L7MP and L3L7CMP take advantage of available L7 paths with no need of administrator permissions and are available on any RAMP-enabled operating system.

```
Host: lia.deis.unibo.it:1234\r\n
Content-Length: 11\r\n
Layer4Protocol: TCP\r\n
\r\n
Hello World
```

Fig. 7. Example of client-side pseudo-HTTP request

5 Experimental Validation and Performance Results

To validate our Internet sharing solution, we have deployed and tested L3SP, L7MP, and L3L7CMP in the multi-hop multi-path spontaneous network depicted in Figure 8 (InternetClient resides on NodeC, InternetService on BNs). The targeted scenario is simple for the sake of briefness and easy interpretation of the results reported in the following, but still complex enough to well point out the RAMP behavior in a real, multi-path, heterogeneous deployment environment. In the following, we mainly concentrate on L7MP and L3L7CMP because L3SP performance almost entirely depends on available bandwidth (limited influence of RAMP performance) and because L3 paths are less frequently used in highly dynamic spontaneous networks.

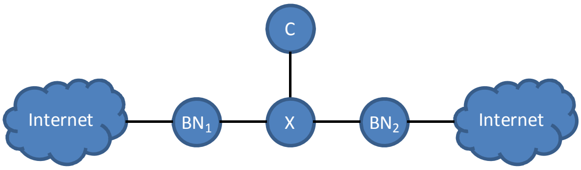


Fig. 8. Testbed deployment scenario

The nodes used in the tests are Core2 2.6GHz laptops with 2.0GB RAM, running MSWinXP (only L7MP) or LinuxDebian (both L7MP and L3L7CMP). NodeC-NodeX link is based on Ethernet (bandwidth limited to 2Mbit/s); all the other single-hop links are on IEEE 802.11b (infrastructure and ad-hoc modes) with CISCO access points and Orinoco cards (available bandwidth set as specified in the following); all the reported performance figures are representative examples selected over 100 runs.

InternetClient takes about 72ms to discover the two InternetServices. The starting value for weights is 0.5 for both BNs, since they are both two-hop distant (*Path-Length* metric). We have tested RAMP while supporting the access of a standard Firefox/Iceweasel Web browser to Google Maps, by exploiting the local InternetClient as proxy. Google Maps, like many Web applications, is characterized by frequent interactions (up to 12 interactions/s), with relatively limited payload (from 1.1 to 29.45KB per request/response, 12.04KB average size, 7.52 standard deviation). This pattern of interaction is particularly challenging for spontaneous networking (frequent different interactions, each one with small-size payload); that is the reason why we have selected it, in order to evaluate the RAMP middleware under stress in a sort of worst case scenario in terms of application traffic type.

In particular, we have collected experimental results about i) the throughput achieved by NodeC and ii) the time evolution of computed weights, when adopting L7MP (Figure 9a) and L3L7CMP (Figure 9b). The goal is to quantitatively evaluate how well RAMP can adapt its behavior dynamically according to actual in-the-field path quality, at the same time by estimating the overhead associated with L7 paths.

About L7MP (Figure 9a), we have initially set BN_1 bandwidth to 125KB/s and BN_2 one to 25KB/s. Throughput depends on many factors, such as Web server load and payload size; these elements have demonstrated to be the main motivation why BN_1 and BN_2 throughputs resulted quite low, independently of RAMP performance.

However, by focusing on the most interesting comparison between the two, BN_1 throughput has demonstrated to outperform BN_2 in many cases, due to the wider bandwidth of the RAMP path toward BN_1 if compared with BN_2. In addition, based on the monitored throughput, after about 20s, InternetClient increases/decreases BN_1/BN_2 weights respectively to 0.81 and 0.19 (*PathThroughput*), thus pushing to exploit BN_1 more frequently than BN_2. At t=105s, we have inverted bandwidth allocation (25KB/s for BN_1, 125KB/s for BN_2) to test the RAMP capability of dynamic adaptation: InternetClient has demonstrated to be able to react promptly with proper weight modifications notwithstanding our lightweight monitoring approach simply based on application-level throughput observation.

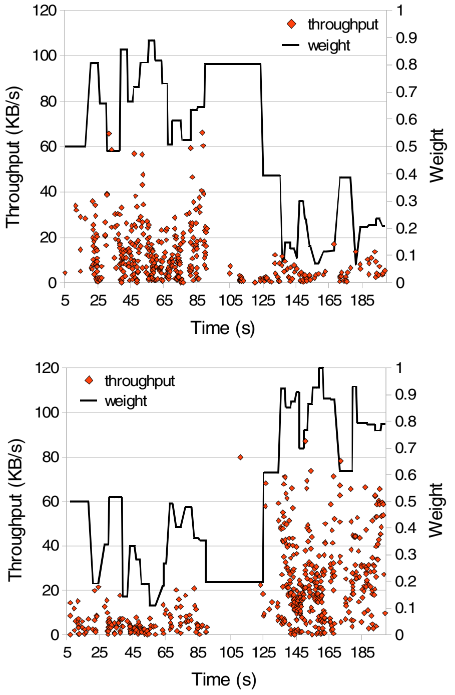


Fig. 9a. Throughput and weights for L7MP BN_1 (up) and BN_2 (down)

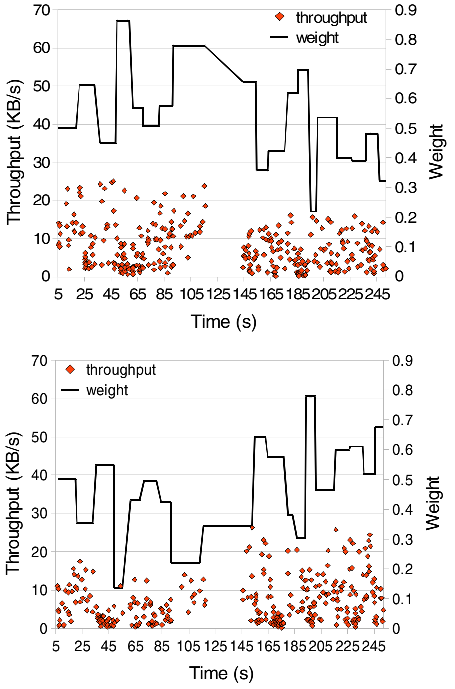


Fig. 9b. Throughput and weights for L3L7CMP BN_1 (up) and BN_2 (down)

About L3L7CMP (Figure 9b), we have allocated the same bandwidth to BN_1 and BN_2 (30 KB/s) and observed RAMP behavior when creating an L3 path toward either BN_1 (before t=125s) or BN_2 (after t=125s). RAMP has demonstrated to require about 298ms to create the two-hop L3 path, measured on InternetClient (from request message to ack reception). It is worth noting that RAMP tends to provide greater priority to L3 paths than to L7 ones, given the usual slightly greater throughput associated to L3 paths (up to 27KB/s vs. 18KB/s). In other words, the RAMP middleware actually perceives that the L3 path provides a slightly greater throughput than the L7 one, and dynamically adapts its behavior by exploiting more frequently the former instead of the latter. However, the throughputs actually achieved via L3 and L7 paths, when measured in-the-field, are frequently almost equivalent (and accordingly the weights tend to become similar), because they tend to mainly depend on Web server load and payload size, as already stated. At the same time, this demonstrates the limited overhead introduced by RAMP when managing packet forwarding at the application level.

6 Related Work

Several proposals have investigated specific partial aspects of more "traditional" multi-hop connectivity: a few recent works are starting to propose the synergic and simultaneous exploitation of heterogeneous wireless interfaces at mobile terminals; most have focused on one specific technology, such as IEEE 802.11 or GPRS/UMTS. However, their primary accent is on seamless connectivity in environments where heterogeneous wireless technologies are integrated. For instance, [8] aims at extending cellular networks via relay stations to increase coverage; [9, 10], instead, specifically address client mobility management in heterogeneous multi-hop networks.

Different aspects of spontaneous networking have been addressed by a number of research activities in the recent literature; here, we focus only on the projects more closely related to RAMP. By focusing on multi-hopping in spontaneous networks, some contributions aim at increasing connection quality via low-level solutions. For instance, [1] improves wireless medium exploitation by opportunistically accessing the available spectrum. [2] optimizes bandwidth allocation by differently managing real-time and best-effort transmissions. The proper support of multi-path connectivity has gained increasing attention only very recently. Some proposals determine the best route towards a destination by exploiting evaluation metrics based on low-level context [11]. Others exploit network-layer context to estimate current path load and to appropriately distribute generated traffic among the available paths [3]. Some proposals specifically focus on multimedia streaming via multi-path channels, with the main scope of improving stream quality via rate allocation algorithms that properly interact with the operating system [4]. Finally, opportunistic and delay-tolerant networking is emerging as an interesting approach for connectivity in highly dynamic spontaneous networks [1]. For instance, [12] supports opportunistic data delivery in intermittently connected mobile ad hoc networks. However, the proposal in [12] is only based on simulations and only considers homogeneous networks with plain addressing.

In short, most related contributions in the literature aim at supporting spontaneous networking mainly in homogeneous networks, by introducing non-standard modifications to layer-2 protocols. In addition, they do not address the heterogeneity issues associated with the exploitation of multiple interfaces, with IP addressing, and with the specific support of Internet connectivity sharing services.

7 Conclusions

The effective and appropriate collaborative sharing of Internet connectivity can be a "killer application" for spontaneous networking, by relevantly promoting its adoption and enabling a better exploitation of the growing amount of computing/memory/bandwidth resources available on portable wireless terminals. This work makes a further step toward this vision and demonstrates that i) both network- and application-layer approaches are suitable and ii) the decision of which mode to adopt should be taken at service provisioning time based on application-specific requirements and spontaneous network performance evaluation. In particular, the RAMP L3SP mode (based on L3 approach) has demonstrated to be adequate for its transparency to application protocols and interaction paradigms; however, it may suffer from frequent topology changes due to the costs of L3 path reconfiguration. Instead, RAMP L7MP and L3L7CMP modes have demonstrated their suitability for highly dynamic network environments: simultaneous exploitation of multiple paths improves connectivity reliability and quality. In particular, the L3L7CMP mode can exploit both L3 and L7 approaches simultaneously. In addition, our in-the-field performance results demonstrate that RAMP imposes limited overhead if compared with more traditional network solutions based only on routing at the operating system-level.

The encouraging results already achieved are stimulating further research activities on spontaneous networking. In particular, we are validating an extended version of the RAMP prototype that supports application-layer splitting of multimedia streams via differentiated paths, in order to both increase throughput and minimize packet loss rate. In addition, we are working on enhancing the RAMP support to peer fairness through the adoption of innovative distributed trust management solutions based on lightweight monitoring/evaluation of users' behavior (resource offers/requests).

References

1. Salameh, H.B., Krunz, M.: Channel Access Protocols for Multihop Opportunistic Networks: Challenges and Recent Developments. IEEE Network 23(4), 14–19 (2009)
2. Wu, H., Liu, Y., Zhang, Q., Zhang, Z.L.: SoftMAC: Layer 2.5 Collaborative MAC for Multimedia Support in Multihop Wireless Networks. IEEE Trans. on Mobile Computing 6(1), 12–25 (2007)
3. Toh, C.K., Le, A.-N., Cho, Y.-Z.: Load balanced routing protocols for ad hoc mobile wireless networks. IEEE Communications Magazine 47(8), 78–84 (2009)
4. Frossard, P., de Martin, J.C., Reha Civanlar, M.: Media Streaming With Network Diversity. Proceedings of the IEEE 96(1), 39–53 (2008)
5. de Amorim, M.D., Ziviani, A., Viniotis, Y., Tassiulas, L.: (eds.) Special Issue on Practical Aspects of Mobility in Wireless Self-organizing Networks. IEEE Wireless Communications 15(6) (December 2008)
6. RAMP Web site, lia.deis.unibo.it/Research/RAMP
7. Bellavista, P., Corradi, A., Giannelli, C.: Mobility-aware Middleware for Self-Organizing Heterogeneous Networks with Multi-hop Multi-path Connectivity. IEEE Wireless Communications 15(6), 22–30 (2008)
8. Le, L., Hossain, E.: Multihop Cellular Networks: Potential Gains, Research Challenges, and a Resource Allocation Framework. IEEE Communications 45(9), 66–73 (2007)
9. Pack, S., Shen, X., Mark, J.W., Pan, J.: Mobility Management in Mobile Hotspots with Heterogeneous Multi-hop Wireless Links. IEEE Communications 45(9), 106–112 (2007)
10. Lam, P.P., Liew, S.C.: Nested Network Mobility on the Multihop Cellular Network. IEEE Communications 45(9), 100–104 (2007)
11. Campista, M.E.M., et al.: Routing Metrics and Protocols for Wireless Mesh Networks. IEEE Network 22(1), 6–12 (2008)
12. Musolesi, M., Mascolo, C.: CAR: Context-Aware Adaptive Routing for Delay-Tolerant Mobile Networks. IEEE Trans. Mobile Computing 8(2), 246–260 (2009)

Session 3: Human-Computer Interface for Mobile Devices

(Chair: Honggang Wang)

Prototyping Convergence Services on Broadband Networks

Alice Motanga, Andreas Bachmann, and Thomas Magedanz

Fraunhofer Institute for Open Communication Systems (FOKUS),
Kaiserin-Augusta-Alle. 31, 10589 Berlin, Germany
{alice.motanga,andreas.bachmann,magedanz}@fokus.fraunhofer.de

Abstract. In today's competitive business environment, operators seek to simplify their network topology in order to cut costs and create a convergent network infrastructure that is secure, easy to manage, always available, and capable of providing bandwidth to new multimedia service traffic loads and changing business needs. This paper gives an overview of broadband networks and how to provision services such as voice, which remains the main revenue generator for operators, on such networks.

Keywords: Open APIs, Multimedia Services, Evolved Packet System, Client Framework.

1 Introduction

Mobile and fixed network consumers have moved from simply using voice and data to more visually oriented, high definition entertainment, video conferencing and integrated services on broadband networks. Broadband networks are networks that can connect user's terminal equipments to network service providers and offer an always on functionality and that has a high capacity for sending and receiving data.

So what is driving broadband networks? In recent years, the Internet usage patterns and behaviour are migrating to the mobile arena. Subscribers are beginning to have the same communication expectations whether at home or on the move. Rather than technology driving user behaviour, this change is user centric, driven by people as consumers and business users. People rapidly take up new services and new ways of using them, and the technology has to evolve to keep up with this. Consumers and business users have similar needs in terms of convenience and being constantly connected.

1.1 Evolved Packet System

The Evolved Packet Core (EPC) network standardized by the Third Generation Partnership Project (3GPP) as part of Release 8 is an efficient standard based all IP network architecture for supporting the next generation of full service broadband. EPC is an important step forward for operators looking to secure success for the long term. EPC together with Long Term Evolution (LTE) access technology form the Evolved Packet Systems (EPS), providing everywhere coverage and always on broadband access for fixed, nomadic and mobile users.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 103–114, 2010.

This enables a richer variety of services and better user experience. Unlike its predecessors, EPC provides support for multiple access technologies and provides mobility between them, allowing subscribers to move between different accesses while providing service continuity. The core network facilitates a fully multi-service converged core, with support for multiple access technology and interworking with legacy 3GPP and non-3GPP networks. It also enables a common core network for Fixed Mobile Convergence (FMC), which significantly reduces the cost of ownership and facilitates development for multi-services subscriber offerings.

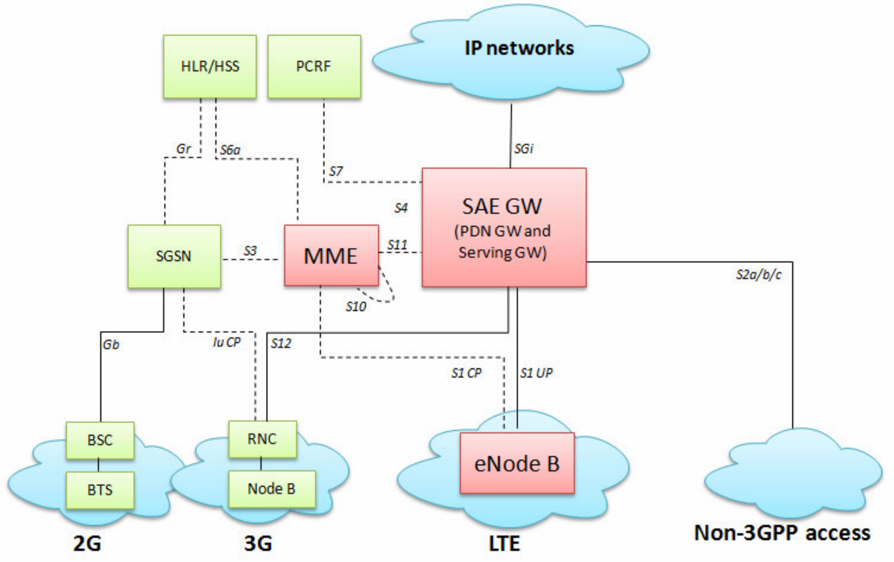


Fig. 1. Overview of the Evolved Packet System[1]

The rest of this paper is divided as follows – chapter 2 presents applications driving the market trend in adopting broadband networks and the challenges faced in replacing legacy applications such as voice and SMS service. Chapter 3 discusses an open standard approach in meeting up to the challenges, while chapter 4 and 5 present a practical approach in crystallize the ideas specified in chapter 3. Chapter 4 seals and concludes this topic.

2 Multimedia Services on Broadband Networks

The Smart devices and smart communication help people to joggle the roles of private and professional life. Today, we can observe how communication is changing the

[1] PDN – Packet Data Network.
SAE – System Architecture Evolution.
MME – Mobility Management Entity.
eNode B – E-UTRAN Node B access.

uptake of new services. ITunes features the worlds' largest video and music catalogue. Its customers have purchased over 4 billion songs as well as renting of purchasing over 125 million TV episodes and 8 million movies. The media is consumed using computers, handsets and conventional devices. The figures for Facebook and YouTube are equally astronomical. As more of these consumer services are migrating to the wireless domain, the community is now on mobile domain, requiring the same facilities both at home and on the move. Together, these statistics point to one fact mobile multimedia communication is happening now across the world and throughout consumers and business groups.

In addition to above described data services (e.g. music and video streaming) and messaging services which belong to the primary focus on enabling an efficient mobile broadband solution, the support for voice and SMS services is also given high priority in the broadband network architecture specification. However, one of the trickiest issues for early broadband adoption is the uncertainty over how voice and SMS services, which are still the key cash flow application for high revenues for most operators.

Conventional telephony communicates using the voice medium only, and connecting only two telephones per user over circuits of fixed bit rates. In contrast, modern communication services depart from the conventional telephony service in three essential aspects; multimedia, multi-point, and multi-rate.

Multimedia voice service may communicate audio, still images, or full motion video or a combination of these media. Each medium requesting different demand on communication qualities such as bandwidth, signal latency within the network, and signal fidelity upon delivery by the network.

Multi-point calls involve the setup of connections among more than two people. These connections can be multimedia. They can be one way or two way communications. These connections may be reconfigured many times within the duration of a call. Traditional voice calls are predominantly two party calls, requiring a point-to-point connection using only the voice medium.

Multi-rate service network is one which allocates transmission capacity flexibly to connections. A multimedia network supports a broad range of bit-rates demanded by connections, not only because there are many communication media types, but also because a communication medium may be encoded by algorithms with different bit-rates. For example, audio signals can be encoded with bit-rates ranging from less than 1 kbit/s to hundreds of kbit/s, using different encoding algorithms with a wide range of complexity and quality of audio reproduction. Similarly, full motion video signals may be encoded with bit-rates ranging from less than 1 Mbit/s to hundreds of Mbit/s.

3 A Standard Based Approach for Voice over Broadband Networks

There are two main solutions for enabling voice services on broadband networks. One solution is to use the IP Multimedia Subsystem (IMS) mechanism specified in 3GPP Release 5 and realize voice using the MultiMedia Telephony (MMTEL) framework introduced in Release 7. The second possibility would be to stick to the old circuit-switched way of providing voice services. The second option would be possible in the EPS network realization by that users temporarily leave the LTE network to perform the voice calls over 2G/3G network, and then return when the voice call is finished. This is not the most elegant of solutions, but it can be realized primarily networks, which lack an IMS infrastructure.

Table 1. EPS solution to voice services

Legacy voice service	Transition Solution	EPS solution
CS voice	CS Fallback (Rel 8)	IMS VoIP (Rel 7)
Supplementary Services	CS Fallback (Rel 8)	Multimedia Telephony (Rel 7)
Emergency Calls with Location Support	CS Emergency Calls (Rel 5)	IMS Emergency Calls with Location Support (Rel 9)

Many initial broadband deployment strategies were to deploy LTE as a data-only network. However voice and, even more importantly, SMS remain the key revenue generating applications for operators. Faced with the risk that large players might delay deployment plans until they have a strong route to voice, the One Voice Initiative [3] was created with endorsement from key operators, with the aim of defining a profile named the Open Voice Profile based on existing 3GPP standards.

The Open Voice Profile defines a minimum mandatory set of features a User Equipment (UE) and network are required to implement in order to guarantee an interoperable, high quality IMS-based telephony service over EPS radio access.

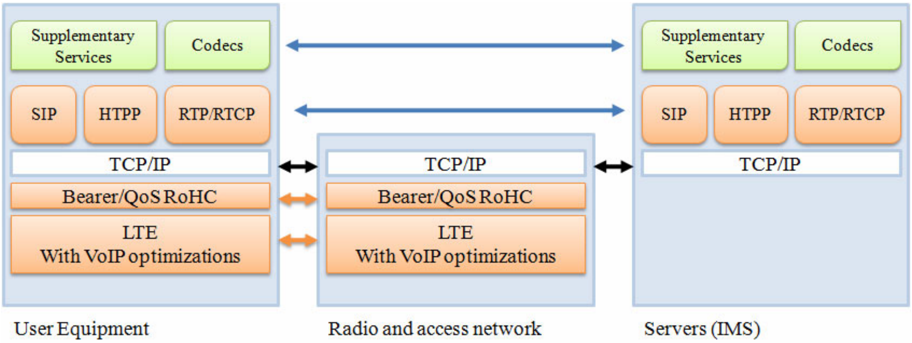


Fig. 2. One Voice Profile for UE and network protocol stacks. Note that the TCP/IP layer also includes UDP and XCAP.

The scope of the profile defines the following aspects:

- IMS capabilities & voice including supplementary services for telephony
- Real-time media negotiation, transport and codec
- LTE radio and Evolved Packet Core capabilities
- Functionality that is relevant across the protocol stack and subsystems.

The profile defines an optimal set of existing 3GPP-specified functionalities that all industry stakeholders, including network vendors, service providers and handset manufacturers, can use to offer compatible LTE voice solutions. This approach will also open the path to service convergence, as IMS is able to simultaneously serve broadband fixed and LTE wireless networks.

From the Open Voice Profile specification, we have implemented a prototype based on the UE specification requirements which offers the minimum set of mandatory service capabilities. The implementation was based on our client service framework myMONSTER [10] (Multimedia Open Services and Telecommunication EnviRonment) software toolkit. myMONSTER is an extendible plug-and-play framework developed by Fraunhofer FOKUS. This toolkit enables the creation of rich terminal applications compliant with NGN, IPTV and Web standards. myMONSTER provides three toolkits – for telecommunication, television and web approaches. We shall present the telecommunication package of the framework in more details in the next chapter.

4 Client Service Creation Toolkit

Parallel to network and service delivery initiatives, there are also some initiative going on in the terminal device arena. One of the most significant responses of industry players to the client development has been the Rich Communication Suite (RCS).

4.1 Rich Communicator Suite

The Rich Communicator Suite Initiative started by a small group of leading industry players in 2007, and in February 2008, it was launched at the Mobile World Congress in Barcelona. The Global System for Mobile Communications Association (GSMA) added RCS to its work program in September 2008 and now more than 70 Converged Solution Providers (CSP) and vendors are part of the saga.

In 2008, the concentration of the RCS features was only on mobile phones, but with the expansion of broadband networks, this quickly expanded to include other platforms with broadband access. To also compliment the efforts of other initiatives

Table 2. Rich Communication Suite Release Overview

Release 1 (12.2008)	Release 2(06.2009)	Release 3(12.2009)
Enhanced phone address book with presence	Broadband Access to RCS features	Enhancement of Release 2 features
Content Sharing	Multi-device environment	
File Sharing	Provisioning and configuration of RCS devices/clients	
Enhanced Messaging (SMS/MMS & Chat)	OMA IM and MMTel integration	

such as Open Voice and Voice over LTE (VoLTE), RCS also integrates open standards from OMA IM and also recently the 3GPP/TISPAN MMTel specifications on their features portfolio.

The advantage of the RCS within the industry is to ensure a steady, open standard conformant implementation on vendor devices for the mass market. However, we find one aspect missing which is pertinent in the overall success to IMS-based multimedia services. The RCS follows already defined open standards to specify services for mobile and fixed terminals, but the actual implementation of these services, again remain left to the different partners to implement on their various devices. From other mobile application development platforms such as the Google Android, multimedia application development is driven by the user development community and not the vendor. Therefore, we find it important that third party developers are given the tools they need to promote and help mass deployment of IMS services on different platforms. The lack of an SDK, which developers can download and quickly integrate on the platforms for third party service integration slows the uptake of IMS based telephony services compared to the use of Web APIs which provide similar service.

4.2 myMONSTER Telecommunication Communicator Suite (TCS)

The Telco Communicator Suite is a Java-based framework that delivers a unified communication experience for all IP networks. It is powerful, yet lightweight enough to run on both fixed and mobile devices. This suite provides developers with high level APIs for easy integration into their applications in order to enrich them with telecommunication aware services.

The services on the TCS framework are modularized and decoupled, giving developers the flexibility and options to extend the framework with their own components. The framework is built in a plug-n-play approach of service bundles known as modules. These modules provide well defined APIs which developers can use to integrate on their own applications known as “add-ons”. Table 3 provides an overview of the basic API offered on the framework. The framework does not only provide protocol stacks for the IMS network but also provide other communication protocol stacks.

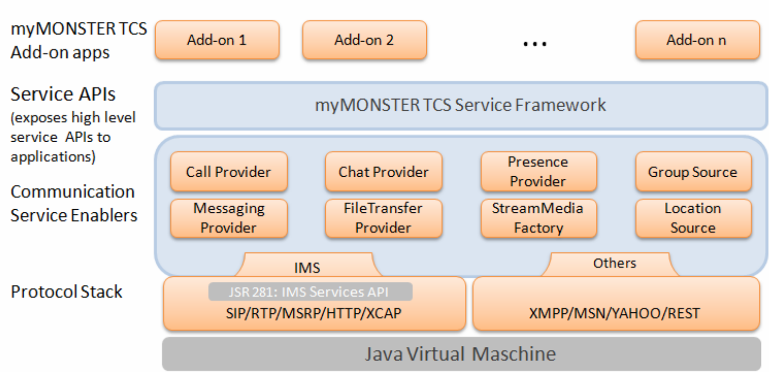


Fig. 3. High level overview of the myMONSTER Telecommunication Communicator Suite framework

Table 3. myMONSTER TCS Communication Service Enablers APIs

API	Description
Call	Creating audio and video calls including call control functions like call-hold, call-resume and call-transfer (IMS VoIP).
Instant Messaging and Chat	Sending instant messages in page mode using SIP and session mode messaging using MSRP (Message Session Relay Protocol) with extensions to OMA IM specification
Presence	Publication of the presence state and getting notification of user presence supporting PIDF, RPIDF and OMA presence
Location	Access to different device location sources like GPS, Cell Id and static locations enables location based services.
Network stored address book	Managing groups for contacts and contact data with local and server-side storage (ext. OMA CAB v1.0)
File Sharing	Creating multimedia sessions for sending and receiving multiple media file types over MSRP

The key benefits and advantages of the myMONSTER TCS client framework include: Shortens development time for third party developers; IMS stack builds on open standards from 3GPP (TS 24.229 [5]) and JSR 281 [6] specifications; multiple target platforms (Linux, Windows Vista /XP/7, Mac, Windows Mobile, and Google Android); decoupling of service logic from the presentation layer enables multiple presentation layers (Swing, SWT, Widgets, embedded) and facilitates branding.

5 Extension on myMONSTER TCS for Voice Provisioning over Broadband

As discussed in chapter 3, Open collaborative discussions concluded that the IMS based solution, as defined by 3GPP, is the most applicable approach to meeting the consumers' expectations for service quality, reliability and availability when moving from existing circuit switched telephony services to IP-based EPS services. This approach will also open the path to service convergence, as IMS is able to simultaneously serve broadband fixed and LTE wireless networks.

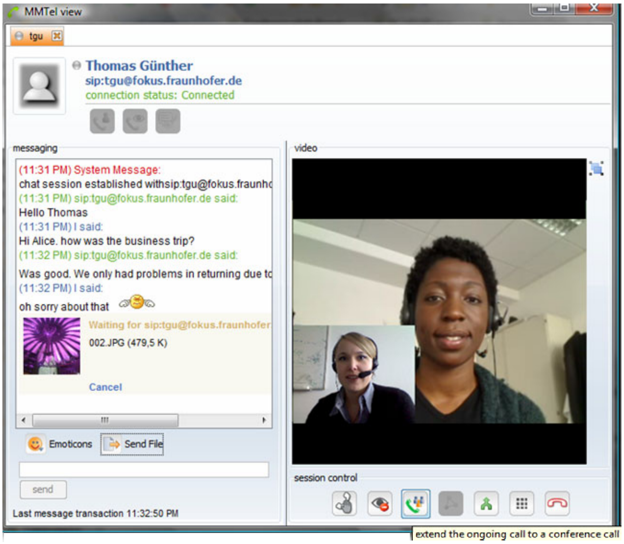
To extend the myMONSTER TCS framework capability for a Voice-over-EPS solution, the following functionalities had to be implemented: MMTel and Supplementary services support; IMS Emergency Calls with Location Support; Session Mobility Manager.

5.1 MMTel and Supplementary Service Add-On Module

The MMTel service module was introduced on the framework to address a new user proposition in which the real time communication between session participants is set according to the telephony paradigm fulfilling quality of service, authentication, authorization, regulatory and efficiency requirements. The new user proposition includes a voice over IP telephony service that can use a set of simulated PSTN/ISDN supplementary services and add and drop a number of different media types during a session to adapt to the current communication need. Examples of media types that can be used in the MMTel session include:

- Voice, both narrow band and wide band quality,
- full-duplex or half-duplex video,
- text, where the characters are either transmitted in real-time when the user types or as pre-typed messages,
- general files that are stored in the receiving terminals memory or files of known file formats.

In addition, the user has the possibility of creating ad-hoc multi party conference sessions, with subscription to the “conference” event package, three way party calls, and extending created peer-2-peer sessions to conference sessions.


MMTel view
tgu
Thomas Günther
sip:tgu@fokus.fraunhofer.de
connection status: Connected
messaging
video
Emoticons
Send File
send
session control
extend the ongoing call to a conference call

(11:31 PM) System Message:
chat session established withsip:tgu@fokus.fraunhc
(11:31 PM) sip:tgu@fokus.fraunhofer.de said:
Hello Thomas
(11:31 PM) I said:
Hi Alice. how was the business trip?
(11:32 PM) sip:tgu@fokus.fraunhofer.de said:
Was good. We only had problems in returning due tc
(11:32 PM) I said:
oh sorry about that
Waiting for sip:tgu@fokus.fraunhofer
002.JPG (479,5 K)
Cancel

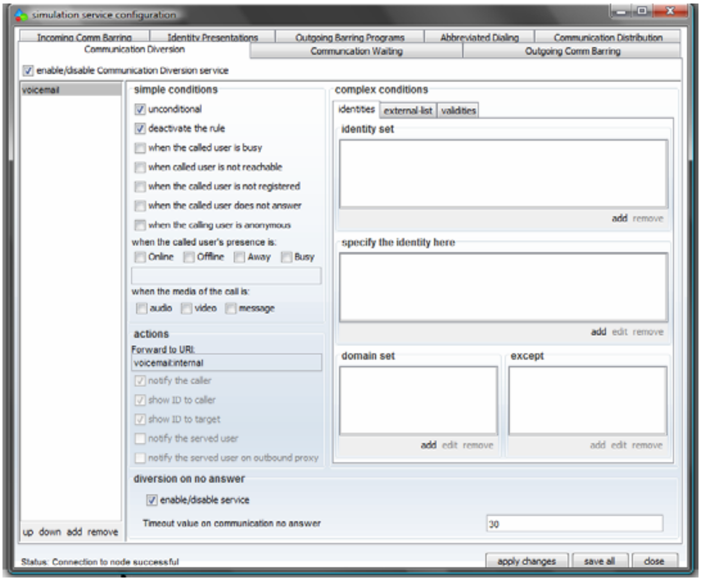
Last message transaction 11:32:50 PM

Fig. 4. Multimedia Telephony View showing an ongoing session with different media types (audio, video, message, files) all within a single established session

Besides the real-time multimedia creation, the MMTel module also provides a complex Ut reference interface (XCAP) implementation towards the application server (XDMS) for configuring supplementary services as defined in 3GPP TS 24.623. Supplementary services are telecommunication services that provide PSTN/ISDN-like service capabilities using session control over IP interfaces and infrastructure. We provided configuration interfaces for multiple service configurations (see Fig. 4).

When an MMTel subscriber registers and is authenticated on the core network, the network interacts with the subscriber's profile on the home registry. If the subscriber's data shows a subscription for the MMTel service it dynamically allocates a Multimedia Telephony Application Server (MTAS) to serve the subscriber on establishing multimedia sessions. When setting up an MMTel session, using the service identifier the core network can determine that session related signaling belongs to an MMTel call, and hence routes the call to the pre-selected MTAS. The MTAS executes the main part of the call control and supplementary services are invoked by the MTAS, if needed.

The development of this extension is performed under the umbrella of an industry Project and is geared toward operators and who provide MMTel service on their broadband networks.



☑ enable/disable Communication Diversion service

voicemail

☑ unconditional
☑ deactivate the rule
□ when the called user is busy
□ when called user is not reachable
□ when the called user is not registered
□ when the called user does not answer
□ when the calling user is anonymous

when the called user's presence is:
□ Online □ Offline □ Away □ Busy

when the media of the call is:
□ audio □ video □ message

Forward to URI: voicemail:internal
☑ notify the caller
☑ show ID to caller
☑ show ID to target
□ notify the served user
□ notify the served user on outbound proxy

☑ enable/disable service
Timeout value on communication no answer 30

Status: Connection to node successful

Fig. 5. Simulation Service configurations interface that the user can use to configure its supplementary service profile

5.2 IMS Emergency Calls and Location Support Add-On Module

IMS VoIP support for emergency calls (including location support) is specified in 3GPP Release 9 which fulfils the last regulatory requirement separating VoIP from CS in 3GPP networks.

During an emergency situation, the UE can register and place a free call to a Public Safety Answering Point (PSAP). The emergency IMS registration can be used only to place emergency calls. The UE acquires queries for location information, and includes this information in the initial request of the emergency call. If the location information is missing in the initial request, the core network can query for the user's location from the access network and refers it in the request. The request is forwarded to the Emergency CSCF (E-CSCF). Upon receiving the emergency related SIP message by the E-CSCF, in case no location information was provided, the E-CSCF queries the Location Retrieval Function (LRF) for the user location. The LRF ensures that the E-CSCF receives the most appropriate PSAP URI, e.g. Police call taker. Then the emergency SIP message is forwarded further to this PSAP.

The myMONSTER emergency services extensions overcomes the problem of the caller location within all IP networks. The location of the device is queried and inserted in the message flow so that the network can map the service request to the nearest PSAP.

The development of this extension is performed under the umbrella of the PEACE European Project [4] and is geared toward operators and safety organizations that have already started the migration of their current emergency system to broadband networks.

5.3 Session Mobility Manger Add-On Module

The Session Mobility Manager on the UE has the role of interacting with the Core Network Mobility Management Components and to provide a seamless experience for the applications running on the client devices, such that operations like network attachment or handovers would be transparently handled. To perform these operations, the Mobility Manager (MM) component on the UE will orchestrate the normal network management procedures. For providing value-added functionality, the Access Network Discovery and Selection Function (ANDSF) situated in the core network assists the Mobility Manager with information and operator pushed policies.

The ANDSF communicates with the MM running on the UE and exchanging information which would enhance both the Always Best Connected (ABC) concepts, but also allows the network operator to manage and enhance connectivity on a

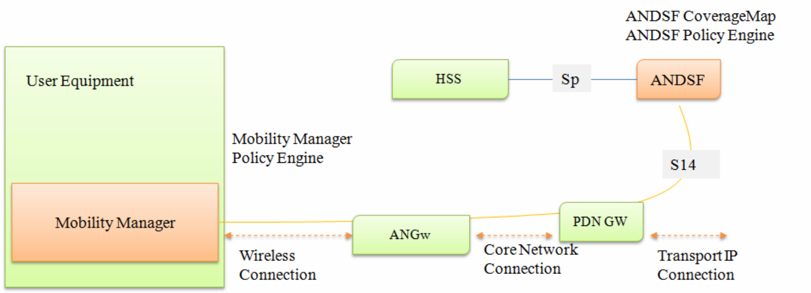


Fig. 6. Session Mobility Support for seamlessly handling session handover and other policy based access network selection functions

multi-access environment as specified in 3GPP TS 23.402 and 3GPP TS 24.312. The ANDSF and the MM communicate through the S14 interface. The transport mechanism is currently limited to a simple XML exchange over TCP interface, with an additional triggering mechanism by simple UDP alerting.

The myMONSTER EPC add-on is a plug-in on the framework which was developed to demonstrate the integration of a mobile client framework with the Mobility Manager. It exposes to the user all the IP connectivity and operator pushed (over the S14 interface) information. It also provides a configuration interface for configuring the behavior of the MM and for manually triggering session handovers, which is demonstrated via a simple video-streaming application.

The development of this extension is performed under the umbrella of the project OpenEPC [12], a prototype reference implementation of the 3GPP Release 8 Evolved Packet Core (EPC) that will allow academic and industrial researchers and engineers around the world to obtain a practical look and feel of the capabilities of the Evolved Packet Core.

6 Conclusion

This paper presented an overview of broadband networks. The EPC network, from 3GPP Release 8 specification offers together with LTE access technology to form the EPS, providing everywhere coverage and always on broadband access for fixed, nomadic and mobile users. This enables a variety of new multimedia. While data services such as E-mail, social networks, video streaming applications, increase in consumer size and leverage the advantages of broadband networks, voice and SMS services which remain their key revenue generating applications for operators are still to be realized.

A group on operators forms an initiative called the Open Voice Initiative, who then defined an Open Voice Profile based on already exciting Multimedia Telephony solutions and Emergency Call solutions on the IMS network as specified by 3GPPP Release 7 and 9 respectively. This profile defines a minimum mandatory set of features a UE and network are required to implement in order to guarantee an interoperable, high quality IMS-based telephony service over EPS radio access.

Based on this profile, we implemented extensions on the myMONSTER TCS framework; a UE implementation for creating applications on the client side, to prototype the requirements of the Open Voice Profile. The extensions were developed and tested within the scope of an industry operator environment, a European project (PEACE) and a reference implementation project (OpenEPC) of the 3GPP Release 8 specification of the Evolved Packet Core.

References

1. Rich Communication Suite Initiative, `http://www.gsmworld.com/our-work/mobile_lifestyle/rcs/`
2. 3GPP TS 23.402: Architecture enhancements for non-3GPP accesses
3. Open Voice Initiative, One Voice; Voice over IMS Profile V1.0.0 (November 2009)

4. PEACE, IP based Emergency Application and Services for Next Generation Networks, http://www.ict-peace.eu/
5. 3GPP TR 24.229: IP multimedia call control protocol based on Session Initiation Protocol (SIP), and Session Description Protocol (SDP), Stage 3
6. Java Specification Requests JSR 281: IMS Services API. JSRs, http://jcp.org/en/jsr/detail?id=281
7. Java Specification Requests JSR 325: IMS Comunication Enablers. JSRs, http://jcp.org/en/jsr/detail?id=325
8. Ericsson, IMS - IP Multimedia Subsystem - The value of using the IMS architecture. White Paper (October 2004)
9. IP Multimedia Subsystem (IMS); Stage 2, Technical Specification Group Services and System Aspects V9.0.0 TS 23.228
10. myMONSTER, Rich client Toolkit, http://www.mymonster.org
11. Open Evolved Packet Core Project, http://www.openepc.net/en/openepc/index.html

Adaptive Online Deployment for Resource Constrained Mobile Smart Clients

Tim Verbelen, Raf Hens, Tim Stevens, Filip De Turck, and Bart Dhoedt

Ghent University - IBBT, Department of Information Technology,
Gaston Crommenlaan 8 bus 201, 9050 Gent, Belgium

Abstract. Nowadays mobile devices are more and more used as a platform for applications. Contrary to prior generation handheld devices configured with a predefined set of applications, today leading edge devices provide a platform for flexible and customized application deployment. However, these applications have to deal with the limitations (e.g. CPU speed, memory) of these mobile devices and thus cannot handle complex tasks. In order to cope with the handheld limitations and the ever changing device context (e.g. network connections, remaining battery time, etc.) we present a middleware solution that dynamically offloads parts of the software to the most appropriate server. Without a priori knowledge of the application, the optimal deployment is calculated, that lowers the cpu usage at the mobile client, whilst keeping the used bandwidth minimal. The information needed to calculate this optimum is gathered on the fly from runtime information. Experimental results show that the proposed solution enables effective execution of complex applications in a constrained environment. Moreover, we demonstrate that the overhead from the middleware components is below 2%.

Keywords: Middleware, Pervasive computing, Offloading, Software partitioning, Smart clients.

1 Introduction

Although mobile devices gain more and more capabilities, mobile applications still have to cope with much less resources than their desktop or server counterparts. Limited memory capacity, CPU speed, network bandwidth and battery power constrain the complexity of the applications. For advanced applications such as augmented reality the programmer has to trade accuracy or robustness for an acceptable framerate [14].

One solution to cope with device limitations is to use the thin client computing model. The mobile device then only handles input from and output to the user, while the application logic is executed on a remote server. This concept dates back to the era of mainframe computers, but recently revived for business desktop applications because it facilitates centralized management of software and reduces hardware cost of client devices. Examples of such systems are Citrix ICA (Independent Computing Architecture) [25] and Sun Ray [24]. The biggest

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 115–128, 2010.

problem to use the thin client setup in a mobile environment is to cope with the varying properties of a wireless network. It is shown that latency is an important limiting factor for thin clients over a WAN [15] and they are also not resilient to data bursts as discussed in [23]. Hence, executing all application logic on the server is not the optimal solution.

Another solution consists of adapting the application to the capabilities of the mobile device by replacing some parts of the software by other, more lightweight components. In [10] a framework is presented which switches between components depending on the context. However, this solution will mostly result in a degraded application and heavily depends on the application developers' willingness to provide different versions of the components. It's also impossible to run an application that needs more resources than the maximum available on the device, which is still fairly limited.

In this paper we propose a middleware solution for smart clients where the application is dynamically divided between the mobile client and a remote server. By choosing the optimal deployment we lower the CPU usage and minimize the consumed bandwidth to be able to run demanding applications on the device. The optimal deployment can change over time as the context in which the application is executed will also change. Contrary to the local adaptation approach discussed above, the proposed solution will not result in a degraded version.

The remainder of this paper is structured as follows. In the next section we discuss related work. Section 3 presents a typical use case used throughout this paper and Section 4 will outline the architecture of the system. In Section 5 the implementation details and design issues of the different components are discussed. Our experimental results are presented in Section 6 and finally Section 7 concludes this paper.

2 Related Work

Since the rise of state-of-the-art middleware such as CORBA [8] and Java RMI [20], research has been done to transform legacy software into distributed applications. JavaParty [17], Doorastha [3] and AdJava [7] make a Java application distributed by preprocessing its source code and generating remote invocation code. The programmer decides which part of the software will run remotely by using special keywords. Big drawbacks of this approach are of course that the source code has to be available and the deployment is fixed at compile time.

Addistant [26] and J-Orchestra [27] try to solve this by manipulating Java bytecodes. The first requires a policy file to decide where to partition, while the latter also does offline profiling of the application to aid finding a good partitioning. A similar approach is used in Coign [12] where an application built from Microsoft COM objects is distributed using offline profiling and binary rewriting. Still these systems end up with static partitions, which are not optimized for the mobile use case.

Gu et al. present an adaptive offloading framework that can offload parts of the software at runtime [9]. The goal is to cope with the limited memory capacity of mobile devices. A fuzzy control model is used to trigger offloading. To get runtime information about the software an application execution graph is maintained by extensive monitoring of objects and method calls, which introduces a significant overhead.

A widely used approach to calculate the optimal partitioning, is to represent the software as a weighted graph, and transform the deployment problem into a graph partitioning problem. This way diverse algorithms from graph theory can be used to solve the problem. Stoer and Wagner [21] describe an algorithm to find the minimum cut to divide the graph in two partitions. The Kernighan-Lin heuristic is an iterative procedure that converges to a local optimum [13].

Ou et al [16] and Han et al [11] present graph partitioning algorithms aimed specificly at the problem of partitioning software for mobile devices. The first describes the (k+1) partitioning algorithm that results in one unoffloadable partition and k disjoint offloadable partitions. The latter transforms the graph to a flow network and computes the maximum flow to find the optimal deployment.

To offload parts of the application there has to be a server infrastructure available. Storz found a synergy between pervasive computing and grid computing, introducing the Grid as a platform for ubiquitous applications [22]. Buyya et al. envision Cloud computing as the technology to offer computing services anywhere in the world on demand [2]. Emerging Cloud platforms like Amazon Elastic Compute Cloud (EC2)[4] or OpenNebula [18] will offer us the necessary computing power to enhance the abilities of our mobile devices.

The solution presented in this paper does not modify the original source code nor Java bytecodes. Instead, it uses the extensible and service oriented architecture of OSGi [1] to offload parts of the software. While others offload to reduce memory usage [9] or battery consumption [11], we investigate how to improve performance for CPU intensive applications, while minimizing the needed bandwidth. We collect data from runtime profiling in order to offload without a priori knowledge of the software and to be able to adapt at runtime when the device context changes.

3 Use Case - Augmented Reality Game

As an example use case we present an augmented reality (AR) game. On a head mounted display the player sees the environment captured by a webcam, augmented with virtual items. The user must be able to move freely in the environment and thus all processing is done by a mobile device that is connected wirelessly to a back end server. Besides the images also other sensor information, such as GPS or accelerometers can be used to determine the location of the player. Objects can be recognized from the image stream and trigger virtual objects to be displayed.

In order to do all this processing and still achieve an acceptable framerate it will be necessary to offload some of the processing components to a remote server. However, a pure thin client model will fail because of the high bandwidth requirements to send all the image and sensor data to the server, and the latency that will be introduced between the capturing the environment and displaying the video stream.

4 Smart Client Architecture

Figure 1 presents the architecture of our management framework. On both client and server a Resource Monitor tracks the resource usage of the system (step 1) and a Bundle Monitor gathers information about individual software bundles (step 2 & 3). The Client Agent will call these bundle monitors to get an overview of the current resource usage and to construct a weighted graph of the components. In this weighted graph, different components are represented by nodes and communication between components results in edges between their corresponding nodes. Nodes are weighted with the CPU usage of the components and edge weights reflect the amount of data exchanged.

This graph is then offered to the Graph Cutter (step 4) that will calculate the best graph cut. This is the cut that minimizes the bandwidth, while making sure the CPU usage on the mobile devices does not exceed a predefined threshold. By putting this threshold on 80% we can make sure there is no resource contention on the client. One could also lower this threshold when the objective is for example to extend battery life.

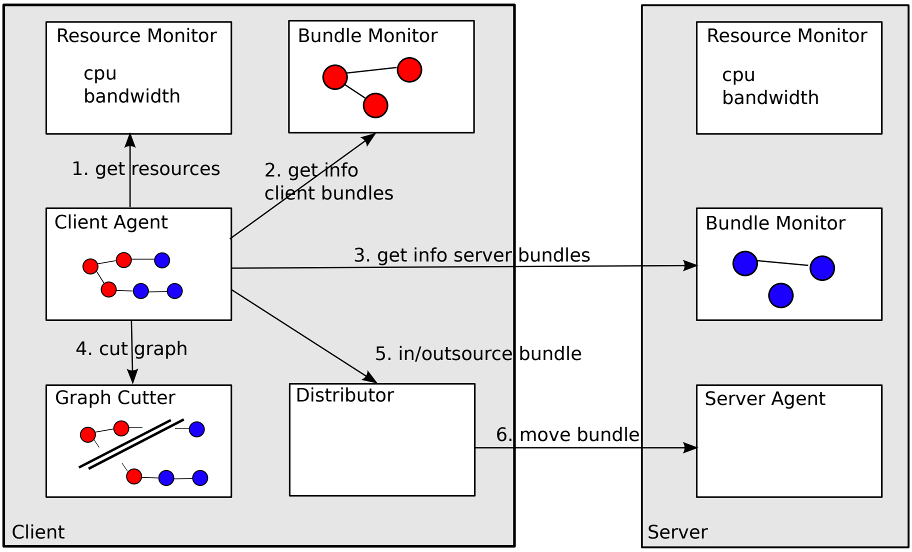


Fig. 1. Smart Client Architecture: The Client Agent gets the resource usage of the system and individual software bundles from the Resource Monitor and Bundle Monitors. The Graph Cutter then calculates the best partitioning after which the Distributor is instructed to in- or outsource some components. The Server Agent takes care of the initialization of bundles at the server side.

Although the Graph Cutter is deployed on the client side in Figure 1, this is not a requirement and the calculation of the best cut can also be offloaded to the server to conserve client resources when the application structure complexity increases. However for small graphs the required CPU time to calculate the best cut is negligible and it isn't worth the communication cost of outsourcing this calculation. Subsequently the Client Agent will instruct the Distributor to migrate some components to or from the server if necessary (step 5). The Server Agent will make sure that the migrated components are started or stopped correctly at the server side (step 6).

The Client Agent performs these actions in a control loop. Periodically it fetches the monitor information and builds up a global graph of the distributed application. It then decides whether to recalculate a better deployment or not.

The Graph Cutter calculates the minimal cut where the sum of the node weights on the client partition cannot exceed a certain threshold. The algorithm used is an extension of the Stoer and Wagner [21] minimum cut algorithm. When the minimum cut found by Stoer and Wagner does not meet the maximum client weight constraint we add the graph together with the found cut to the queue. For each of the edge found in the cut, we investigate the graph with that edge's weight set to infinity. That way this edge will not be in the new solution. If the new solution still does not meet the constraint we add it to the queue. This algorithm will search in a breadth-first manner to find a cut that satisfies the maximum client weight contraint. It is shown in pseudocode below. The *MINCUT* subroutine calls the minimum cut algorithm of Stoer and Wagner and *threshold* represents the maximum client weight contraint.

```
$INF\_CUT(G : Graph, threshold : Number)$
$Graph\ G', G''$
$GraphCut\ result, previous$
$result \leftarrow MINCUT(G)$
if $result.GET\_CLIENT\_WEIGHT \leq threshold$ then
  return $result$
else
  $Queue.ADD(result, cut)$
  while $Queue \neq \emptyset$ do
    $G', previous \leftarrow Queue.POP\_FIRST()$
    for all $edges\ e \in previous$ do
      $G'' \leftarrow G'$
      $G''.SET\_EDGE\_WEIGHT(e, \infty)$
      $result \leftarrow MINCUT(G'')$
      if $result.GET\_CLIENT\_WEIGHT \leq threshold$ then
        return $result$
      end if
      $Queue.ADD(result, G'')$
    end for
  end while
end if
```

5 Implementation Details

In this section we discuss the implementation issues for the architecture components depicted in Figure 1.

5.1 Core

A high level view of our implementation is shown in Figure 2. The middleware builds upon OSGi [1], a popular module management API. We adapted the OSGi framework to get runtime information about method calls between software bundles. This allows for fine grained monitoring necessary for making the right outsourcing decisions.

OSGi adopts a service oriented model in which an application is built from loosely coupled components called bundles. OSGi bundles communicate through services, which are Java classes published under a service interface in a central service registry. Through this service registry bundles look up services they want to use. OSGi provides a light-weight component model that is well suited for use on mobile devices. We use the OSGi bundle as a unit of deployment that can be deployed either at the client or at the server.

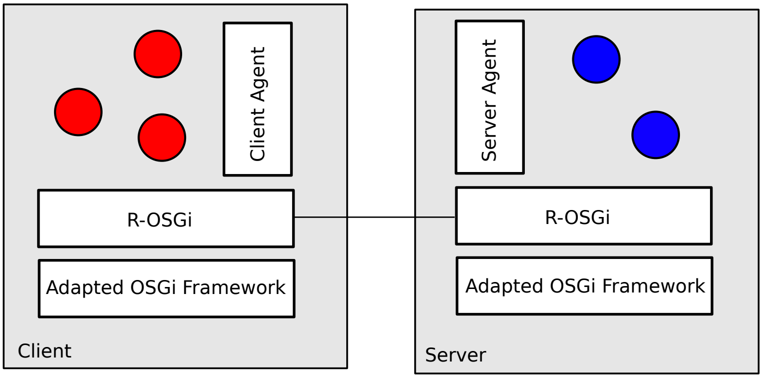


Fig. 2. The client and server run R-OSGi upon an adapted OSGi framework. The agents on both machines monitor the resources and deploys the application bundles (represented by the circles) according to the optimal partition.

On top of OSGi we use R-OSGi [19], which extends the OSGi paradigm for distributed systems. R-OSGi manages interaction between bundles located on different devices by maintaining its own service registry for services that are remotely available. When a bundle requests such a remote service, R-OSGi will generate a local proxy bundle that exposes that service interface locally. When this proxy bundle is called, R-OSGi initiates a remote invocation to forward the method call to the original bundle. This is the core for our management bundles at the client and the server. These management bundles will gather the information about the different running application components, build up the weighted graph, calculate the best graph cut and migrate bundles from the client to the server or vice versa.

5.2 Resource Monitor

The Resource Monitor tracks the resource usage of the system over time. At predefined time intervals it fetches the used memory, the percentage of CPU usage and the number of bytes sent and received over the network.

To get this information this component has to interact with the operating system which necessitates a platform-dependent solution. We implemented it by reading the */proc/* filesystem on Linux based machines. Alternatively, one could also use native bindings through JNI to interface with the underlying operating system.

5.3 Bundle Monitor

The Bundle Monitor monitors bundle-specific information to estimate the node and edge weights for the application graph. Each time interval we calculate the percentage of CPU time used by each bundle and the amount of data exchanged between all bundles in order to be able to assign graph node weights and edge weights respectively.

To gather this information we adapted the Felix OSGi implementation [5] to intercept all calls between bundles. Instead of registering a service object bound to a certain service interface, we create a dynamic proxy for this interface and register this proxy as service object. The proxy will then send events to *MonitoredCallListeners* when a method is called and then forward the call to the original service object. The event will notify a listener of the method called, the thread in which the method was called and the arguments used or the object returned. Our Bundle Monitor listens to these events and calculates the size of the data exchanged as if it would have been serialized. This represents the bandwidth cost of an edge if a bundle would be outsourced.

The *java.lang.management.ThreadMXBean* is used to calculate the CPU usage, which exposes an interface to the JVM and gives us the CPU usage of each thread. For each thread we keep a bundle call stack. When we receive an event of a bundle method call, we push this bundle on the thread's stack, and when we receive an event of a bundle method return, we pop it off the stack. Thus we have to account the execution time of the thread to the bundle that was on top of the stack on that moment. However, we still have to find the bundle that started a thread, since that does not necessarily involve a bundle method call. We do this using the fact that every bundle in OSGi has its own classloader. By matching the classloader that loaded the Thread object to the bundle classloaders we can identify the root bundle of each thread call stack. That way we can map the execution time in a thread to execution time in a bundle. The accuracy of the measurements is dependend on JVM implementation and of course the underlying operating system. Experiments show that we reach an accuracy of tens of milliseconds.

5.4 Client Agent

The Client Agent fetches the information of all monitor bundles periodically and uses it to build up a weighted graph of the application. When the CPU usage exceeds the defined threshold it will request the Graph Cutter to calculate a better graph cut and migrate the necessary bundles.

5.5 Graph Cutter

The Graph Cutter implements the algorithm discussed in Section 4. We implemented two optimizations to the algorithm. As queue we implemented a priority queue that is sorted on the client weight of the cut. This ensures that cuts are processed in order of increasing cut weight.

A second optimization is pruning of the search tree by keeping track of the graphs that already have been partitioned by the minimum cut algorithm. This prevents that equivalent graphs (i.e. with the same edge configuration) are visited more than once by $MINCUT$.

5.6 Distributor and Server Agent

The Distributor and Server Agent will handle the migration of the bundles. At this moment only the migration of stateless software bundles is supported. Stateful migration would mean that a bundle has to be able to serialize its state on the client and restore this state on the server side. Also state changes at the client during the migration have to be propagated to the server. This introduces many difficulties and is considered as future work.

When a bundle is outsourced from the client to the server:

- The Distributor will send the .jar file to the server.
- The Server Agent generates proxy bundles of the services used by the migrated bundle that are only available on the client.
- The Server Agent installs and starts the migrated bundle and makes its services remotely available.
- The Distributor generates proxy bundles of the migrated bundle.
- The Distributor uninstalls the local version of the migrated bundle.

The R-OSGi bundle takes care of the proxy generation, remote invocation and remote service lookup.

Some optimizations can be done to this system by also considering duplication instead of migration. The server could for example keep a bundle cached when it is moved back to the client. When later the client wants to outsource it again to the server this would cut back the migration cost.

6 Prototype Evaluation Results

6.1 Evaluation Setup

We evaluate our framework on a Nokia N900 mobile device with a 600 MHz ARM Cortex A8 processor and 256 MB RAM. This device runs Maemo 5 Linux

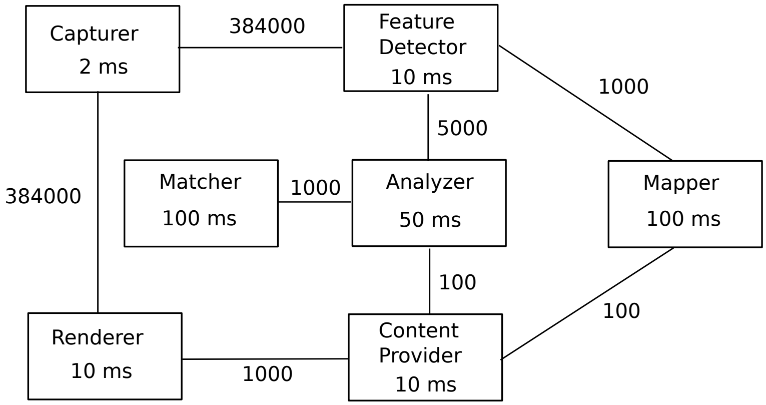


Fig. 3. The architecture and cost graph of our AR application. Different components are annotated with the time it takes to execute a method call and the edges are annotated with the size of the method's arguments in bytes.

as operating system and we used Sun Java SE for Embedded 6 JVM [6]. It also has a camera capable of video recording at a resolution of 800x480. The server machine is equipped with an Intel Core 2 DUO P8400 CPU clocked at 2.26GHz and runs Ubuntu Linux.

To illustrate the operation of the proposed solution, we created a dummy AR application based upon the use case discussed in Section 3. Figure 3 presents the architecture of this application.

The application consists of three concurrent threads. A first thread starts with the Capturer that simulates the fetching of 800x480 images of the camera. It pushes these frames to the FeatureDector and the Renderer. The Renderer will then request the ContentProvider for virtual content and renders it together with the image from the camera on the display to create the augmented reality effect. A second thread starts with the FeatureDetector. This thread grabs the latest image pushed by the Capturer and does a rough first detection step. It then sends image parts to the Analyzer which will analyze it further and generate a pattern characterizing the properties of the image part. This will be matched against a list of known patterns to recognize certain objects by the Matcher. When an object is found it will be notified to the ContentProvider to activate some virtual content. A third and final thread is started by the Mapper component. This thread gets feature points as input of the FeatureDetector and uses an iterative optimization algorithm to estimate the position of the camera in the 3D space. This data is used by the ContentProvider to estimate a correct pose for the virtual content. The more steps the Mapper can perform, the better the estimation of the pose will be.

We implemented this application as stub components that generate a predefined CPU load. We estimated the time to render or to detect some feature rather small (10 ms), and identified some more CPU intensive actions such as the analyzing of an image patch (50 ms), the matching of a pattern (100 ms) or an iteration in the Mapper thread (100 ms). Moreover the Mapper thread will be greedy and try to fill up all remaining CPU time to get an accurate solution of many iterations. We also estimated the size of the data exchanged between the components. In the example the Capturer will push 800x480 uncompressed grayscale images, while the FeatureDetector will only send cropped parts of 5000 bytes to the Analyzer and some feature points totalling 1000 bytes to the Mapper. The other edges are estimated in a similar way.

We started this application on the mobile device and measured CPU and bandwidth usage of the system. As monitor interval we used one second. We also recorded the number of calls per second to the Renderer, which would reflect the frames per second shown in a real application. After one minute, we activated the Client Agent with a threshold of 80% for the CPU load.

6.2 Experimental Results

Beforehand one can easily see that it will be difficult to get good performance of this application by running it on the mobile device. Depending on the thread scheduling strategy of the JVM, we expect a low framerate, a low rate of analyzing features or a low accuracy of the Mapper as not all threads can be active all the time. It's also clear that a thin client approach would introduce an image stream to the server as input and an augmented image stream back to the client as output and thus would have too high bandwidth requirements.

The resulting graph of the CPU usage is presented in Figure 4. Four phases are marked in the figure. The first minute the Client Agent is inactive and the usage on the client is 100%. In the second phase the Client Agent calculates a

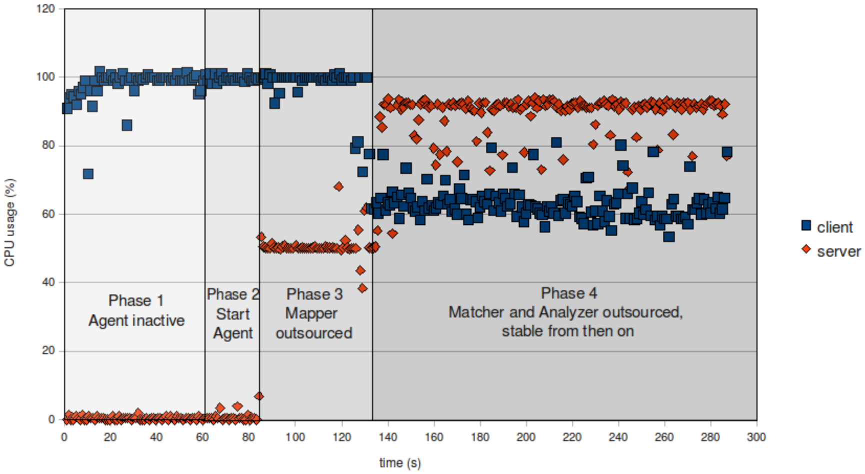


Fig. 4. CPU usage over time. After one minute the Client Agent is activated and starts offloading components to the server until the CPU usage on the client is below the threshold of 80%.

new deployment and decides to outsource the Mapper component. When this outsourcing is complete we see in in the third phase that the CPU usage on the server rises to 50%. This is because the Mapper tries to do as many iterations as possible and thus it uses a complete core of the dual core processor of the server. However, the CPU usage on the client remains 100% so there is still resource contention. The Client Agent will recalculate again and decides to outsource two more components: the Matcher and the Analyzer. When these are outsourced we see in the fourth and final phase that the resource usage on the server rises even more to 90% , but more important, the CPU usage of the client lowers to 60%. Now the CPU usage of the client is below the defined threshold of 80% and the Client Agent will not attempt to outsource any more components.

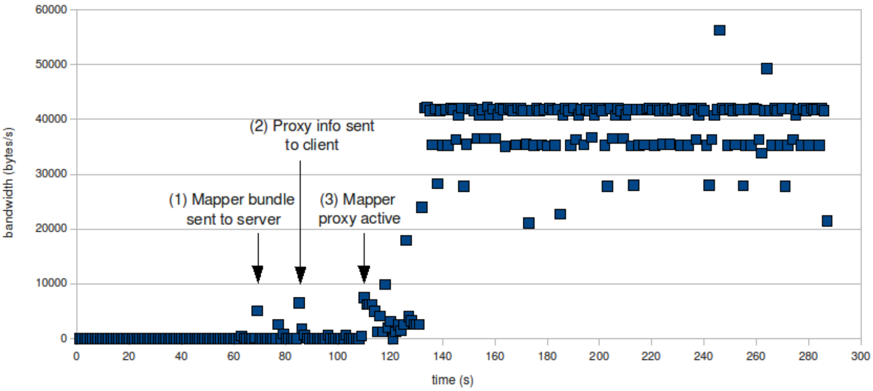


Fig. 5. Bandwidth usage over time. Peaks around 69, 85 and 120 seconds show the outsourcing process of the bundles. The more bundles are outsourced, the more bandwidth is used for the remote method calls.

Figure 5 shows us the bandwidth usage over time. The first 60 seconds there is no bandwidth usage as the Client Agent is inactive. When the Client Agent is activated, there is a peak of around 5 kilobytes after 69 seconds, which is the size of the Mapper jar file that is sent to the server (1).

After 80 seconds the Mapper bundle is started at the server and the the client fetches proxy information from the server, which explains the second peak at 85 seconds (2). However, after the migration the bandwidth drops to zero until 110 seconds (3). This shows that although the Mapper bundle is allready started at the server, the proxy bundle has to be generated and started at the client. Because of the resource contention at the client side it takes a while before the proxy is ready to use.

One also notices the 2 peaks around 120 seconds that represent the sending and receiving of the Matcher and Analyzer bundles and their proxies respectively. After that the communication between the client and server bundles uses about 40 kilobytes per second. Note that this much smaller then the bandwidth required to send the whole video stream in a thin client configuration.

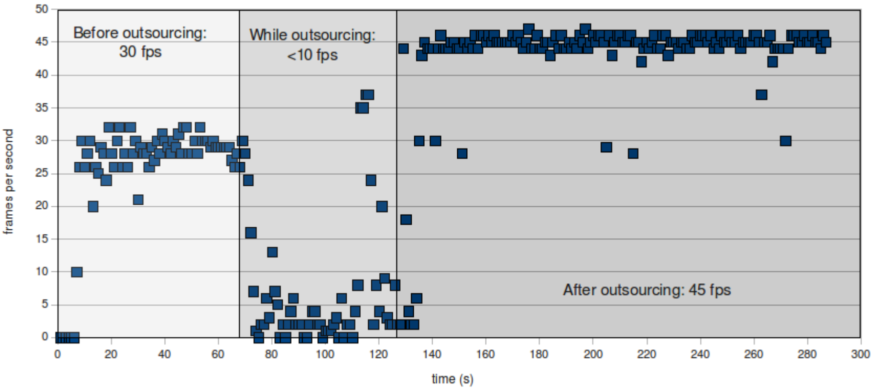


Fig. 6. Frames per second over time. During the outsourcing of the bundles there is a drop in performance. However, when the outsourcing is done there is a gain of 15 frames per second.

Figure 6 presents the frames that would be rendered per second, measured by counting the calls to the Renderer each second. The outsourcing of bundles causes a temporary drop in the performance, because the CPU is used for generating and starting the proxy bundles. However, after the outsourcing of the three bundles the system stabilizes and we see a gain of 15 frames per second while the mobile device only uses 60 % CPU.

Note that a component configured to use 50 ms CPU time will use 50 ms whether it is on the client or the server. Thus, in case of a real application there would also be a performance gain due to the higher clock frequence of the processor at the server side.

Of course the monitoring of the bundles also introduces a certain overhead. However, in our experiments the BundleMonitor never uses more than 2% CPU, which is better than the 7% stated in [9], where a more fine grained monitoring solution is used. To get more detailed data about the monitoring overhead we conducted the following experiment. We used a dummy application of two components that execute 1000 method calls. We ran this application on the mobile device using both the unmodified Felix OSGi framework and our modified framework performing monitoring. By comparing the execution time of the application in the two configurations we can estimate the overhead per method call. We find values between 20 and 40 ns overhead per method call. This shows a very small overhead, knowing that only method calls between different bundles are monitored.

7 Conclusion and Future Work

In this paper we presented an offloading framework for mobile devices that partitions an application and outsources components to a remote server. Built upon the OSGi framework it uses runtime monitoring information to decide which bundles should be outsourced. By calculating the best partition that restricts the client's CPU usage and minimizes the bandwidth the framework converges to the best deployment of the application, without a priori knowledge. Experimental results with a relevant use case of augmented reality show the effectiveness of our approach, while the overhead introduced by monitoring is small.

Future work consists of optimizing the framework in order to make the impact of migrating bundles as small as possible, for example by caching or pro-active generation the of proxy bundles. In the future we also want to migrate the state of components and deal with fault-tolerance in case of network failures.

References

1. The OSGi Alliance. OSGi Service Platform, Core Specification, Release 4, Version 4.2. aQute (September 2009)
2. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J., Brandic, I.: Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems 25(6), 599–616 (2009)
3. Dahm, M.: Doorastha – a step towards distribution transparency. In: JIT (2000)
4. Amazon elastic compute cloud (EC2), http://www.amazon.com/ec2/
5. Apache Felix, http://felix.apache.org/site/index.html
6. Sun Java SE for Embedded 6, http://java.sun.com/javase/embedded/index.jsp
7. Fuad, M.M., Oudshoorn, M.J.: Adjava: automatic distribution of java applications. In: Proceedings of the Twenty-Fifth Australasian Conference on Computer Science, ACSC 2002, pp. 65–75. Australian Computer Society, Inc., Australia (2002)
8. Object Management Group. Common object request broker architecture: Core specification, http://www.corba.org
9. Gu, X., Messer, A., Greenberg, I., Milojicic, D., Nahrstedt, K.: Adaptive offloading for pervasive computing. IEEE Pervasive Computing 3(3), 66–73 (2004)
10. Gui, N., Sun, H., De Florio, V., Blondia, C.: Accada: A framework for continuous context-aware deployment and adaptation. In: Guerraoui, R., Petit, F. (eds.) SSS 2009. LNCS, vol. 5873, pp. 325–340. Springer, Heidelberg (2009)
11. Han, S., Zhang, S., Cao, J., Wen, Y., Zhang, Y.: A resource aware software partitioning algorithm based on mobility constraints in pervasive grid environments. Future Gener. Comput. Syst. 24(6), 512–529 (2008)
12. Hunt, G.C., Scott, M.L.: The coign automatic distributed partitioning system. In: Proceedings of the Third Symposium on Operating Systems Design and Implementation, OSDI 1999, Berkeley, CA, USA, pp. 187–200. USENIX Association (1999)
13. Kernighan, B.W., Lin, S.: An efficient heuristic procedure for partitioning graphs. Bell System Technical Journal 49(2), 291–307 (1970)
14. Klein, G., Murray, D.: Parallel tracking and mapping on a camera phone. In: Proc. Eigth IEEE and ACM International Symposium on Mixed and Augmented Reality ISMAR 2009, Orlando (October 2009)
15. Lai, A.M., Nieh, J.: On the performance of wide-area thin-client computing. ACM Trans. Comput. Syst. 24(2), 175–209 (2006)
16. Ou, S., Yang, K., Zhang, J.: An effective offloading middleware for pervasive services on mobile devices. Pervasive Mob. Comput. 3(4), 362–385 (2007)
17. Philippsen, M., Zenger, M.: Javaparty – transparent remote objects in java. Concurrency: Practice and Experience 9(11), 1225–1242 (1997)

18. OpenNebula Project, http://www.opennebula.org/
19. Rellermeyer, J.S., Alonso, G., Roscoe, T.: R-osgi: distributed applications through software modularization. In: Cerqueira, R., Campbell, R.H. (eds.) Middleware 2007. LNCS, vol. 4834, pp. 1–20. Springer, Heidelberg (2007)
20. Java RMI, http://java.sun.com/javase/technologies/core/basic/rmi/index.jsp
21. Stoer, M., Wagner, F.: A simple min-cut algorithm. J. ACM 44(4), 585–591 (1997)
22. Storz, O., Friday, A., Davies, N.: Towards 'Ubiquitous' ubiquitous computing: an alliance with the grid. In: First Workshop on System Support for Ubiquitous Computing Workshop (Ubisys 2003) in association with Fifth International Conference on Ubiquitous Computing. Citeseer, Washington (2003)
23. Sun, Y., Tay, T.T.: Analysis and reduction of data spikes in thin client computing. J. Parallel Distrib. Comput. 68(11), 1463–1472 (2008)
24. Sun Ray Sun Microsystems, http://www.sun.com/sunray
25. Citrix Systems, www.citrix.com
26. Tatsubori, M., Sasaki, T., Chiba, S., Itano, K.: A bytecode translator for distributed execution of "legacy" java software. In: Knudsen, J.L. (ed.) ECOOP 2001. LNCS, vol. 2072, pp. 236–255. Springer, Heidelberg (2001)
27. Tilevich, E., Smaragdakis, Y.: J-orchestra: Enhancing java programs with distribution capabilities. ACM Trans. Softw. Eng. Methodol. 19(1), 1–40 (2009)

Addressing Challenges with Augmented Reality Applications on Smartphones

J. Benjamin Gotow, Krzysztof Zienkiewicz, Jules White,
and Douglas C. Schmidt

Vanderbilt University, Nashville, TN USA
{ben.gotow,krzysztof.k.zienkiewicz,j.white,d.schmidt}@vanderbilt.edu

Abstract. The popularity of smartphones equipped with GPS and geomagnetic sensors has spurred mobile application developer interest in augmented reality (AR), which presents highly contexualized, spatially relevant information that enhances user knowledge of their immediate surroundings. AR applications typically mesh relevant information with user views of the physical world. Prior research has focused on interfaces built with custom hardware, but a smartphone equipped with GPS, a camera, and a geomagnetic sensor is an attractive alternative to traditional solutions. These devices can be programmed to present context-sensitive information to users without needing custom hardware.

This paper examines three key challenges facing AR developers on mobile devices and presents solutions applicable to modern mobile platforms, such as Apple's iPhone and Google Android-based smartphones. First, we investigate methods of filtering raw sensor data and present an algorithm that eliminates sensor noise. Second, we explore the process of implementing a "magic lens" interaction metaphor by overlaying perspective-rendered graphics on the device's camera using OpenGL and UIKit. Third, we provide an efficient technique for fetching and caching geographically tagged points of interest from a server.

Keywords: smartphones, mobile devices, magic-lens, augmented reality, geomagnetic sensor filtering.

1 Introduction

Augmented reality (AR) overlays highly contexualized, spatially relevant information on user views of the physical world [1,2]. In a typical mobile AR application, users point their smartphones at objects of interest and view the augmented display that is drawn on the phone's screen. The display provides additional information about their environment, *e.g.*, to make them aware of information that is not immediately visible, such as the dates of upcoming events or ratings of nearby restaurants.

AR has been used to create mixed reality video games for use in education [3,4] and handheld tools for underground infrastructure visualization [5]. It has also made inroads in the medical domain. For example, AR has been used to give

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 129–143, 2010.

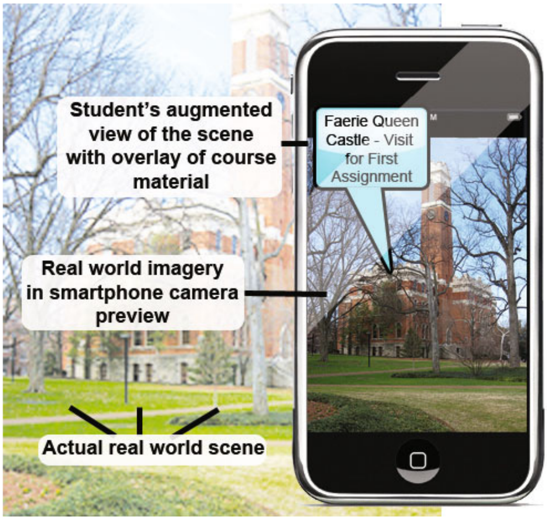

Student's augmented view of the scene with overlay of course material
Faerie Queen Castle - Visit for First Assignment
Real world imagery in smartphone camera preview
Actual real world scene

Fig. 1. An AR Application Overlaying Labels on Real-world Objects

surgeons information about the position of internal organs and the adjustments needed for needle biopsy [2].

Meshing content onto users' views of their environments (*e.g.*, as shown in Figure 1) is a fundamental challenge of AR, requiring methods for determining user locations and estimating the area within their field of view. In applications where environments contain known identifiable markers (such as 2D barcodes) image analysis of these markers can be used to infer the camera's position and frame of reference. Markers are typically designed for ease of recognition and planted in fixed locations within the environment. This class of solutions for pre-prepared environments has been studied extensively [6,1,7,8].

In open environments that have not been previously instrumented with markers, onboard sensors or natural feature recognition can be used. Identifying naturally occurring features of an environment and inferring the user's location is computationally intensive, however, and traditional solutions have been relegated to research labs due to the high cost of the custom hardware required [1]. Today's smartphones are an attractive alternative to custom hardware since they are equipped with Internet access, cameras, and GPS and geomagnetic sensors. The prevalence of smartphones—combined with the ease with which new software can be delivered—makes them a promising platform for building AR applications and conducting future research.

Due to data caching and processing power requirements, natural feature recognition is beyond modern mobile device capabilities. Approaches that centralize data processing [9] are undesirable for consumer mobile applications due to the high cost of scaling server-based solutions and the relatively low-bandwidth networks connecting mobile devices to servers. Several applications perform detection and pose estimation of 2D barcodes and fiducial markers [10,7], which can be used in AR applications that display special content on objects branded with 2D markers. Marker tracking, however, is a specialized use-case not suited for general-purpose, open environment applications, such as providing nearby restaurant reviews or information about events in a city.

GPS and geomagnetic sensors in modern smartphones require significantly less processing power and can work in open environments lacking custom markers needed for image analysis. The use of GPS and geomagnetic sensors in commodity smartphones are, however, accompanied by significant challenges [9], such as the limited accuracy of the GPS sensors in and the noise present in sensor data. For example, the noise in geomagnetic heading values can cause jitter in onscreen information presentation.

The paper provides the following contributions to R&D on mobile AR:

– We present an algorithm that filters sensor data in real time, eliminating noise and allowing for a smooth display based on GPS and geomagnetic sensor data alone. We show that limited processing speeds are not a barrier to filtering sensor data necessary to create smooth AR displays.
– We show that a large number of geographically tagged data points can be stored on a central server, retrieved, and cached using geographic ranges without the processing required to compare latitude/longitude values under the bandwidth constraints typical of modern commodity smartphone hardware.

The remainder of this paper is organized as follows: Section 2 examines key challenges facing developers of AR applications on modern smartphone devices; Section 3 presents solutions to these challenges based on our *Vanderbilt AR Toolkit* (VART), including an approach for efficiently storing and retrieving geotagged data, a filter for effectively reducing geomagnetic sensor noise, and methods for creating perspective rendered overlays in real-time; Section 4 evaluates the benefits of our solutions empirically by analyzing database query speed for point-of-interest retrieval and quantifing the benefits of our sensor filtering algorithm; Section 5 compares VART with related work; and Section 6 presents concluding remarks.

2 Challenges of Mobile AR Application Development

This section presents four key challenges facing developers of AR applications for modern smartphone platforms.

2.1 Challenge 1: Mobile 3D Solutions are Non-optimal and Hard to Mesh with Camera Imagery

The magic lens interaction metaphor [11,12] (where widgets are placed above content to reveal hidden information) is common in AR applications, but is hard to produce on resource-constrained smartphones. Information displayed over the camera preview must be transformed and rendered in real-time according to information about the user's position, orientation, and heading within the environment. Rendering accuracy is important since AR applications offer a rich user experience by precisely associating overlaid information with elements in user surroundings.

Overlaying information directly on top of physical objects obviates the need for context in the information displayed and results in more intuitive data presentation. User experience thus deteriorates quickly when accuracy is lost. Incorrectly aligned overlays provide misleading information because the context assumed by the user is not accurate. Previous AR applications have achieved fast rendering using OpenGL or by moving processing to a server and streaming video to embedded devices [13].

Graphics libraries (such as OpenGL) are available on modern smartphone platforms and can render three dimensional models in real-time. On most devices, pixel fragment processing is done on dedicated graphics hardware, so rendering does not block other CPU-intensive operations, such as the loading of points of interest (POI). The use of OpenGL on smartphone platforms introduces other challenges, however, *e.g.*, rendering content and displaying it over live camera video requires integrating low-level services provided in mobile OS APIs.

Using OpenGL to display interface elements is also undesirable on modern mobile platforms. Once perspective-rendered content is displayed onscreen, it is hard to perform hit testing because OpenGL ES 1.1 does not provide APIs for “picking mode” or “selection” used to determine the geometry at particular screen coordinates. When controls are rendered in a perspective view, it is hard to determine whether touch events lie within the control bounds. While OpenGL supports perspective 3D rendering under the processing constraints typical of modern mobile smartphones, it is not optimal. Section 3.1 describes how VART addresses this challenge with an alternative graphical solution employing nested view objects that display perspective distorted content while preserving user interaction with overlaid visuals.

2.2 Challenge 2: Real-Time Estimation of Frame of Reference is Computationally Demanding

AR requires high-performance techniques for mapping a virtual environment onto the real-world coordinate space. As users move their smartphones, the virtual viewport must update quickly to reflect changes in the camera’s orientation, heading, and perspective, so it is essential to gather information about the device’s physical position in the environment in real-time. Traditional approaches [6,1] to frame of reference estimation depend on identifiable tokens embedded in the environment or computationally-intensive image processing of natural markers.

Image processing techniques must be optimized extensively to fit within the hardware constraints imposed by mobile devices. Detection and frame of reference estimation of identifiable markers (such as two-dimensional barcodes) is an option for closed environments that can be instrumented with such markers. This approach, however, is less suitable for AR applications in outdoor environments since instrumenting the environment with markers prior to the applications use is unlikely.

Attempts to perform natural feature detection in open environments on commodity mobile devices have been largely unsuccessful [9] since they use large amounts of cached data and significant processing power. Devising a strategy for determining the device's position, heading, and orientation with high accuracy is a significant challenge given the limited processing capabilities of mobile devices. Section 3.2 describes how VART addresses this challenge using GPS and geomagnetic sensors for frame of reference estimation.

2.3 Challenge 3: Geomagnetic Sensor Noise Makes Orientation Estimation Hard

Modern mobile smartphones contain a number of sensors that are applicable for AR applications. For example, cameras are ubiquitous and accelerometers and geomagnetic sensors are available in many smartphones. Geomagnetic sensors provide information about user headings, which can be combined with GPS data to estimate field of view.

The geomagnetic sensors in popular mobile devices present unique problems, however, since they do not provide highly accurate readings. To map the virtual AR environment into a real-world coordinate space, sensor data must be accurate and free of noise that causes jitter in rendered overlays. The reduction of noise thus represents a significant challenge confronting AR software.

The Savitzky-Golay smoothing filter [14] is a natural approach to removing sensor noise. This filter leverages the fact that data from most types of rotations can be modeled by a fairly small number of standard equations. Different regression tests can thus be run iteratively on a portion of the most recent data to identify the regression with the highest coefficient of determination and use the resulting equation to adjust the incoming point. Unfortunately, the Savitzky-Golay smoothing filter is not usable in mobile AR application since running a single regression algorithm is expensive and doing it multiple times for a single incoming point at 40 Hertz is infeasible. Section 3.2 describes how VART addresses this challenge via an algorithm that efficiently filters sensor noise within the processing constraints of modern mobile smartphones.

2.4 Challenge 4: Filtering Geotagged POIs by Proximity is Computationally Intensive

Mobile AR applications focus on providing information about immediate user vicinity. In areas of high information density (such as a city) there may be a dozen POI within a few hundred feet of a user. Efficiently storing a large number of geotagged points and retrieving those most relevant to individual users is hard due to the large number of comparisons necessary to identify which item(s) are near user(s). Geotagged points change frequently, so mobile devices need to query a central database server regularly to retrieve information about nearby POI.

Unfortunately, there are several problems with this straightforward approach. Querying a database of geotagged points by specifying latitude/longitude ranges is not practical for mobile applications with many users. It is inefficient to place bounds on two numeric columns in a large data set because comparisons must be performed on each row to compile the result. Databases index content for faster retrieval, but numerical values cannot be efficiently preprocessed for faster comparison. While speed problems could be mitigated by subdividing points into separate tables based on geographic region, a popular AR application might offer thousands of POI within a small geographic area. A different approach is thus required to obviate the need for complex database queries involving numerical ranges.

Requesting and retrieving data on a mobile smartphone is also problematic for several reasons. WiFi and cell network connectivity consumes battery rapidly and users may observe rendering interruptions or a drop in frame rate as data from remote servers is received and processed. Caching data on the mobile device partially alleviates the need for network retrieval. This approach is also problematic, however, since it is hard to aggregate geotagged points and filter them in a latitude/longitude window with limited processing power. Section 3.3 describes how VART addresses this challenge by quantizing geotagged points into geographic blocks and fetching, caching, and filtering on the block level, which consumes less processing as users navigate their environment.

3 The Vandy AR Toolkit

This section describes our solutions to the challenges presented in Section 2 based on the *Vanderbilt AR Toolkit* (VART) for iPhone and Android smartphone platforms.[1]

3.1 Using Hardware Accelerated 3D APIs to Display Perspective Rendered Content

Section 2.1 describes that meshing perspective rendered graphics onto smartphones is hard due to limited control over their camera image. It is also hard to determine what object in 3D space users are interacting with since screen coordinates do not map directly to coordinates in the 3D environment once a projection has been applied. Hardware-accelerated rendering can be achieved using the OpenGL graphics library on some mobile platforms but fails to adequately address the challenge in Section 2.1. Below we present an alternative solution based on nested view objects that display perspective distorted content while preserving user interaction with overlaid visuals.

An alternate approach utilizing nested views. To easily enable user interaction with rendered content, VART employs nested view objects to which a 4x4 visual transformation matrix is applied. When the view hierarchy is rendered, the transformation matrix is applied to each view allowing for basic perspective distortion of the content rendered in each view. The benefits of this approach are that hit testing can be achieved by applying the transformation matrix to incoming touch locations and platform-standard view objects allow the display of standard graphical interface elements.

[1] VART is open-source software available at `code.google.com/p/vuphone`

We use Apple's UIKit framework to implement this solution on the iPhone. UIKit provides sophisticated APIs for building graphical user interfaces composed of nested views. Each view has bounds declared relative to its parent and draws itself. All views may contain subviews; interaction events proceed down a call chain to the lowest view capable of handling an event of that type.

UIKit also allows an AR application to specify a 4x4 visual transformation matrix for each view, which supports basic perspective graphics. The transformation matrix is applied to graphics output when each view draws its respective content and is also applied to user interaction events as they are passed into the view stack. Since the transforms are applied to events, hit testing is handled transparently regardless of the transformation matrix.

We created a transformation matrix approximating distortion from a camera lens and used it to render buttons and other controls with a perspective projection applied. This solution obviates the need for other graphics libraries, such as OpenGL. It also enables user interaction with rendered content, which is important for mobile AR applications.

Meshing content with the camera image. Meshing rendered content with imagery from the smartphone camera required overcoming platform-specific issues on the iPhone and Android platforms. For example, restrictions built into Apple's 3.0 iPhone OS prevent camera image data from being used in the graphics pipeline. Since direct access to the camera image in memory is not provided, however, it is not possible to use OpenGL or another graphics library to display the camera image and the rendered POI elements.

Although Apple provides an API to take individual frames from the phone's camera, this approach yields low frame rates unsuitable for an AR application. Using a single graphics pipeline to draw the camera image and the overlaid content seemed attractive since images can be distorted and adjusted prior to their display, but our inability to pull frames from the camera rapidly made this option unappealing. Instead, a generic camera preview can be used to display the camera image on the screen separately, which provides little flexibility since image data cannot be manipulated and its display is beyond the application's control. Transparent content can be displayed in a layer on top of the camera preview, however, achieving the desired effect.

3.2 Using GPS and Geomagnetic Sensors to Estimate of Device Position and Orientation

Section 2.2 identified problems using traditional image recognition techniques for device position and orientation estimation on mobile devices. We now present an alternative made possible by the sophisticated sensors in commodity smartphone hardware. Prior work has focused on using specifically designed markers and token detection for location and frame of reference estimation on mobile phones [2,6,10], which provide accurate position and pose estimation of markers placed in user environments.

Our intended use of AR to display nearby POI does not require highly accurate pose estimation or position information. Instead, geographic location

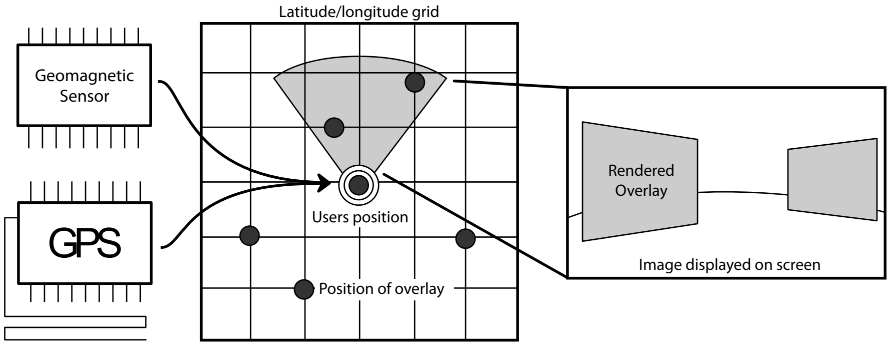


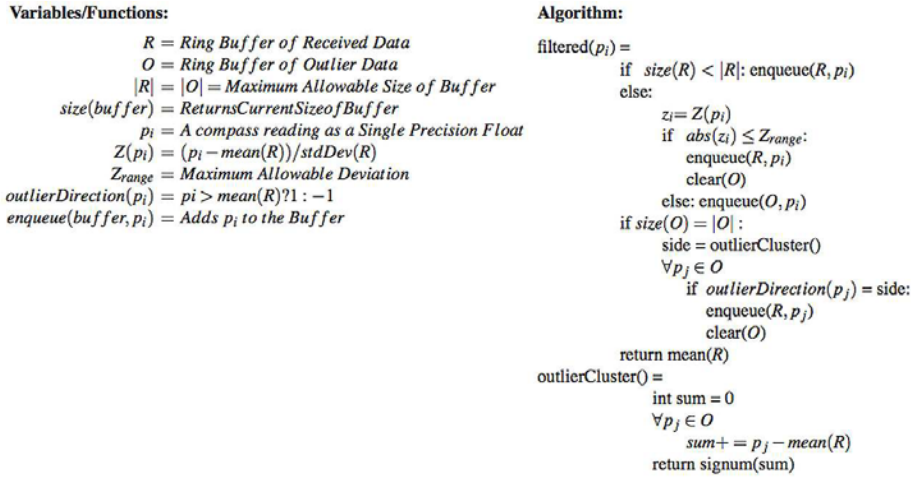
Fig. 2. Sensors Identify Device Frame of Reference and Update Screen

information within a few meters and heading data within a few degrees is acceptable. To avoid computationally expensive image processing and the need for environmental markers, therefore, VART uses onboard GPS and geomagnetic sensors sensors available in modern commodity smartphones to pinpoint the user on a latitude/longitude grid and compute the POI within their field of view. Points are then rendered over a camera image, allowing use of the phone as a lens to view an augmented version of the world, as shown in Figure 2. iPhone and Android devices feature GPS hardware and geomagnetic sensors, and both operating systems provide APIs for accessing data from this hardware within third-party applications.

Prevailing issues: sensor noise and accuracy. We tested Android and iPhone smartphones (such as the iPhone 3G and the Android G1) that are representative of modern mobile phone technology. While these devices offer an impressive range of features, their reliance on commodity sensor hardware is problematic. For example, these smartphones incur a great deal of input noise and have less accurate hardware than traditional mobile AR systems [13]. The geomagnetic sensors of these phones were noisy, even when the phone was lying flat on a table. This variance in heading information yielded visual jitter and degraded the presentation of POI on-screen. The GPS sensors in these smartphones also provided less accurate readings than dedicated AR devices, which often utilize GNSS surveying equipment with accuracy to one centimeter.

Although 100% accuracy is not required, reasonable accuracy is helpful in AR applications that overlay 3D geometry to match real-world features. These applications often render geometry on top of a view of the phone's camera. When user location and heading cannot established with high accuracy, overlaid geometry may be misaligned.

Possible approaches to filtering sensor noise. We identified several algorithms to optimally filter incoming heading data. Ideally, an algorithm should work in the following conditions: (1) device is held steady, (2) device is rotated at a uniform speed, and (3) device is moved semi-randomly. Pattern recognition algorithms run on the most recent data would select the correct filter. Unfortunately, such algorithms require computing power beyond the capacity of modern smartphones.



Variables/Functions:

$$R = Ring\ Buffer\ of\ Received\ Data$$
$$O = Ring\ Buffer\ of\ Outlier\ Data$$
$$|R| = |O| = Maximum\ Allowable\ Size\ of\ Buffer$$
$$size(buffer) = ReturnsCurrentSizeofBuffer$$
$$p_i = A\ compass\ reading\ as\ a\ Single\ Precision\ Float$$
$$Z(p_i) = (p_i - mean(R))/stdDev(R)$$
$$Z_{range} = Maximum\ Allowable\ Deviation$$
$$outlierDirection(p_i) = pi > mean(R)?1 : -1$$
$$enqueue(buffer, p_i) = Adds\ p_i\ to\ the\ Buffer$$

Algorithm:

```
filtered($p_i$) =
        if  $size(R) < |R|$: enqueue($R, p_i$)
        else:
            $z_i = Z(p_i)$
            if  $abs(z_i) \le Z_{range}$:
                enqueue($R, p_i$)
                clear($O$)
            else: enqueue($O, p_i$)
        if $size(O) = |O|$ :
            side = outlierCluster()
            $\forall p_j \in O$
                if  $outlierDirection(p_j)$ = side:
                    enqueue($R, p_j$)
                    clear($O$)
        return mean($R$)
outlierCluster() =
            int sum = 0
            $\forall p_j \in O$
                $sum+ = p_j - mean(R)$
            return signum(sum)
```

Fig. 3. The Compass Filtering Algorithm

If we eliminate pattern recognition (which is the most processor intensive part of the above algorithm) another approach emerges: the Savitzky-Golay smoothing filter [14]. Unfortunately, this technique is not usable in an AR applications on smartphones for the reasons discussed in 2.3.

A lightweight and portable solution. The compass filtering algorithm shown in Figure 3 extends Finite Impulse Response filters [15], with added statistical analysis for data exclusion and outlier analysis. It can be customized for different noise levels by a small list of parameters. The filter structure shown in Figure 3 contains two ring buffers of set capacity, one for the recent data and one for outlier data. The filter starts in an uninitialized state, accepting all incoming points and enqueuing them into the data buffer. After we reach capacity, each new point's z-score (which is a statistic for measuring the deviation of a point from the mean of the sample) is calculated. If the z-score is within an acceptable range, we enqueue the corresponding reading into the data buffer and clear the outlier buffer. Otherwise, we enqueue the reading into the outlier buffer.

If the outlier buffer reaches its capacity, we determine the direction of the outliers by computing on which side of the mean the majority of the outliers lie. We then enqueue all of the outliers in this majority to the data buffer, thus flushing it, and clear the outliers buffer. We repeat this process each time a new sensor reading is available. When asked for the filtered value, we return the mean of the data buffer.

Calculating mean and standard deviation are the most computationally expensive operations in our compass filtering algorithm. After the initialization stage, however, these calculations can be optimized to constant time operations by keeping track of the current sum and variation. When a new point comes in, therefore, we only have to remove the old point from the current sum/variation and add in the new one. Our approach lends itself to extension via subclassing so that the filter parameters can vary dynamically.

3.3 A Grid-Based Approach to Data Storage and Retrieval

Section 2.4 described the problems surrounding the storage and retrieval of many POI. Below we present a highly scalable solution to the problem of data retrieval, caching, and filtering on both the server and mobile device using a grid-based approach that progressively loads content from web sources based on GPS coordinates.

A mapping function generates discretized x,y values for each POI based on the latitude/longitude pair such that multiple POI in the same geographic region share the same x,y value. A basic function might round latitude and longitude values, giving all POI in the same lat/lon minute the same x,y pair.

Each block in the x,y grid contains all points within a specific geographical area, and may be loaded by querying the database for the indexed coordinate values. Indexing the contents of the database using discretized values obviates the need for numeric comparison and queries bounded by latitude and longitude values. Queries may specify an exact block index such as (x=1, y=2) and retrieve a group of points within a predefined geographic area.

Dividing available content into a latitude/longitude grid and fetching it in discrete blocks has several advantages. Information can be requested by specifying an index to a particular block within the grid and stored based on grid coordinates, alleviating complex retrieval queries on a central server. Caching retrieved data is also straightforward since data can be stored and retrieved on the device based on the block index. Purging cached data based on its distance from the user's current location does not require iterating through each cached point. Instead, entire blocks can be quickly purged based on their discretized latitude and longitude values.

Dividing content into geographic blocks maps well onto the presentation space, where POI must be displayed/hidden as the user moves toward/away from an area. Blocks may be partitioned into a small geographic size so that a fixed number of blocks are displayed at a time, corresponding to a few miles in each direction. Filtering blocks of points is much more efficient than processing each point and also requires constant evaluation time, regardless of the number of points present in the area. Hiding and showing POI one-by-one can yield poor application performance in high data density areas.

4 Empirical Results

This section presents empirical data that evaluates our techniques and algorithms described in Section 3.

4.1 Evaluating the Compass Filtering Algorithm

Below we present an experiment assessing the efficacy of our VART compass filtering algorithm described in Section 3.2. We sample a typical geomagnetic sensor and demonstrate favorable results produced in real-time.

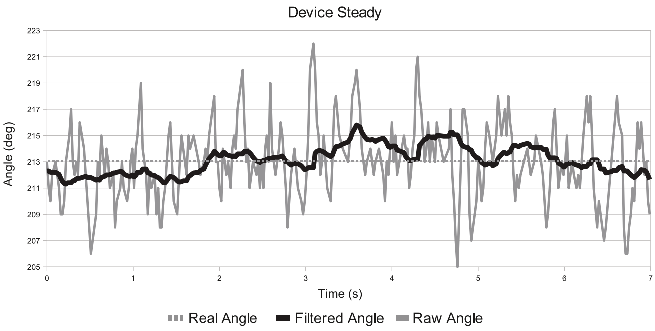


Fig. 4. Noise Observed When the Device is Held Steady

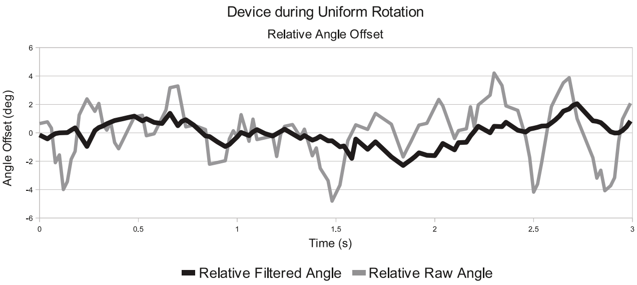


Fig. 5. Relative Angle Offset When the Device is Under Uniform Rotation

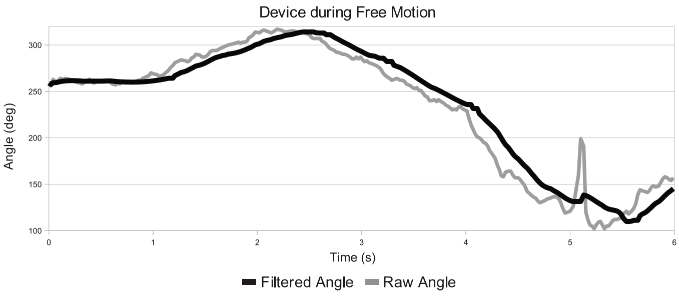


Fig. 6. Noise Observed When the Device is Experiencing Freehand Motion

Experimental setup. We used an Android Dev Phone 1 running Android 1.5 to collect measurements. The sensor was sampled at highest possible rate (roughly 36 Hz). The data was stored to an SD card via a Java application.

Raw sensory data was saved while the device was rotated at a uniform angular velocity on a magnetically insulated rotating mechanism. An adapter was then used to feed this time-stamped data into the filtering application in real-time. The resulting filtered measurements were then recorded and plotted on a time versus angle graph from which noise reduction was then calculated.

Hypothesis. Plotting the raw sensory output provides a general idea of the noise levels to reduce. The noise is most visible when the device is held steady. When it is rotated at a uniform angular velocity, noise becomes almost nonexistent. An effective filter must therefore eliminate the corrupted data while preserving accurate measurements.

Analysis of results. Analysis of our filter on real-life data suggests that we are doing just that. Figures 4, 5, and 6 depict a graphical representation of our filter's performance. In a worst-case scenario (*i.e.*, when the device is held steady) we achieved a 60% noise reduction. When the data is most accurate (*i.e.*, rotation at uniform angular velocity) we still eliminate over half the noise.

The results in Figures 4, 5, and 6 show a significant reduction of sensor noise when our algorithm is employed, even in the worst case scenario of the device lying stationary. Prior to filtering, the geomagnetic sensor was shown to produce values ranging ±4.8° at rest and we reduced the margin of error to ±2°. This reduction is a significant improvement, confirming that our filtering algorithm is effective and reduces overlay jitter observed by the end user of a mobile AR application.

Using data from the geomagnetic sensor of a commodity Android smartphone, we confirm that sensor noise is problematic. Our experiment also demonstrates that this algorithm is within the processing capabilities of modern smartphones. AR applications can therefore be developed to leverage GPS and geomagnetic sensors for frame of reference estimation without significant levels of jitter due to noise in orientation information.

4.2 Evaluating Database Retrieval of Quantized Data Points

Section 3.3 presented an approach to efficiently storing and retrieving data points using a grid based on latitude and longitude values. To test that avoiding numeric comparisons improves retrieval speed, we designed an experiment for measuring the speed of various database queries on a large set of geocoded points.

Experimental setup. All experiments were conducted on an Apple Powerbook with a 2.53 GHz Intel Core 2 Duo processor, 4 gigabyes of 1067MHz DDR3 RAM running OS X version 10.6.2 and mySQL 5. Queries were executed and timed using PHP 5.3.0. The results of each query were not processed, so recorded times indicate time spent performing queries only.

A single table containing 1,000,000 rows was created in a mySQL database on the machine executing the queries. Each row in the table consisted of an integer id, double latitude value, and an integer block id. Blocks were assigned based on the latitude values so that 1000 blocks were evenly filled with 1000 rows each. A standard mySQL index was created on the block id column.

Hypothesis. Storing geotagged points in discrete blocks within the database and retrieving them based on block index is much faster than performing numerical queries that specify upper and lower bounds on latitude, longitude values.

Analysis of results. We ran three types of queries on the database and noted average response time in microseconds for 200 queries, each returning 1,000 matching rows from 1,000,000 rows, as shown in Table 1.

Table 1. Average Query Response Time

Query Type	Response Time
Latitude range (latitude column indexed)	581700 µs
Latitude range (latitude column not indexed)	284600 µs
Specific latitude block	5209 µs

Our results confirmed that querying a large data set of geocoded points based on latitude/longitude values is a performance issue and that our solution presented in Section 3.3 offers dramatic performance benefits by organizing data points into discretized numerical blocks. Retrieving records within a single numerical block was exponentially faster than querying for the equivalent range of latitude values. Since AR applications generally load a number of records at a time, the loss of granularity in queries for discrete blocks is not an issue and this approach will dramatically decrease server load. The poor performance observed when querying for numerical ranges suggests that mobile smartphones should not cache a large number of POI without the optimization described in Section 3.3.

5 Related Work

This section compares our work on smartphone-based AR applications with related work. The techniques employed in this paper are inspired by earlier work in mobile AR, data filtering, and magic lenses. Location and frame of reference estimation is a fundamental issue in AR and has been addressed in two primary ways in recent literature. In applications where environments can be instrumented with easily identifiable markers or contain a limited number of known natural features, image analysis techniques are optimal and provide highly accurate results. This class of solutions has been studied in great detail [6,1,7,8].

Techniques utilizing sensors present a viable alternative in open, unprepared environments and have been presented in [5]. Likewise, [9] presents a hybrid approach using image recognition to refine frame of reference information derived from onboard inertial sensors. Although this approach helps increase accuracy in open uncontrolled environments, future research is needed to reduce the requirement for large amounts of pre-prepared environment data. It is likely that methods for filtering sensor data (such as our solution in 3.2) partially obviate the need to refine sensor data using image analysis.

To cache and retrieve points of interest rapidly, we employ a coordinate quantization technique similar to "loxels" [16], which organized image descriptors used for pose estimation by natural feature recognition into a location-based grid that could be loaded incrementally and provided the inspiration for our method of POI storage. We adapt this approach to store and retrieve geotagged data points and quantify the benefit of querying a database based on discrete blocks instead of numeric latitude and longitude ranges.

Our approach to data smoothing leverages qualities of the Savitzky-Golay filter [14]. It takes advantage of the fact that data from most types of rotations can be modeled by a fairly small number of standard equations. This insight provided the inspiration for our filter for applications where the regression required by Savitzky-Golay filter is not tractable due to constraints on processing power.

Another approach to filtering utilizing a Kalman filter is found in [17]. When multiple sensors such as accelerometers and magnetometers are available, Kalman filters can fuse multiple sensor signals into a single estimate of heading. These filters have been shown to have low calculation times, on the order of 1/10th of a sec. on a 50MHz processor [17], but require multiple sensors for fusion and are thus not well suited for mobile phones.

6 Concluding Remarks

Today's smartphones are promising platforms for AR applications since they are portable, ubiquitous, and provide the processing power and sensor capabilities necessary for AR applications. This paper identified several challenges in developing AR applications for the iPhone and Android platforms and showed how our *Vanderbilt AR Toolkit* (VART) provided acceptable solutions to these challenges. Our work on VART has yielded the following lessons learned:

- **POI retrieval based on numeric geographic ranges is infeasible.** Retrieving geotagged points from a database table within a specific numeric geographic range is costly. Quantizing points into a grid of discrete blocks is a more efficient solution.
- **Discretizing POI locations eliminates costly comparisons.** Discretizing latitude and longitude values allows geotagged points to be indexed and retrieved rapidly from a database.
- **Hit testing in OpenGL is laborious.** Hit testing three dimensional content rendered in OpenGL is hard due to the lack of selection and picking modes in OpenGL ES.
- **Sensor data requires processing to remove noise.** Raw data from smartphone geomagnetic sensors contains significant noise that results in jitter in rendered overlays unless corrected.
- **Existing smartphone platforms are capable of delivering magic lens AR**, but additional work is needed to identify other forms of AR that can be supported, *e.g.*, a hybrid form of AR utilizing onboard sensor data and flucidial marker detection could allow for impressive massively-multiplayer AR games.

References

1. Wagner, D., Pintaric, T., Ledermann, F., Schmalstieg, D.: Towards massively multi-user augmented reality on handheld devices. In: Gellersen, H.-W., Want, R., Schmidt, A. (eds.) PERVASIVE 2005. LNCS, vol. 3468, pp. 208–219. Springer, Heidelberg (2005)

2. Azuma, R.: A survey of augmented reality. Presence 6, 355–385 (1995)
3. Kirkley, S.: Creating next generation blended learning environments using mixed reality, video games and simulations. Tech. Trends 49(3), 42–53 (2004)
4. Huang, J.-Y., Tung, M.-C., Keh, H.-C., Wu, J.-J., Lee, K.-H., Tsai, C.-H.: A 3d campus on the internet — a networked mixed reality environment, pp. 282–298 (2009)
5. Schall, G., Mendez, E., Kruijff, E., Veas, E., Junghanns, S., Reitinger, B., Schmalstieg, D.: Handheld augmented reality for underground infrastructure visualization. Personal Ubiquitous Comput. 13(4), 281–291 (2009)
6. Wagner, D.: Multiple target detection and tracking with guaranteed framerates on mobile phones. In: ISMAR 2009 (2009)
7. Mohring, M., Lessig, C., Bimber, O.: Video see-through ar on consumer cell-phones. In: Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR 2004, pp. 252–253. IEEE Computer Society, Washington (2004)
8. Schmalstieg, D., Wagner, D.: Experiences with handheld augmented reality. In: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2007, pp. 1–13. IEEE Computer Society, Washington (2007)
9. Zhou, Z., Karlekar, J., Hii, D., Schneider, M., Lu, W., Wittkopf, S.: Robust pose estimation for outdoor mixed reality with sensor fusion. In: Proceedings of the 5th International Conference on Universal Access in Human-Computer Interaction. Part III, UAHCI 2009, pp. 281–289. Springer, Heidelberg (2009)
10. Schmalstieg, D., Wagner, D.: Mobile phones as a platform for augmented reality. In: IEEE VR 2008 Workshop on Software Engineering and Architectures for Realtime Interactive Systems, pp. 43–44. Shaker Publishing, NY (2009), `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.144.6950`
11. Bier, E.A., Stone, M.C., Pier, K., Buxton, W., Derose, T.D.: Toolglass and magic lenses: The see-through interface, pp. 73–80. ACM Press, New York (1993)
12. Looser, J.: Ar magic lenses: Addressing the challenge of focus and context in augmented reality. Master's thesis, University of Canterbury (2007)
13. Perritaz, D., Salzmann, C., Gillet, D., Naef, O., Bapst, J., Barras, F., Mugellini, E., Abou Khaled, O.: 6th sense— toward a generic framework for end-to-end adaptive wearable augmented reality, pp. 280–310 (2009)
14. Savitzky, A., Golay, M.J.E.: Smoothing and differentiation of data by simplified least squares procedures. Analytical Chemistry 36(8), 1627–1639 (1964), `http://dx.doi.org/10.1021/ac60214a047`
15. Rabiner, L.R.: Theory and Application of Digital Signal Processing. Prentice-Hall, Inc., Englewood Cliffs (1975)
16. Takacs, G., Chandrasekhar, V., Gelfand, N., Xiong, Y., Chen, W.-C., Bismpigiannis, T., Grzeszczuk, R., Pulli, K., Girod, B.: Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In: Proceeding of the 1st ACM International Conference on Multimedia Information Retrieval, MIR 2008, pp. 427–434. ACM, New York (2008)
17. Harada, T., Mori, T., Sato, T.: Development of a tiny orientation estimation device to operate under motion and magnetic disturbance. Int. J. Rob. Res. 26(6), 547–559 (2007)

Location Cognition for Wireless Systems: Classification with Confidence

Stefan Aust[1], Tetsuya Ito[1], and Peter Davis[2]

[1] NEC Communication Systems, Ltd., Kanagawa, Japan
aust.st@ncos.nec.co.jp, ito.tts@ncos.nec.co.jp
[2] Telecognix Corporation, Kyoto, Japan
davis@telecognix.com

Abstract. Location cognition is a challenging task in cognitive wireless systems when there is no explicit location information system available, such as Global Positioning System (GPS) or dense wireless beacons. This paper decribes a simple-but-effective method of real-time location cognition which can be used by wireless devices in WLAN systems without depending on any location service infrastructure. The method is based on monitoring, learning and recognizing the statistics of received data traffic, with an awareness of the confidence in the recognition result. It uses the property that traffic statistics such as average and variance of throughput are correlated with the location of the transmission. Locations are recognized by comparing monitored statistics with a set of reference distributions and identifying the best match. A measure of the confidence in the location classification result is obtained by comparing matches with multiple candidate locations. It is demonstrated that the method can be implemented as middleware for use with WLAN devices and used to recognize multiple locations, indoor and outdoor. It is also demonstrated that the method can be used to detect the distance between a sender and receiver.

Keywords: monitoring, statistics, fingerprinting, classification, location cognition, confidence, cognitive radio systems.

1 Introduction

Acquiring location information in cognitive wireless systems is a required function in cognitive radio (CR) systems [1]. The FCC has recently regulated that TV white space CRs need to know their geo-location and avoid interfering as secondary user (SU) with the primary user (PU) by considering the coverage area of TV broadcast services. It is standardized using GPS service for positioning with an extension for systems which are not always able to acquire position information [2]. How a CR system should work when there is poor reception of GPS signals, for example at indoor locations, remains open. Indoor as well as outdoor location information is important for many other wireless functions, such as navigation, context-aware applications [3], pervasive computing [4] and movements of users, such as fire fighters [5]. GPS is widely used but can be

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 144–157, 2010.

applied for outdoor applications only. Other methods exist which complement the use of GPS for providing location information for example, using beacons, triangulation or a centralized database (DB) of so-called fingerprints of signals from surrounding access points (APs).

We propose a novel approach for location cognition which can be independent of GPS, APs or any centralized DB. In particular, we propose a novel location cognition method for multiple locations based on monitoring, learning and recognizing the characteristics of data transmission between radio terminals. This is a form of so-called *fingerprinting*, where the statistics which are obtained during an off-line "learning" phase are used to identify a location. Our method includes an estimation of the confidence in the location cognition, which enhances the use of the method in practical scenarios. We show that location cognition can be done with confidence either inside or outside a building, using reference distributions held by each device and without using additional infrastructure. In particular, we show that the method can be implemented as wireless middleware, which we call a Location Cognition Engine (LOC), supporting WLAN IEEE 802.11 standards. We show that it can be used to classify multiple indoor and outdoor locations, and also recognize distances between sender and receiver terminals.

2 Related Work

With the use of the Global Positioning System (GPS) in outdoor environments the location problem can be easily solved. GPS is widely accepted as a useful positioning system and is applied in car navigation systems and military scenarios. However, even GPS or Galileo may have situations where shadowing or interference reduces the accuracy of these services [5]. Work has started that aims to extend GPS/Galileo to improve the positioning in outdoor and indoor environments [6]. The work aims to combine the GPS/Galileo system with other location information sources to fulfill location detection requirements efficiently. Other solutions and methods have been developed for indoor localization, using WLAN, infrared (IR), ultrasound, FM radio, fingerprinting, sensor networks, ultra-wideband (UWB), Bluetooth, magnetic signals, vision analysis and triangulation, etc. Each technology has unique advantages in performing location sensing, but also has some intrinsic limitations. Localization methods which combine one or more of these technologies to increase location performance have also been proposed [5].

Comparing with outdoor, indoor location is usually more difficult due to the complexity of the surrounding physical environment, such as walls and doors, which influence the propagation of electromagnetic waves and can result in complex multi-path effects. A comprehensive study which is given in [7] shows details of indoor positioning systems (IPS) for wireless personal networks, addressing security, privacy, cost, performance, robustness, complexity and limitations. The authors in [7] pointed out that for indoor applications new challenges arise for IPS. IPS considers only indoor environments such as inside a building and extensions for location cognition in outdoor environments are not included. The authors classify four techniques for indoor positioning estimations, triangulation, fingerprinting, proximity and vision analysis, and show that each technique has its limitations, but combinations of positioning systems can significantly improve the quality of position estimates. In [8] an FM indoor positioning system (FINDR) was proposed. The power consumption of the proposed location method uses 15mW for FM transmission whereas for WLAN about 300mW is used which was the motivation for the authors to utilize FM as an energy efficient indoor positioning system. The location method uses the signal-to-noise ratio and the received signal strength of FM transmissions. FM transmitters have to be pre-installed and manually tuned to broadcast-free frequencies. A k-nearest neighbor classifier was applied for position classification. The accuracy of this system was reported to be 4.5m (at 95% confidence).

Fingerprinting methods for indoor positioning are widely applied [9], [10]. Fingerprinting positioning technique uses pre-measured location related data, including two phases, an off-line training phase and an on-line phase. The authors in [11] proposed an enhanced fingerprint-type technique based on trilateration through Received Signal Strength (RSS) values obtained in real-time in indoor locations. The method estimates the propagation models that best fit the propagation environments. The authors conducted a vast amount of measurements to obtain accurate values for their proposed RSS log-normal path-loss model that uses a constant value which depends on averaged fast and slow fading, gains and transmitted power. The authors concluded that this constant is known beforehand and should be valid for different environments. We conclude that none of the proposed location methods provides a solution for a combined location cognition in indoor and outdoor environments.

3 Proposal: Location Cognition Engine (LOC)

We aim to provide a versatile Location Cognition Engine (LOC) which is user friendly and reliable in the cognition of both indoor and outdoor locations. The original purpose of the Location Cognition Engine (LOC) was the use in cognitive wireless systems, to support autonomous selection of wireless channels avoiding interfered or busy channels. However, the LOC could also be integrated in various other kinds of services and applications which require location awareness. In order to achieve versatile, reliable and user-friendly location cognition, we propose a method based on monitoring, learning and recognizing the characteristics of data transmission between radio terminals, combined with estimates of the confidence in the learnt reference data and the result of the location recognition.

We show that a wireless device can recognize its location on-line by comparing the statistical distribution of its data transmissions with a set of previously (i.e., off-line) acquired reference distributions. This is a form of so-called *fingerprinting*, where the statistics which are obtained during an off-line "learning" phase are used to identify a location. In comparison to other fingerprinting methods which use RSS fingerprints of specific positions of multiple indoor WLAN APs to obtain a detailed signal strength map, our proposed location cognition is fully independent of any pre-installed WLAN APs. Another difference is that the fingerprints are based on high-level transmission characteristics, rather than physical layer information such as signal strength or delay. Moreover, we introduce parameters to indicate confidence in the set of fingerprints and the confidence in the best fit.

In [12] we argued that monitored traffic of constant data transmission show characteristic distributions. It was shown that it is possible to monitor data transmissions and obtain statistical quantities, such as mean, variance and standard deviation, which characterize distributions of received data at each different location. In particular, the received number of data packets (we called it *TxCount*) has been used as monitoring parameter. Further, we proposed in [12] the use of the Jeffrey Divergence (JD) with Gaussian approximation to calculate dissimilarities between monitored data and reference data. Figure 1 shows the idea of identifying a location by comparing an input distribution with a set of reference distributions. The reference distribution with the least dissimilarity is selected as the best fitting distribution, which identifies the location. The reference distributions (RD) at different locations were obtained during the off-line phase, including data pre-processing, outlier-filter and removing transition states. The input data is frequently monitored during the on-line phase, for example once-per-second, and statistics of the input data are calculated on-line with a sliding window.

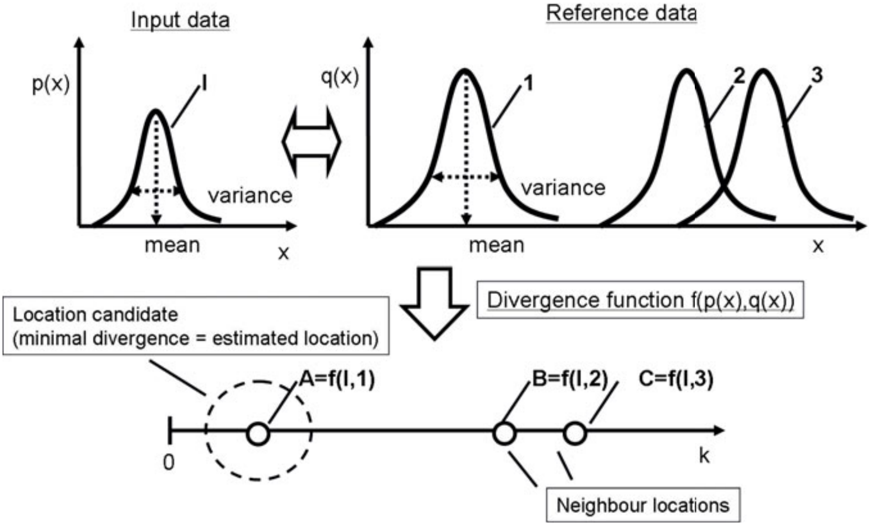


Fig. 1. Details of the proposed entropy-based location cognition method

The location recognition procedure is conducted in two steps. First, the LOC executes the recognition of the generic location. In the second step, the LOC executes the recognition of the location and the distance between transmitter and receiver. One *generic RD set* is used for the estimation of the location and one *complete RD set* including distributions at different distances is used for the estimation of the distance. The generic RD at each location is obtained by selecting the distribution with the maximum static confidence value.

4 Estimation of Location Confidence

In addition to identifying a location by choosing the best fit which we introduced in [12], we introduce new parameters which describe the confidence of the training data sets and the LOC location classification. We contribute two new parameters called the *static confidence* and the *dynamic confidence* which have been developed to classify the confidence in the static set of training data or reference data and the confidence during the on-line classification of the monitored transmission to recognize the location of the wireless terminal. The proposed location method does not require an exhaustive learning at all points in the space. This is not desirable for an ad hoc user. However, the confidence can be used to measure the reliability of location classification over the entire space including intermediate points between the learnt points. The confidence shows the user when the location classification is reliable.

Figure 2 shows details of the implemented confidence estimator. The dynamic confidence is calculated using input data and the reference data in the on-line mode. The result is a time-variant value with a range of [0-100]. A confidence close to 100 indicates a large distance (large dissimilarities) between best fit and other templates, indicating high location confidence. A confidence close to 0 indicates a small distance (small dissimilarities), indicating low location confidence. In Fig. 2 an example is shown with a candidate (A) and two possible neighbor divergences (B), (C). Neighbor (B) has the smallest distance a to (A) and its distance contributes to the degree of confidence. If the distance a is close to the divergence of (A) the confidence will be low. If a is larger than the divergence of (A) the confidence increases. Neighbor (C) has the distance b to (A) which is larger compared to (B). The confidence increases when the distance of b increases. The confidence decreases when the number of neighbor candidates increases. Distances between all distributions are considered during the calculation of the dynamic confidence. The following algorithm has been implemented to calculate the dynamic confidence dc at sample k for n reference locations

$$a_k = \sum_{i=1}^{n} \frac{1}{|1 - JD_{i,k}/JD_{min,k}|} \tag{1}$$

with

$$dc_k = 100 \cdot \left(1 - e^{-\left(\frac{\alpha}{a_k}\right)}\right). \tag{2}$$

The calculation of the distance between minimum divergence and neighbor divergence is executed in Eq. 1 and applied in Eq. 2. In Eq. 1 calculating distance between best fit $JD_{min,k}$ and neighbor candidate $JD_{i,k}$ is followed by normalizing each distance with JD_{min} to obtain the relative distance. The result in Eq. 1 increases when distances decrease (higher weight for small distances) and

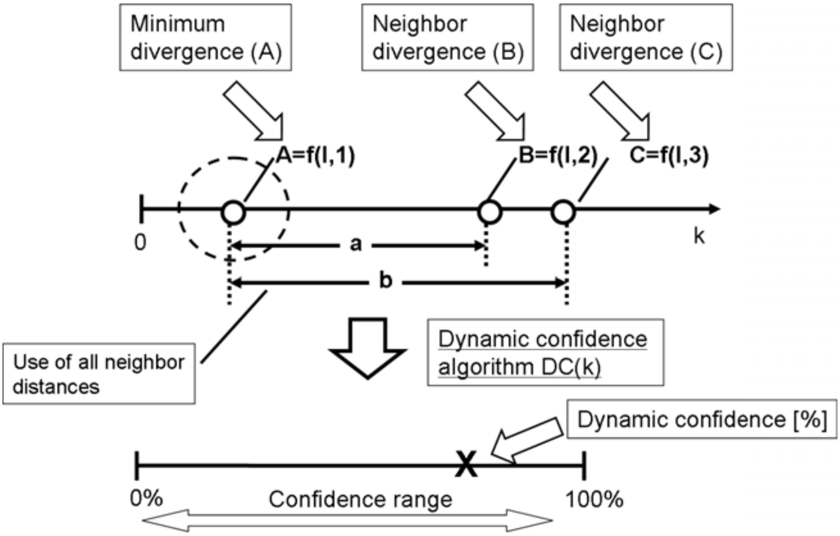


Fig. 2. Distance-based calculation of dynamic confidence

includes all distances. The sum is applied in Eq. 2 so that *dc* will decrease when number of distances increase. The algorithm uses a negative exponential function to converge the result toward 100 when distances increase (100% confidence) and toward 0 if the distances decrease (0% confidence would equal total similarity). The α-value was applied to the *dc* calculation for optimal scaling. Optimal scaling is achieved when *dc* increases towards 100 for large distances and reduces significantly if distances have been found small. Graphical analysis was performed prior to the implementation of the dynamic confidence algorithm identifying the optimal α-value (graphs excluded for reasons of brevity). The value of $\alpha = 0.5$ was found to allow an optimal recognition of dissimilarities.

In addition to the dynamic confidence which varies due to the time variant input data, the LOC has an intrinsic classification ability that depends on the selected distribution. In Eq. 3 the static confidence *sc* of the LOC is shown which can be estimated for different locations *l* selecting a candidate distribution *D* from the off-line learning phase. *sc* describes the ratio of true classifications *T* and the sum of true classifications and false classifications *F* for a given set of distributions *D*. The result is a value with a range of [0-1]. An increased number of true classifications and a reduced number of false classifications increase the *sc* value.

$$sc = \frac{\sum_l T_l|_D}{\sum_l T_l|_D \ + \sum_l F_l|_D}. \tag{3}$$

5 Implementation and Testing of a Prototype

In this section we describe the implementation and testing of a prototype of the location cognition (LOC). Acquisition of data during the off-line phase, selection of the reference distributions and then on-line location cognition are described. Transmission data was obtained by transmitting data between a pair of WLAN terminals at different locations under various conditions. The LOC engine was implemented in our wireless middleware in Linux, using kernel version 2.6, and includes a monitoring module, a location module that includes the LOC and a decision module for adaptive channel selection. We extended our WLAN IEEE 802.11 driver to obtain the TxCount value from the wireless device. The LOC uses the TxCount value which counts the number of received data packets. We have also implemented an outlier filter for data pre-processing. Transmission between transmitter and receiver was line-of-sight (LOS) in each case. The following five locations were used for testing of the LOC:

- Indoor locations

1. lab: an environment in a laboratory similar to an office environment (size: 10m x 20m x 3m).
2. corridor: a corridor inside the office building, consisting of doors, walls and windows (size: 3m x 30m x 3m).
3. entrance: an entrance hall in the office building with height of 10 m, mainly consisting of glass doors and large windows (size: 25m x 25m x 10m).

- Outdoor locations

1. building: a location beside the office building, 5m from the building.
2. road: a location at a road 50m from the office building.

At each location, transmission data were obtained for multiple distances in steps of 5m, namely 5m, 10m and 15m. Data were obtained for both short packets (200 bytes) and long packets (1500 bytes) capturing the effect of different packet sizes. A single UDP stream was sent continuously for the duration of 300s, increasing the transmission rate until maximum throughput is obtained. Transmission was monitored at the receiver in intervals of 1 sec at maximum throughput to obtain a reference distribution.

5.1 Selection of Reference Distributions

The mean and deviation of the reference distributions are shown in Fig. 3 and Fig. 4, respectively. The top graphs show the results for short packets obtaining max throughput at 7 Mbps before saturation occurred. The bottom graphs show the results for long packets at 28 Mbps before saturation occurred. Note that for all RDs the mean shows a higher number of successfully received data packets for outdoor locations, indicating a better quality link with less interference and multi-path effects. Regarding the indoor locations, the RDs for the entrance show the highest number of received data packets, followed by the RDs for lab and corridor which show the lowest number of received data packets. An explanation is that for the entrance less background wireless activities lead to a higher number of successfully received data packets. However, due to the exposed multi-path environment at the entrance (floor, doors, and windows) the number

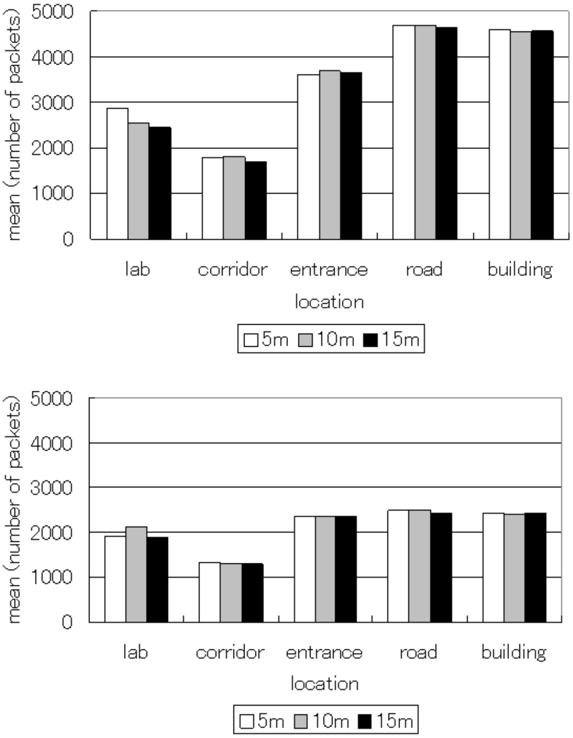


Fig. 3. Mean (number of packets) for short packets (top graph) and long packets (bottom graph)

of received packets is less than outdoors. Multi-path fading and increased wireless activities result in lower throughput in particular for the lab and the corridor as shown in Fig. 3. The characteristics for receiving data of long data packets are similar for short packets, except that the maximum value for outdoor is reduced for long data packets.

In Fig. 4 the standard deviation of all locations and different packet sizes are shown. It shows the highest value for the lab location for short packet lengths and decreases at the locations corridor and entrance. For the outdoor locations the standard deviation decreases significantly, and the location building shows a higher standard deviation than road. A conclusion is that multi-path fading at the long side of the building lead to higher standard deviation. This conclusion is valid for short packets including all distances. For the corridor location a lower standard deviation can be observed for short distances. For long packet length (1500 byte) the results show a different characteristic. The standard deviation is significantly reduced for all locations, expect for the lab location. Our explanation for this is that it is due to the significantly larger background wireless activity inside the lab.

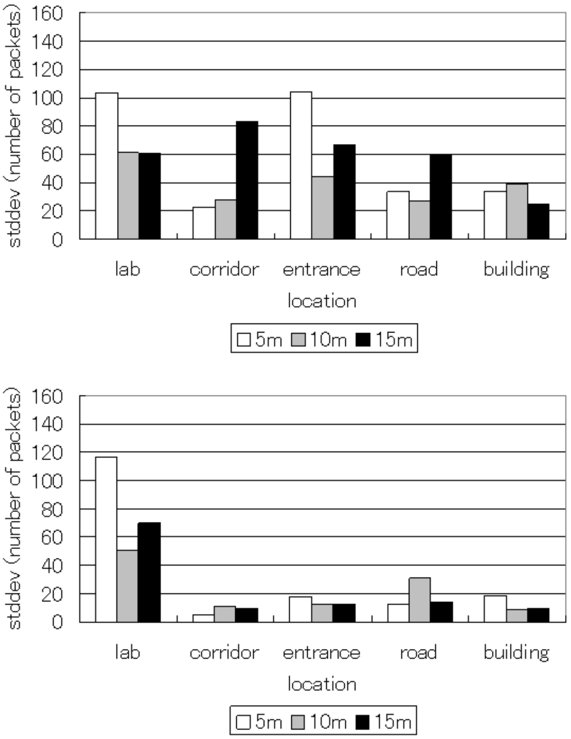


Fig. 4. Standard deviation (number of packets) for short packets (top graph) and long packets (bottom graph)

In Fig. 5 and Fig. 6 the relation between mean and standard deviation of the RDs are presented. Fig. 5 shows the results for the selected generic RD at each location, i.e., the RD at each location with a maximum static confidence value. Fig. 6 shows the results for the complete set of RD at each location and distance. Lab and corridor show the largest dissimilarities whereas entrance and building shows the smallest dissimilarities. Using Eq. 3 the result of static confidence for RDs in Fig. 5 was sc=1.0 for small packet length and sc=0.93 for long packet length. For RDs in Fig. 6 it was sc=0.76 for short packet length and sc=0.73 for long packet length.

5.2 Emulation of Location Cognition

During an emulation phase the LOC prototype was used to test the accuracy of the location cognition. In order to obtain quantitative results, emulations were run based on 3 trials at the 5 locations, lab, corridor, entrance, building and road at 3 different distances. The LOC used the pre-selected RDs of each location with a maximum of static confidence. Moreover, the LOC used an extended set of RDs which are used to identify the distance between the sender and

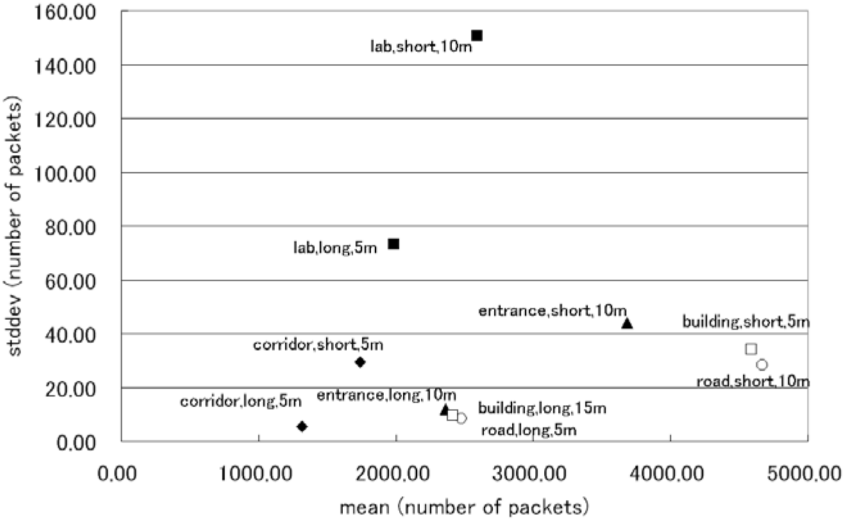


Fig. 5. Mean and standard deviation (stddev) of all RDs selected for location cognition including short and long packet sizes at 5 different locations (sc=1.0 for short packets, sc=0.93 for long packets)

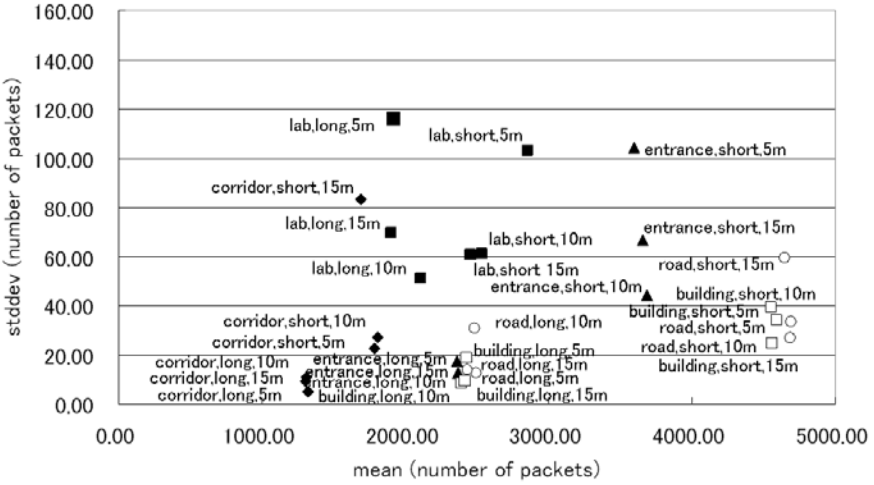


Fig. 6. Mean and standard deviation (stddev) of all RDs selected for location and distance cognition including short and long packet sizes at 5 different locations including the cognition of 3 different distances (sc=0.76 for short packets, sc=0.73 for long packets)

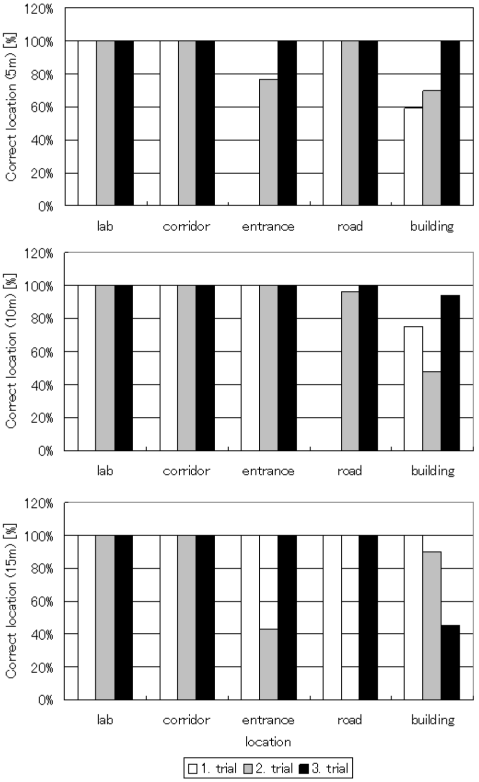


Fig. 7. Correct location cognition for all locations and distance (1500 byte, 3 trials)

receiver. The extended RD set contains additional distributions of each location for the distances 5m, 10m and 15m. We evaluated the accuracy of the location cognition by counting the number of correct classifications for multiple tests at the same location. Location cognition tests have been conducted including short packet length and long packet length for 3 trials (graphs excluded for reasons of brevity). The ratio of correct location cognition when transmitting short packets have shown 100% correct location for all 3 trials, expect for lab 15m (90%, 2. trial), road 15m (90%, 3. trial), and building (91%, 3. trial). We show the results for location cognition of long packets in Fig. 7. From the graph it can be observed that the location lab and corridor are judged correctly (100%), where as location entrance for 5m (0%, 77%) and 15m (43%) shows a reduced location cognition performance. We conclude that the entrance and building RD show similar statistics for long packets which lead into false detection.

In Fig. 7 the location road at 10m (1. trial) and 15m (2. trial) was judged as building, which can be counted as successful outdoor location cognition. Next, we discuss the accuracy of the combined classification of location and distance, in particular for the location building and entrance and for long packets (again,

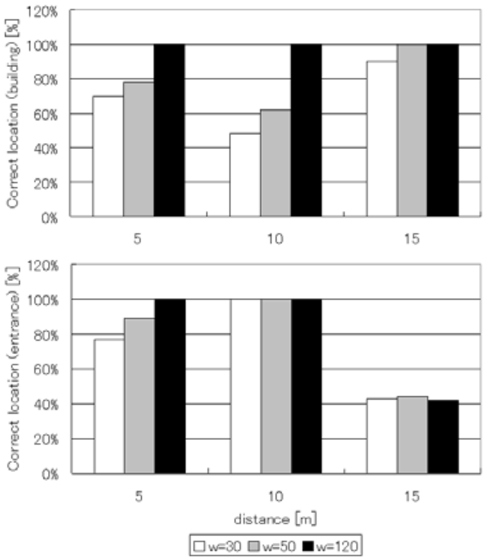
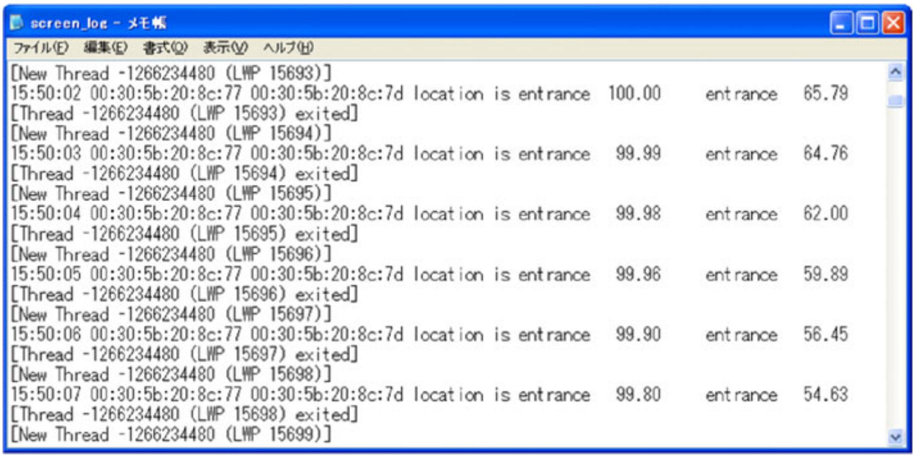


Fig. 8. Results of increased window size to improve combined location and distance for long packets (2. trial)



```
[New Thread -1266234480 (LWP 15693)]
15:50:02 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance  100.00     entrance    65.79
[Thread -1266234480 (LWP 15693) exited]
[New Thread -1266234480 (LWP 15694)]
15:50:03 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance   99.99     entrance    64.76
[Thread -1266234480 (LWP 15694) exited]
[New Thread -1266234480 (LWP 15695)]
15:50:04 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance   99.98     entrance    62.00
[Thread -1266234480 (LWP 15695) exited]
[New Thread -1266234480 (LWP 15696)]
15:50:05 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance   99.96     entrance    59.89
[Thread -1266234480 (LWP 15696) exited]
[New Thread -1266234480 (LWP 15697)]
15:50:06 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance   99.90     entrance    56.45
[Thread -1266234480 (LWP 15697) exited]
[New Thread -1266234480 (LWP 15698)]
15:50:07 00:30:5b:20:8c:77 00:30:5b:20:8c:7d location is entrance   99.80     entrance    54.63
[Thread -1266234480 (LWP 15698) exited]
[New Thread -1266234480 (LWP 15699)]
```

Fig. 9. On-line display of the LOC engine showing time-stamp, sender and receiver MAC address, location cognition, and dynamic confidence

graphs for short packets are excluded for reasons of brevity). These two locations are difficult to judge for the LOC.

In Fig. 8 it can be observed that the cognition is almost higher than 60% at 5m distance for both locations, having a window size of 30. The distance classification can be improved when the window size was increased to 50. Up to 80% successful location and distance classification can be observed. Similar improvements can be observed at 10 and 15, except the classification for the entrance at 15m, which remains unchanged (only observed for this particular trial). We conclude that the distance cognition has high success for a large variation of locations and distances.

Finally, Fig. 9 shows a screen shot of our implemented LOC engine including confidence values. Each line shows the results for a single cognition event. The first item is a time-stamp, the second and third items are MAC addresses of sender and receiver, and the following items show location classification results followed by confidence values. Two location cognition results can be observed, one for the generic location classification, obtained using the generic set of reference distributions, and the other for the extended location classification obtained using the extended set of reference distributions including distance.

6 Conclusions

Location cognition in wireless systems is an important problem and a challenging task. Various methods have been proposed to recognize the location of a terminal in specific environments. However, multi-location identification for indoor and outdoor is needed for future wireless networks. We proposed a simple-but-effective solution for recognition of multiple locations in indoor and outdoor environments, which does not need any infrastructure. The method is based on learning and recognizing the location dependent differences in wireless transmission characteristics based on a fingerprinting like method. We implemented a location cognition engine and demonstrated it is able to recognize different indoor/outdoor locations. We also implemented a novel distance cognition function to indicate the distance between sender and receiver. Finally, we proposed novel system parameters which report the confidence of the location cognition process.

The location cognition engine as it is described in this paper has been fully implemented in our Linux wireless middleware supporting the current IEEE 802.11a/b/g/n standards. The LOC uses a novel monitoring parameter, the number of received data packets, instead of RSS values. We have implemented an outlier filter for data pre-processing, and fast online location cognition. The proposed LOC is highly versatile and can be applied to detect indoor/outdoor environments for cognitive radios or location based services. All performance tests have been conducted in real WLAN environments including typical dynamics such as changes of wireless activities or surrounding environments. It can be concluded that multi-location cognition has been tested successfully in scenarios including both indoor and outdoor locations, and has the potential to be an integral part of intelligent WLAN systems in the future.

References

1. Arslan, H.: Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems. Springer, Heidelberg (2007)
2. Shellhammer, S.J., Sadek, A.K., Zhang, W.: Technical Challenges for Cognitive Radio in the TV White Space Spectrum. In: Information Theory and Applications Workshop, San Diego, pp. 323–333 (2009)
3. Kim, H.H., Ha, K.N., Lee, K.C.: Resident Location-Recognition Algorithm using a Bayesian Classifier in the PIR Sensor-Based Indoor Location-Aware System. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 39(2), 240–245 (2009)
4. Pan, J.J., Kwok, J.T., Yang, Q., Chen, Y.: Multidimensional Vector Regression for Accurate and Low-Cost Location Estimation in Pervasive Computing. IEEE Transactions on Knowledge and Data Engineering 18(9), 1181–1193 (2006)
5. Fischer, C., Gellersen, H.: Location and Navigation Support for Emergency Responders: A Survey. IEEE Pervasive Computing 9(1), 38–47 (2010)
6. Indoor Galileo/GPS Indoor Navigation & Positionierung (in German), funded by the Deutsches Zentrum fuer Luft & Raumfahrt (DLR) and the German Ministry, http://www.indoor-navigation.de
7. Gu, Y., Lo, A., Niemegeers, I.: A Survey of Indoor Positioning Systems for Wireless Personal Networks. IEEE Communications Surveys & Tutorials 11(1), 13–32 (2009)
8. Papliateseyeu, A., Kotilainen, N., Mayora, O., Osmani, V.: FINDR: Low-cost Indoor Positioning Using FM Radio. LNCS, Social Informatics and Telecommunications Engineering, vol. 7. Springer, Heidelberg (2009)
9. Kjaergaard, M.B., Treu, G., Ruppel, P., Kuepper, A.: Efficient Indoor Proximity and Separation Detection for Location Fingerprinting. In: 1st International Conference on Mobile Wireless Middleware, Operating Systems, and Applications, vol. 278, Innsbruck (2008)
10. Honkavirta, V., Perala, T., Ali-Loevtty, S., Piche, R.: A Comperative Survey of WLAN Location Fingerprinting Methods. In: 6th Workshop on Positioning, Navigation and Communication, Hannover (2009)
11. Mazuelas, S., Bahillo, A., Lorenzo, R.M., Fernandez, P., Lago, F.A., Garcia, E., Blas, J., Abril, E.J.: Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks. IEEE Journal of Selected Topics in Signal Processing 3(5), 821–831 (2009)
12. Aust, S., Matsumoto, A., Ito, T., Davis, P.: Supervised Classification Using Jeffrey Divergence for Location Cognition in Cognitive Radio. In: 12th International Symposium on Wireless Personal Multimedia Communications, Sendai (2009)

Session 4: Mobile Intelligent Middleware

(Chair: Ying Cai)

Towards an Elastic Application Model for Augmenting Computing Capabilities of Mobile Platforms

Xinwen Zhang, Sangoh Jeong, Anugeetha Kunjithapatham, and Simon Gibbs

Computer Science Lab., Samsung Information Systems America, San Jose, CA, USA
{xinwen.z,sangoh.j,anugeetha.k,s.gibbs}@samsung.com

Abstract. We propose a new elastic application model that enables the seamless and transparent use of cloud resources to augment the capability of resource-constrained mobile devices. The salient features of this model include the partition of a single application into multiple components called weblets, and a dynamic adaptation of weblet execution configuration. While a weblet can be platform independent (e.g., Java or .Net bytecode or Python script) or platform dependent (native code), its execution location is transparent – it can be run on a mobile device or migrated to the cloud, i.e., run on one or more nodes offered by an IaaS provider. Thus, an elastic application can augment the capabilities of a mobile device including computation power, storage, and network bandwidth, with the light of dynamic execution configuration according to device's status including CPU load, memory, battery level, network connection quality, and user preferences. This paper presents the motivations, concepts, typical elasticity patterns, and cost consideration of elastic applications. We validate the augmentation capabilities with an implemented reference architecture and example applications.

1 Introduction

Applications on smartphones traditionally are constrained by limited resources such as low CPU frequency, small memory, and a battery-powered computing environment. For example, the iPhone 3G is equipped with 412MHz CPU, 512MB RAM, and a battery allowing about 5 hours of talking time. The new Samsung Galaxy Android phone has 528MHz CPU, 128MB RAM, and battery offering about 6.5 hours of talk time. Both devices have up to 7.2 Mbps 3G data network connection. Compared to today's PC and server platforms, these devices still cannot run compute-intensive applications such as complex media processing, search, and large-scale data management and mining.

Cloud computing delivers new computing models for both service providers and individual consumers including infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS), which enable novel IT business models such as resource-on-demand, pay-as-you-go, and utility-computing [7]. From the perspective of service providers, cloud computing is often viewed as a vast and scalable platform for service delivery. We suggest a new perspective, one tuned to the needs of mobile devices. We consider cloud computing as a means to extend or augment the capabilities of resource constrained devices.

There are several approaches to realize this perspective. One approach is to duplicate the runtime environment of the device in the cloud and then run the application either

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 161–174, 2010.

on the device or in the cloud. The off-device runtime environment is sometimes called a "surrogate" [13], a "clone" [10], or a cloudlet [15]. Virtual machine technology is often used to host and isolate the off-device runtime so making this approach fit well with emerging IaaS platforms such as Amazon EC2 [1]. Running a device clone in the cloud has some attractive properties such as enhanced CPU and memory resources which lead to better performance. Furthermore, applications do not need any modification – the clone and the physical device can run identical binaries. However, this approach has disadvantages too. First, the application on the clone may need to access the physical hardware on the device. For example, consider a GPS application or simply the question of how an application running in the clone interacts with the user. It is certainly possible to transfer device I/O between the device and clone environment over the network, but this may impact responsiveness and battery use. Secondly, simply replacing one processor with another fails to take full advantage of cloud compute resources. Ideally, a cloud application should be able to run in a highly parallel fashion distributed over many cloud nodes. Thirdly, completely duplicating a device and running it on the cloud increases the complexity of device management. For example, the cloud system needs similar security protection and data privacy control as those on the device since it runs all possible applications with data resources from the original device.

The above considerations lead us to focus on application level augmentation instead of cloning a complete device environment. Often these applications are data-parallel with high compute-to-communication ratio. Examples include media processing, search, and data mining. Our goal is to design an architecture and related middleware to enable *elastic applications* which consist of multiple components called *weblets*, each of which can be launched on a mobile device or in the cloud. The decision of where to launch a weblet is based on application configuration and/or the status of the device such as its CPU load and battery level. Ideally the application model could also support migration of weblets between the device and cloud platform during runtime. While offloading and delegating computing have been proposed by many researchers [11,9,13,12], the novelty of our approach lies in enabling flexible and optimized elasticity by considering multiple factors including device status, cloud status, application performance measures, and user preferences (e.g., different running modes of an application including power-saving mode, high speed mode, low cost mode, offline mode, or in terms of expected application throughput).

To enable this new application model, many challenges exist in different areas, including management of heterogeneous computing environments, data management and communication dependencies between weblets, state synchronization between weblets, and cost-effective dynamic execution configuration. The middleware should provide infrastructure for seamless and transparent execution of elastic applications and offer convenient development support. This paper first gives the concepts and typical elasticity patterns (Section 2). We then focus on the optimization of cost-effective execution configuration by considering multiple factors (Section 3), which we believe is one of the most critical and unique components of the application model. We then present a high-level description of an implemented reference framework including deployment and runtime architecture and software development kit (SDK) (Section 4). We then show some experimental results which confirm the augmentation capabilities of our approach (Section 5).

2 Concepts and Elasticity Patterns

2.1 Concepts and Benefits

We define elastic applications as having two properties. First, following the client/server split of traditional web applications, an elastic application is split or partitioned so that execution occurs partially on the device and partially on the cloud. Previous work has proposed many mechanisms for splitting an application into modular components for remote execution or *cyber foraging* purposes, such as [8,9,11,12,14,16]. For elastic devices we assume application developers can determine how to organize weblets based on their functionalities and runtime behaviors such as computation demand, data dependency, and communication need, which we believe should be part of high-level design consideration of an application. Elastic middleware should provide necessary SDK and tools allowing developers to implement and test their designs. One principle for partitioning applications is that each weblet should have minimum dependency on others. This is not only for robustness but decreases communication overhead between weblets during runtime.

Second, the *execution configuration* of an elastic application is not static, instead it is determined when the application is launched and potentially modified during runtime. By execution configuration, we mean the assignment of application partitions to execution units (e.g., cores or virtual machines), either on the device or in the cloud. The left hand side of Figure 1 shows some possible execution configurations for an application using three weblets.

There are several benefits that the elastic application concept offers to mobile users and application developers deriving from coarse-grained application partitioning and dynamic configuration. First, elastic applications are not constrained by the compute capabilities of today's mobile platforms and can be configured to take advantage of multiple processing cores when available. If more compute (or storage) is needed then this can be obtained from the cloud. As devices become more powerful, compute and storage can shift back to the device. On the other hand, mobile device compute and storage need not be designed to satisfy the most demanding applications. Device resources can be modest (and less power consuming) since the more demanding applications can acquire resources from the cloud. From a performance perspective, the ability to allocate resources in the cloud and migrate functionality gives the device great flexibility. For example, performance can be increased or optimized to fit various goals (such as responsiveness, monetary cost, or power consumption). Furthermore, application components that are partitioned for migration can also be replicated. The failure then of one instance of a replicated component need not compromise the application. Also, the elastic application model offers a testbed for future technologies of mobile devices. Applications that run on the cloud today can move to the device in future products. This greatly extends the lifetime of applications and reduces development costs.

Table 1. Weblets vs. Web Services

Weblets	Web Services
HTTP (REST interface)	HTTP (REST or SOAP interface)
single client	many clients
client is application root or other weblet	clients are generally browsers or other web services
short-lived & long-lived requests	generally short-lived requests
dynamic endpoints (may migrate)	fixed endpoints
lifetime is client dependent	lifetime is client independent
runs on servers or client (cloud or device)	runs on servers
push to client possible	not available or non-standard

2.2 Elasticity Patterns

We now consider elastic applications and weblets in more detail. Our motivation for using weblets is that developers are familiar with the web application model and so can easily transition from the client/server partitioning of web applications to the more general form of partitioning found in elastic applications. Furthermore, programming methods used for web applications, for example AJAX and REST, are adapted by weblets. To see the similarities and differences of web applications and elastic applications, it is interesting to compare weblets with traditional web services. We highlight some areas for comparison in Table 1.

In designing a web application, a key issue is determining what logic will run on the server and what on the client. For early web sites, the client was mainly used for rendering and input, but now with JavaScript, AJAX, and plug-ins such as Flash and Silverlight, many tasks can be performed by the client. With elastic applications there is a similar issue, but because several weblets can be created by a single application, the topology of elastic applications is more varied. It appears these topologies fall into some common patterns, what we call *elasticity patterns*, several of these are shown on the right hand side of Figure 1 and briefly summarized as follows.

Replication Patterns: Pools and Shadowing. Weblet replication refers to running multiple weblets with the same interface, i.e., accepting the same types of request. There are two forms of replication: *pools* and *shadowing*. Weblet pools allow an application to leverage cloud CPU cycles and augment its throughput. With this pattern, the application issues requests that are routed to weblets as they become available. Weblet pools are well suited for applications that are easily divided into similar tasks, for example processing sets of images or scanning sets of files. Closely related to pools is shadowing in which the same request is sent to a set of replicated weblets in parallel. Shadowing can be used for fault tolerance and latency control. For example, shadowing a weblet on the device with a copy on the cloud can help the application recover from loss of network connectivity or loss of battery power. Shadowing can also enable more flexible latency control for an application, e.g., the device can use the earliest response from multiple shadowed weblets on the cloud.

Splitter Pattern. With the splitter pattern, a set of worker weblets perform variant implementations of a shared interface. For example, the workers may encapsulate adapters to access different social networks, or codecs to process different media formats. The

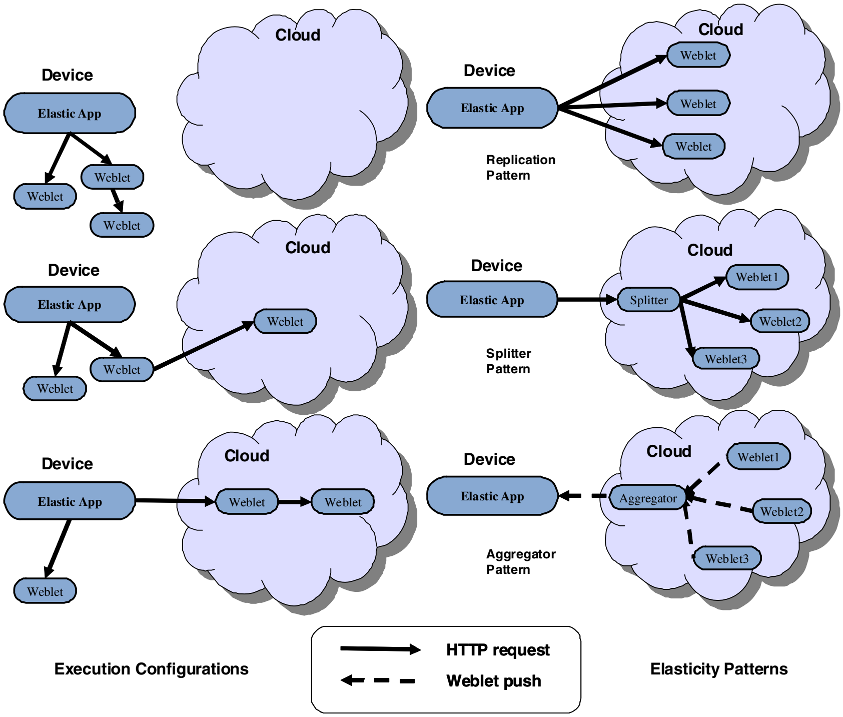


Fig. 1. Execution configurations and elasticity patterns

application is decoupled from the various implementations by a splitter weblet that routes requests to appropriate workers. This pattern increases application extensibility since new worker weblets are added without changing the application structure. Splitting can also enhance the user experience by converging multiple services on a single device. For instance, in the case where the worker weblets access different social networks, the splitter weblet's interface provides a unified or converged interface to a range of social networking services.

Aggregator Pattern. An elastic application can also aggregate computations from multiple worker weblets. In this pattern, an aggregator weblet collects information from multiple worker weblets and uses *weblet push* to relay this information to the device. For example, an application can run multiple weblets in the cloud as background threads that monitor the user's web accounts (e.g., emails or instant messages), the aggregator weblet pushes events (such as account activity) to the device. In some cases the splitter and aggregator patterns are combined or overlaid, the splitter pushes requests to the workers while the aggregator pushes events back to the device.

3 Cost Optimization for Elastic Applications

3.1 Cost Model

The augmented computation of an elastic application is not free but introduces costs to the mobile device and user, which depends on when and where a weblet is running and communications within weblets or between weblets and Internet. Furthermore, elastic applications can exhibit variant runtime behaviors with dynamic execution configurations, such as power consumption, monetary consummation, application performance, and even security and privacy properties. Therefore, the dynamic execution configuration of an elastic application is decided based on some cost saving objectives, which form a cost model in our framework. As Figure 2 shows, the cost model takes inputs of sensor data from both device and cloud sides, and runs optimizing algorithms to decide execution configuration of applications. Device and cloud related data such as battery level, network conditions, device loads, cloud loads and other performance data including current latency of the application, are obtained from appropriate sensing modules. The output of the cost model is possible actions that lead to the optimal execution configuration for the application, such as allocating resources on the cloud, launching/migrating weblets on/to device and/or cloud, selecting/switching between different network interfaces, replicating and shadowing weblets on cloud, etc.

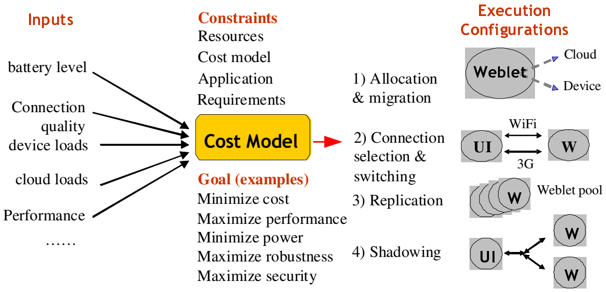


Fig. 2. Cost model of elastic applications

An important part of the cost model is choosing the attributes or objectives that should be optimized. We consider the following four attributes in our current elastic application framework, while new cost objectives can be integrated easily.

Power Consumption. Each application/weblet running on a mobile device consumes battery power by using CPU cycles, memory and radio module for communication with peer weblets on the cloud and/or external web services. The power consumption of a weblet on the device heavily depends on the I/O operations it performs [17,5,4]. In addition, different communication channels, such as W-CDMA, WiFi (802.11/a/b/g/n) etc., consume different power [2,6,3]. Considering the above, it is evident that although launching/migrating weblets to clouds should ideally save power consumption of computation on the device, the power consumption of network interfaces may override the benefits of the migration.

Monetary Cost. Execution of a weblet on a cloud platform may involve a monetary cost for the application user, based on the exact resources consumed on the platform. Usually, a commercial cloud service provider measures the cost of a computing task based on the amount of CPU cycles, storage, and communication traffic (in and out) of a cloud platform [1]. The monetary cost of a weblet running on the cloud platform is determined by the size of the input data consumed by the weblet (including those from peer weblets on the device for the same application and external web services), total execution time of the weblet on the cloud platform, data size/rate for intra-cloud communication between this weblet and others within the same cloud service provider (if applicable), and any other attributes that affect these parameters, such as network status affecting data transmission rate.

Performance Attributes. As an elastic application potentially runs across different platforms, latency is an important design consideration. There are different aspects of latency, such as impact on the user experience when using the application's UI and network latency with different network connections and traffic status, and the application latency to finish a particular computing task. Throughput also can be an important objective for some applications. For example, an application that does image analysis to find similar pictures from a large database needs maximum throughput. To achieve this, the heavy computing tasks are be launched or migrated to the cloud, although there is a tradeoff between doing this and the data communication overhead: too much communication may slow down the overall application throughput. Given this, building a good performance model is more challenging than power and monetary aspects. In general, to optimize latency, throughput and some application-specific options, CPU cycles and memory used by the weblets, along with the available network bandwidth for communication between the device and the cloud should be carefully evaluated.

Security and Privacy. Security is increasingly concerned in web-based computing systems. A mobile device potentially contains many user secrets and privacy-sensitive data, such as: contacts, SIM information, credit card details and many other credentials that may be needed to consume web services. Naturally, a mobile user may trust her device more than the cloud platform which is controlled by a third-party service provider. As launching or migrating a weblet to the cloud may also require offloading user data to the cloud, the user security and privacy concerns are even higher with an elastic mobile device. A weblet on the device or the cloud may need to access external web services on behalf of the user. For cost modeling purposes, we need to evaluate if a weblet requires any user data and if the user has strong concerns about offloading such data to the cloud. If the user has concerns over doing this, the weblet that requires this data should be launched on the device only and never migrated. Furthermore, during runtime, if a weblet needs to acquire external user data from other web services, which usually requires user credentials (username/password, public key certificate, or any other security credentials), the weblet may have to be migrated back to the device.

3.2 Optimizing Execution Configuration with Cost Objectives

Once a cost model is developed for a particular application, a mechanism is needed for efficient and intelligent dynamic execution configuration, e.g., via some lightweight machine learning algorithms at the device side. In our implementation of one elastic application, we use Naïve Bayesian Learning techniques to find the optimal weblet configuration (# of weblets on device and cloud), given device status (in terms of CPU, memory and network consumptions), user preference (in terms of expected # of images that should be concurrently processed), and history data of the application.

As Figure 3 shows, a vector '$\mathbf{x}$' consists of values representing device status components such as the upload bandwidth, throughput, power level, memory usage and file cache. A vector '$\mathbf{z}$' consists of values representing user's preferred setting for cost objectives including monetary cost, power consumption, and processing speed. The configuration variable 'y' has values from 1 to N (max number of possible configurations), where each value maps to a specific configuration pair. Given all these data, the following expression can be applied to determine the most optimal configuration.

$$y^* = \underset{y}{\operatorname{argmax}}\, p(y) \prod_{i=1}^{L} p(x_i|y) \prod_{j=1}^{M} p(z_j|y) \tag{1}$$

In the above expression, x_i is the i-th status component value that can have different number of states for each component and z_j is a j-th preference component, where $i \in \{1, 2, \cdots, L\}$ and $j \in \{1, 2, \cdots, M\}$, with L and M representing the number of components in the status vector and the number of components in the preference vector, respectively.

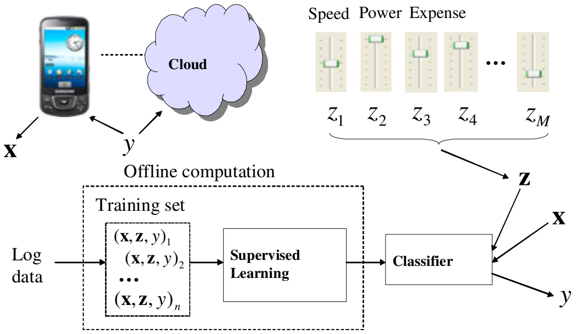


Fig. 3. Weblet scheduling through Machine Learning techniques

Note that it is relatively easy to determine dynamic configurations in this application since it has only one type of weblet. For a general application with multiple types of weblets, each having different runtime behaviors, the optimization can be very complex and the computation itself may override the cost savings. Considering that an elastic application can be installed and executed by many users on similar devices, a service-oriented cost optimization implementation can save computation cost for the device.

4 Reference Implementation and Application Development

4.1 Reference Architecture

To experiment with this new application model, we have developed a reference framework including application bundle, architecture, and some example elastic applications. Our framework works with Amazon EC2 and S3. Figure 4 shows the main functional components.

In our current framework design, a typical elastic application consists of a UI component, one or more weblets, and a manifest. Weblets are autonomous software entities that run either on the device or cloud and expose RESTful web service interfaces via HTTP. The manifest is a static XML file that contains metadata for the application.

It could be used to specify any requirements and constraints for the application and the individual weblets, such as: the digital signature needed to download/migrate the weblets, requirements for compute power, network and storage, time limits for weblet execution, maximum instances of the weblet that can be launched on the device and the cloud, if a weblet can be launched/migrated to the cloud and specifics about handling data required/generated by the application/weblets etc.

On the device side, the key component is the device elasticity manager (DEM) which is responsible for configuring applications at launch time and making configuration changes during run time. The configuration of an application includes: where the application's components (weblets) are located, whether or not components are replicated or shadowed (e.g., for reliability purposes), and the selection of paths used for communication with weblets (e.g., WiFi or 3G if such a choice exists). Each device also provides sensing data on the device such as processor type, utilization, and battery state. This data is made available to the elasticity manager and is used to determine when and where a new weblet instance should be launched.

The cloud elasticity service (CES) consists of the cloud manager, application manager, and sensing information collection. The cloud manager is responsible for allocating resources from, and releasing to, underlying cloud nodes. It maintains usage information, including compute, bandwidth and storage, for the various weblets running on the cloud. The application manager provides functions to install and maintain applications on behalf of elastic devices, and helps launch weblets on different cloud nodes. Sensing information refers to the collection of operational data on the cloud platform. These data are made available to the cloud manager to assist it in tracking usage. As a service provider, the CES exports a web service, referred to as the cloud fabric interface (CFI) to elastic devices and applications. A node manager on each cloud node oversees resources associated with a particular node (server) within the cloud. It communicates

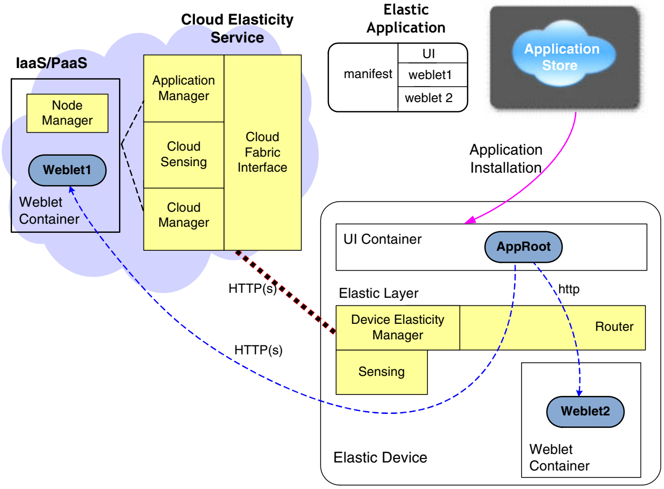


Fig. 4. Reference architecture for elastic application

directly with the cloud manager and application manager. Each node runs one or more weblet containers which are the weblet runtime environments hosted on an Amazon EC2 instance.

4.2 SDK Development

We have implemented a preliminary SDK for according to the reference architecture, which is used to develop the basic interfaces of weblets in our example applications. Using this SDK as a base, developers can build elastic applications in high-level languages such as JavaScript, Java, and C#. Currently the SDK has C# bindings; however we plan to extend it to other languages.

A typical elastic application includes a `AppRoot` component and one or more weblets. The `AppRoot` is the part of the application that provides the user interface and issues requests to weblets. All of these are packaged into one bundle, which includes the binaries of weblets and a manifest describing the application, and most importantly, the developer-signed hash values of the individual weblets. Figure 5 shows a state diagram illustrating the lifecycle of a weblet, including the various states that a weblet can be in and the actions that cause the state transitions. A weblet is an independent functional unit of an application that performs computing, storing, and networking tasks. It resembles an embedded or dedicated web server and presents a web service interface (i.e., it is accessed via HTTP). In our SDK, an abstract class called `AbstractWeblet` is defined to represent the core behavior of weblets. Other specific types of weblets can be implemented as subclasses of `AbstractWeblet` and extend its methods as required. Each weblet is associated with a weblet type and identified through a unique id. Once an application has defined one or more weblet types, it can use the DEM to create instances (i.e., to create specific weblets) and issues requests to these weblets.

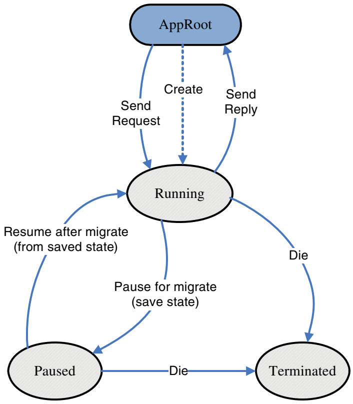

AppRoot
Create
Send Request
Send Reply
Running
Resume after migrate (from saved state)
Die
Pause for migrate (save state)
Paused
Die
Terminated

Fig. 5. Lifecycle of a weblet. A weblet is always created by the `AppRoot`, and can be in state of `Running`, `Paused`, or `Terminated`.

The DEM can decide to migrate a running weblet from the device to the cloud or vice-versa; weblet migration is transparent to the application. When a weblet is running on device and the DEM decides to migrate it to a cloud node, the DEM issues a `Pause` request to the weblet, this causes the weblet to close its request interface, release resources and save state. The DEM then sends the saved state to the cloud via the CFI. After the state has been transferred to the cloud, the weblet is resumed and restores itself from the saved state. The CFI returns the new connection information for the weblet (e.g., IP address, port, and session tokens) to the DEM so that the DEM may continue to route requests to the weblet on cloud.

5 Elastic Applications and Experimental Validation

5.1 Example Applications

To demonstrate the elastic application model, we have developed several test applications with our SDK and deployed on the reference architecture with Amazon EC2. The simplest is an image processing application in which various filtering operations are applied to set of images. Following the replication pattern, a weblet pool is created on the cloud; images are then processed in parallel by pool members. The application can adjust the size of the pool, so it is possible to compare throughputs for different execution configurations. For example, the application running on a mobile device can be configured to offer the same throughput or greater as the application running on a PC.

A second example is a form of augmented reality in which real-world objects are detected and enhanced. This application runs tracking and rendering on the device and uses the splitter pattern with a set of matcher weblets on the cloud. Each matcher searches for different objects within video frames. The splitter collects information on identified objects and relays this to the device for rendering. By running the matchers in the cloud, many more objects can be detected (per unit time) than when the application runs fully on the device.

5.2 Experimental Validation

We validate the elasticity of our framework by using the aforementioned image processing application as benchmark. This application consists of only one type of weblet called `ImageWeblet`. Its functionality is to perform image filtering with an algorithm specified by the user. The weblet is replicated on the device and the cloud, as and when required. The total number of weblet instances spawned depends on application load and the number of weblets in the cloud, both specified by the user. The application UI enables the user to do the following configurations during runtime: online (can launch weblet at cloud) or offline (all weblets are running on the device) mode of the application, number of weblets to run on the cloud (if in online mode), the filtering algorithm to be used, and the number of images (workload) to process at the same time. The images used in by the experiment are 24-bit color with size 240 x 360. Figure 6 shows a snapshot when it is running on a Samsung Galaxy smart phone with Android 1.6.

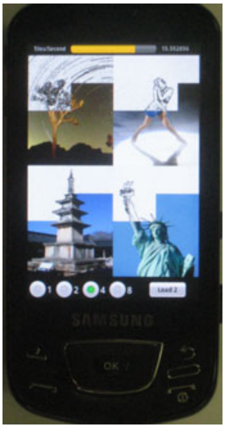


Fig. 6. Snapshot of elastic image processing application on Samsung Galaxy

The goal of our validation is to compare the performance of an elastic device (ED) and a non-elastic device (NED) running the same image processing application. For the elastic device, the application uses an in-house cloud comprising of 8 Linux boxes. A non-elastic version of the application is also running independently in order to compare it with the elastic version. Essentially, the non-elastic version uses only the device to run weblets, whereas the elastic version uses both the device and the cloud. The setup also includes PCs to host the CFI and a performance monitor application. The CFI is implemented with PHP scripts on a Linux server with Apache and MySQL.

The performance monitor collects several measurements, including the available upload/download bandwidth (KB/sec), application workload (number of images to be processed) and throughput (the number of image tiles processed/sec), average CPU usage (%), and available memory (MB), from the test device and from the cloud. In addition, it also maintains information about the total number of weblets started for the application and the individual number of weblets running on the device and the cloud.

Each configuration has a unique composition of device weblets and cloud weblets. We set the maximum number of weblets as 16 and consequently, more than 100 different configurations are possible. The configuration specifying 1 device weblet & 0 cloud weblets is considered the default configuration for the non-elastic device. Among all possible configurations, we chose the 74 configurations where the number of device weblets is less than or equal to 4 (due to limitations with CPU utilization) for the experimental analysis. For each configuration, the data was collected 20 times and the average values were considered for final comparisons.

Figure 7 shows the performance of the elastic device over 74 configurations. In comparison with the throughput of about 6 tiles/sec for the default/non-elastic device configuration (1 device weblet, 0 cloud weblets), the throughputs of all other configurations are better. We can observe that the throughput for the configuration with 0 device weblet and 16 cloud weblets has the highest throughput among the 74 configurations tested. The configuration with 16 device weblets has the best performance, as there are a total of 16 images in load 3. A surprising observation is that the configuration with 8 weblets performed better than configurations with 9-15 weblets (a result of internal application logic). This indicates that an intelligent weblet scheduling is essential to identify the most efficient weblet configuration.

CPU usage is more predictable overall, in that more device weblets lead to more CPU usage. However, the trend is interesting when comparing the number of device weblets. For configurations with up to 2 device weblets, running more cloud weblets leads to more CPU usage. For configurations with 3 and 4 device weblets, a general

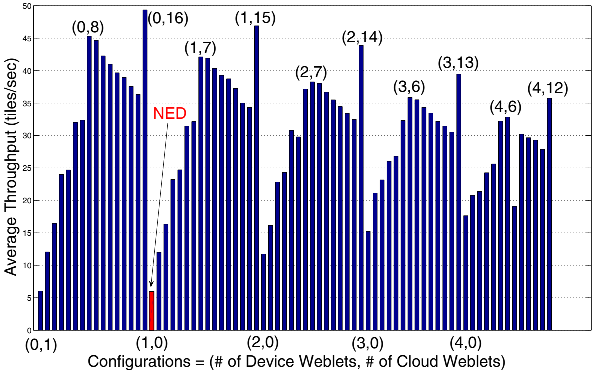


Fig. 7. Throughputs vs. configurations

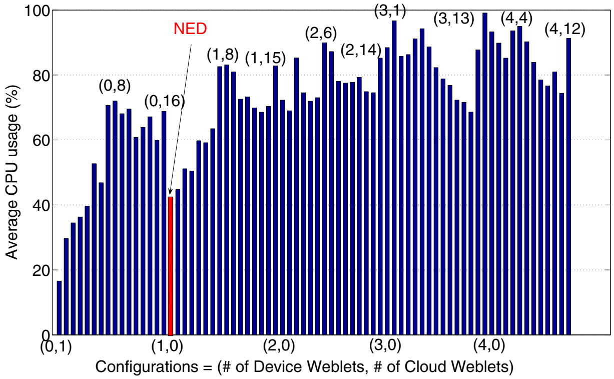


Fig. 8. CPU usage vs. configurations

trend is that running more cloud weblets reduces the CPU usage. By combining CPU usage data in Figure 8 with the throughput data in Figure 7, we are able to identify the configurations that lead to low CPU usage and high throughput: for instance, configurations (0,2), (0,3) and (0,4) have lower CPU usage (than that of a non-elastic device) and higher throughput. This results in more available CPU cycles for other applications and improves multi-tasking capabilities.

6 Conclusions and Future Research Themes

We propose an elastic application programming model aiming to remove the constraints of specific mobile platforms by providing a distributed framework that extends the device into the cloud. The salient feature of this model is that it offers a range of elasticity patterns between resource-constrained devices and Internet-based clouds. Each pattern in turn can be realized by several execution configurations. A comprehensive cost model is used to dynamically adjust execution configurations thus optimizing application performance in terms of a set of objectives. We present the high level design of elasticity framework and primitive experimental results with an example application.

There are a set of directions that need further research efforts. First of all, we use a simple weblet launching scheduling mechanism in our example application, while a general cost optimization engine is very desired for elastic applications with comprehensive considerations based on our cost model. Further, as aforementioned in the elasticity patterns, weblets of a single application may share application data and state. Since weblets run in different locations, it is desirable to replicate data to increase performance, but then data integrity and synchronization become issues. As another issue, code and computation migration is a traditional problem in many systems [9,18]. How to support runtime weblet migration thus enhance mobile user experience but at the same time achieve the transparency and seamlessness is challenging. Furthermore, integrity and data security of weblets running on cloud are essential problems for many applications. We have designed a lightweight protocol to distribute shared secrets and session keys between weblets for mutual authentication purposes [19]. However, how to build strong trust between weblet runtime environments on cloud and device is an open problem.

References

1. Amazon ec2, http://aws.amazon.com/ec2/
2. Rfmd data sheet, http://www.rfmd.com/databooks
3. Wifi power consumption analysis, http://nesl.ee.ucla.edu/fw/documents/reports/2007/poweranalysis.pdf
4. Samsung corp., flash/smartmedia/filesystem memory databook (2000)
5. Samsung semiconductor dram products (2001), http://www.usa.samsungsemi.com/products/family/browse/dram.htm
6. Analog devices data sheet, analog device inc. (2003), http://www.analog.com/productselection/pdf
7. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G., Patterson, D.A., Rabkin, A., Stoica, I., Zaharia, M.: Above the clouds: A berkeley view of cloud computing. Technical Report UCB/EECS-2009-28, University of California, Berkeley (February 2009)
8. Balan, R., Flinn, J., Satyanarayanan, M., Sinnamohideen, S., Yang, H.: The case for cyber foraging. In: Proc. of the 10th ACM SIGOPS European Workshop (2002)
9. Balan, R.K., Satyanarayanan, M., Park, S., Okoshi, T.: Tactics-based remote execution for mobile computing. In: Proc. of the 1st International Conference on Mobile Systems, Applications, and Services, pp. 273–286 (2003)
10. Chun, B.-G., Maniatis, P.: Augmented smartphone applications through clone cloud execution. In: USENIX HotOS XII (2009)
11. Gu, X., Messer, A., Greenberg, I., Milojicic, D., Nahrstedt, K.: Adaptive offloading for pervasive computing. IEEE Pervasive Computing, 66
12. Hunt, G.C., Scott, M.L., Hunt, G.C., Scott, M.L.: The coign automatic distributed partitioning system. In: Proc. of the 3rd Symposium on Operating Systems Design and Implementation, pp. 187–200 (1999)
13. Porras, O.R.J., Kristensen, M.D.: Middleware for Network Eccentric and Mobile Applications. In: Dynamic Resource Management and Cyber Foraging, Springer Press, Heidelberg (2008)
14. Rellermeyer, J.S., Alonso, G., Roscoe, T.: R-osgi: distributed applications through software modularization. In: Cerqueira, R., Campbell, R.H. (eds.) Middleware 2007. LNCS, vol. 4834, pp. 1–20. Springer, Heidelberg (2007)
15. Satyanarayanan, M., Bahl, P., Caceres, R., Davies, N.: The case for vm-based cloudlets in mobile computing. IEEE Pervasive Computing (4) (2009)
16. Sousa, J., Garlan, D.: Aura: an architectural framework for user mobility in ubiquitous computing environments. In: Proc. of the 3rd Working IEEE/IFIP Conference on Software Architecture (2002)
17. Vijaykrishnan, N., Kandemir, M., Irwin, M., Kim, H., Ye, W.: Energy-driven integrated hardware-software optimizations using simplepower. In: Proc. of the Int. Symposium on Computer Architecture (2000)
18. Xian, C., Lu, Y.H., Li, Z.: Adaptive computation offloading for energy conservation on battery-powered systems. In: ICPADS (2007)
19. Zhang, X., Schiffman, J., Gibbs, S., Kunjithapatham, A., Jeong, S.: Securing elastic applications on mobile devices for cloud computing. In: Proc. of ACM Cloud Computing Security Workshop (2009)

A Self-organizing Approach for Building and Maintaining Knowledge Networks

Gabriella Castelli, Marco Mamei, and Franco Zambonelli

DISMI - Dipartimento di Scienze e Metodi dell'Ingegneria
University of Modena and Reggio Emilia
Via Amendola 2 , 42100 Reggio Emilia, Italy
{gabriella.castelli,marco.mamei,franco.zambonelli}@unimore.it

Abstract. Pervasive and mobile devices can generate huge amounts of contextual data, from which knowledge about situations occurring in the world can be inferred for the use of pervasive services. Due to the overwhelming amount of data and the distributed and dynamic nature of pervasive systems, this may be not a trivial task. Indeed the management of contextual data should be run by a dedicate middleware layer, i.e., knowledge networks in charge of organizing and aggregating such data to facilitate its exploitation by pervasive services. In this paper we introduce a unsupervised, distributed and self-organizing approach to build and maintain such a layer based on simple agents that organize and extract useful information from the data space. We also present a Java-based implementation of the approach and discuss experimental results.

Keywords: Distributed Middleware, Knowledge Networks, Context Awareness.

1 Introduction

Pervasive and mobile devices and Web 2.0 are already able to generate an overwhelming amount of data about users and context, from which knowledge about situations and facts occurring in the world should be inferred for the use of pervasive and mobile services. A service in need of understanding what is happening around can access the produced pieces of data and analyze them to realize what is the current situation of its context. Nevertheless there are a number of complexities inherent in this process, such as the communication efforts to retrieve the useful knowledge out of an overwhelming amount of data, and the computational efforts to analyze, relate and aggregate such information.

Accordingly, a real challenge for future pervasive services is the investigation of principles, algorithms, and middleware infrastructures, via which this growing amount of distributed information can be properly represented, organized, aggregated, and made more meaningful, so as to facilitate the successful retrieval by pervasive services [7,3]. Many approaches [10,4] are currently going in the direction of adopting specific middleware layer, i.e., a knowledge network layer embedding data and algorithms, and providing effective access to such data by services.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 175–188, 2010.

Unlike other approaches (e.g., centralized and/or deterministic) to data organization and aggregation [24,5], we propose a distributed self-organized approach to organize, link, and aggregate, related items of contextual information. This choice better suits the decentralized, dynamic, and unpredictable nature of pervasive and mobile systems. In particular, in this paper:

1. We present a middleware architecture and prototype to store and manage contextual data coming from diverse pervasive devices structured according to the W4 data model [9]
2. We present an original self-organized algorithmic approach to perform knowledge networking over a massive amount of distributed pieces of knowledge stored in the above middleware
3. We show a several experiments we performed to test the system and discuss evaluation results.

The remainder of this paper is organized as follows. Section 2 briefly summarizes the W4 data model that is used to represent contextual data provided by pervasive devices and the middleware architecture. In Section 3 introduce the W4 Knowledge Networks idea, describe the algorithmic approach to organize isolated and distributed pieces of paper into networks of correlated data items. Section 4 presents the performance evaluation. Section 5 discusses related work, and finally Section 6 concludes.

2 The W4 Model and Architecture

We adopt the W4 data model to represent and structure the data to to illustrate our self-organizing approach for building and maintaining knowledge networks. The W4 data model has been firstly described in [9]. Here we shortly summarize its key features and then give an overall architectural view of the W4 middleware, its API, ant its implementation.

2.1 The W4 Data Model and Architecture

The proposed W4 model starts from the consideration that any elementary data atoms as well as any higher-level piece of contextual knowledge, in the end, represents a fact which has occurred in the world. Such facts can be expressed by means of a simple yet expressive four-fields tuples (Who, What, Where, When): *"someone or something (Who) does/did some activity (What) in a certain place (Where) at a specific time (When)"*.

More in particular the four-fields of the W4 data model each describes a different aspect of a contextual fact:

– The *Who* field associates a subject to a fact. The *Who* field is represented by a type-value pair.
– The *What* field describes the activity performed by the subject. It is represented as a string containing a predicate:complement statement.

- The *Where* field associates a location to the fact. In our model the location may be a physical point represented by its coordinates, a geographic region, or it can also be a place label.
- The *When* field associates a time or a time range to a fact. This may be an exact time/time range or a context-dependent expression, e.g., *now*.

The way it structures and organizes information makes the W4 data model able to represent data coming from very heterogeneous sources and simple enough to promote ease of management and processing (although we are perfectly aware that it cannot capture each and every aspect of context, as freshness of data, reliability, access control, etc).

2.2 The W4 API

In the W4 data model, we rely on the reasonable assumption that software drivers are associated with data sources and are in charge of creating W4 tuples and inserting them in some sorts of shared data spaces that are distributed in physical world.

The interface to access the W4 middleware took inspiration from tuple-space approaches [1] and consists in two basic operation:

```
void inject(KnowledgeAtom a);
KnowledgeAtom[] read(KnowledgeAtom a);
```

The *inject* operation is equivalent to a tuple space “out” operation: an agent accesses the closest data space to store a W4 tuple there.

The *read* operation is used to retrieve tuples from the closest data space via querying. A query is represented in its turn as a W4 template tuple. Upon invocation, the read operation triggers a pattern-matching procedure between the template and the W4 tuples that already populate the data space. Pattern-matching operations work rather differently from the traditional tuple space model and may exploit differentiated mechanisms for the various W4 fields. Read operations can involve searching in multiple tuple spaces, as explained in Section 3.3.

In [7] we provide several examples of knowledge representation and knowledge generation using the W4 data model.

2.3 Architecture and Implementation

Figure 1 depicts the overall architecture of a W4 system.

At the bottom there are diverse data sources that produce data formatted according the W4 data model and feed a number of W4 tuple spaces. We assume that software drivers gather information from all the available devices (e.g., RFID tags, GPS devices, Web services), and combine them with the goal of producing a W4 tuple as accurate and complete as possible.

The W4 system is made up by a number of distributed W4 tuple spaces. Those tuple spaces can be both local tuple spaces hosted by personal devices

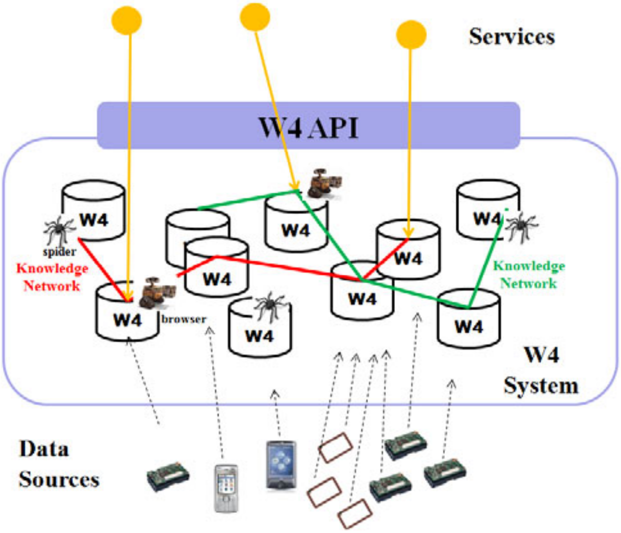


Fig. 1. The W4 System Architecture

and shared tuple spaces acting as public accessible servers. In the systems there are a variety of agents that access the tuple spaces via the W4 API in order to organize the data layer. In particular:

- *Spiders*: are able to jump from a tuple space to another and link tuples that are related into knowledge networks
- *Browsers*: can browse a knowledge network to solve a query and to infer new tuples

Many W4 Knowledge Networks can be realized and coexist in the W4 system, each realizing a specific view over the data. Those agents (i.e., spiders and browsers) and the algorithms to create and manage knowledge networks will be presented in Section 3.

Finally, at the top there are the various services that access the W4 system to retrieve data, to whom the internal W4 system and data location are completely transparent. Indeed they can act over the system submitting queries to the closest W4 tuple space via the W4 API.

We developed a prototype implementation of the described architecture in a small pervasive computing testbed by extending the LighTS Tuple Space [2], a light weight tuple space implementation particularly suitable for context-aware application , and by realizing spiders and browsers as simple Java agents.

The implemented W4 middleware runs on laptops and on PDAs equipped with wireless interface and J2ME-CDC (Personal Profile) Java virtual machine.

3 W4 Knowledge Networks

Although the W4 data model proved to be rather flexible to manage contextual data, the idea is to exploit the W4 structure to access and exploit distributed contextual data in a more sophisticated and effective way. More specifically we propose general-purpose mechanisms and policies to link together knowledge atoms, and thus form W4 Knowledge Networks in which it will be possible to navigate from a W4 tuple to the others. Moreover, new information could be produced combining and aggregating existing tuples while navigating the space of W4 tuples.

The basic ideas towards the realization of W4 knowledge networks had been anticipated also in [9] and [7], but only in this paper they are eventually realized and evaluated.

3.1 The W4 Knowledge Networks Idea

The W4 Knowledge Networks approach is based on the consideration that a relationship between knowledge atoms can be detected by a relationship (i.e., a pattern-matching) between the information contained in the atoms fields. In particular, for the W4 data model, we can identify two types of pattern matching relations between knowledge atoms:

- *Same value – same field*: We can link together those W4 tuples in which the values in the same field match according to some pattern-matching function. In this way, we can render complex concepts related to groups of W4 tuples, e.g. *All students (same subject) who are attending a class (same activity) at the same room (same location).*
- *Same value – different field*: We can link atoms in which the same information appears in different fields augmenting the expressive level of the information contained in the W4 tuples. For example, a knowledge atom having *When: 18/09/2009* can be linked with another atom like *Who: Fall Class Begin* , to add semantic information to that date.

Exploiting those correlations it is possible to find the relationships between one particular W4 tuple with other tuples in the data space, which may then be used to create a web of linked information both to more effectively navigate in the space of information (e.g., for effectively gathering information correlated to a specific context) and as a basis for more elaborated inference and reasoning (e.g., for representing in a comprehensive and expressive way complex situations).

3.2 The W4 Knowledge Networks Algorithm

An unsupervised, distributed and self-organizing approach to generate and maintain the knowledge networks' layer is clearly required by the decentralized nature of pervasive computing systems and the overwhelming amount of generated data, which prevent the use of a centralized process for data management. To this end, we adopt a swarm-based approach relying on a two-phase process.

The first phase is the identification of all possible correlations of interest between knowledge atoms, and the creation of links between W4 atoms. This can be done by a number of simple agents, which we call *spiders* as they weave their webs between correlated tuples. Each spider is associated with a pattern matching function that takes a W4 tuple and matches it against a W4 tuple used as a template. The function returns a boolean value meaning wether the matching is successful or not, and accordingly suggesting creating a link or not. Obviously the simpler the matching function is (i.e., few fields of a W4 tuple are involved), the more the resulting net of links can be reusable. E.g., A1 is in charge of linking together all the tuples with corresponding *who* fields, for instance all the tuples whose who field corresponds to *user:Gabriella*, while another spider agent can search tuples with corresponding where field.

Spiders continuously surf W4 tuple spaces in order to retrieve tuples that fulfill the specific relationship, those tuples are virtually linked together thus creating a W4 knowledge network for the given relationship. To this end spiders must be capable of analyzing W4 tuples stored in different tuple spaces and building correlation networks that extends over distributed tuple spaces. For this reason, spiders are realized in terms of weakly mobile java agents [13].

The spiders' algorithm follows:

```
define:
    rel; //the relation to be satisfied
    knet; //the knowledge network reference

Main:
    Do forever:
        TupleSpace ts = random();
        move (ts);
        tuple t[] = ts.read(rel);
        knet.add(t);

    Done;
```

The spider chooses a random tuple space and checks if any tuple in the tuple space fulfills the given relationship. If it is positive the tuples are added to the knowledge network *knet* by adding a reference to the last tuple space that was found earlier, i.e., drawing a link between the last tuple space added to the knowledge network *rel* and the current one. This process continuously repeats. In this way, a single knowledge network of links between correlated tuples is generated. More spiders can work concurrently both on the same relationship or on different ones, building the knowledge networks layer in a self-organizing fashion.

The second step is the generation of new knowledge atoms, by analyzing which of the identified links can lead to a new W4 atom as a process of merging related atoms. This activity is performed by another class of agents, called *browsers*. Browser agent surf the knowledge networks trying to generate new W4 atoms. Each browser is capable of inferencing a specific type of relationship. The browsers' algorithm follows:

```
define:
    rel; // the relation that the browser is capable to infer

Main:
    Do forever:
        TupleSpace ts = random();
        tuple t = ts.random();
        ts.add (GenerateNewKnowledge(t));
    Done;
```

The browser chooses a random tuple *t* in the system, and locates all the knowledge networks in which the tuple *t* is involved. Then the browser start to browse each of the found knowledge networks. For each tuple *ti* found in a related knowledge networks, the browser checks if he is able of generate a new W4 atom carrying higher knowledge. If positive, the new atom is generated and added to the current tuple space. The issue of bounding the amount of knowledge generation has been discussed in [8].

3.3 Using W4 Knowledge Networks

The idea at the base of the W4 Knowledge Networks approach is that spiders and browsers continuously surf, analyze, correlate and infer new knowledge. In this way new tuples are linked to the knowledge networks of interest and new knowledge networks can be realized as soon as they become of interest for services that access the data middleware. At the same time, browsers can exploit the knowledge networks to browse among tuples that are somehow related and possibly infer new knowledge to be injected in the system in form of a W4 tuple.

Although the knowledge networks can be used as the basis for knowledge reasoning, even when new data are not generated, the web of links between atoms can be fruitfully used during querying to access and retrieve contextual information more effectually. When a query is submitted to the W4 tuple space system, a query-solving agent capable of browsing knowledge networks, i.e. a query solving browser created in order to solve a query, analyze the query template and determine one or more knowledge networks to which the matching tuples should belong. Then the query solving browser choose a random W4 tuple space in the system and scans it until he finds an entry point for one of the identified knowledge networks, i.e. a tuple belonging to one of those knowledge networks. When the entry point is found, the agent starts to jump from the entry point tuple to the other tuples in the identified knowledge network, checking if they matches the template and finally returns the retrieved tuples. This is beneficial for services because fewer read operations have to be performed when exploiting knowledge networks instead of a set of data spaces in which information is not related to each other.

4 Performance Evaluation

To assess the W4 approach presentation, we report several experimental results we performed to evaluate the effectiveness and feasibility of the proposed approach and compare it to other solutions.

To test the approach, we developed a simulated environment based on the Repast framework [http://repast.sourceforge.net] and integrated with the actual prototype presented in Section 2.3. The simulated environment is used to generate a huge amount of data that are needed in order to properly test our middleware. We represented a virtual campus with a number of users (i.e., professors, students, administrative staff, etc.) each moving in the environment and performing their day by day activities. The virtual campus is split in 100 zones, each of them holds a private W4 tuple space that stores all the tuples generated in it. Periodically a W4 tuple for each user is generated on the basis of the current position, activity and time. In this scenario many tuples are stored in the W4 tuple spaces, and services may find difficult to access those data.

4.1 Efficiency

The first set of experiments aims to measure the efficiency of the W4 system in retrieving information and comparing it with an exhaustive search in tuple spaces and with an hash based approaches based on the performance of the above systems when a non destructive query is submitted to the system.

The exhaustive search is performed on a tuple space that embeds the W4 data model facilities but not the W4 knowledge networks mechanisms. When a query is submitted to this simplified W4 tuple space system, a query agent chose a random tuple space in the system and scans it seeking for the W4 tuples that fulfill the query template. Then a random tuple space is chosen again, until the whole system is scanned.

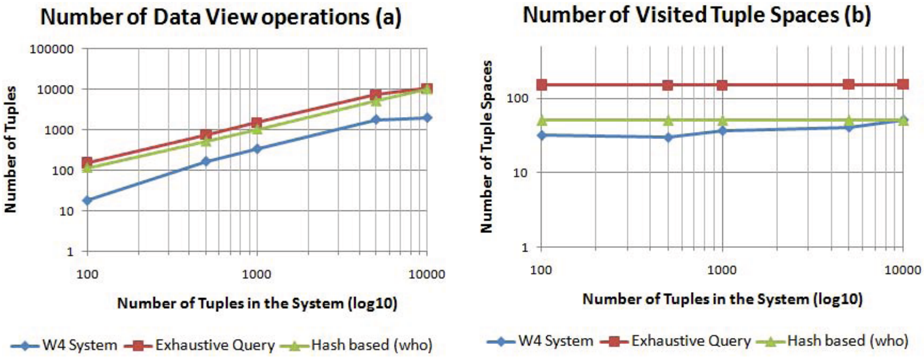


Fig. 2. Efficiency: (a) Number of view operations done by query-solving agents. (b) Number of view operations done by query agents.

Hash based tuple space is a well known and popular technique for data indexing in distributed environment. Here we follow an approach similar to [15] in which a single field of the tuple structure is used for the hashing operation and indexing purpose. When a tuple is injected in the distributed system, the hashing operation is performed over the designated field and the tuple is then stored on the resulting tuple space. Then when a query is submitted to the system, the same hashing operation is performed on the query template and the result indicates the tuple space to scan for results. In this simulations we considered the hash performed on the *who* field. Similar results are achieved considering the others fields of a W4 tuple. Of course other hash based approaches that use tuple spaces exist. We intentionally didn't consider approaches that consider to hash more than one W field at a time, because the issues that would arise in their distributed management make them ineffective in highly dynamic scenarios.

The experiment works as follow: we fed the W4 system with fixed amount of tuples and made them organized in W4 knowledge networks. Then, for the sake of experiments, we stopped the data sources and submitted to the system the following complex query: "Retrieve all the users that were near agent A5 was, on time timeT1". For a W4 System this means that the following two queries should be subsequently resolved:

```
query1 (who, what, where, when) = (user:A5, *, ?var1, timeT1)
```

The second query looks as follow:

```
query2 (who, what, where, when) = (*, *, var1, timeT1)
```

Here both the *where* is automatically considered as a bounding box and the *when* field is automatically transformed in a time interval. To solve such a query two knowledge networks must been investigated: one relating all tuples from *user:A5*, and one relating all tuples that refers a specific region of the space.

We run the simulations 15 times and depicted the average values. Figure 2 (a) shows the number of tuple spaces visited by the query-solving agent in the considered systems. The W4 tuple space systems performs better than the other considered approaches. Indeed, in the medium case, the exhaustive search has to query half the number of tuple spaces in the system to solve the first sub-query and the whole number of the tuple spaces to solve the second ones. The hash approach works better than the exhaustive query because one of the sub-query is solved thanks to the hashing operation, nevertheless the other sub-query have to be solved traditional as in the case of the exhaustive search. However exploiting the W4 Knowledge Networks is even better because the number of accessed tuple spaces is determined by the number of tuple spaces involved in the knowledge networks of interest.

Figure 2 (b) shows the number of read operation performed. Also in this case the W4 tuple space system performs better then the other systems. As in the previous case, the exhaustive search have to access half the number of tuples in the systems to solve the first sub-query, and all the tuples in the system to solve the second one. Here the performance of the hash based systems can be significantly different depending if the hashing is performed on the who field or on the where field. However, when data are accessed on multiple semantic dimensions, the W4 system performs better because all the fields are considered equally important when building knowledge networks.

4.2 Effectiveness

Provided that the W4 approach exhibits a good behavior in accessing contextual data (i.e., the access costs are lower than the other considered approaches), we run a second set of experiments to test the effectiveness of the knowledge networks' approach, in terms of accuracy of provided results when the knowledge networks algorithms are running, i.e. the fraction of the documents that are relevant to the query that are successfully retrieved (also called "recall" in information retrieval).

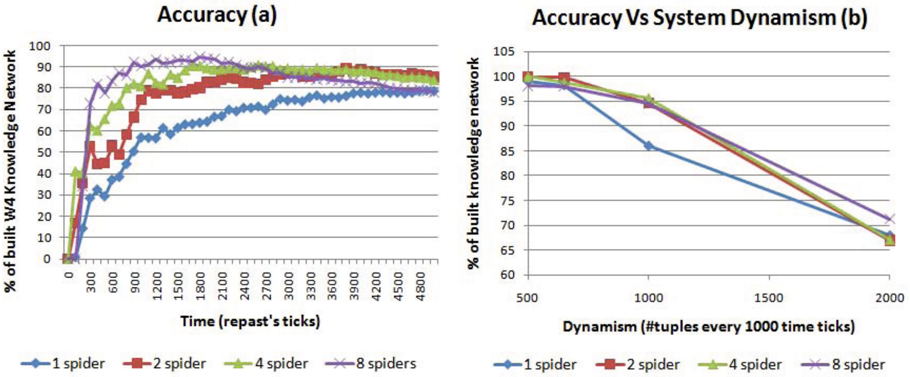


Fig. 3. (a) Accuracy of the indexation over time. (b) Accuracy of the indexation Vs dynamism of the system.

In order to test the effectiveness of the approach, we started feeding the system with W4 tuples generated by the simulated environment and let the knowledge networks keep organizing. Periodically we checked the content of a specific knowledge networks respect to the content of the whole W4 system to measure the percentage of tuples that has been indexed.

Of course the indexing works as quicker as more spiders are involved in, to this end Figure 3 (a) compares the results obtained when a different number of spiders is running. The obvious result is that the more spiders are working, the quicker the knowledge network reaches its indexation level. We can see that it takes a certain amount of time for the knowledge network to reach its stable value of indexation that is in the satisfactory range of 80-90% depending on the number of spiders run. This suggests that the W4 system could be improved by taking into account the W4 tuples' injection rate in order to autonomously determine the right number of spiders running.

Accordingly to this observation, we performed a second set of experiments varying the dynamism (i.e. the tuples injection rate) of the system respect to

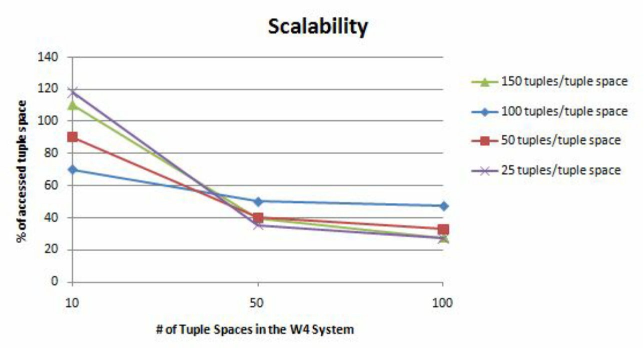


Fig. 4. Scalability of the w4 systems respect to the number of tuple spaces in the system (number of tuples per tuple space fixed)

the number of spiders running, the results is quite interesting because they give an idea of the number of spiders that should run simultaneously in relation to the dynamism of the system. As we expected, Figure 3 (b) shows that as the dynamism of the system increases, i.e. tuples are injected in the W4 system more quickly, the percentage of indexation decrease. That is, when the tuple injection rate increase, it may be needed to run more spiders and browsers to keep the good level of indexation.

4.3 Scalability

Another key factor in distributed systems is their ability to scale. To test further the system scalability we performed another set of experiments fixing the number of tuples per tuple space, and varying the number of tuple space in the W4 system. We performed the measurements as described for the efficiency experiments, and measured the percentage of tuple spaces accessed to solve the query (the number of tuples accessed is not represented because it is highly correlated, as shown in Figure 2). Figure 4 depicts results. We can see that the performances improve when the number of tuple spaces in the system increase. This is due to the fact that the more the system is wide and distributed, the more selective the knowledge networks can be. Indeed knowledge networks approach can be useful only if accessing the knowledge networks allows to skip out accessing the majority of tuple spaces (and then tuples) in the systems, i.e., the knowledge networks extends over a limited subset of tuple spaces of the whole system. In other words the W4 approach makes sense in a distributed environment rather than in a centralized and static one.

5 Related Work

Context is a very fluid notion and although several researchers claim that it is very hard to abstract it in terms of variables and data models [12], it is also a widespread opinion that a more pragmatic perspective should be adopted. Early works in this area, as from Schmidt et al. [23] and Dey et al. [11], concentrates on the issue of acquiring context data from sensors and of processing such data but they generally miss in identifying a uniform model to describe the data and analyzing the issues at the middleware level. Some recent proposals, such as [24,5] focus on providing models for contextual data that adopt a uniform well-defined structure. Indeed, our W4 proposal accounts for a very similar structuring for contextual information, and enriches it further with a well-defined API, and with the possibility of linking data atoms and of providing application-specific views to services.

An increasing number of research works get inspiration from tuple space middleware models [1] and propose representing and storing contextual information in the form of tuples to be stored in distributed tuple spaces. Egospaces [16] adopts this perspective, without committing to a specific pre-defined structure for context tuples, which can make it difficult for services to uniformly deal with tuples represented in different formats. Other proposal, such as The Context Fabric model [14] rely on well-structured context tuples. Recent proposals focusing on sensor networks, suggest exploiting a tuple-based approach to provide application-specific views on sensorial data [19]. In general we consider tuple-based approaches very suitable for organizing and accessing contextual information, but we also think that there is need of more structuring and flexibility than those exhibited by the existing approaches.

In the above described work, the issue of relating contextual data atoms with each other and of providing different views to different applications is not generally addressed. More recently, other proposals have adopted a similar endeavor but have considered the issue of adopting specific ontologies to model context information and enable other than efficient querying also efficient context-reasoning [22,17]. Although such approaches tend to be too application-specific, they attribute the importance of linking independent atoms of contextual information (with ontological relations) and of reasoning not only on individual data items but also on their relations, an idea which is fully shared by our knowledge network vision. Other proposals experience different techniques for context reasoning. Many works, such as and [21], are focused on situation learning and situation relationships in smart environment. Other works, such as [20] propose predicate logic as an effective language for context-aware reasoning. The W4 Knowledge Networks approach we propose aims to be more general and proposes an approach different from traditional ones, considering self-organizing agents. Campbell et al. [6] consider the possibility of extracting higher-level knowledge from raw sensed data merging feature vectors in an opportunistic fashion for people-centric application. The idea of merging and considering data coming from diverse sources is shared with the W4 Knowledge Networks approach. However in the W4 approach we go further considering multiple knowledge views that can be accessed by multiple services.

Obviously also other areas of research contributed towards the realization of our knowledge networks vision, in particular data mining and pattern discovery and granular computing. See [7] for a critical survey.

6 Conclusion and Future Works

Despite the promising results achieved so far in the study of the W4 self-organized knowledge networks algorithms, some research issues still have to be faced. In particular, more experiments should be done to evaluate properly the overhead coasts.

Moreover, in the current implementation of the W4 system, the number of tuples stored in the system is constantly increasing as new data are injected in the system. There is the need for a ”garbage collection” solution and we plan to experiment with a concept of knowledge tuple fading as introduced in [18]. Finally, security and privacy issues need to be analyzed w.r.t. accessing W4 tuples and their relations.

References

1. Ahuja, S., Carriero, N., Gelernter, D.: Linda and friends. Computer 19(8-9), 26–34 (1986)
2. Balzarotti, D., Costa, P., Picco, G.P.: The LighTS Tuple Space Framework and its Customization for Context-Aware Applications. International Journal on Web Intelligence and Agent Systems 50(1-2), 36–50 (2007)
3. Bettini, C., Brdiczka, O., Henricksenc, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive and Mobile Computing (in press)
4. Bicocchi, N., Castelli, G., Mamei, M., Rosi, A., Zambonelli, F., Baumgarten, M., Mulvenna, M.: Knowledge networks for pervasive services. In: Proceedings of the 2009 International Conference on Pervasive Services, ICPS 2009, pp. 103–112. ACM, New York (2009)
5. Bravo, J., Hervs, R., Snchez, I., Chavira, G., Nava, S.: Visualization services in a conference context: An approach by rfid technology. Journal of Universal Computer Science 12(3), 270–283 (2006)
6. Campbell, A., Eisenman, S., Lane, N., Miluzzo, E., Peterson, R., Lu, H., Zheng, X., Musolesi, M., Fodor, K., Ahn, G.-S.: The rise of people-centric sensing. IEEE Internet Computing 12(4), 12–21 (2008)
7. Castelli, G., Mamei, M., Zambonelli, F.: Engineering contextual knowledge for autonomic pervasive services. International Journal of Information and Software Technology 52(8-9), 443–460 (2008)
8. Castelli, G., Menezes, R., Zambonelli, F.: Self-organized control of knowledge generation in pervasive computing systems. In: ACM Symposium on Applied Computing, March 8-12 (2009)
9. Castelli, G., Rosi, A., Mamei, M., Zambonelli, F.: A simple model and infrastructure for context-aware browsing of the world. In: Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications, PERCOM 2007, Washington, DC, USA, pp. 229–238. IEEE Computer Society Press, Los Alamitos (2007)

10. Clark, D.D., Partridge, C., Ramming, J.C., Wroclawski, J.T.: A knowledge plane for the internet. In: Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, SIGCOMM 2003, pp. 3–10. ACM, New York (2003)
11. Dey, A.K., Abowd, G.D., Salber, D.: A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Human Computer Interaction 16(2), 97–166 (2001)
12. Dourish, P.: What we talk about when we talk about context. Personal Ubiquitous Computing 8(1), 19–30 (2004)
13. Fuggetta, A., Picco, G.P., Vigna, G.: Understanding code mobility. IEEE Transactions on Software Engineering 24, 342–361 (1998)
14. Hong, J.I.: The context fabric: an infrastructure for context-aware computing. In: extended abstracts on Human factors in computing systems, CHI 2002, pp. 554–555 (2002)
15. Jiang, Y., Xue, G., Jia, Z., You, J.: Dtuples: A distributed hash table based tuple space service for distributed coordination. In: Grid and Cooperative Computing, 2006, pp. 101–106 (October 2006)
16. Julien, C., Roman, G.-C.: Egospaces: facilitating rapid development of context-aware mobile applications. IEEE Transactions on Software Engineering 32(5), 281–298 (2006)
17. Lee, D., Meier, R.: Primary-context model and ontology: A combined approach for pervasive transportation services. In: Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, 2007. PerCom Workshops 2007, pp. 419–424 (2007)
18. Menezes, R., Wood, A.: The fading concept in tuple-space systems. In: Proceedings of the 2006 ACM Symposium on Applied Computing, Dijon, France, pp. 440–444. ACM Press, New York (2006)
19. Mottola, L., Picco, G.P.: Logical neighborhoods: A programming abstraction for wireless sensor networks. In: Gibbons, P.B., Abdelzaher, T., Aspnes, J., Rao, R. (eds.) DCOSS 2006. LNCS, vol. 4026, pp. 150–168. Springer, Heidelberg (2006)
20. Ranganathan, A., Campbell, R.H.: An infrastructure for context-awareness based on first order logic. Personal Ubiquitous Comput. 7(6), 353–364 (2003)
21. Reignier, P., Brdiczka, O., Vaufreydaz, D., Crowley, J.L., Maisonnasse, J.: Context-aware environments: from specification to implementation. Expert Systems: The Journal of Knowledge Engineering 24(5), 305–320 (2007)
22. Roussaki, I., Strimpakou, M., Kalatzis, N., Anagnostou, M., Pils, C.: Hybrid context modeling: A location-based scheme using ontologies. In: IEEE International Conference on Pervasive Computing and Communications Workshops, vol. 1, pp. 2–7 (2006)
23. Schmidt, A., Aidoo, K.A., Takaluoma, A., Tuomela, U., Laerhoven, K.V., de Velde, W.V.: Advanced interaction in context. In: Gellersen, H.-W. (ed.) HUC 1999. LNCS, vol. 1707, p. 89. Springer, Heidelberg (1999)
24. Xu, C., Cheung, S.C.: Inconsistency detection and resolution for context-aware middleware support. In: Proceedings of the 10th European Software Sngineering Conference Held jointly with 13th ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp. 336–345 (2005)

Dynamic Adaptive Middleware Services for Service Selection in Mobile Ad-Hoc Networks

Rogério Garcia Dutra and Moacyr Martucci Jr.

Department of Computer and Digital Systems Engineering (PCS),
Escola Politécnica, Universidade de São Paulo,
Av. Prof. Luciano Gualberto, trav 3, no 158, São Paulo, SP, Brasil
rogdutra@gmail.com, moacyr.martucci@poli.usp.br

Abstract. Dynamic adaptive service selection became a key necessity for most mobile middlewares based on functional services properties. Well-grounded algorithms for datamining were used for unsupervised selection of services clusters with similar non functional properties, adaptive induction of decision trees for supervised selection of quality of service (QoS) parameters relationship and adaptive fuzzy inference to manage uncertainty in QoS measures. These algorithms, encapsulated as services, compose a middleware solution for mobile ad-hoc networks service selection, using Service-Oriented Architecture (SOA) approach.

Keywords: Adaptation and Service Selection; Mobile Ad-Hoc Network QoS Awareness; Datamining Algorithms for Unsupervised and Supervised Learning; Fuzzy Inference; Service- Oriented Architecture (SOA) Middleware.

1 Introduction

Well-grounded datamining algorithms are ready-made tools, that have been peer-reviewed, widely used and tested to tackle data heterogeneity and sparseness, supporting the discovery process of knowledge-based interactions and relationships from high volume databases.

Pervasive computing using mobile devices faces similar challenges, where heterogeneity is characterized by lack of service and communication standards. A *service* is any tangible or intangible facility a device provides that can be useful for any other device. Services comprise those for software and hardware resources, which move at high or low speeds or even remain stationary, entering and leaving the system when switched on or off in the network. Sparseness and uncertainty capabilities for service discovery, are key challenges in the mutable nature of mobile ad-hoc networks (MANETs).

The nodes of MANETs intercommunicate through single-hop and multihop paths in a peer-to-peer fashion. Intermediate nodes between a pair of communicating nodes (providers and consumers) act as routers. The nodes are mobile, so the creation of routing paths is affected by the addition and deletion of nodes. The topology of the network may change rapidly and unexpectedly, once no fixed infra-structure is used.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 189–202, 2010.

In MANETs environment, service discovery would enable devices and services to properly discover, configure, and communicate with each other. Discovery comprises search and selection. These two mechanisms can be independent or integrated. For example, a consumer might first search for all instances of a service provider and then select a suitable service, or might perform the two functions simultaneously. Although service selection is a basic feature for service discovery approaches, it has been underestimated or simply ignored in most of discovery solutions found in literature.

Usually, a consumer issues a query to search services based on functional properties, advertised by service providers or intermediate nodes in the network, resulting in a set of similar services. To complete the discovery process, a selection based on additional service non functional properties is necessary. If this selection is not performed properly, the search will generate non optimized results, causing an unnecessary overhead in MANETs environment or low Quality of Service (QoS) perception from the consumer point of view.

To overcome these challenges, this paper propose a novel selection solution called Dynamic Adaptive Middleware Services for Service Selection (DAMS-SS) in MANETs, to satisfy the following requirements:

- Cluster search results, based on unsupervised learning of Self-Organizing Map algorithm, without consumer interaction or hard-coded assumptions;
- Define hierarchical cluster relationships, using adaptive and incremental supervised learning of an Adaptive Decision Tree algorithm;
- Adapt consumer service request, managing uncertainty in QoS metrics definitions from the consumer perspective, using a Fuzzy Inference algorithm.

The expected benefits from DAMS-SS solution are:

- Enhance service selection capabilities of existing functional middleware solutions, encapsulating datamining algorithms as middleware services based on a service architecture;
- Transform data gathered from MANETS into comprehensible information to support consumer decision on best service choice selection;
- Propose a structured process for service search refinement combined with a reactive and proactive selection method.

The rest of this paper is organized as follows. Section 2 describes the related work and current research about service selection and algorithms used for service mining. Section 3 describes the proposed middleware architecture and the service selection process. Section 4 presents the results and section 5 the conclusion and future work.

2 Related Work for Service Selection and Mining Algorithms

Although service selection is a basic feature for service discovery approaches, it has been many times underestimated or simply ignored. In a service discovery survey [1] for MANETs, 12 works were evaluated, but only 3 proposed selection algorithms, although service search and selection are often integrated. In [2] demonstrated that proper integration improves overall network performance by localizing network communication, thus reducing interference and allowing multiple concurrent transmissions in different parts of the network.

However, none of these works proposed any kind of intergration of service mining techniques, to enhance service selection. In [3], a middleware that exploits machine learning techniques to learn how to perform "on the fly" translations across ontologies, removing the unrealistic assumption that devices exchange knowledge by means of a shared ontology (or a set of statically known ones). It uses Kohonen's *Self-Organizing Map* (SOM) [4] for service cluster exploration, due to its unsupervised learning capability. The drawback of SOM is the inability to manage uncertainty in data and explain the relations of gathered clusters, due to opaque nature of its algorithm, as described in section 2.1.

To manage uncertainty, usually generated by non clearly defined service consumer requests, [5] proposed a fuzzy service adaptation engine for context-aware mobile computing middleware biased on known set of rules or policies extracted from historical data. The dependence from historical data generates a hard issue, due to its unpredictable nature of MANETs. To overcome this issue, an *Adaptive Network-based Fuzzy Inference Systems* (ANFIS) [6], combining supervised neural networks and fuzzy logic, is necessary to generate a set of rules from the SOM cluster results, to avoid a historical data analysis, as described in section 2.3.

This set of rules, based on all rules combinations, can potentially generate a huge amount of possibilities, causing a overhead in the fuzzy inference engine and impacting in ANFIS performance. To mitigate this pitfall, the rules must be induced and evaluated incrementally based on SOM cluster relationships. An *Adaptive Decision Tree* algorithm [7] can perform this task, as described in the section 2.2.

2.1 Self-Organizing Map (SOM) Algorithm

Due to its unsupervised learning capabilities, the Self-Organizing Map (SOM) is usual tool chosen for exploratory phase of datamining [8]. It projects input space on prototypes of a low-dimensional regular grid that can be effectively used to visualize and explore properties of the data. When the number of SOM units is large, to facilitate quantitative analysis of the map and the data, similar units need to be grouped, i.e., clustered. A two-stage procedure—first using SOM to produce the prototypes that are then clustered in the second stage—is found to perform well when compared with direct clustering of the data and to reduce the computation time.

First, a large set of prototypes—much larger than the expected number of clusters—is formed using SOM or some vector quantization algorithm. The prototypes can be interpreted as "protoclusters", which are in the next step combined to form the actual clusters. Each data vector of the original data set belongs to the same cluster as its nearest prototype.

SOM consists of a regular, usually two-dimensional (2-D), grid of map units. Each unit $\boldsymbol{i}$ is represented by a prototype vector $\boldsymbol{m_i = [m_{i1},\ldots,m_{id}]}$, where $\boldsymbol{d}$ is the input vector dimension. The units are connected to adjacent ones by a neighborhood relation. The number of map units, which typically varies from a few dozen up to several thousand, determines the accuracy and generalization capability of SOM. During training, the SOM forms an elastic net that folds onto the "cloud" formed by the input data. Data points lying near each other in the input space are mapped onto nearby map units. Thus, SOM can be interpreted as a topology preserving mapping from input space onto the 2-D grid of map units.

SOM could be trained iteratively, where at each training step, a sample vector is randomly chosen from the input data set. Distances between and all the prototype vectors are computed. The bestmatching unit (BMU), is the map unit with prototype closest to the sample vector, using a distance measure, usually Euclidian distance. The update rules and training parameters for vector prototypes can be found in [8], since clustering SOM units after SOM training is better than clustering services directly.

The primary benefit of the two-level approach is the reduction of the computational cost. Another benefit is noise reduction. The prototypes are local averages of the data and, therefore, less sensitive to random variations than the original data, although it cannot avoid uncertainty in data.

The two main ways to cluster data, make the partitioning are hierarchical and partitive approaches. The hierarchical methods can be further divided to agglomerative and divisive algorithms, corresponding to bottom-up and top-down strategies, to build a hierarchical clustering tree. Partitive clustering algorithms divide a data set into a number of clusters, typically by trying to minimize some criterion or error function. The number of clusters is usually predefined, but it can also be part of the error function [9].

Partitive methods are better than hierarchical ones in the sense that they do not depend on previously found clusters. On the other hand, partitive methods make implicit assumptions on the form of clusters. To select the best one among different partitionings, each of these can be evaluated using some kind of validity index.

Several indices have been proposed [10], [11]. The Davies–Bouldin index [12] was used, because is suitable for evaluation of k-means [9] partitioning because it gives low values, indicating good clustering results for spherical clusters. If desired, some vector quantization algorithm, e.g., k-means, can be used instead of SOM in creating the first abstraction level. Other possibilities include the following. Minimum spanning tree SOM [13], neural gas [14], growing cell structures [15], and competing SOM's [16] are examples of algorithms where the neighborhood relations are much more flexible and/or the low-dimensional output grid has been discarded.

In any case, the pruning produces a more clear dendrogram but still does not provide a unique partitioning. This is not a problem if the objective is only to find a set of interesting groups. If necessary, a final partitioning can be selected kind of interactive tool, such as a decision tree, since SOM opaque algorithm cannot incrementally define the relationship of gathered clusters.

2.2 Adaptive Decision Tree (ADAPTREE) Algorithm

According [7], decision trees are widely used as a knowledge representation tool in machine learning, mainly for their clean graphical representation, which captures, some aspects of the human decision process. Algorithms for inducing decision trees from examples, a major problem in machine learning, include ID3 and C4.5 [17], CART [18] and ITI [19], the last one being an incremental inducer.

ADAPTREE is a decision tree induction algorithm based on adaptive finite state automata, extended with some non-syntactical characteristics to handle continuous features and statistical generalization. ADAPTREE can be incrementally trained to solve classification problems. The general idea is to handle each training and testing instance as a string and consider the decision tree itself as a special kind of adaptive finite state automaton, with the initial state of this automaton corresponding to the root of a classical decision tree.

The core of the ADAPTREE learning strategy, including the generalization mechanism based on conditional probabilities estimates, which are calculated dynamically, is incremental. Hence, the training and testing instances could be presented one by one, interchangeably. In fact, the strategy could also be viewed as a different kind of instance-based learning [20], with some resemblance to the approaches based on k-d trees [20]. However, if the attributes are to be reordered, the automaton should suffer some major modifications from time to time.

Datasets containing continuous value attributes should also present a problem to the incrementality of ADAPTREE, at they must be previously discredited. In the current implementation, a discretization method, described in [19], is automatically performed on each continuous feature present in the dataset, before the learning takes place. This method employs a supervised (use class information) approach based on recursive entropy minimization and a minimum description language (MDL) stopping criteria [21], and so, depends on the existence of some training examples (supervised learning).

ID3 algorithm was originally used as tree induction algorithm by ADAPTREE, but according [17], if an attribute has a large number of possible values, the higher will be the information gain. To avoid this kind of distortion, C4.5 algorithm uses the ratio of entropy information gain, generating better classification results compared to ID3. To capture this benefit, the original ADAPTREE was modified to use the ratio of entropy information gain, but without pruning the branches as in C4.5 algorithm, resulting in small trees due to the incremental strategy.

Due to mutable nature of MANETs, the training set is seldom not known, requiring an unsupervised learning technique as SOM algorithm, to define the number of clusters or classes and map each service to only one class. This kind of map is called crisp clustering, where each data sample belongs to exactly one cluster. Fuzzy clustering [22] is a generalization of crisp clustering where each sample has a varying degree of membership in all clusters, as described in the following section.

2.3 Adaptive Network-Based Fuzzy Inference Systems (ANFIS) Algorithm

According [6], fuzzy if-then rules or fuzzy conditional statements are expressions of the form IF A THEN B, where A and B are labels of fuzzy sets [23] characterized by appropriate membership functions. Due to their concise form, fuzzy if-then rules are often employed to capture the imprecise modes of reasoning that play an essential role in the human ability to make decisions in an environment of uncertainty and imprecision.An example that describes a simple fact is “If pressure is high, then volume is small”, where pressure and volume are linguistic variables [24], high and small are linguistic values or labels that are characterized by membership functions.

Another form of fuzzy if-then rule, proposed by Takagi and Sugeno [25], has fuzzy sets involved only in the premise part. Where, again, high in the premise part is a linguistic label characterized by an appropriate membership function. However, the consequent part is described by a nonfuzzy equation of the input variable, velocity.

Both types of fuzzy if-then rules have been used extensively in both modeling and control. Through the use of linguistic labels and membership functions, a fuzzy if-then rule can easily capture the spirit of a “rule of thumb” used by humans. Basically a fuzzy inference system is composed of five functional blocks:

- A rule base containing a number of fuzzy if-then rules;
- A database which defines the membership functions of the fuzzy sets used in the fuzzy rules;

- A decision-making unit which performs the inference operations on the rules;
- A fuzzyfication interface which transforms the crisp inputs into degrees of match with linguistic values;
- A defuzzification interface which transform the fuzzy results of the inference into a crisp output.

Several types of fuzzy reasoning [26] have been proposed in the literature, but Takagi and Sugeno's (TS) fuzzy inference system (FIS) is used, once the output of each rule is a linear combination input variables plus a constant term, and the final output is the weighted average of each rule's output. Due to its simplicity and high performance, TS FIS is suitable for ANFIS building.

The acronym ANFIS derives its name from adaptive neuro-fuzzy inference system. Using a given input/output data set, the toolbox function ANFIS constructs a fuzzy inference system (FIS) whose membership function parameters are tuned (adjusted) using either a back propagation algorithm alone, or in combination with a least squares type of method. This allows your fuzzy systems to learn from the data they are modeling.

A network-type structure similar to that of a neural network, which maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs, can be used to interpret the input/output map.

The parameters associated with the membership functions will change through the learning process. The computation of these parameters (or their adjustment) is facilitated by a gradient vector, which provides a measure of how well the fuzzy inference system is modeling the input/output data for a given set of parameters. Once the gradient vector is obtained, any of several optimization routines could be applied in order to adjust the parameters so as to reduce some error measure (usually defined by the sum of the squared difference between actual and desired outputs). ANFIS uses either back propagation or a combination of least squares estimation and backpropagation for membership function parameter estimation.

In the proposed architecture for service selection, ANFIS uses the If-Then rules derived from ADAPTREE, where the service clusters were resulted from SOM unsupervised learning algorithm.

3 DAMS-SS Architecture and Selection Process

Search and selection tasks are normally performed at a intermediate layers between the network and application layer, named middleware. To reduce middleware complexity, service architectures were proposed to establish standards for service provisioning and consumption.

Service-Oriented Architecture (SOA) is a logical way of designing a software system to provide services to either end-user applications or to other services distributed in a network, via published and discoverable standard interfaces [27]. In a most known implementation of a Service-Oriented Architecture, a service description is based on the a standard called *Web Services Definition Language* (WSDL) to enable service discovery based on functional attributes in internet. Since WSDL supports only syntactical discovery, other languages were proposed to support semantic discovery based on functional ontologies, where *Ontology Web Language for Services* (OWL-S) [27] is one of the most known languages used for this purpose.

Additionally to service language definitions, the Quality of Service (QoS) description publishes important functional and non-functional service quality attributes. WSDL did not support non-functional attributes and OWL-S ontologies can be extend to support QoS features. Other descriptions languages were created to support both attributes, such as *Web Service Offering Language* (WSOL) [28].

From the service consumer perspective [29], the following four generic QoS criteria are considered for non terminal nodes in MANETs:

- **Availability** – The availability of a service is the probability that service is usable.
- **Price** – The price of a service is the fee that a service requester has to pay for using the service. The value of this QoS parameter is given by the service provider.
- **Reliability** - The reliability of a service is the probability that a service request is correctly responded, namely, the requester has received the expected results, within the maximum expected time frame indicated in the service description.
- **Execution Delay** - The delay of a service is a measure of duration between the time point when a service request is sent out and the time point when the results are received by the requester.

According [29], these criteria could be applied for atomic service or composed service selection. From the network nodes perspective, the following secondary criteria could be considered as a search refinement:

- **Node availability -** The availability of a node to its neighbor nodes is highly related to the node's mobility—here it is assumed that battery power is not a concern and as such a node will not out of reach, due to battery exhaustion.
- **Network Delay -** The delay of a packet in a network is the time it takes the packet to reach the destination after it leaves the source. Similar to that in fixed networks, the delay in wireless mobile networks is the sum of time spent at each relay on the route.
- **Network reliability –** The reliability is measured in this paper by the packet loss rate of the wireless link via which the service host is connected to the outside world.

Publication of such information about available services provides the necessary means for discovery, selection, binding, and composition of services. A discovery agency or service registry stores published descriptions, where *Universal Description Discovery & Integration* (UDDI) [30] is the registry standard in SOA for Web Services. A centralized UDDI discovery agency model implies some challenges to support dynamic discovery in MANETs, once it does not define any mechanism to propagate discovery queries toward the site where the corresponding information is stored. The efficacy of queries depends on the completeness of the information stored on the registry that is being contacted, on the precision of the query, and also on the ability of the requester to explicitly query different registries.

In order to implement a novel QoS based selection in MANETs, the proposed architecture extends SOA concepts, wrapping the combination of mining algorithms as middleware services to support and enhance exiting functional search solutions, as described in the following.

Dynamic Adaptive Middleware Services for Service Selection (DAMS-SS) architecture and process are described in the picture 1.

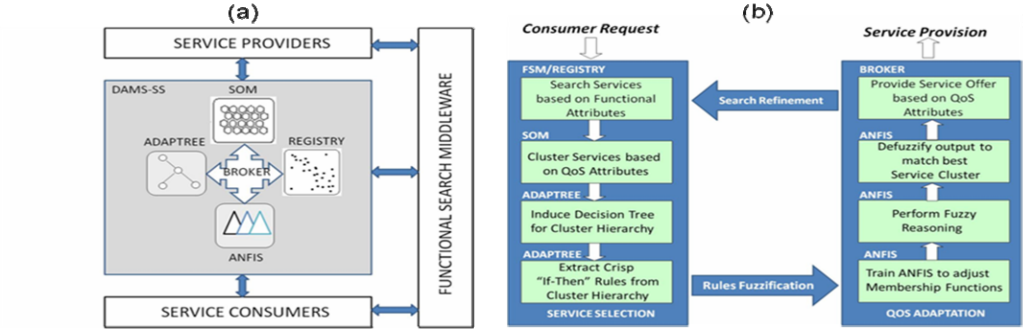

(a)
SERVICE PROVIDERS
DAMS-SS
SOM
ADAPTREE
REGISTRY
BROKER
ANFIS
SERVICE CONSUMERS
FUNCTIONAL SEARCH MIDDLEWARE
(b)
Consumer Request
FSM/REGISTRY
Search Services based on Functional Attributes
SOM
Cluster Services based on QoS Attributes
ADAPTREE
Induce Decision Tree for Cluster Hierarchy
ADAPTREE
Extract Crisp "If-Then" Rules from Cluster Hierarchy
SERVICE SELECTION
Search Refinement
Rules Fuzzification
Service Provision
BROKER
Provide Service Offer based on QoS Attributes
ANFIS
Defuzzify output to match best Service Cluster
ANFIS
Perform Fuzzy Reasoning
ANFIS
Train ANFIS to adjust Membership Functions
QOS ADAPTATION

Fig. 1. DAMS-SS Architecture (a) and Selection Process (b)

The main components in fig 1(a) are SOM, ADAPTREE and ANFIS algorithms encapsulated as services, using a service description language. These services can be assessed by service providers and consumers directly through the BROKER service, to perform the selection. Additionally, a service REGISTRY was necessary to store temporary the services resulted from a functional query from consumers to its Functional Search Middlewares (FSM). Many FSM solutions can be found in the literature, as SAMOA [31], MOBISOC [32], CAMPE [33], for example.

Based on SOA guidelines, the Web service description used is WSDL and BROKER must allow distributed, content-based, publish and subscribe messaging for query and advertise in MANETs dynamic environments.

There are many BROKER proposals in the literature, but REDS [34] fulfills all the requirements above. REDS, differently from the other similar middleware, natively offers the possibility of replying to messages. This feature can be very useful to transfer the result of a query from the resource owner to the requester, thus completely addressing the requirements of query–advertise.

Another key feature of REDS is the internal structure of its brokers, which are organized as a set of modules that implement well-defined interfaces and encapsulate the major aspects of Content-Based Routing (CBR). CBR holds the promise of addressing, at the same time and with a single routing infrastructure, the two issues we identified in SOAs: providing support to an asynchronous, publish–subscribe interaction style, and enabling complex searches to be executed in a completely distributed environment.

Fully content-based WS-Notification subscriptions and messages can be easily managed by *RedsNotificationBroker* (i.e., routed by REDS), while precise queries for exactly the services required can be formulated via *RedsUDDINode*, which routes them toward the right service providers. *RedsUDDINode* module was used also to implement service REGISTRY, as a distributed UDDI searched services.

As in datamining, the selection is a cyclic process, which starts with a consumer issuing a request and ends with a service offer to match it, as describe in Fig. 1(b).

In a reactive selection mode, a service consumer issues a service request based on functional attributes. The FSM reacts, searching in REGISTRY first, than in MANET nodes, services whose functional attributes best match the consumer request. The select services are stored or updated in the service REGISTRY.

In a proactive selection mode, FSM search available services within a search scope: context-based (e.g. SAMOA), people-based (e.g. CAMPE) or a combination of both (e.g. MOBISOC), instead reacting to a service request.

SOM service reads the REGISTRY, using BROKER services, and cluster available services using its unsupervised learning algorithm. The clusters are used to induce an adaptive decision tree using ADAPTREE service, generating a service cluster hierarchy, composed by QoS attributes on non terminal nodes, and service classes on terminal nodes (leafs).

The search refinement will be necessary when MANET became very instable, and only QoS parameters perceived by the user are not enough to fulfill consumer demands. SOM can cluster services based on both types of QoS criteria, if necessary to speed-up the search process, or use first user parameters than network parameters in an iterative mode.

Reading top-down the decision tree, If-Then rules are created by ADAPTREE and fuzzyfied to train ANFIS based on Membership Functions (MF) adjustments. Once adjusted, the MFs will allow ANFIS to match the best service cluster to consumer request fuzzy QoS statements as follows:

From users perspective:

- The highest availability and reliability;
- The lowest price and execution delay.

From Network perspective:

- The highest node availability and network reliability;
- The lowest network delay.

The ANFIS output defuzzification based on QoS parameters will set the boundaries of a “*n*” dimensional (N-D) fuzzy decision surface or region, where *n* is equal to the number of input parameters.

4 Implementation and Results

To implement a prototype of SOM and ANFIS services, Matlab® release 2006a SOM Toolbox 2.0 [36] and FUZZY Toolbox [6] was used. Matlab® was used also to simulate a MANET environment [37]. ADAPTREE was implemented using WEKA function libraries [38] and free software available from the author [7].To test core service selection algorithms, we assume that the functional search was done by some FSM mentioned earlier, and BROKER already stored the searched services in REGISTRY. SOM service evaluates 4 QoS parameters from user´s perspective of a sample 469 services [29] database, and identify 4 clusters, based on Davies-Bouldin index minimum value, as illustrated in Fig. 2(a).

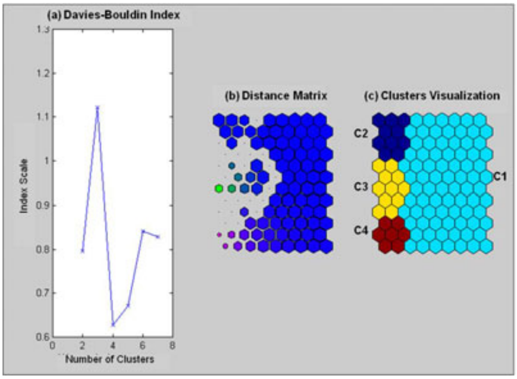


Fig. 2. SOM service results: (a) Davies-Bouldin index, (b) Distance Matrix and (c) Cluster Visualization

Fig. 2(b) illustrates the distance matrix between SOM units for cluster distance definition [8] and (c) the 4 clusters (C1 to C4) visualization over SOM units matrix. SOM unsupervised training algorithm identifies a Cluster for each service, used as input for ADAPTREE supervised training, as illustrated in Fig. 3.

In Fig. 3, "Ci:*X*" represents the number of services in each Cluster "i". "Low" and "High" represents the fuzzyfication rule of each induced branch of Decision Tree. *Info(X,T)* represents the information entropy measure used by ADAPTREE to induce another tree node, where "*X*" represents the number of services in each tree node and "*T"* the total amount of searched services in REGISTRY.

Reading the Decision Tree from root, then top-down scan until the leafs, the fuzzy "If-Then" can be collected as described in figure 4(a). The tree graphical representation

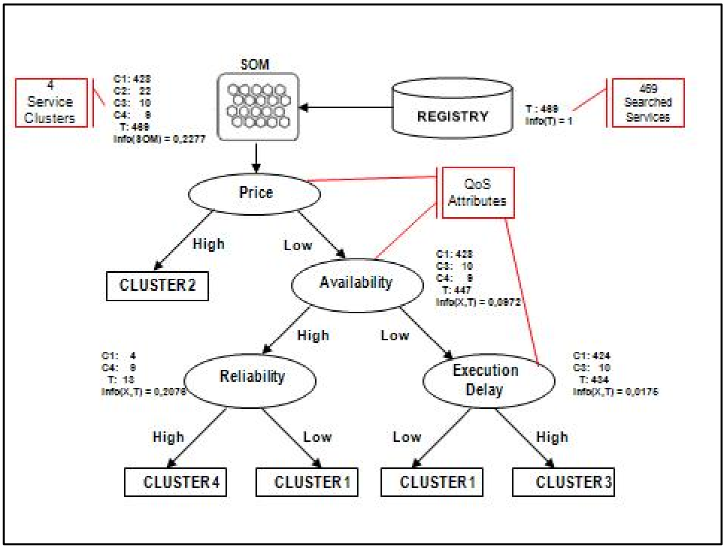
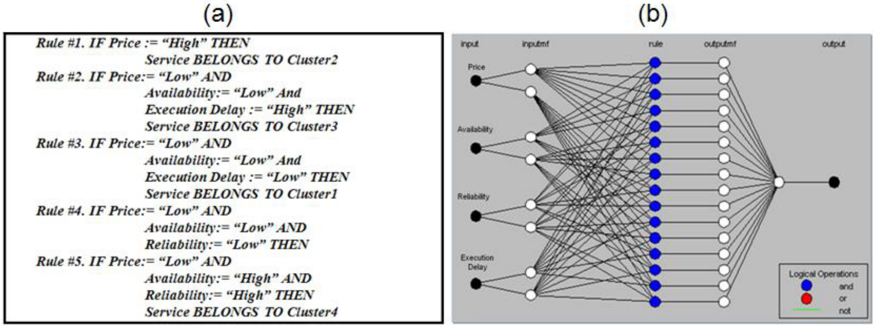


Fig. 3. Induced Decision Tree from SOM Clusters



Rule #1. IF Price := "High" THEN
Service BELONGS TO Cluster2
Rule #2. IF Price:= "Low" AND
Availability:= "Low" And
Execution Delay := "High" THEN
Service BELONGS TO Cluster3
Rule #3. IF Price:= "Low" AND
Availability:= "Low" And
Execution Delay := "Low" THEN
Service BELONGS TO Cluster1
Rule #4. IF Price:= "Low" AND
Availability:= "Low" AND
Reliability:= "Low" THEN
Rule #5. IF Price:= "Low" AND
Availability:= "High" AND
Reliability:= "High" THEN
Service BELONGS TO Cluster4

Fig. 4. Fuzzy "If-Then" rules from induced Decision Tree generated from ADAPTREE (a) and (b) structure of ANFIS system induced by ADAPTREE

is not necessary to generate the rules described in figure 4(a), but helps to understand ADAPTREE rule induction mechanism based on entropy gain.

The 5 rules described in figure 4(a), are a subset of 16 possible rules combining the 4 parameters with possible value "High" and "Low. These rules were used as inputs to train ANFIS output membership functions (MF), as illustrated in fig. 4(b).The "AND" logical operator was used to compute the "min" function of the 4 inputs, due to the inductive nature of the decision tree. The 469 services set in REGISTRY was divided into 3 parts randomly: 200 services for ANFIS training;169 services for ANFIS testing and 100 services for ANFIS checking.

The output results from each set of training can be found in Fig. 5, where "Output numbers" represent the identification of each service cluster (from 1 to 4).

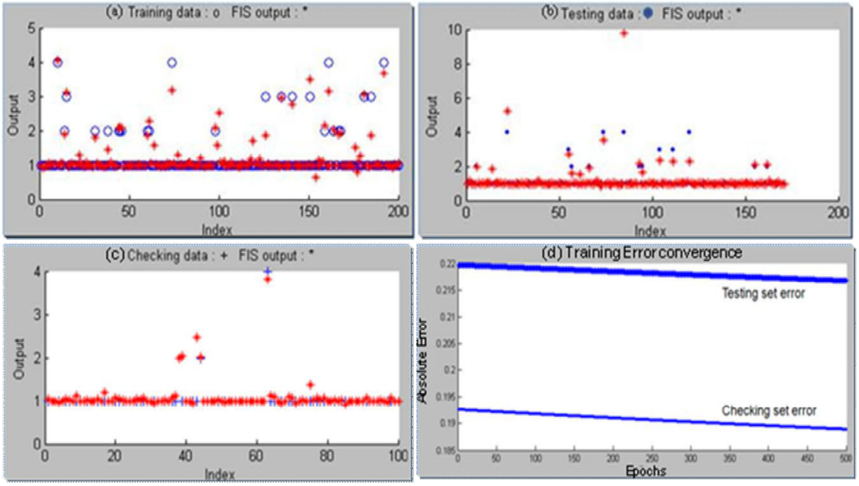


Fig. 5. (a) Output results from training and (b) Testing data set (c) Output results from checking data set and (d) Training Error convergence

In fig. 5(a), the output of ANFIS is compared to its related service cluster, as defined by SOM service. In fig. 5(b), another set of services is used for testing ANFIS after MF training. In fig. 5(c), another set of services, different for the training set and testing set, is used for checking ANFIS output accuracy. The testing and checking error decrease is slow, as illustrated in fig. 5(d), not justifying a increase in the number of training epochs.

Once ANFIS supervised training is finished, the output defuzzification, based on membership functions, set the boundaries of a fuzzy decision surface, as illustrated in fig. 6. In the fig. 6(a), Price and Availability QoS parameters were combined to define a 3-D fuzzy decision surface. In the fig. 6(b), Price and Execution Delay were combined to define another 3-D fuzzy decision surface. The combination of all 4 QoS input parameters and output, set the boundaries of a 5-D fuzzy decision region, where consumer request will be adapted to select the best service cluster based on consumer demands.

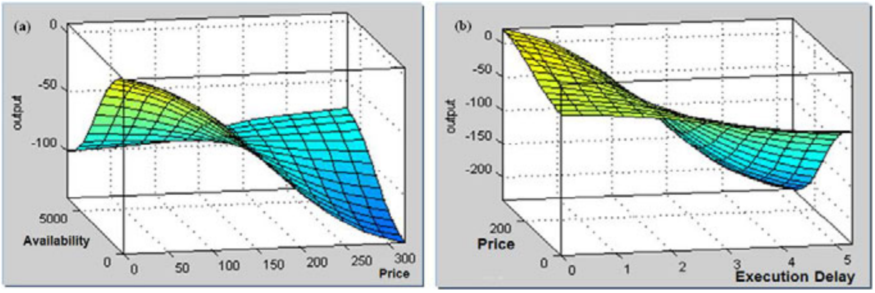


Fig. 6. Fuzzy decision surface combining different service QoS parameters

5 Conclusion and Future Work

Dynamic Adaptive Middleware Services for Service Selection (DAMS-SS) is a novel proposal to support service discovery in MANETs, based on a combination of tree well-grounded dataming algorithms. A Proof of Concept (PoC) implementation demonstrated DAMS-SS feasibility to enhance service selection process, combining unsupervised and supervised algorithms to perform fuzzy clustering based on incremental rule induction.

The proposed reactive and proactive selection method, supported by a service oriented architecture solution, where datamining algorithms were encapsulated as middleware services, refining current functional search middleware solutions.

The combination of SOM, ADAPTREE and ANFIS transformed data, gathered from MANETs, into comprehensible information to support consumer decision on best service choice selection, while reducing the drawbacks of standalone service mining implementations, as describe in the literature. Comparisons to other approaches were not possible, due to the novelty and uniqueness nature of proposed service mining algorithms combination.

Our future work includes a implementation on real mobile devices to evaluate algorithms memory consumption and possible performance issues. To measure the trade-off between accuracy and usability, we intend to investigate and experiment with more QoS parameters, for example, networks parameters, evaluating the advantages and disadvantages of proposed iterative selection process for MANETs environments.

Although designed for MANETs, the proposed solution could also be used for service selection in distributed environments with fixed infra-structure networks, to support discovery in other service-oriented architectures, such as cloud computing, once the combination of mining algorithms could be encapsulated using any service language description.

References

1. Mian, A.N., et al.: A Survey of Service Discovery Protocols in Multihop Mobile Ad Hoc Networks
2. Varshavsky, A., et al.: A Cross Layer approach to Service Discovery and Selection in Manets. In: Proc. 2nd Int'l Conf. Mobile Ad-Hoc and Sensor Systems (MASS 2005). IEEE Press, Los Alamitos (2005)
3. Capra, L.: MaLM: Machine Learning Middleware to Tackle Ontology Heterogeneity. University College London (2005)
4. Kohonen, T.: Self-Organizing Maps. Springer, Heidelberg (1995)
5. Cheung, R., et al.: A fuzzy service adaptation engine for context-aware mobile computing middleware. International Journal of Pervasive Computing and Communications 4(2), 147–165 (2008)
6. Jang, J.S.R.: Adaptive Network-base Fuzzy Inference System. IEEE Transactions on Systems, Man, and Cybernetics 23(3) (May/June 1993)
7. Pistori, H., Neto, J.J., Pereira, M.C.: Adaptive Non-Deterministic Decision Trees: General Formulation and Case Study. INFOCOMP Journal of Computer Science, Lavras, MG (2006)
8. Vesanto, J., Alhoniemi, E.: Clustering of the Self–Organizing Map. IEEE Transactions on Neural Networks 11(3), 586–600 (2000)
9. Buhmann, J., Kühnel, H.: Complexity optimized data clustering by competitive neural networks. Neural Comput. 5(3), 75–88 (1993)
10. Bezdek, J.C.: Some new indexes of cluster validity. IEEE Trans. Syst., Man, Cybern. B 28, 301–315 (1998)
11. Milligan, G.W., Cooper, M.C.: An examination of procedures for determining the number of clusters in a data set. Psychometrika 50(2), 159–179 (1985)
12. Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Trans. Patt. Anal. Machine Intell. PAMI-1, 224–227 (1979)
13. Kangas, J.A., Kohonen, T.K., Laaksonen, J.T.: Variants of self-organizing maps. IEEE Trans. Neural Networks 1, 93–99 (1990)
14. Martinez, T., Schulten, K.: A neural-gas network learns topologies. In: Kohonen, T., Mäkisara, K., Simula, O., Kangas, J. (eds.) Artificial Neural Networks, pp. 397–402. Elsevier, Amsterdam (1991)
15. Fritzke, B.: Let it grow—Self-organizing feature maps with problem dependent cell structure. In: Kohonen, T., Mäkisara, K., Simula, O., Kangas, J. (eds.) Artificial Neural Networks, pp. 403–408. Elsevier, Amsterdam (1991)
16. Cheng, Y.: Clustering with competing self-organizing maps. In: Proc.Int. Conf. Neural Networks, vol. 4, pp. 785–790 (1992)

17. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Francisco (1992)
18. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth and Brooks, Monterey (1984)
19. Utgoff, P.E., et al.: Decision tree induction based on efficient tree restructuring. Machine Learning 29(1), 5–44 (1997)
20. Basseto, B.A., Neto, J.J.: A stochastic musical composer based on adaptative algorithms. In: Anais do XIX Congresso Nacional da Sociedade Brasileira de Computação, SBC 1999, PUC-RIO, Rio de Janeiro, Brazil, vol. 3, pp. 105–130 (July 1999)
21. Jackson, Q.T.: Adaptive predicates in natural language parsing. Perfection (4) (2000)
22. Costa, E.R., Hirakawa, A.R., Neto, J.J.: An adaptive alternative for syntactic pattern recognition. In: Proceeding of 3rd International Symposiumon Robotics and Automation, ISRA, Toluca, Mexico, pp. 409–413 (September 2002)
23. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
24. Zadeh, L.A.: Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans. Syst., Man, Cybern. 3, 28–44 (1973)
25. Takagi, T., Sugeno, M.: Derivation of fuzzy control rules from human operator's control actions. In: Proc. IFAC Symp. Fuzzy Inform., Knowledge Representation and Decision Analysis, pp. 55–60 (July 1983)
26. Lee, C.C.: Fuzzy logic in control systems: Fuzzy logic controller-Part I. IEEE Trans. Syst., Man, Cybern., 20, 404–418 (1990)
27. Papazoglou, M.P; et al, Service-Oriented Computing Research Roadmap. In: Dagstuhl Seminar Proceedings 05462 Service Oriented Computing (SOC) 2006
28. Patel, K.: Improvements on WSOL Grammar and Premier WSOL Parser. Research Report. SCE-03-25 (2003)
29. Yang, K., et al.: QoS-Aware Service Selection Algorithms for Pervasive Service Composition in Mobile Wireless Environments. Mobile Netw Appl. (2009)
30. Clement, L., et al.: UDDI Version 3.0.2, Tech. Rep., OASIS (2004), `http://uddi.org/pubs/uddi-v3.0.2-20041019.htm` (last access in January 2010)
31. Bottazzi, D., Montanari, R., Toninelli, A.: Context-Aware Middleware for Anytime, Anywhere Social Networks. IEEE Intelligent Systems 22(5), 23–32 (2007)
32. Gupta, A., Kalra, A., Boston, D., Borcea, C.: MobiSoC: A Middleware for Mobile Social Computing Applications. In: Mobilware 2009 (2009)
33. Bottazzi, D., Montanari, R., Giovanni, R.: A self-organizing group management middleware for mobile ad-hoc networks. Computer Communications 31, 3040–3048 (2008)
34. Cugola, G., Nitto, E.D.: On adopting Content-Based Routing in service-oriented architectures. Information and Software Technology 50, 22–35 (2008)
35. Yang, K., et al.: QoS-Aware Service Selection Algorithms for Pervasive Service Composition in Mobile Wireless Environments. Mobile Netw Appl. (2009)
36. Vesanto, J., Alhoniemi, E., Himberg, J., Parhankangas, J.: Som Toolbox 2.0 BETA online documentation (1999), `http://cis.hut.fi/projects/somtoolbox`
37. Free software available in, `http://wireless-matlab.sourceforge.net/` (last access in January 2010)
38. Free software available in, `http://www.cs.waikato.ac.nz/ml/weka` (last access in January 2010)

Session 5: Location-Aware and Context-Aware Networking and Computing
(Chair: Chris Thompson)

An Object-Oriented Model in Support of Context-Aware Mobile Applications

Felix Dobslaw[1], Aron Larsson[1,2], Theo Kanter[1], and Jamie Walters[1]

[1] Mid Sweden University, Sweden
[2] Stockholm University, Sweden
{felix.dobslaw,aron.larsson,theo.kanter,jamie.walters}@miun.se

Abstract. Intelligent and context-aware mobile services require users and applications to share information and utilize services from remote locations. Thus, context information from the users must be structured and be accessible to applications running in end-devices. In response to this challenge, we present a shared object-oriented meta model for a persistent agent environment. The approach enables agents to be context-aware facilitating the creation of ambient intelligence demonstrated by a sensor-based scenario. The agents are context-aware as agent actions are based upon sensor information, social information, and the behavior of co-agents.

Keywords: Context-Modeling, Context-Awareness, Ontologies, Ubiquitous Computing.

1 Introduction

The growing number of mobile devices and users; the increase in the devices' computational power and the use of Internet as a platform together with new and efficient sensor techniques opens new possibilities for creating intelligent sensor-based mobile services. Recently, vendors of mobile devices such as smart phones and PDAs have started to embed sensors including accelerometers, GPS, light, and short-range radio in the devices. Thereby we are in a position to further exploit access to sensor information in intelligent services. Users of electronic services are also expecting applications and services to provide more value by means of taking into account social and real-time information about the user's real (or perceived) environment.

With the proliferation of mobile services and on-line communities, there is a growing need to be able to provide more customized services based on user's context information in a broader sense than by purely physical data. For instance, services ranging from simple ones such as delivering customized and location-based advertisements, to more complex tasks including social interactions over time and access to shared digital content and media in an intelligent manner. Access to both sensor information and social contexts can substantially increase the perceived value of mobile services to users. Sensor information may originate from user devices equipped with sensors or from sensor networks surrounding

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 205–220, 2010.

the user; providing a physical context. On-line services that manage a social context further provide individuals with an accessible *virtual life*. The exploitation of such information in order to provide intelligent services to users is commonly referred to as ambient intelligence [1]. Ambient intelligence requires a context-aware environment since context information describing the physical context of a user originates from sensor information. In such an environment, the integration of sensors and sensor information is an important feature in order to create physical situational context-awareness.

1.1 Motivation

The work behind this paper is mandated by the fact that mobile services benefit from having access to information regarding both a user's situation and intentions, as well as information regarding the user's social activities in order to support the user in achieving tasks. In order to reach this, a framework enabling intelligent mobile services must be able to support reasoning with sensor information and social information simultaneously. For this purpose, we advocate a common context model incorporating both these aspects of contextual data.

The interest in ambient intelligence as a fundamental facilitator of intelligent mobile services has increased substantially; and the idea of utilizing sensor information in mobile devices is not novel. As for technical solutions enabling the sharing of physical context data; the MobiLife project [2] realized the provisioning sensor information via 3G mobile systems. However, the solution proposed is based upon web-services incapable of maintaining the correspondence between the location of the end-devices and the reachability of sensor information.

Other approaches include SenseWeb [3] and AmbieSense [4]. With respect to the former, applications can initiate and access sensor information streams from shared sensors across the entire Internet. The SenseWeb infrastructure provides support for an optimal sensor selection for each application and the efficient sharing of sensor streams among multiple applications based on web services. However, this is achieved using centralized solutions which negatively impacts on scalablilty. The approach of the AmbieSense project involves the use of so called *context tags* which is a special-purpose hardware device (and a small wireless web-server) mounted in the surroundings and communicating with adjacent mobile devices. In this way, the relevant information can be provided to mobile users based upon the users' current context. While this provides an acceptable means of gathering context information, the requirements with respect to specific hardware, will impact on user adoption and leaves open the possiblity of deriving context in a more seamless manner based on information that is derivable through the deployment of software over existing mobile devices.

With respect to context models and agent environments in [5], an ontology based context model and an architecture for the specification of intelligent context-aware services is presented. The approach to context modeling presented in this paper goes beyond existing approaches, emphasizing the model's extendibility combined with the ability to translate context information with similar semantics but differing syntaxes. It also addresses the potential issues with regards to query processing times that are inherent with ontology approaches.

The involving of sensing and reacting on the environment has much in common with the field of artificial intelligence; in particular intelligent agent theory and multi-agent systems. In this area, context-awareness of an intelligent agent refers to the agent having the ability to sense and react (perform actions) based upon the state of the environment, cf. [1,6]. Portable devices, such as mobile phones, could then be made seemingly context-aware to the extent that software services implemented in the devices are triggered by context-aware agents and their actions.

For a distributed agent environment to operate, a number of basic issues need to be resolved in order to enable and facilitate reasoning in such systems. These basic issues, already highlighted in [7] and [8], include: 1) How to make agents in a distributed system context- aware, 2) How to enable these agents to communicate and interact, and 3) How to facilitate for these agents to reason about sensor and social information. Of particular concern and what constitutes the main challenge in this paper is to show how the first issue is resolved combining sensor and social data.

An additional challenge involves providing the means for the spontaneous sharing of context information from any locally available sensor among agents, also enabling sharing of social information with other agents and enable reasoning with such contextual information. Sharing of remote and local context information requires a context model with the ability to capture the context of users or services and also maintain and update this context during which the service operates. In order to enable reasoning, a context-base and general context model also serving as an agent environment for persistent and context-aware agents is proposed in this paper.

The paper is structured as follows. Section 2, Approach, describes the proposed CII-model and its components in detail. This is followed by an account for how the model could serve as an agent environment and a scenario, highlighting the models features. Section 3, Realization of the Approach, first explains the general architecture for a context-base based on the CII-model, followed by how this context-base provides a necessary foundation for the desired functionality of the MediaSense framework. Section 4 concludes the paper with a summary of its contributions and an outlook on future research and development.

2 Approach

In this paper, a user is the holder of a mobile device and a service is the result of a set of actions initiated by a set of agents which are exploited by a particular application. We conform to the definition of context as given by [9], however slightly extending it to include interactions between applications as this is an important feature of the proposed framework. Thus, semantically speaking, context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application or between two applications; including the user and applications themselves.

Our approach for meeting our objectives comprises of a distributed system of agents as first-class objects reasoning over sensor and social information. We conform to the agent paradigm given in [10]; in that an agent is an entity that can perceive its environment through sensors and acts upon that environment through actuators. Thus agents are defined as autonomous entities that make decisions upon perceived stimuli and desires. The environment, or the shared structure of the agents, is the base for enabling agent reasoning and intelligent services. The proposed agent environment (The Context Information Integration Model) operates on an object-oriented context-base organized as an ontology that supports integration of context from customized multi-dimensional domains. The agents are software objects spawned in the context-base; able to move between end devices to be executed locally as a service. They can reason upon the global context by remote access to the object meta model. See Figure 1 for an illustration of the overall approach.

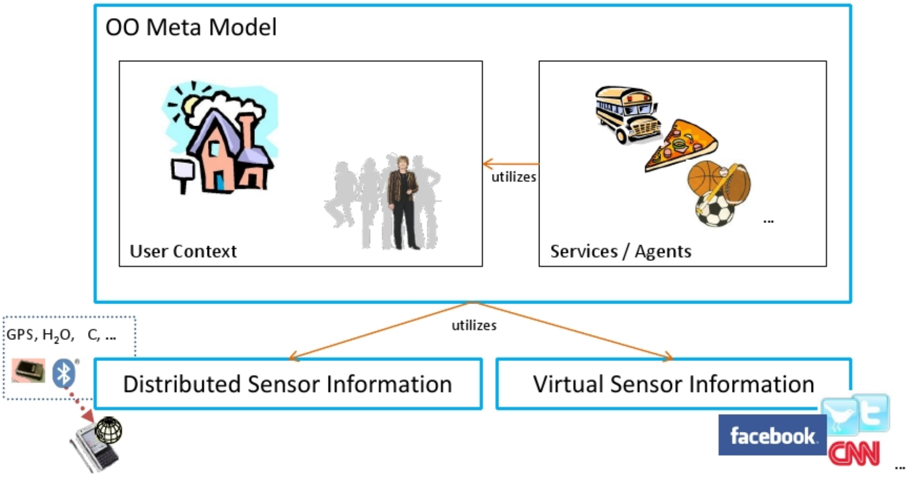


Fig. 1. Illustrative overview of the approach. A user's context is captured from a multitude of sources. Agents are persistent and active objects in the context-base.

2.1 Ontologies

An ontology is a 'formal, explicit specification of a shared conceptualization' [11]. This shared conceptualization or understanding can help with overcoming communication barriers between people, organizations and software systems. Each party in a plot tends to have a particular viewpoint on a matter and diverse, possibly overlapping concepts. This fact constitutes misunderstandings and thereby poor communication, implying 'difficulties in system specification' [11].

The most basic exemplification regarding that, is the bidirectional communication between two individuals not sharing a common language. The language in this scenario contains none or partly overlapping concepts. The application of an ontology would allow for the specification of a meta-language (a shared understanding, an Esperanto) and solve related problems, given that the metalanguage is sufficiently rich in expression. Concepts and relationships for ontologies can vary in representation from highly informal (e.g. spoken language) to rigorously formal (formal semantics, axioms, theorems and proofs of important properties), depending on the purpose of the model. As different problem domains may have syntactically different concepts with the same or similar semantics, enabling overarching context-awareness in several problem domains simultaneously requires some technique for translating between syntaxes and units. When assimilating syntactically different concepts with the same or similar semantics, a translation is required to enable their content to be translated from and to one another. A pattern for the generic resolution regarding this problem is used in the proposed CII-model. This pattern is referred to as Inter-Lingua. Inter-Lingua describes the necessity for one central, globally accepted representation. For each syntactical representation of a concept not conforming with the standard, a translator to the inter-lingua concept and vice versa must exist. In this manner, a valid translation from each model to another is guaranteed. This considerably reduces the system's translation complexity (see [11]).

2.2 The Context Information Integration Model

The Context Information Integration (CII) model, is an ontology with respect to the above definition. Figure 2a shows the concepts in UML class-diagram notion. Top-down, the model is partitioned into three components: *Internet of Things*, *Sensed Context* and *Context Meta Data*. Within the *Internet of Things*, concrete instances of concepts and their relationships to each other can be modeled. The result is a graph where nodes represent concept instances and the directed edges are Resource Description Framework (RDF) triples $\langle subject, predicate, object \rangle$, the standard for resource interrelations on the Semantic Web [12].

An example of a resulting graph can be seen in Figure 2b. Sensed Context synthesizes all physical, virtual and logical sensors as classified in [13]. An example of a physical sensor is a thermometer whereas a virtual sensor could be a Twitter feed. Logical sensors are more sophisticated, integrating different data sources or sensors; for instance, in order to draw conclusions about the state of health of a patient. The Context Meta Data component specifies guidelines for the extension and integration of new concepts into the existing ontology. The following subsections provide a component wise explanation of the concepts.

Sensed Context. `Information Sources` are the perceiving objects in the model. Each `Information Source` holds one value at a time; an instance of `Context Value`. An `Information Source` could be any physical sensor, such as a sensor for GPS, but it could also be any type of source on a local machine or the Internet revealing information such as user profiles and preferences. `Context`

Values represent sensed context data in the model, for instance, a GPS coordinate $(123.23, -12.34, 144.40) \in \text{longitude} \times \text{latitude} \times \text{altitude}$. Furthermore, the model allows for Information Source inheritance, and thereby its extension by specific sub-concepts. GPS Sensor and Camera are sub-concept examples for Information Source.

Internet of Things. Entities represent the things in the model to which context can be attached. Predicates give relations between two Entities a semantic meaning, forming subject-predicate-object triples that allow for social information. The order in which they are written is relevant (symmetry is not implied). Each Entity can be related to multiple other Entities in two ways, either as a subject or an object of a triple. Entities in the model allow for inheritance in the same way as Information Sources do.

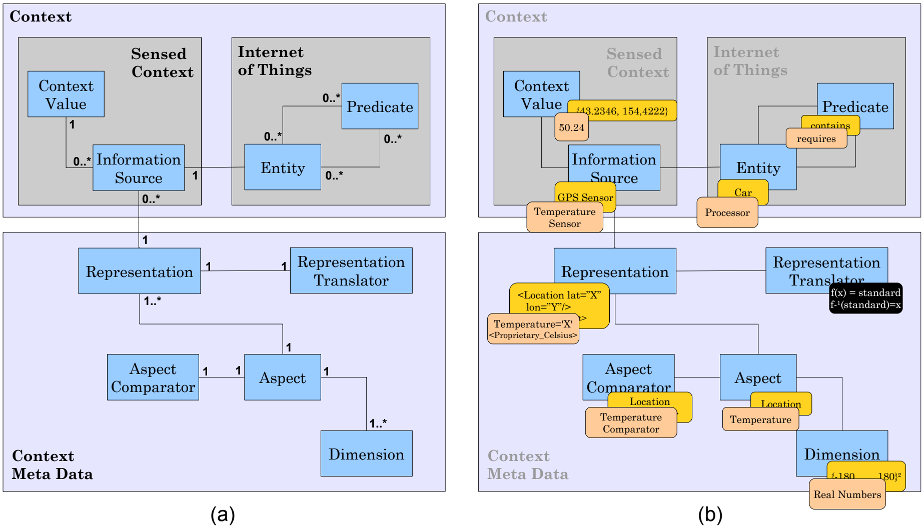


Fig. 2. (a) Shows the CII-model, represented as a UML class diagram. The concepts (classes) are categorized according to their role in the model as *Sensed Context*, *Internet of Things* and *Context Meta Data*. (b) An example of how model components could look like. A simplified scenario could involve a processor with a temperature sensor (red) and a Car with a GPS sensor (orange).

Context Values can be attached to and detached from Entities. The same accounts for relations between Entities, allowing for context dynamically changing over time.

Context Meta Data. The central meta data concept in CII is Aspect. Such an Aspect could be *location*. Each Aspect has any desired amount of Dimensions and Representations, requiring at least one of each. One particular Representation has to be specified as the Aspects *standard*. The *standard* serves as the *inter-lingua* for the integration of different Representations, and thereby resolves any syntactical, quantitative, qualitative or unit based distinctions with respect to one Aspect.

The Dimensions specify the valid domain for all Context Values of a particular Aspect. The three Dimensions for Aspect *location* could be called longitude, latitude and altitude; assuming context values within the domain $\{-180, \ldots, 180\}$ for longitude and latitude, as well as "meters over sea level" for altitude ($\mathbb{R}$). The largest valid domain for a Context Value is the Cartesian product of all its Dimensions.

Representations are templates for the presentation of sensed context data. Each Information Source holds a Context Value of one particular Representation. An example Representation for the Aspect *location* could be called *gmlPoint*, where the Context Values template might look like the standard Google representation for location data:

```
<position>
    <gml:Point srsDimension="3">
    <gml:pos>X Y Z</gml:pos>
    </gml:Point>
</position>
```

X, Y, Z stand for the longitude, latitude and altitude values, respectively.

Representation Translators are responsible for the translation from one particular Representation of an Aspect to the standard Representation (*mapToStandard*) and vice versa (*mapFromStandard*). In the following example two Representations, *gmlPoint* and *proprietary*, for an Aspect *location*, are considered. *gmlPoint* serves here as the *standard*. The Representation Translator for *proprietary* has to supply the two functions

$$f_{trans}(x_{proprietary}) = y_{gmlPoint} \tag{1}$$

$$f^{-1}_{trans}(y_{gmlPoint}) = x_{proprietary} \tag{2}$$

The Representation Translator ensures the transitive translation closure for all Representations of one particular Aspect, meaning that the model itself is able to translate between all types of Representations for that Aspect. This is a key feature for the integration of context in different formats and units.

The Aspect Comparator enables CII to operate between Context Values of the same Aspect even if they have different representations (units, syntaxes). It incorporates the functionality of ordering context values with respect to the shared Dimensions. Therefore, it is required to specify an Aspect Comparator for each Aspect. The Aspect Comparator enables that CII, without knowing anything about the *Aspects* or context contents, is capable of optimizing its retention. The Aspect Comparator provides two functions. One to compare two Context Values, returning their order in relation to an implicit but well-defined metric (*compareTo*). The other function returns the relative distance between two Context Values with respect to that particular metric (*getDistance*). This data could be of special interest to applications that only know little about a certain, newly explored, Aspect. For instance, an agent could use distances and orders for reasoning and decision making. Naturally, comparison is only possible for measurable data. Figure 2b adds two modeling examples to the general model figure. By extending the CII-model, its semantic expressiveness increases.

2.3 How Context Is Expressed

The real essence of the model is to provide `Entities` with contextual meaning. It is important to link the structures of the CII-model to these meanings, so as to obtain a correlation between the syntactical constructs and their semantic interpretation. This is done in the following definitions.

Definition 1. *The **context** of an `Information Source` at a moment in time is defined by its `Context Value`.*

Definition 2. *The **simple-context** of an `Entity` at a moment in time is the set of all attached `Information Sources` contexts.*

Definition 3. *The **context** of an `Entity` at a moment in time is its simple-context including the simple-context of all `Entities` in its network proximity under consideration of the type of their relationships. How proximity is defined, depends on the eye of the beholder.*

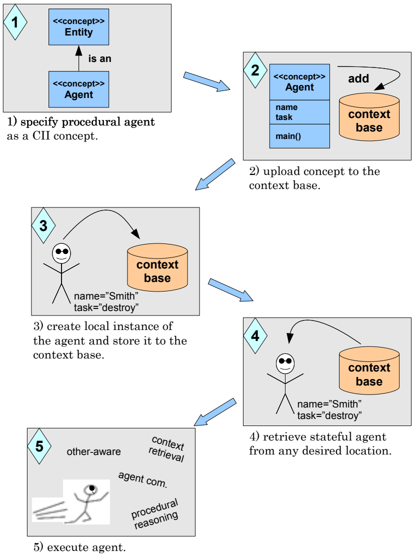
The context of an `Entity` is not expressed in an entirely explicit manner. Context may depend on the subjective interest of the user or other stakeholders, which is why context is partially implicit in the CII-model. That is also the reason why *situations*, unlike other approaches [14], are not taken into account inside the CII-model.

The model viewed at a moment in time always represents a snapshot of the current overall context. It is possible to extend CII, expressing historical context for `Information Sources`. When extending CII with respect to historical context, the 1..1 relation between `Information Source` and `Context Value` becomes a 1..$*$ correlation. Timestamps mark the `Context Values`, so as to allow the determination of their time-frame of validity.

2.4 The CII-Model as an Agent Environment

We assume an environment in which many users (possibly millions) are concurrently following certain interests, where the agent's main objectives are to serve these interests as well as possibly fulfilling their own desires. The importance of the environment in multi-agent systems has been raised in [15,16], arguing in favor of the environment as a first-order abstraction having two different roles: 1) providing the surrounding conditions for agents to exist and 2) to provide an exploitable design abstraction for building multi-agent applications. Herein, the environment is defined by the CII-model extended with protocols for agent communication, conforming with the Agent Communication Language (ACL) agents communication standards from the Foundation for Intelligent Physical Agents (FIPA) [17]. The CII-model serves as a first-class abstraction: it is an independent building block holding different responsibilities than those of the agents within the system. This means that the behavior of the system is not only a result of the combined actions and desires of the agents; as rules for agent activities with respect to, e.g., accessibility and communication, are defined by the CII-model as the environment.

In this framework, physically an agent is a first-class object represented as an active `Entity` in the context-base. Thus, the existence of the environment is not dependent on the agents in the system, the agents are rather seen upon as a part of the environment. By active, we mean that an agent, as oppose to passive entities (or ordinary objects), is capable of perceiving events, perform actions and make commitments, cf. [18]. Hence, an agent is viewed as a specialization of an object's building on an object-oriented approach, analogous to the agent paradigm of multi-agent Systems Engineering (MaSE) for the analysis and design of multi-agent systems in general, see [19]. This can be realized by hybrid agents that are part of the CII-model, as well as the Java Agent Development Environment (JADE), by inheritance. JADE is an object oriented agent environment, based on the standards imposed by the FIPA [20]. Of further importance, as an agent is represented as an `Entity`, is that the agent's internal states can be persisted which serves as a condition for agent recovery, resilience, accessibility and mobility.



1) specify procedural agent as a CII concept.

2) upload concept to the context base.

3) create local instance of the agent and store it to the context base.

4) retrieve stateful agent from any desired location.

5) execute agent.

Fig. 3. Illustration of how the agent paradigm concept can be enabled by the CII-model. 1) The agent's internal logic is locally defined, 2) The agent class is persisted, 3) Agent instances can be created and persisted, 4) Agents can be globally retrieved. Other-awareness can be enabled as each agent has an accessible internal state as well as access to the global context information.

2.5 Scenario

A potential scenario benefiting from the CII-model would be a traveler/commuter scenario. Commuters are interested in traveling conveniently, regarding price or (and) travel time. Since public traffic requires the concurrent usage of resources (streets, rails etc.) which is affected by dynamically changing conditions (weather, building sites etc.), planning a multi-hop trip in advance adds a realistic chance of delayed arrivals and subsequently missed connections.

Travelers, equipped with mobile devices, require a service that keeps them up to date about the current opportunities in case of an unplanned schedule change. We suggest the modeling of this service as an agent, such as discussed in 2.4.

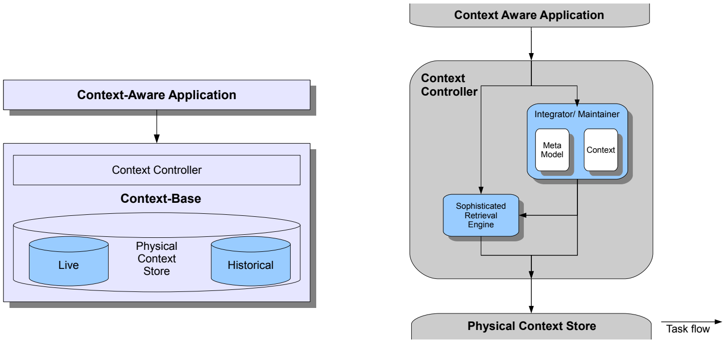
When changing from regional busses to local busses, in a city, providers don't always share the same data model or services for the finding of direct connections. However, they could offer agents with a standard interface to allow for context-based recommendations for ad-hoc journeys. Busses are equipped with sensors such as GPS, etc. These sensors are represented in the model as sub-concepts of `Information Source`; `Bus` and `Traveler` as sub-concepts of `Entity`. Each instance is represented by an unique identifier. Travelers are registered to be on a bus by adding RDF triples to the context model $\langle travelerID, isOn, busID\rangle$. In 3.3 we present an approach for how the travelers devices can interact with the sensors, retrieving their context values, updating them to the context model. In response to the state of their computing environment with respect to resources, agents may be executed locally on a device; be opportunistically mobile or in some cases have their state persisted until resources become available.

The advantage of modeling in CII here is that context information can be arbitrarily shared with other domains, since the model is based on the same ontological meta concepts. Car rental, emergency, or event-services could derive their conclusions from, or manipulate the same context pool. Not only could the idea be further extended to air traffic, cab services or trains; with the case of a bus accident, passengers with special needs could be identified since the context database would reveal that they were on the bus. Thus, the ambulance rushing to the crash site could be equipped with the required medication or apparatus. The extendibility of the model allows for all stakeholders to support different formats and units for how the context is expressed, so that existing modules can remain untouched (e.g., kml vs. gml for location information).

3 Realization of the Approach

3.1 The Context-Base

Our proposed architecture for a context-base is data-centric, comprising three layers: the context-aware application, the context-base controller and the physical context storage, see Figure 4a. A data-centric approach is applicable to our problem of global context provisioning, since it addresses the communication via a common repository [21]. However, this repository does not have to reside on a central point. The context-base can be viewed as a middle-ware that hides complexity regarding the distribution of data, processing, application and control from context consumers/providers, providing a high level of distribution transparency. From the perspective of a context-aware application, the context-base is a rich service for the management of context and context models. The application could be: a) manually controlled by end users, or b) perform automated, context related tasks. Developers of context-aware applications either reuse existing domain-model concepts from the context-base or extend the global model. Figure 4b shows the context controller with a higher granularity. The controller has two main tasks with respect to the functionality outlined in 2.2:



(a) (b)

Fig. 4. (a) Views the proposed layered architecture for context attainability, utilized via a context-base, encapsulating the CII-model. (b) Depicts the context controller component. It is responsible for the creation and maintenance of the meta model as well as the integration of domain model components and likewise for the provision of the retrieval engine to the context-aware applications.

1. Handling data operations, mediating between external requests (data from information sources) and the physical data store.
2. Maintaining the context model including alterations, extensions and validity enforcement inside the context-base.

For both tasks, a sophisticated retrieval engine is essential, allowing for access to the global context. The controller is preferably thin and stateless in order to enable the distribution of multiple controllers. The idea behind using multiple controllers is to support concurrent access to the global context-base while offering high scalability. Where the Context Integrator/Maintainer is dealing with Task 1, Task 2 is dealt with by the Meta Model Integrator/Maintainer component. It is crucial to deal with both resilience and robustness as, within a distributed scope, flawed model extensions should only have local impact and should not harm the overall usage of the context-base.

3.2 User Interfaces

Two interfaces are presented, placing the CII functionality at the disposal of the context prospects. From the perspective of the context-base, there are two user groups: context providers and context consumers. Providers create domain models and reuse existing model extensions, and thus modify the schema itself. Consumers query, add, modify, and remove context. Hence, consumers only request or change the context information itself. Context providers supply the consumers with services/applications that are, to some extent, context-based or context-aware. The interface is split in two with respect to this classification (see Figure 5).

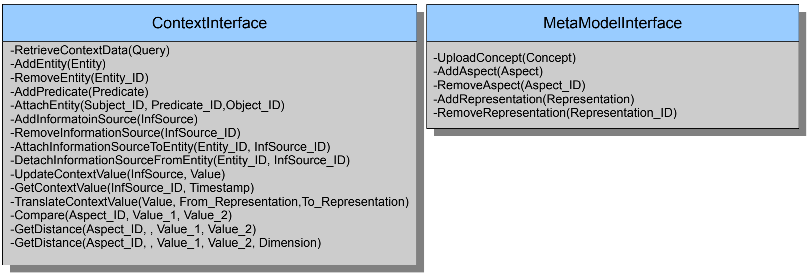

ContextInterface
-RetrieveContextData(Query)
-AddEntity(Entity)
-RemoveEntity(Entity_ID)
-AddPredicate(Predicate)
-AttachEntity(Subject_ID, Predicate_ID,Object_ID)
-AddInformatoinSource(InfSource)
-RemoveInformationSource(InfSource_ID)
-AttachInformationSourceToEntity(Entity_ID, InfSource_ID)
-DetachInformationSourceFromEntity(Entity_ID, InfSource_ID)
-UpdateContextValue(InfSource, Value)
-GetContextValue(InfSource_ID, Timestamp)
-TranslateContextValue(Value, From_Representation,To_Representation)
-Compare(Aspect_ID, Value_1, Value_2)
-GetDistance(Aspect_ID, , Value_1, Value_2)
-GetDistance(Aspect_ID, , Value_1, Value_2, Dimension)
MetaModelInterface
-UploadConcept(Concept)
-AddAspect(Aspect)
-RemoveAspect(Aspect_ID)
-AddRepresentation(Representation)
-RemoveRepresentation(Representation_ID)

Fig. 5. The two interfaces that allow for context access. ContextInterface (left) allows for the treatment of context, where MetaModelInterface (right) dealing with the treatment of the meta model.

Context Interface. The interface should be made available to all applications having an interest in context storage and retrieval. It allows for context retrieval via a dedicated method *RetrieveContextData* that returns objects. Concrete `Entities` and `Information Sources` can be initialized and stored in the context-base together with their interrelations. `Context Values` can be updated via the ID:s of their respective `Information Sources` (*UpdateContextValue*). Current or historical values can be retrieved by the provision of the ID and the desired timestamp of validity (*GetContextValue*). For the launching of agents, methods for translation, comparison, and distance computation are provided.

Meta Model Interface. The functionality of the Meta Model Interface should be restricted to a system modeler with permission to alter the meta model. `Aspects` and `Representations` can be added, modified, and removed on demand. The most significant method is *UploadConcept*, which allows for a domain level modeler to extend the model with new sub-concepts of `Entity` and `Information Source`, or the addition of `Aspect Comparators` and `Representation Translators`. In an object oriented environment, these concepts would be presented as classes following a standard interface. This integration would be made possible by adaptive software techniques such as a combination of computational reflection, automated compilation and dynamic class loading.

3.3 The Extendable Model Agent Environment

Providing the necessary structure and technical architecture for enabling intelligent services, focusing on the incorporation and utilization of sensor information is the initial main issue of the MediaSense project [22]. These solutions can at present be summarized into the use of a wireless sensor network gateway (WSNG), utilizing low-energy Bluetooth communication for providing mobile devices with various forms of sensor information, in combination with the Distributed Context eXchange Protocol (DCXP) enabling distributed sharing of context information in real-time. Hence, agents update their state and behavior in response to exchanges of sensor information from other agents or sensor

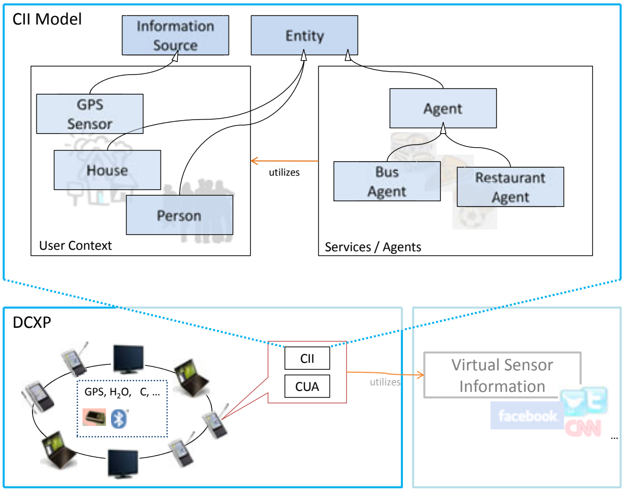


Fig. 6. The DCXP overlay and CII architecture in coordination as a means to globally share context related information that can be persisted and retrieved on demand

sources via DCXP. The framework is event-driven in that it supports publication, discovery, and (conditional) subscription to sensors. The context information may thus be collected from sensors (or sensor networks) that are wirelessly attached to mobile devices through the WSNG.

Although the framework outlined in this paper is done relative to the MediaSense framework, in this paper we define a service as a product of an application in order to serve a user. Hence, this paper treats the architecture of, and conditions for, enabling ambient intelligence, conforming to the paradigm of multi-agent systems and distributed artificial intelligence, with the possibility of utilizing the technical solutions proposed in the MediaSense framework. See Figure 6 for the principal enrollment and utilization of agents in the model. For our purposes of omnipresent availability of global sensor information, the extendable model agent environment is realized on the DCXP framework through the layering of the model on top of the architecture; with the DCXP protocol for the distribution of context information to mobile devices in an overlay network (see [23]). DCXP creates a peer to peer (P2P) overlay network of context-aware nodes, capable of exchanging context information via a conditional publisher/subscriber principle, enforcing a loose coupling. From this perspective however, the interface is architecture independent and simply requires a real-time means of collecting and disseminating the information to support the model. This functionality is realized by the Context User Agent (CUA) which is co-located with the CII platform. Within this distributed model, the CII can exist on multiple nodes, and by extension within multiple domains while having access to the common sensor information. This creates multiple sensor domains with agents located within the domains.

Consequently, the context-base itself can be seen as a service which can be utilized in any service oriented architecture. Combined with the CII-model, the context can be persisted and retrieved from any stationary or mobile node of the system. The resulting architecture can be considered a shared data space with an emphasis on context, due to the integration of the data centric context-base and DCXPs event based approach. This allows for agents and services to communicate decoupled in time since all context, including their own, is persistent. This information could be fused with virtual sensor information such as Twitter feeds.In the MediaSense framework, each node (e.g., a mobile device) in the DCXP overlay can have the functionalities for context propagation and query execution.

4 Concluding Remarks

In this paper we proposed the CII meta model and architecture for the representation of context information through the implementation of an ontology motivated object platform. Such a platform, we envision provides a concrete foundation for building dynamic agent based services and supporting time critical context dependent applications. The model, while being strict with respect to the fundamental concepts, provides for an extendable environment for realizing continually evolving object domains reflecting the inherent fluidity of observable context information. It provides for methods to translate, compare and order context values with respect to the heterogeneity of dependent applications and services.

With respect to the model as an agent environment, it exists as a first-order abstraction with two different roles: a container for agents and other entities (and their relations), and an exploitable abstraction for building multi-agent applications with context-aware agents. The context-base allows for agent persistence, including their internal states, as active first-class objects. This provides the basis for agent recovery, resilience, accessibility and mobility. Furthermore agents derive their context awareness from their ability to act upon sensor information within the model and may subsequently trigger other agents and services within the system. We provide a generic context information layer that has no explicit control of how and what agents infer from or reason upon the context.

The model's applicability was explored in the bus scenario; demonstrating the value it adds to large scale context modeling. We have presented a framework that enables for the implementation of intelligent services in systems of mobile devices. The framework is presented relative to the MediaSense framework for intelligent delivery of any information to any host, anywhere, based on context-aware information regarding personal preferences, presence information, and sensor information. In the MediaSense framework, sensor information from sensors in the vicinity of mobile users is perceived and shared using the distributed context exchange protocol (DCXP).

The model was realized with respect to the MediaSense framework creating an instance of CII in support of context aware applications and services. MediaSense with the underlying DCXP protocol provided the means for the real-time dissemination of context information in support of CII. CII was developed in Java, utilizing the DataNucleus access platform [24]. This enabled two separate supported services to be implemented: the first, a real-time vehicular safety system exploring the ability to monitor and feedback driving and traffic conditions to commuters in real time; and secondly a presence profile service allowing for dynamic generation and activation of presence profiles in response to changes in context information, supporting seamless media transfer between users' devices. Within this implementation, the context model is centralized with respect to a CUA, however further research will involve the distribution of the CII-model across the entire domain space thus realizing a common, global, content-centric object space.

Acknowledgements

The work was financed in part by the EU Regional Fund and the County Administrative Board of Västernorrland.

References

1. Remagnino, P., Foresti, G.L.: Ambient Intelligence: A New Multidisciplinary Paradigm. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans 35(1), 1–6 (2005)
2. Floréen, P., Przybilski, M., Nurmi, P., Koolwaaij, J., Tarlano, A., Wagner, M., Luther, M., Bataille, F., Boussard, M., Mrohs, B., Lau, S.: Towards a Context Management Framework for MobiLife. IST Mobile & Wireless Communications Summit (2005)
3. Grosky, W., Kansal, A., Nath, S., Liu, J., Zhao, F.: Senseweb: An infrastructure for shared sensing. Multimedia 14, 8–13 (2007)
4. Göker, A., Watt, S., Myrhaug, H.I., Whitehead, N., Yakici, M., Bierig, R., Nuti, S.K., Cumming, H.: An Ambient, Personalised, and Context-Sensitive Information System for Mobile Users. In: Proceedings of Second European Symposium on Ambient Intelligence, pp. 19–24 (2004)
5. Strang, T., Linnhoff-Popien, C., Korbinian, F.: CoOL: A Context Ontology Language to enable Contextual Interoperability. In: Stefani, J.-B., Demeure, I., Hagimont, D. (eds.) DAIS 2003. LNCS, vol. 2893, pp. 236–247. Springer, Heidelberg (2003)
6. Schilit, B., Adams, N., Want, R.: Context-Aware Computing Applications. In: Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pp. 85–90 (1994)
7. Bond, A.H., Gasser, L.: An analysis of problems and research in DAI. Readings in Distributed Artificial Intelligence, 3–35 (1988)
8. Sycara, K.P.: Multiagent Systems. AI Magazine 19(2), 79–92 (1998)
9. Dey, A.K.: Understanding and using context. In: Personal and Ubiquitous Computing, vol. 5, pp. 4–7 (2001)
10. Russel, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, Upper Saddle River (1995)

11. Uschold, M., Gruninger, M.: Ontologies: Principles, methods and applications. Knowledge Engineering Review 11, 93–136 (1996)
12. W3C: Resource description framework (rdf), http://www.w3.org/RDF/
13. Indulska, J., Sutton, P.: Location management in pervasive systems. In: Proceedings of the Australasian Information Security Workshop, CRPITS 2003, pp. 143–151 (2003)
14. Pollich, J., et al.: AmbieSense Reference Information Model (2004)
15. Weyns, D., Schumacher, M., Ricci, A., Viroli, M., Holvoet, T.: Environments in Multiagent Systems. The Knowledge Engineering Review 20(2), 127–141 (2005)
16. Weyns, D., Omicini, A., Odell, J.: Environment as a First-Class Abstraction in Multiagent Systems. Autonomous Agents and Multi-Agent Systems 14(1), 5–30 (2007)
17. Foundation for Intelligent Physical Agents: FIPA ACL Message Structure Specification (2002)
18. Wagner, G.: The Agent-Object-Relationship Metamodel: Towards a Unified View of State and Behavior. Information Systems 28(5), 475–504 (2003)
19. DeLoach, S.A., Wood, M.F., Sparkman, C.H.: Multiagent Systems Engineering. International Journal of Software Engineering and Knowledge Engineering 11(3), 231–258 (2001)
20. Bellifemine, F., Poggi, A., Rimassa, G.: Developing multi-agent systems with a FIPA-compliant agent framework. Software - Practice and Experience 31(2), 103–128 (2001)
21. Tanenbaum, A.S., van Steen, M.: Distributed Systems, 2nd edn. Prentice Hall International, Englewood Cliffs (2006)
22. Kanter, T.G., Österberg, P., Walters, J., Kardeby, V., Forsström, S., Pettersson, S.: The MediaSense Framework. In: Proceedings of the Fourth International Conference on Digital Telecommunications, pp. 144–147 (2009)
23. Kanter, T.G., Pettersson, S., Forsström, S., Kardeby, V., Norling, R., Walters, J., Österberg, P.: Distributed Context Support for Ubiquitous Mobile Awareness Services. In: Proceedings of Fourth International ICST Conference on Communications and Networking in China (2009)
24. Dobslaw, F.: An Adaptive, Searchable and Extendable Context Model, enabling cross-domain Context Storage, Retrieval and Reasoning. Master Thesis (2009)

A Context Aware Interruption Management System for Mobile Devices

Sina Zulkernain, Praveen Madiraju, and Sheikh Iqbal Ahamed

Dept. of Mathematics, Statistics & Computer Science
Marquette University, Milwaukee, WI – 53233, USA
{sina.zulkernain,praveen.madiraju,sheikh.ahamed}@marquette.edu

Abstract. To prevent unwanted interruptions from cell phones, this paper proposes a system solution considering user's unavailability. We first look at desirable characteristics of the system, then design a system architecture which takes as input user preferences, relevant context information and then produces as output if an incoming call should be allowed to ring. We also present a case study application that benefit by using the interruption management system. Finally, we discuss evaluations of the system by (i) evaluating the prototype and (ii) undertaking cognitive walkthroughs of the application.

Keywords: Context Aware System, Interruption, Ubiquitous Computing, Unavailability.

1 Introduction

Cellular phones have only been in use for mass communication during the last decade or so. However, the International Telecommunication Union estimates that cellular subscriptions worldwide have reached approximately 4.6 billion by the end of 2009 [39]. Along with its major role as a phone, cell phones have features like text messaging, voice messaging, data transferring and even the Internet. So it is not too astonishing to realize the rapid growth in mobile phone production. In a very short time period phones have come a long way, and the newer versions are smartphones with a number of applications built in like camera, games, GPS, calendar, alarm clock, notes, speech recognizer, touchpad etc. There has been a tremendous growth in smartphone applications too. Recent statistics indicate that there are over 100,000 active iPhone applications [42]. The massive number of users and enormous number of applications make cell phone a device integrated to our daily life. A University of Michigan study [40] shows that 83% people think cell phones make life easier and they choose it over the Internet.

Definitely with mobile phones, there is the obvious benefit of all the moment communication, but irrespective of time and place, we do expect a phone to ring. A ringing phone interrupting at an inopportune moment can be very disruptive to the current task or social situation [17]. In a survey of 1000 senior executives, it was reported that undesirable interruptions constitute 28 percent of the knowledge worker's day, which translates to 28 billion wasted hours to companies in the United

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 221–234, 2010.

States alone [34]. It results in a loss of 700 billion dollars per year, considering an average labour rate of $25 per hour for information workers [43]. A University of Oxford experiment suggests that in cognitively demanding situations, the advantage that 18-21 year olds enjoy over 35-39 year olds is reduced by an interruption caused by electronic communication technology [41]. Interruptions are mostly not beneficial to the immediate task and moving them a few minutes into the future could greatly benefit many users [2]. Undesired disruption causes interrupted users to take up to 30% longer to complete and commit up to twice the number of errors [4]. In order to mitigate the aforementioned problems, we propose a mobile interruption management system that will decide in real-time whether the user should be interrupted or not.

1.1 Necessary Characteristics of the System

Analyzing different scenarios, we identified the desired characteristics of an interruption management system in our earlier work [35]. In short the required characteristics are:

Mobility (C1). The system must be installable on a small handheld device being mindful of data transference costs, memory and CPU limitations.
Customizable (C2). Rules and outcomes must be customizable by and for each user.
Adaptable (C3). The system must be able to change itself to different environments from CPU power, screen size to input methods.
Context Aware (C4). The system has to be aware of its contexts i.e. take inputs from its surroundings.
Automated (C5). The system must make decisions all by itself without user interaction.
Unavailability Aware (C6). The system should take into account different modes of unavailability like audio, visual or touch by changing the interruption method to ring, vibrate or go silent.

1.2 Similar Researches

Several research studies have investigated the issue of interruption management in general [8, 15, 16, 20] and also specific to mobile devices [11, 17]. Dekel et al. [11] built an application that minimizes mobile phone interruptions by changing profile settings intelligently. Savioja [32] addresses different kinds of alarms for different types of interruptions in control room environments. Khalil & Connelli [25] use calendar information of the phone to minimize disruptions. Marti & Schmandt [28] devised an application for a group setting where a phone had to get all of the members' votes before ring. Also a methodology and design process for building interruption aware system is proposed in [15]. However, the distinguishing aspect of our work in comparison to the aforementioned ones is we have identified desirable characteristics of the system and show that our solution satisfies all of them.

The system we propose uses the capabilities from ubiquitous computing and context aware systems to programmatically learn about the environment and achieve our goals. Modelling context information and software engineering framework for context aware pervasive computing are already built in [18, 19]. We also have location and environment aware handheld systems [24] and frameworks in development for generalizing the sensor interfaces [14]. We have distributed resource discovery [33] and trust models for anonymous sensors [3]. Altogether, the avenue is clear for revolutionary system development awaiting only the sensor deployments.

1.3 Contribution of This Paper

The contribution of this paper is an intelligent interruption management system for mobile phones. The system is intelligent because of its adaptability, awareness for both context and unavailability, and also automatic decision making capability.

In our earlier poster paper [35], we proposed preliminary system architecture for an interruption management system with initial results. In this paper, we are extending on our previous work and furthermore provide the following contributions:

- Designed and developed system architecture for the system.
- Carefully surveyed the state of the art.
- Implemented a case study application using the developed system architecture.
- Evaluated the prototype application.
- Gathered users' feedback regarding the usability of the case study application.

The rest of the paper is organized as follows. Section 2 provides the system architecture and Section 3 focuses on the case study implementation. Evaluation of the system is done in Section 4 followed by related works in Section 5. Finally, Section 6 concludes our findings and paves the way for future works.

2 General System Architecture

In this section, we present our proposed general system architecture for mobile interruption management system. The general components of the system are shown below in Figure 1.

The large unit on the right, labeled Unavailability System is the system installed on the handheld. The system is divided into three tiers. The first tier includes the Context Service Interface in the upper left and Context Data Store in the upper right of the main unit. Upon reception of data, this tier collects and stores them in a continual process. In the middle tier, we have the User Interface Component that saves user preferred configurations in the Training Rules Data Store. The main component of the third tier is the Tree Generator. The tree generator collects the available context information from the first tier and user preferences from the second tier. This is the processing stage whereupon the decision structures are made. Tree Generator provides its decisions to the Content Provider Interface. The Content Provider Interface then instructs the Ringer Application on the phone to either ring or not ring. Finally, Ringer Application on the left is external to our system, but internal to the handheld's operating system.

Tier 1 (Context Information). Context Service Interface in Figure 1 is a persistent service on the mobile device that actively searches for context information from publicly available sensors. It aggregates information from the internal sources and generates a Context object to be stored in the Context Data Store. This object is passed to the Tree Generator (Decision Tree tier) for processing.

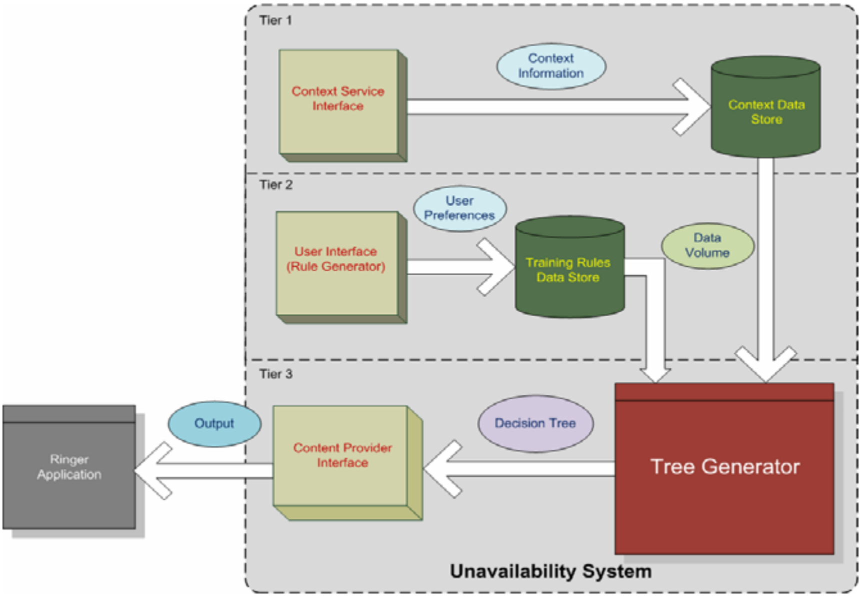


Fig. 1. General System Architecture

Tier 2 (User Preferences). Each user has a set of preferences about when to let calls through or when to deny them in any circumstances. These rules appear contradictory between users. So we need some sort of training data that needs to be collected either through a survey or another form of user inputs e.g. from the user interface.

Tier 3 (Decision Tree). The Tree Generator receives inputs of Context Data Point from Tier 1 and user preferences from Tier 2 and then generates a decision tree structure. A decision tree is a structure of conditional code that classifies a data set into one or more categories. A decision making piece of code is one that takes in context data such as sensory data and then activates the correspondingly correct action.

Due to space constraints, we are omitting a detailed description of the system architecture; however interested readers may refer to our earlier poster paper in [35] for a thorough discussion of the system architecture.

3 Case Study Implementation

Here we present a case study implementation where a private party (anonymous for privacy reasons) wants to send all its sales employees performance metrics each hour. These metrics represent their production, sales and the employee's current rank compared to others. They use it to make their sales force competitive within the company. Each sales person has a specified work area and the company wants to send the metrics to him/her whenever s/he is in the zone. Some sales people work indoors and they are mostly in the managerial positions. Now even though a sales person may be in his/her work area, s/he might be busy in a meeting with superiors. Whenever the sales person is in some scheduled event, the company does not want to send the metrics. Based on these requirements, we developed our prototype application. Whenever the sales person is in their designated area, not busy in a scheduled event and also when the time is between his/her work hours, we show the metrics on the device. But when any of these cases fail, we do not provide the metrics.

The prototype is developed on the Android, the operating system of a new class of smartphones which was designed primarily at Google in participation with the Open Handset Alliance. The reason for choosing Android is that it is Linux to the core and entirely open sourced. Most importantly when there is no call (in this case when no data is sent from the server); our application can run as a background process using minimal CPU and battery resources. The application needs to be installed in the receiver phone and Android is the only platform that allows full control of the ringer actions i.e. the interrupter. In the future, we also plan to implement it on other platforms as well. For the prototype, we used three contexts: location, schedule, and day of week along with time of day. We used Google Calendar as our scheduler. So our assumption is whoever uses our system will have some sort of scheduler where the application can query into. We used GPS service provided by Google to identify location and system clock service to find day and time of the week. Now we show step by step screen shots (see Table 1 – Figure T1-T8) to explain how our application is working.

Figure T1. The application first looks for its office location. It knows from its data storage what user's office address is and pinpoints that location using Reverse GPS service. Figure T1 in Table 1 shows user's office address.

Figure T2. The application then looks for user's current location. It uses GPS service and shows a region of radius 1000 meters where user could be. Figure T2 in the table shows that region by a black circle. Measuring the distance from the centre of this circle to user's office location, the application decides that user is at office.

Figure T3. Now the application uses the system clock to get today's day of week and current time. From its data storage the application determines that it is user's working hour.

Figure T4. It shows user's scheduler, in this case the Google Calendar. The application now queries the calendar to get user's current schedule.

Figure T5. The application sees that user has no event specified at the current time. So it decides to show the salesperson metrics sent from the office.

Figure T6. This figure shows the things the application considered before showing the metrics. The first line shows user's current location in latitude and longitude and user's current address in next. Then it shows the distance between user's current location and his/her office location. Next line shows current day and time. The application then shows user's status. The last line in the figure shows the metrics.

Figure T7. Now we make a change in the calendar and put an event there. Now the user is supposed to be busy. So the application sees that user is busy and now takes a decision not to show the metrics to the user.

Figure T8. This figure shows in the last line the event the user is currently attending. Also it shows the time left for this event to finish.

Table 1. Screenshots of the application

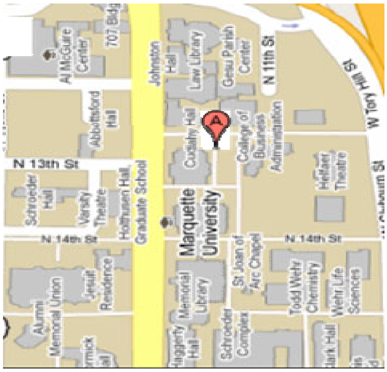
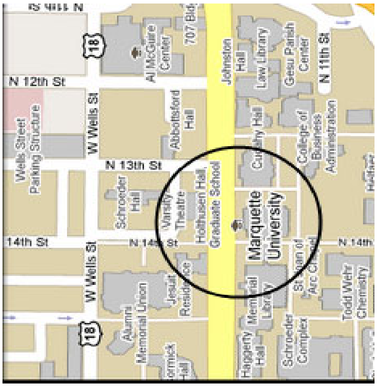
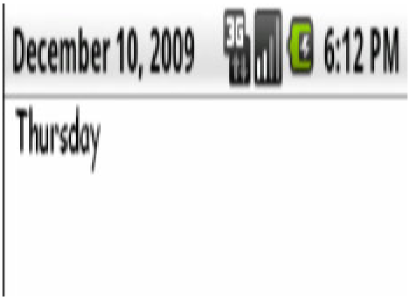
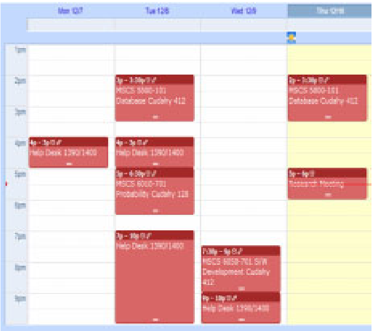
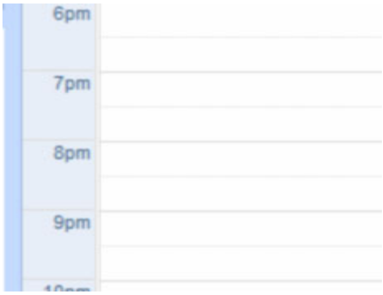
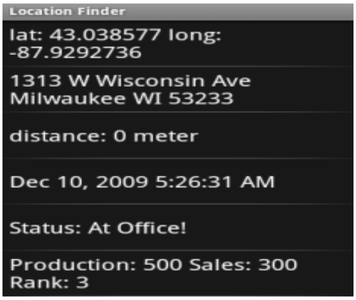
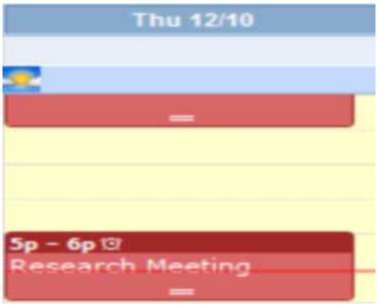
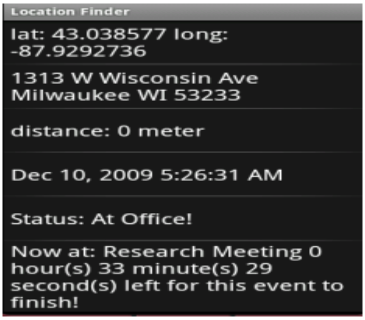
Figure T1  A Al McGuire Center 707 Bldg Johnston Hall Law Library Gesu Parish Center N 11th St W Tory Hill St Abbottsford Hall Cudahy Hall College of Business Administration N 13th St Schroeder Hall Varsity Theatre Holthusen Hall, Graduate School Marquette University Helfaer Theatre N 14th St Alumni Memorial Union Jesuit Residence Haggerty Hall Memorial Library Schroeder Complex St Joan of Arc Chapel Todd Wehr Chemistry Wehr Life Sciences	**Figure T2**  N 12th St 18 Wells Street Parking Structure W Wells St Al McGuire Center 707 Bldg Johnston Hall Law Library Gesu Parish Center N 11th St Abbottsford Hall Cudahy Hall College of Business Administration N 13th St Schroeder Hall Varsity Theatre Holthusen Hall, Graduate School Marquette University 14th St N 14th St Alumni Memorial Union Jesuit Residence Haggerty Hall Memorial Library Schroeder Complex St Joan of Arc Chapel Todd Wehr Chemistry
Figure T3  December 10, 2009 6:12 PM Thursday	**Figure T4**  Mon 12/7 Tue 12/8 Wed 12/9 Thu 12/10 1pm 2pm 3pm 4pm 5pm 6pm 7pm 8pm 9pm 2p – 3:30p MSCS 5800-101 Database Cudahy 412 2p – 3:30p MSCS 5800-101 Database Cudahy 412 4p – 5p Help Desk 1390/1400 4p – 5p Help Desk 1390/1400 5p – 6:30p MSCS 6000-701 Probability Cudahy 118 5p – 6p Research Meeting 7p – 10p Help Desk 1390/1400 7:30p – 9p MSCS 6050-701 S/W Development Cudahy 412 9p – 10p Help Desk 1390/1400
Figure T5  6pm 7pm 8pm 9pm	**Figure T6**  Location Finder lat: 43.038577 long: -87.9292736 1313 W Wisconsin Ave Milwaukee WI 53233 distance: 0 meter Dec 10, 2009 5:26:31 AM Status: At Office! Production: 500 Sales: 300 Rank: 3

Table 1. (*Continued*)

Figure T7	**Figure T8**
 	 Location Finder lat: 43.038577 long: -87.9292736 1313 W Wisconsin Ave Milwaukee WI 53233 distance: 0 meter Dec 10, 2009 5:26:31 AM Status: At Office! Now at: Research Meeting 0 hour(s) 33 minute(s) 29 second(s) left for this event to finish!

4 Evaluation

We have already done a simulation of our system for scalability in our earlier work [35]. Here we provide a detailed evaluation the system by evaluating the prototype and cognitive walkthroughs of the application.

4.1 Prototype Evaluation

In our system, data volume received is calculated below 50 kilobytes at any given environmental change. We have over 64MB to use and the memory usage requires a fraction of that. With respect to data transference, this is a perfect conformity to characteristic C1 (mobility). The system is installed on a cell phone with dimensions only 117.7 mm × 55.7 mm ×17.1 mm and in respect to size and portability, it conforms to C1 (mobility). To acquire information, our system uses sensors built into the device to gather data as to the local context. So it is C4 (context aware). Also, the unavailability system accepts varying sensor sets in differing situations and environments. Our prototype platform has built in GPS for instance, and this is one of the only data sources that can be assumed to exist for all deployments and use cases. Thus the system is C3 (adaptable). Decision tree traversal is a linear process. So the CPU power usage is very low along with the battery concerns. Our system keeps a ready to use decision tree so that when a call comes in, it can immediately make a decision and prevent the interruption without interference by the user. This prevents interruption and satisfies C5 (automated). With each decision tree having a different structure due to the per user customization, we have easily satisfied C2 (customizable). The user interface component is a constant time computation; again less CPU and memory usage. This system also utilizes the varying modes of unavailability; vision, hearing and touch. These levels of permissions for incoming communication attempts make the system smarter and more user friendly, providing appropriate attention to the different sorts of unavailability modes and thus satisfies

C6 (unavailability aware). Thus, all the characteristics outlined in Section 1 have been realized in the solution we have proposed.

4.2 Cognitive Walkthrough

To get the proper assessment of our application, we used the cognitive walkthrough strategy. We did a survey on a group of 30 people on the usability and usefulness of our application. First we explained the problem, briefly went over some of the issues we addressed and then showed the prototype application demo. The distribution of the participants is as follows: 17 undergraduate students, 8 graduate students, 2 faculty members, 2 entrepreneurs, and 1 other.

We handed 5 questions about the application over to each participant and requested them to answer them on a scale of 1 to 5. The questionnaire for the survey is given below:

Question 1. Overall, how would you rate the services? (*1 = Very Poor, 5 = Excellent*)

Question 2. What is the effectiveness of this application? (*1 = Not Effective at all, 5 = Very Useful*)

Question 3. How easy is it to give the input? (*1 = Very Hard, 5 = Very Easy*)

Question 4. Will you pay to use this application? (*1 = Definitely Not, 5 = Definitely Yes*)

Question 5. Would you recommend this application to a friend? (*1 = Surely Not, 5 = Surely Yes*)

The survey results are shown in Figure 2:

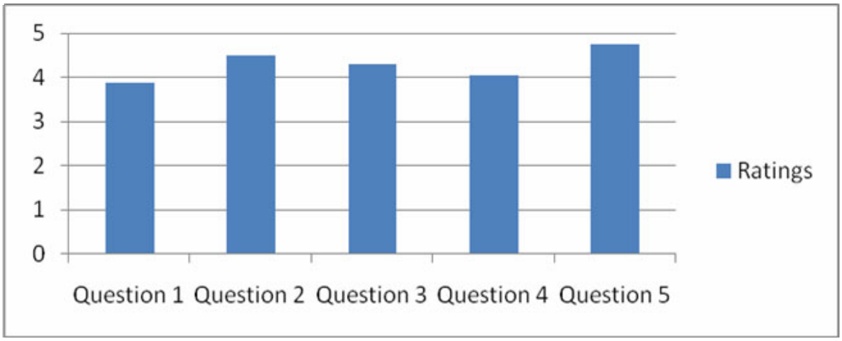


Fig. 2. Survey Results

The results from Figure 2 indicate that participants were enthusiastic about the application and its usability.

5 Related Works

When a call is made to a phone, the system decides beforehand not to the let the call go through if it is a costly interruption, The cost of interruption (COI) is a function of immediate task and the user's state of mind, which can also be seen as a function of the task at hand. A proper ubiquitous computing system can theoretically understand the task at hand and infer the user's state of mind and therein get a measure of COI. Hence the survey of literature spans into areas related to COI, interruption management and context aware systems.

5.1 Cost of Interruption

Adamczyk [1] measures the effects of interruption in terms of task performance, emotional state and social attribution. The study also aims to find the most suitable time to interrupt the user. Several researchers have addressed the issue of cost of interruption [5, 23, 16]. Mark [27] measures the COI based on additional time required to reorient back to the primary task and mental stress brought upon the interruptee. A user's pupil size increases due to the mental processing efforts and there is an upper bound on how much it can grow. Bailey [5] shows this could be a possible way to measure a user's mental stress and hence decide whether interruption could be detrimental or a bit refreshing anyway. The bottom line is to defer interruption when COI is high. This has been shown to not only increase worker efficiency, but also benefit morale [2].

5.2 Interruption Management

To manage interruption, first we need to specify the factors that make interruption a burden. Horvitz et al. [21] describe a system that builds decision-theoretic models by asking users about their perceived interruptibility during a training phase. Ho & Intille [20] consider 11 factors that impact the perceived burden of interruption. The authors suggest that an exhaustive model of interruptibility should include a weighted sum of the factors.

Next we need to use context aware services to manage interruption. Abundant body of literature has studied the issue of context management for personal computers [13, 22, 9]. Baladauf et al. [6] presents a survey of context aware systems. The typical contexts included are: location, time, day, and proximity. In relation to interruption management, several researchers have proposed other meaningful contexts. Petersen [30] mentions the challenges to face when pervasive computing becomes a reality and a part of our everyday life. Godbole & Smari [15] consider three types of contexts namely relational, social and interruptee's cognitive context to solve the interruption problem.

5.3 Context Aware Systems

A context aware system is a computing resource with knowledge of its environment and its user's situation. Research in autonomic computing [38] recognizes the complexity involved with applying or interfacing such a system with human users. The problem arises when we expect the context aware system and the pervasive environment to combine into one intelligent environment. Ziebart et al's [38] work on Learning Automation Policies serves to solve this problem. In a context aware system, information will come in from many sources rather than only one or two streams of input. The Context Toolkit is a java based library that facilitates development and deployment of context-aware systems [12].

With context aware systems assumed and available, the next step is aware and adaptive services. In [10], context awareness is extended into the service oriented architecture. Our unavailability system will require such rich information sources to properly diagnose a given situation. Privacy is a topic that is closely related to personal unavailability. In [26], interpersonal relationships in regard to data privacy preferences have been addressed. A system called Lilsys, which reads motion, sound and door-closed-state has been constructed in [8] to build a qualitative measure of user unavailability. Also, automated preference control on mobile devices has been tackled in [7].

5.4 Interruption Associated with Mobile Devices

There have been several works on how to manage interruption at inopportune moments using smartphones. Yu et al. [37] define user preference, terminal capability, location, time, activity and so on as context dimensions for smartphones. In [31], the authors suggest that an interruption technology adapting its response considering a person's feelings is likely to improve people's experience with that technology. Godbole & Smari [15] survey the type and extent of desired information about the incoming cell phone call. Guzman et al. [17] studied the context information users consider when they make a call and also the context information they wish others consider when they receive a call. In [36], the authors group the strategies for interruption management by filtering calls based on caller's identity, situation and time, and, status message sharing e.g. current location, activity etc. As users tackle

Table 2. Comparison of Various Interruption Management Systems

Characteristics → **Research Works** ↓	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
Toninelli et al. [36]	Y	Y	N	Y	Y	N
Godbole & Smari [15]	N	N	N	Y	Y	N
Picard et al. [31]	N	N	Y	N	Y	N
Bailey et al. [5]	N	N	Y	Y	Y	N
Ho & Intille [20]	N	N	Y	Y	Y	Y
Mark et al. [27]	N	N	N	Y	Y	N
Dekel et al. [11]	Y	Y	N	Y	Y	N
Guzman et al. [17]	Y	N	N	Y	Y	N
Khalil & Connelli [25]	Y	Y	N	N	Y	Y
Marti & Schmandt [28]	Y	Y	N	N	N	Y
Our System	Y	Y	Y	Y	Y	Y

interruption by taking some actions themselves, Toninelli et al. [36] suggest that the intelligent system should learn how the users act in some situations, learn from them and later take actions like them.

The aforementioned research works give us a solid basis for (i) which context needs to be considered, and (ii) how to evaluate such context. However, the chief distinguishing aspects of our work are (i) system architecture and prototype implementation with performance evaluations, and (ii) identification of desirable characteristics of the problem solution. In Table 2, we present a comparison of different interruption management system against the desirable characteristics.

6 Conclusion

In this paper, we have presented the design, development and evaluation of an intelligent interruption management system. The system architecture considers context information and user preferences and automatically filters out interruptions for mobile devices. We also presented a prototype case study that implements the system architecture. The system is fully analysed and the performance evaluations indicate that it is efficient to run within the constraints of a handheld device.

We plan to extend our work with additional features. The caller can be notified of the receiver's current state if s/he is not picking up. The receiver may not want to disclose this information to everyone. In some cases, s/he might just want the caller to know that s/he is "busy", wherein the other cases such as to a spouse s/he would like to inform the caller specifically of his/her current state. Again, this information can be passed to the caller in a simple text message or there can be a user interface for the caller in our application where this information is viewed. Secondly, receiver can inform the caller when to try calling again. Acquiring information from the user's task scheduler, our system can know when the current task is going to finish and notify the caller accordingly. In some cases, the receiver may just fail to notice that there is a call. In that case, the system can encourage the caller to try again instantaneously.

We also plan to formalize the model for unavailability which takes into account context-aware services such as location based services. As a part of our goal, we are currently working toward a mathematical formulation of Cost of Interruption (COI). We also like to explore possible applications of our system in different application domains from cell phones to instant messaging, email clients, and social networking. These are some areas which operate by interrupting a user and we plan to incorporate our unavailability feature to them so that the cost of interruption is kept to a minimum.

Acknowledgements. We thank Karl Stamm for his initial work on interruption management described in [35].

References

1. Adamczyk, P.D., Bailey, B.P.: If not now, when? the effects of interruption at different moments within task execution. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vienna, Austria, pp. 271–278 (2004)
2. Adamczyk, P.D., Iqbal, S.T., Bailey, B.P.: A method, system, and tools for intelligent interruption management. In: Proceedings of the 4th International Workshop on Task Models and Diagrams, pp. 123–126 (2005)

3. Ahamed, S.I., Sharmin, M., Ahmed, S.: A Risk-aware Trust Based Secure Resource Discovery (RTSRD) Model for Pervasive Computing. In: Proceedings of the 2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications, pp. 590–595 (2008)
4. Bailey, B.P., Konstan, J.A.: On the need for attention aware systems: Measuring effects of interruption on task performance, error rate, and affective state. Journal of Computers in Human Behavior 22(4), 709–732 (2006)
5. Bailey, B.P., Iqbal, S.T.: Understanding changes in mental workload during execution of goal-directed tasks and its application for interruption management. ACM Trans. Comput.-Hum. Interact. 14(4), 1–28 (2008)
6. Baldauf, M., Dustdar, S., Rosenberg, F.: A Survey on Context Aware Systems. International Journal of Ad Hoc and Ubiquitous Computing 2(4), 263–277 (2007)
7. Bayley, C., Jernigan, C., Lin, J., Shu, J., Wright, C.: Talk Android (2008), `http://www.talkandroid.com/android-forums/android-market-reviews/495-locale.html`
8. Begole, J., Matsakis, N.E., Tang, J.C.: Lilsys: Sensing Unavailability. In: Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, pp. 511–514 (2004)
9. Brown, P.J.: The Stick-e Document: a framework for creating context-aware applications. Electronic Publishing, Palo Alto (1996)
10. Conlan, O., Power, R., Higel, S., O'Sullivan, D., Barrett, K.: Next generation context aware adaptive services. In: Proceedings of the 1st International Symposium on Information and Communication Technologies, pp. 205–212 (2003)
11. Dekel, A., Nacht, D., Kirkpatrick, S.: Minimizing mobile phone disruption via smart profile management. In: Proceedings of the 11th International Conference on Human-Computer interaction with Mobile Devices and Services, Bonn, Germany, pp. 1–5 (2009)
12. Dey, A.K.: Enabling the use of context in interactive applications. In: Extended Abstracts on Human Factors in Computing Systems, CHI 2000, The Hague, The Netherlands, pp. 79–80 (2000)
13. Dey, A.K., Salber, D., Abowd, G.D.: A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. The Human-Computer Interaction (HCI) Journal, 97–166 (2001)
14. Dey, A., Mankoff, J., Abowd, G., Carter, S.: Distributed mediation of ambiguous context in aware environments. In: Proceedings of the 15th Annual ACM Symposium on User interface Software and Technology, Paris, France, pp. 121–130 (2002)
15. Godbole, A., Smari, W.W.: A Methodology and Design Process for System Generated User Interruption based on Context, Preferences, and Situation Awareness. In: IEEE International Conference on Information Reuse and Integration, pp. 608–616 (2006)
16. Grandhi, S.A., Schuler, R.P., Jones, Q.: To answer or not to answer: that is the question for the cell phone users. In: Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems, pp. 4621–4626 (2009)
17. De Guzman, E.S., Sharmin, M., Bailey, B.P.: Should I call now? Understanding what context is considered when deciding whether to initiate remote communication via mobile devices. In: Proceedings of Graphics Interface 2007, Montreal, Canada, pp. 143–150 (2007)
18. Henricksen, K., Indulska, J.: A software engineering framework for context-aware pervasive computing. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications (PerCom), pp. 77–86 (2004)

19. Henricksen, K., Indulska, J.: Modeling and using imperfect context information. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, pp. 33–37 (2004)
20. Ho, J., Intille, S.S.: Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Portland, Oregon, USA, pp. 909–918 (2005)
21. Horvitz, E., Koch, P., Apacible, J.: BusyBody: creating and fielding personalized models of the cost of interruption. In: Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, pp. 507–510 (2004)
22. Hull, R., Neaves, P., Bedford-Roberts, J.: Towards situated computing. In: Proceedings of International Symposium on Wearable Computers (1997)
23. Iqbal, S.T., Bailey, B.P.: Leveraging characteristics of task structure to predict the cost of interruption. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Montréal, Québec, Canada, pp. 741–750 (2006)
24. Jiang, X., Chen, N.Y., Hong, J.I., Wang, K., Takayama, L., Landay, J.A.: Siren: Context aware Computing for Firefighting. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 87–105. Springer, Heidelberg (2004)
25. Khalil, A., Connelly, K.: Improving cell phone awareness by using calendar information. In: Proceedings of INTERACT, Rome, Italy (2005)
26. Lederer, S., Mankoff, J., Dey, A.K.: Who wants to know what when? Privacy preference determinants in ubiquitous computing. In: Extended Abstracts on Human Factors in Computing Systems, CHI 2003, Ft. Lauderdale, Florida, USA, pp. 724–725 (2003)
27. Mark, G., Gudith, D., Klocke, U.: The cost of interrupted work: more speed and stress. In: Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, pp. 107–110 (2008)
28. Marti, S., Schmandt, C.: Giving the caller the finger: collaborative responsibility for cellphone interruptions. In: Extended Abstracts on Human Factors in Computing Systems, CHI 2005, Portland, OR, USA, pp. 1633–1636 (2005)
29. McCrickard, D.S., Chewar, C.M., Somervell, J.P., Ndiwalana, A.: A model for notification systems evaluation—assessing user goals for multitasking activity. ACM Trans. Comput. Hum. Interact. 10(4), 312–338 (2003)
30. Petersen, S.A., Cassens, J., Kofod-Petersen, A., Divitini, M.: To be or not to be aware: Reducing interruptions in pervasive awareness systems. In: Proceedings of the Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies UBICOMM, pp. 327–332 (2008)
31. Rosalind, P.W., Karen, L.K.: Relative subjective count and assessment of interruptive technologies applied to mobile monitoring of stress. International Journal of Human-Computer Studies 65(4), 361–375 (2007)
32. Savioja, P.A.: Necessary Interruptions? Seminar on User Interfaces and Usability, HUT, Sober IT, pp. 121-900 (2004)
33. Sharmin, M., Ahamed, S.I., Ahmed, S., Li, H.: SSRD+: A Privacy-aware Trust and Security Model for Resource Discovery in Pervasive Computing Environment. In: Computer Software and Systems Conference, pp. 67–70 (2006)
34. Spira, J.B., Feintuch, J.B.: The Cost of Not Paying Attention: How Interruptions Impact Knowledge Worker Productivity, Basex (2005)
35. Stamm, K., Ahamed, S.I., Madiraju, P., Zulkernain, S.: Mobile Intelligent Interruption Management (MIIM): A Context Aware Unavailability System. In: Proceedings of the 25th Annual ACM Symposium on Applied Computing, Sierre, Switzerland (2010)

36. Toninelli, A., Khushraj, D., Lassila, O., Montanari, R.: Towards Socially Aware Mobile Phones. In: 7th International Semantic Web Conference (2008)
37. Zhiwen, Y., Xingshe, Z., Daqing, Z., Chung-Yau, C., Xiaohang, W., Ji, M.: Supporting context-aware media recommendations for smart phones. IEEE Pervasive Computing 5(3), 68–75 (2006)
38. Ziebart, B.D., Roth, D., Campbell, R.H., Dey, A.K.: Learning Automation Policies for Pervasive Computing Environments. In: Proceedings of the Second International Conference on Automatic Computing,, pp. 193–203. IEEE Computer Society, Washington (2005)
39. International Telecommunication Union, http://www.itu.int/newsroom/press_releases/2009/39.html
40. University of Michigan News Service, http://www.ur.umich.edu/0607/Apr02_07/02.shtml
41. Disruptive communication and attentive productivity, http://www.iii-p.org/research/disrupt_comm_report_v2.pdf
42. 148 Apps.biz, http://148apps.biz/app-store-metrics
43. Bureau of Labor Statistics, http://www.bls.gov/

Applying Behavioral Detection on Android-Based Devices

Asaf Shabtai and Yuval Elovici

Deutsche Telekom Laboratories at Ben-Gurion University, and
Department of Information Systems Engineering
Ben-Gurion University, Be'er Sheva, 84105 Israel
{shabtaia,elovici}@bgu.ac.il

Abstract. We present Andromaly - a behavioral-based detection framework for Android-powered mobile devices. The proposed framework realizes a Host-based Intrusion Detection System (HIDS) that continuously monitors various features and events obtained from the mobile device, and then applies Machine Learning methods to classify the collected data as normal (benign) or abnormal (malicious). Since no malicious applications are yet available for Android, we evaluated Andromaly's ability to differentiate between game and tool applications. Successful differentiation between games and tools is expected to provide a positive indication about the ability of such methods to learn and model the behavior of an Android application and potentially detect malicious applications. Several combinations of classification algorithms, feature selections and the number of top features were evaluated. Empirical results suggest that the proposed detection framework is effective in detecting types of applications having similar behavior, which is an indication for the ability to detect unknown malware in the Android framework.

Keywords: Intrusion Detection, Mobile Devices, Machine Learning, Malware, Security, Android.

1 Introduction

Personal Digital Assistants (PDAs), mobile phones and recently smartphones have evolved from simple mobile phones into sophisticated yet compact minicomputers which can connect to a wide spectrum of networks, including the Internet and corporate intranets. Designed as open, programmable, networked devices, smartphones are susceptible to various malware threats such as viruses, Trojans, and worms, all of which are well-known from desktop platforms. These devices enable users to access and browse the Internet, receive and send emails, SMSs, and MMSs, connect to other devices for exchanging information/synchronizing, and activate various applications, which make these devices attack targets [1], [2].

A compromised smartphone can inflict severe damages to both users and the cellular service provider. Malware on a smartphone can make the phone partially or fully unusable; cause unwanted billing; steal private information (possibly by Phishing and Social Engineering); or infect every name in a user's phonebook [3].

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 235–249, 2010.

The challenges for smartphone security are becoming very similar to those that personal computers encounter [5]. Common desktop-security solutions are being developed for mobile devices. Botha et al. [6] analyze common desktop security solutions and assess their applicability to mobile devices. Nevertheless, some of these solutions (such as antivirus software) are inadequate for use on smartphones as they consume too much CPU and memory and might result in rapid draining of the power source. In addition, most antivirus detection capabilities depend on the existence of an updated malware signature repository, therefore the antivirus users are not protected whenever an attacker spreads a previously un-encountered malware. Since the response time of antivirus vendors may vary between several hours to several days to identify the new malware, generate a signature, and update their clients' signature database, hackers have a substantial window of opportunity [7]. Some malware instances may target a specific and relatively small number of mobile devices (e.g., for extracted confidential information or track owner's location) and will therefore take quite a time till they are discovered.

In this research we describe a framework for detecting malware on Android mobile devices in the form of a Host-based Intrusion Detection System (HIDS). This is accomplished by continuously monitoring mobile devices to detect suspicious and abnormal activities. The framework relies on a light-weight agent that samples various system metrics, analysis of the sampled measurements and inference about the state of the device. The main assumption is that system metrics such as CPU consumption, number of sent packets through the Wi-Fi, number of running processes, battery level etc. can be employed for detection of previously un-encountered malware by examining similarities with patterns of system metrics induced by known malware [4], [8]. The primary goal of the study is to find the optimal mix of: a classification method, feature selection method and the number of monitored features that yields the best performance in accurately detecting new malware on Android. Since no malicious applications are yet available for Android, we evaluated Andromaly's ability to differentiate between game and tool applications.

2 Related Work

Our overview of related academic literature indicates that most extant research on protection of mobile devices has focused on applying and evaluating HIDS. These systems, using anomaly- or rule-based detection methods, extract and analyze (either locally or by a remote server) a set of features indicating the state of the device. Several systems are reviewed in this section.

Artificial Neural Networks (ANNs) were used in order to detect anomalous behavior indicating a fraudulent use of the operator services (e.g., registration with a false identity and using the phone to high tariff destinations) [9]. The Intrusion Detection Architecture for Mobile Networks (IDAMN) system [10] offers three levels of detection: location-based detection (a user active in two different locations at the same time); traffic anomaly detection (an area having normally low network activity, suddenly experiencing high network activity); and detecting anomalous behavior of individual mobile-phone users. Yap et al. [11] employ a behavior checker solution that detects malicious activities in a mobile system. They present a proof-of-concept scenario using a Nokia Mobile phone running a Symbian OS. In the demonstration, a behavioral detector detects a simulated Trojan attempting to use the message server component without authorization to create an SMS message. Cheng et al. [4] present SmartSiren, a collaborative proxy-based virus detection and alert system for smartphones. Single-device and system-wide abnormal behaviors are detected by the joint analysis of communication activity of monitored smartphones. Schmidt et al. [12] monitored a smartphone running a Symbian OS in order to extract features that describe the state of the device and which can be used for anomaly detection. These features were collected by a Symbian monitoring client and forwarded to a Remote Anomaly Detection System (RADS). The gathered data were used for anomaly detection methods in order to distinguish between normal and abnormal behavior.

An interesting behavioral detection framework is proposed [13] to detect mobile worms, viruses and Trojan horses. The method employs a temporal logic approach to detect malicious activity over time. An efficient representation of malware behaviors is proposed based on a key observation that the logical ordering of an application's actions over time often reveals malicious intent even when each action alone may appear harmless.

Special effort has been devoted to Intrusion Detection Systems (IDS) that analyze generic battery power consumption patterns to block Distributed Denial of Service (DDoS) attacks or to detect malicious activity via power depletion. Kim et al. [14] presented a power-aware, malware-detection framework that monitors, detects, and analyzes previously unknown energy-depletion threats. Buennemeyer et al. [15] introduced capabilities developed for a Battery-Sensing Intrusion Protection System (B-SIPS) for mobile computers, which alerts when abnormal current changes are detected. Nash et al. [16] presented a design for an intrusion detection system that focuses on the performance, energy, and memory constraints of mobile computing devices. Jacoby and Davis [17] presented a host Battery-Based Intrusion Detection System (B-BID) as a mean of improving mobile device security. The basic idea is that monitoring the device's electrical current and evaluating its correlation with known signatures and patterns, can facilitate attack detection and even identification.

Hwang et al. [18] evaluated the effectiveness of Keystroke Dynamics-based Authentication (KDA) on mobile devices. Their empirical evaluation focused on short PIN numbers (four digits) and the proposed method yielded a 4% misclassification rate.

The aforementioned frameworks and systems proved valuable in protecting mobile devices in general however, they do not leverage Android's capabilities to their full extent. Since Android is an open source and extensible platform it allows to extract as much features as we would like. This enables to provide richer detection capabilities, not relying merely on the standard call records [9], or power consumption patterns [15]-[18].

3 Anomaly Detection Framework for Android

Google's Android is a comprehensive software framework targeted towards such smart mobile devices (i.e., smartphones, PDAs), and it includes an operating system, a middleware and a set of key applications. Android emerged as an open-source, community-based framework which provides APIs to most of the software and hardware components. Specifically, it allows third-party developers to develop their own applications. The applications are written in the Java programming language based on the APIs provided by the Android Software Development Kit (SDK), but developers can also develop and modify kernel-based functionalities, which is not common for smartphone platforms.

We developed a lightweight Host-based Intrusion Detection System (in terms of CPU, memory and battery consumption) for Android-based mobile devices. The basis of the intrusion detection process consists of real-time, monitoring, collection, preprocessing and analysis of various system metrics, such as CPU consumption, number of sent packets through the Wi-Fi, number of running processes and battery level. System and usage parameters, changed as a result of specific events, may also be collected (e.g., keyboard/touch-screen pressing, application start-up). After collection and preprocessing, the system metrics are sent to analysis by various detection units, namely processors, each employing its own expertise to detect malicious behavior and generate a threat assessment (TA) accordingly. The pending threat assessments are weighted to produce an integrated alert and also includes a smoothing phase (combining the generated alert with the past history of alerts) in order to avoid instantaneous false alarms. After the weighting phase, a proper notification is displayed to the user. Moreover, the alert is matched against a set of automatic or manual actions that can be undertaken to mitigate the threat. Automatic actions include among others: uninstalling an application, killing of a process, disconnecting of all radios, encrypting data, changing firewall policies and more. A manual action can be uninstalling an application subject to user consent.

The components of the agent are clustered into four main groups (see Figure 1): Feature extraction, processors, agent service, and the Graphical User Interface (GUI). The *Feature Extractors* communicate with various components of the Android framework, including the Linux kernel and the Application Framework layer in order to collect feature metrics, while the Feature Manager triggers the Feature Extractors and requests new feature measurements every pre-defined time interval. In addition, the *Feature Manager* may apply some pre-processing on the raw features that are collected by the Feature Extractors.

A *Processor* is an analysis and detection unit. It is preferred that the processor will be provided as pluggable external component which can be seamlessly installed and un-installed. Its role is to receive feature vectors from the Main Agent Service, analyze them and output threat assessments to the Threat Weighting Unit. Each processor may expose an advanced configuration screen. Processors can be either: rule-based, knowledge-based, or classifiers/anomaly detector based on Machine Learning (ML) methods.

The *Threat Weighting Unit* (TWU) obtains the analysis results from all active processors and applies an ensemble algorithm (such as Majority Voting, Distribution Summation etc.) in order to derive a final coherent decision regarding a device's infection level. The Alert Manager receives the final ranking as produced by the TWU. It can then apply some smoothing function in order to provide a more persistent alert and to avoid instantaneous false alarms. Examples of such functions can be moving average and leaky-bucket. The smoothed infection level is then compared with pre-defined minimum and maximum thresholds.

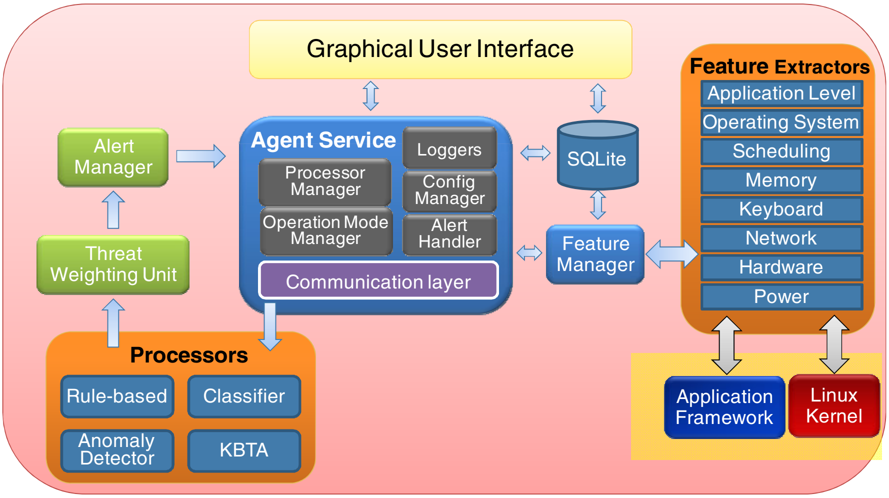


Fig. 1. The Andromaly Framework

The *Main Agent Service* is the most important component. This service synchronizes feature collection, malware detection and alert process. The Agent Service manages the detection flow by requesting new samples of features, sending newly sampled metrics to the processors and receiving the final recommendation from the *Alert Manager*. The *Loggers* provide logging options for debugging, calibration and experimentation with detection algorithms. The *Configuration Manager* manages the configuration of the agent (such as active processors, active feature extractors, alert threshold, active loggers, sampling temporal interval, detection mode configuration, etc.). The *Alert Handler* triggers an action as a result of a dispatched alert (e.g., visual alert in the notification bar, uninstalling an application sending notification via SMS or email, locking the device, disconnecting any communication channels). The *Processor Manager* registers/unregisters processors, and activates/ deactivates processors. The Operation Mode Manger changes the agent from one operation mode to another based on the desired configuration. This will activate/ deactivate processors and feature extractors. Changing from one operation mode to another (i.e. from Full Security mode to Normal mode) is triggered as a result of changes in available resources levels (battery, CPU, Network).

The last component is the *Graphical User Interface* which provides the user with the means to configure agent's parameters, activate/deactivate (for experimental usage only), visual alerting, and visual exploration of collected data.

4 Detection Method

4.1 Using Machine Learning for Behavioral Analysis

The evaluation of Machine Learning classifiers is typically split into two subsequent phases: training and testing. In the first phase, a training-set of games and tools feature vectors is provided to the system. These feature vectors are collected during the activation of both game and tool applications. The representative feature vectors in the training set and the real class of each vector (as game/tool) are assumed to be known and enable to calibrate the detection algorithms (such as a Decision Trees, or Bayesian Network). By processing these vectors, the algorithm generates a trained classifier.

Next, during the testing phase, a different collection (the testing-set) containing both game and tool applications feature vectors is classified by the trained classifier. In the testing phase, the performance of the classifier is evaluated by extracting standard accuracy measures for classifiers. Thus, it is necessary to know the real class of the feature vectors in the test-set in order to compare it real class with the class that was derived by the trained classifier.

Based on previous experience and after weighing the resource consumption issue, we decided to evaluate the following candidate classifiers: k-Means [19], Logistic Regression [20], Histograms [21], Decision Tree [22], Bayesian Networks [23] and Naïve Bayes [24].

4.2 Feature Selection

In Machine Learning applications, a large number of extracted features, some of which redundant or irrelevant, present several problems such as - misleading the learning algorithm, over-fitting, reducing generality, and increasing model complexity and run-time. These adverse effects are even more crucial when applying Machine Learning methods on mobile devices, since they are often restricted by processing and storage-capabilities, as well as battery power. Applying fine feature selection in a preparatory stage enabled to use our malware detector more efficiently, with a faster detection cycle. Nevertheless, reducing the amount of features should be performed while preserving a high level of accuracy.

Three feature selection methods were applied to the datasets: Information Gain (IG), Chi-Square (CS) and Fisher Score (FS). These feature selection methods follow the Feature Ranking approach and, using a specific metric, compute and return a score for each feature individually. Chi-Square [25] measures the lack of independence between a feature f and a class C. The Fisher Score [26] expresses the difference between two classes relative to a specific feature taking into account the mean and standard deviation of the feature's values in different classes. If the absolute difference between the feature's mean values in the two classes is small, and the sum of the feature's standard deviations of the two classes is large, the feature is not considered discriminative. Information Gain [27] determines the amount of information which a feature provides about a class by measuring how well it separates the training examples according to their target classification. In a more formal definition, Information Gain quantifies the expected reduction of Shannon's Entropy [27] caused by partitioning the examples according to a selected feature.

A problem was raised when we had to decide how many features we would choose for the classification task from the feature selection algorithms' output ranked lists. In order to avoid any bias by selecting an arbitrary number of features, we used, for each feature selection algorithm, three different configurations: 10, 20 and 50 features that were ranked the highest out of the 88 features ranked by the feature selection algorithms.

5 Evaluation

In order to evaluate our behavioral detection framework we performed two experiments. The research questions that we attempt to answer using the experiments are described in subsection 5.1. In subsection 5.2 we describe the dataset created for the experiments. Finally, subsection 5.3 describes the scheme of the two experiments and the obtained results.

5.1 Research Question

We evaluated the capability of the proposed HIDS framework to classify applications through two experiments, aimed at answering the following questions:

1) Is it possible to detect unknown instances of known application types on Android devices?
2) Which classifier is most accurate in detecting malware on Android devices: Decision Tree (DT), Naïve Bayes (NB), Bayesian Networks (BN), k-Means, Histogram or Logistic Regression (LR)?
3) Which number of extracted features and feature selection method yield the most accurate detection results: 10, 20 or 50 top- features selected using Chi-Square, Fisher Score or InfoGain?
4) What are the specific features that yield the maximum detection accuracy?

In order to perform the comparison between the various detection algorithms and feature selection schemes, we employed the following standard metrics: the True Positive Rate (TPR) measure, which is the proportion of positive instances classified correctly; False Positive Rate (FPR), which is the proportion of negative instances misclassified; and the Total Accuracy, which measures the proportion of absolutely correctly classified instances, either positive or negative.

5.2 Creating the Dataset for the Experiments

Since no standard dataset was available for this study, we had to create our own datasets. For the experiments, 23 games and 20 tools were collected, 11 of them were available on the Android framework, while the rest were obtained from the Android Market (Appendix A). All games and tools were verified to be virus-free before installation by manually exploring the permissions that the applications required, and by using a static analysis of dex files.

The aforementioned applications (i.e., 23 games, and 20 tools) were installed on five Android devices. The five devices were similar in the platform (HTC G1) with the same firmware and software versions. The five devices were used regularly by different users and thus varied in the amount and type of applications installed as well as usage patterns. The HIDS application, which continuously sampled various features on the device, was installed and activated on the devices under regulated conditions, and measurements were logged on the SD-card.

Each of the five Android devices had one user who used each of the 43 applications for 10 minutes, while in the background the malware detection system collected new feature vectors every 2 seconds. Therefore, a total of approximately 300 feature vectors were collected per each application and device. All the vectors in the datasets were labeled with their true class: ‘game’ or ‘tool’. The table in Appendix A presents the number of vectors collected for each malicious, tool and game applications on the two tested devices.

The extracted features are clustered into two primary categories: Application Framework and Linux Kernel. Features belonging to groups such as Messaging, Phone Calls and Applications belong to the Application Framework category and were extracted through APIs provided by the framework, whereas features belonging to groups such as Keyboard, Touch Screen, Scheduling and Memory belong to the Linux Kernel category. A total of 88 features were collected for each monitored application (see Appendix B).

5.3 Experiments and Results

The purpose of the experiments was to evaluate the ability of the proposed detection methods to distinguish between games and tools applications. The following two experiments examine the performance of the detection system in different situations. For each experiment we used datasets extracted from 5 different devices, on which we evaluated 6 detection algorithms, 3 feature selection methods, and 3 sizes of top features groups (10, 20 and 50) as presented in Table 1.

Table 1. Experiments descriptions

Exp.	# of detection algorithms	# of feature selection methods	# of devices	# of top feature groups	# of iterations	Total number or funs	Testing on applications not in training set
I	6	3	5	3	20	5,400	-
II	6	3	5	3	20	5,400	+

Experiment 1

The purpose of this experiment is to evaluate the ability of each combination of detection algorithm, feature selection method, and number of top features to differentiate between game and tool applications when training set includes all game/tool applications. The training set contained 80% of the feature vectors of both the game and tool applications. The testing set contained the rest 20% feature vectors of the same game and tool applications (Figure 2a). The feature vectors were assigned in a random fashion. This experiment was repeated for each device 20 times, with different allocations of the training and testing set which results in a total of 5,400 runs (Table 1).

Experiment 2

The purpose of this experiment is to evaluate the ability of each combination of detection algorithm, feature selection method, and number of top features to differentiate between game and tool applications not included in the training set. The configuration of this experiment resembles the first one. However, unlike the first experiment the training set contained feature vectors clusters for 80% of all games and 80% of all tools. The testing set contained feature vectors clusters of the rest of the 20% games and 20% tools that were not included in the training set on the same device

(Figure 2b). This examined the ability of the different algorithms to detect unknown applications. This experiment was repeated for each device 20 times, with different allocations of the training and testing set which results in a total of 5,400 runs (Table 1).

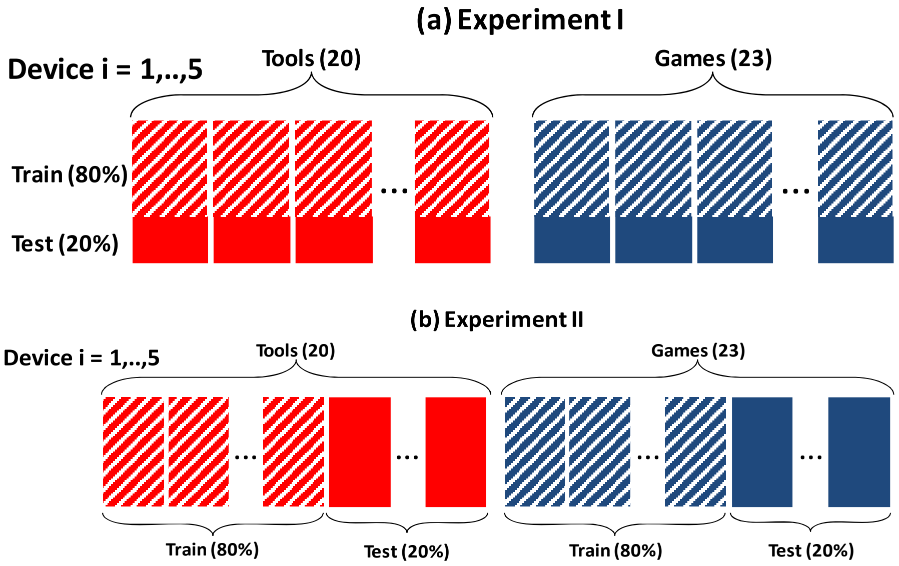


Fig. 2. Illustration of the datasets in each experimental configuration

Figure 3 presents, for each experiment, the average Accuracy and FPR, of the five devices when combined together, for each detector. The results show that the best detectors (classifiers) in experiment I and II are Decision Tree, Logistic Regression and Bayesian Networks. Additionally, we observed that in experiment I the five devices had similar results for BN, DT, Logistic Regression and NB. For experiment II the five devices had similar results for Logistic Regression and NB.

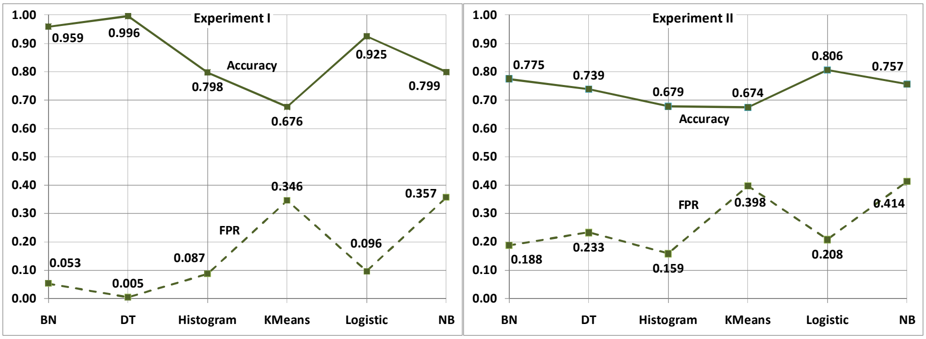


Fig. 3. Average Accuracy and FPR for each one of the detectors

Table 2 presents the averaged Accuracy and FPR of the three feature selection methods and the three top numbers of selected features. The results indicate that for experiment I, InfoGain with top 20 features outperformed all the other combinations. For experiment II Fisher Score with top 10 outperformed all the other combinations.

Table 2. Averaged Accuracy and FPR for each one of the feature selection methods and top features

Exp	Feature Selection Method	FPR			Accuracy		
		10	20	50	10	20	50
I	ChiSquare	0.160	0.134	0.172	0.852	0.876	0.850
	FisherScore	0.152	0.174	0.167	0.857	0.860	0.857
	InfoGain	0.155	**0.129**	0.174	0.850	**0.877**	0.850
II	ChiSquare	0.270	0.258	0.280	0.732	0.751	0.725
	FisherScore	**0.250**	0.263	0.268	**0.750**	0.750	0.735
	InfoGain	0.265	0.263	0.282	0.729	0.747	0.724

Figure 4 present the selected features for each feature selection method. The top 10 selected features for each feature selection method were picked as follows. All of the feature selection methods ranked each feature that they pick. A higher rank indicates that the feature differentiates better between game and tool applications using the feature selection method. For each device we listed the top 10 features in a descending order according to their rank for each feature selection method. Corresponding to the features' rank, each feature was given a score starting from 10 to 1 (10 for the most significant feature according to the feature selection method). Then, for each feature selection method we calculated the sum of scores over all the devices for each feature selection method. Additionally, we omitted features with a low score and features that were chosen only for one device. The features in the graph are ordered primarily by the number of devices that selected the feature and then by their score. From Figure 4 we conclude that Chi-Square and InfoGain graded the same top 10 selected features with a very similar rank. Both of them assigned the highest ranks to the following features: Load_Avg_15_mins, Total_Entities, Running_Processes, Mapped_Pages, Battery_ Voltage, Context_Switches, Avg_Key_Dwell_Time, Schedule_Calls and Anonymous_ Pages.

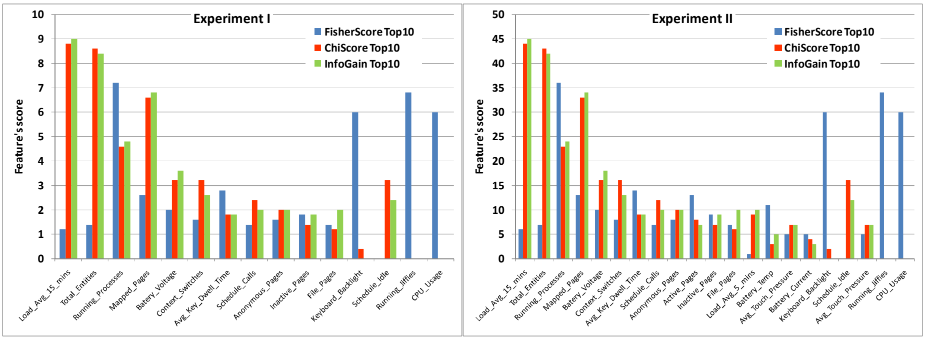


Fig. 4. Best selected features

6 Discussion and Conclusions

We presented an HIDS framework for Android which employs Machine Learning and tested various feature selection methods and classification/anomaly detection algorithms. The detection approach and algorithms are light-weight and run on the device itself. There is however also an option to perform the detection at a centralized location, or at least report the analysis results such a location. This can be useful in detection of malware propagation patterns across a community of mobile devices. As the framework is open, it is also possible to accommodate additional malware detection techniques (e.g., knowledge-based inferences [28]).

Table 3 presents the best configurations, which outperformed all other configurations in terms of ability to differentiate between game and tool applications for each of the experiments (averaged over the five devices and all iterations).

Table 3. Averaged Accuracy, FPR, TPR and AUC of the best configuration in each experiment

Experiment	Best Configuration	TRP	FPR	AUC	Accuracy
I	DT\J48 InfoGain 20	0.997	0.004	0.998	0.997
II	LR FisherScore 20	0.828	0.199	0.888	0.818

From the results we conclude that anomaly detection is feasible on Android devices. The successful differentiation between games and tools provides a positive indication about the ability of such methods to learn and model the behavior of an Android application and potentially detect malicious applications. Furthermore, the fact that the detection can be effective even when using a small number of features (20 features were sufficient in both experiments) and simple detection algorithms ensure that stringent resource constraints (i.e., CPU, battery) on the device are met. These findings are congruent with earlier work which noted that most of the top ten applications preferred by mobile phone users affected the monitored features in different patterns [12]. This observation strengthens the viability of anomaly detection techniques for detection malware on mobile devices.

Looking at the performance of detectors on each of the five devices separately, it also is evident that they exhibit similar performance. This supports the external validity of our experiments by indicating that the selected features in our experiments are not user-, or configuration-dependent, and that we can learn on a set of devices and detect effectively even on other devices. Additionally, there is high similarity in the features that were selected on each of the experiments. We suggest that it is due to a unique and persistent behavior of applications across devices.

Several avenues can be explored for future research. First and foremost we would like to understand whether we can train the classifiers on a set of devices and can still detect effectively on other devices. Second, we can alert about a detected anomaly when it persists. Third, we can add a temporal perspective by augmenting the collected features with a time stamp (e.g., use the change of the battery level in the last 10min rather than the level of the battery at a certain point in time), or logging sequences of events (e.g., a Boolean feature that is true if there was an access to an SD-card concurrently with a high volume of network traffic via Wi-Fi). Finally, we can focus on monitoring and detection of malicious processes rather than monitoring the whole system. This will enable to isolate the activities of specific applications.

References

1. Leavitt, N.: Mobile phones: the next frontier for hackers? Computer 38(4), 20–23 (2005)
2. Shih, D.H., Lin, B., Chiang, H.S., Shih, M.H.: Security aspects of mobile phone virus: a critical survey. Industrial Management & Data Systems 108(4), 478–494 (2008)
3. Piercy, M.: Embedded devices next on the virus target list. IEEE Electronics Systems and Software 2, 42–43 (2004)
4. Cheng, J., Wong, S.H., Yang, H., Lu, S.: SmartSiren: virus detection and alert for smartphones. In: Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (2007)
5. Muthukumaran, D., et al.: Measuring integrity on mobile phone systems. In: Proceedings of the 13th ACM Symposium on Access Control Models and Technologies (2008)
6. Botha, R.A., Furnell, S.M., Clarke, N.L.: From desktop to mobile: Examining the security experience. Computer & Security 28, 130–137 (2009)
7. Dagon, C., Martin, T., Starner, T.: Mobile phones as computing devices the viruses are coming. Pervasive Computing, 11–15 (2004)
8. Emm, D.: Lasco: the hybrid threat. Computer Fraud and Security (2005)
9. Moreau, Y., Preneel, B., Burge, P., Shawe-Taylor, J., Stoermann, C., Cooke, C.: Novel techniques for fraud detection in mobile telecommunication networks. In: ACTS Mobile Summit (1997)
10. Samfat, D., Molva, R.: IDAMN: An intrusion detection architecture for mobile networks. IEEE Journal on Selected Areas in Communications 15(7), 1373–1380 (1997)
11. Yap, T.S., Ewe, H.T.: A mobile phone malicious software detection model with behavior checker. In: Shimojo, S., Ichii, S., Ling, T.-W., Song, K.-H. (eds.) HSI 2005. LNCS, vol. 3597, pp. 57–65. Springer, Heidelberg (2005)
12. Schmidt, A., Peters, F., Lamour, F., Albayrak, S.: Monitoring smartphones for anomaly detection. In: Proceedings of the 1st International Conference on Mobile Wireless Middleware,Operating Systems, and Applications (2008)
13. Bose, A., Hu, X., Shin, K.G., Park, T.: Behavioral detection of malware on mobile handsets. In: Proceeding of the 6th International Conference on Mobile Systems, Applications, and Services (2008)
14. Kim, H., Smith, J., Shin, K.G.: Detecting energy-greedy anomalies and mobile malware variants. In: Proceeding of the 6th International Conference on Mobile Systems, Applications, and Services (2008)
15. Buennemeyer, T.K., et al.: Mobile device profiling and intrusion detection using smart batteries. In: International Conference on System Sciences, pp. 296–296 (2008)
16. Nash, D.C., et al.: Towards an intrusion detection system for battery exhaustion attacks on mobile computing devices. In: Pervasive Computing and Communications Workshops (2005)
17. Jacoby, G.A., Davis, N.J.: Battery-based intrusion detection. In: Global Telecommunications Conference (2004)
18. Hwang, S.S., Cho, S., Park, S.: Keystroke dynamics-based authentication for mobile Devices Computer & Security 28, 85–93 (2009)
19. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering. ACM Computing Surveys 31(3), 264–296 (1999)
20. Neter, J.K., Nachtsheim, M.H., Wasserman, W.: Applied Linear Statistical Models. McGraw-Hill, New York (1996)
21. Endler, D.: Intrusion detection: Applying machine learning to solaris audit data. In: Proceedings of the 14th Annual Computer Security Applications Conference (1998)

22. Quinlan, J.R.: C4.5: Programs for machine learning. Morgan Kaufmann Publishers, Inc., San Francisco (1993)
23. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kauhmann, San Francisco (1988)
24. Russel, S., Norvig, P.: Artificial Intelligence: A modern approach. Prentice Hall, Englewood Cliffs (2002)
25. Imam, I.F., Michalski, R.S., Kerschberg, L.: Discovering Attribute Dependence in Databases by Integrating Symbolic Learning and Statistical Analysis Techniques. In: Proceeding of the AAAI 1993 Workshop on Knowledge Discovery in Databases (1993)
26. Golub, T., et al.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286, 531–537 (1999)
27. Shannon, C.E.: The mathematical theory of communication. The Bell system Technical Journal 27(3), 379–423 (1948)
28. Shabtai, A., Kanonov, U., Elovici, Y.: Intrusion Detection on Mobile Devices Using the Knowledge Based Temporal-Abstraction Method. Journal of Systems and Software, (2010) doi:10.1016/j.jss.2010.03.046

Appendix A - List of Used Applications

	Application	Device1	Device2	Device3	Device4	Device5
Games	abduction	322	197	205	257	341
	armageddonoid	343	222	198	264	326
	battleformars	544	208	269	294	582
	bonsai	355	287	267	253	304
	breadfactory	300	208	241	246	291
	connect4	308	204	225	249	289
	flyinghigh	279	250	206	234	365
	froggo	308	185	241	263	301
	hangemhigh	342	348	274	255	339
	labyrinthlite	317	297	244	252	324
	lexic	354	266	222	273	347
	minesweeper	342	267	249	285	541
	mushroom	251	172	193	199	209
	pairup	410	275	298	259	582
	picacrossexpress	300	357	255	302	332
	smarttactoe	298	233	254	262	300
	snake	294	254	250	233	314
	solitaire	334	378	281	288	439
	switcher	303	231	269	236	395
	tankace	336	287	250	262	330
	throttlecopter	321	230	228	210	280
	trap	454	245	189	251	398
	wordpops	301	314	302	320	267
Tools	browser	307	274	218	336	300
	calculator	296	251	266	276	365
	calendar	319	233	250	270	341
	camera	302	251	249	260	294
	contacts	303	230	218	256	680
	email	219	250	445	275	339
	im	371	245	238	251	385
	iofilemanager	295	235	241	289	284
	maps	342	216	245	278	429
	messaging	322	247	255	278	296
	music	304	251	272	256	343
	mytracks	356	233	269	271	543
	noteeverything	329	212	287	275	408
	oxforddictionary	323	275	272	283	374
	pdfviewer	280	249	248	263	319
	phonalyzer	304	240	268	290	318
	phone	125	224	140	110	244
	tasks	300	226	250	263	302
	voicememo	312	230	270	253	269
	weather	372	242	272	269	297

Appendix B - List of Collected Features

Collected Features (88)

Touch screen:
- Avg_Touch_Pressure
- Avg_Touch_Area

Keyboard:
- Avg_Key_Flight_Time
- Del_Key_Use_Rate
- Avg_Trans_To_U
- Avg_Trans_L_To_R
- Avg_Trans_R_To_L
- Avg_Key_Dwell_Time
- Keyboard_Opening
- Keyboard_Closing

Scheduler:
- Yield_Calls
- Schedule_Calls
- Schedule_Idle
- Running_Jiffies
- Waiting_Jiffies

CPU Load:
- CPU_Usage
- Load_Avg_1_min
- Load_Avg_5_mins
- Load_Avg_15_mins
- Runnable_Entities
- Total_Entities

Messaging:
- Outgoing_SMS
- Incoming_SMS
- Outgoing_Non_CL_SMS

Power:
- Charging_Enabled
- Battery_Voltage
- Battery_Current
- Battery_Temp
- Battery_Level_Change
- Battery_Level

Memory:
- Garbage_Collections
- Free_Pages
- Inactive_Pages
- Active_Pages
- Anonymous_Pages
- Mapped_Pages
- File_Pages
- Dirty_Pages
- Writeback_Pages
- DMA_Allocations
- Page_Frees
- Page_Activations
- Page_Deactivations
- Minor_Page_Faults

Application:
- Package_Changing
- Package_Restarting
- Package_Addition
- Package_Removal
- Package_Restart
- UID_Removal

Calls:
- Incoming_Calls
- Outgoing_Calls
- Missed_Calls
- Outgoing_Non_CL_Calls

Operating System:
- Running_Processes
- Context_Switches
- Processes_Created
- Orientation_Changing

Network:
- Local_TX_Packets
- Local_TX_Bytes
- Local_RX_Packets
- Local_RX_Bytes
- WiFi_TX_Packets
- WiFi_TX_Bytes
- WiFi_RX_Packets
- WiFi_RX_Bytes

Hardware:
- Camera
- USB_State

Binder:
- BC_Transaction
- BC_Reply
- BC_Acquire
- BC_Release
- Binder_Active_Nodes
- Binder_Total_Nodes
- Binder_Ref_Active
- Binder_Ref_Total
- Binder_Death_Active
- Binder_Death_Total
- Binder_Transaction_Active
- Binder_Transaction_Total
- Binder_Trns_Complete_Active
- Binder_Trns_Complete_Total

Leds:
- Button_Backlight
- Keyboard_Backlight
- LCD_Backlight
- Blue_Led
- Green_Led
- Red_Led

Context-Enhanced Web Service Invocations in Mobile Business Processes

Heinz-Josef Eikerling

Fachhochschule Osnabrück – University of Applied Sciences,
Laborbereich Technische Informatik / Institute of Computer Engineering
Barbarastraße 16, D- 49076 Osnabrück, Germany
h.eikerling@fh-osnabrueck.de

Abstract. We present a mechanism which transparently enhances service invocations by contextual data (e.g., location / positioning data gained through other information that is accessible within the invocation chain): the context is applied and extracted in a way such that the called service(s) can be completely agnostic to the intricacies of context encoding, transmission, and processing. This is particularly important when accessing services from within a mobile environment, since mobile processes are sensitive to contextual data like location, user and device status by nature. We intercept the service request / response handling prior to calling the business logic and the delivering of the response to the calling application, respectively. The proposed approach turns out to be rather applicable in Field Force Automation scenarios and efficient which is proved by some experimentation.

Keywords: mobile business processes, context, web services.

1 Introduction

In today's business environment, efficiency and business success increasingly relies on how companies manage to seamlessly integrate employees into the business process, independent from where they are and which device they are using. This is frequently referred to as *user mobility*. In addition to the mobility of employees, within a business process a company has to deal with mobile assets, i.e. moving / movable objects like a specific medical device within a hospital (*asset mobility*). Being able to dynamically involve both as *context data* in business processes offers a huge potential to optimize process execution speed, process cost, and asset utilization.

The above aspects have to be aligned to the tendency to wrap certain functionalities as services in order to foster reusability and gain flexibility to move services to external service providers [10]. Services can perform everything ranging from simple functions to complicated business processes. They allow business entities to deliver their key offerings over the Internet [8]. As an underlying technology, web services (WS) and the underlying service-oriented architecture (SOA) paradigm both have gained attraction. Particularly mobile applications are nowadays built using services [4], i.e., applications consume services and may also offer services to other mobile devices within a certain range.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 250–262, 2010.

Contextual data may help to tackle the problems linked to the integration of services, similar as for humans where it expands the conversational bandwidth [1] and enables to use implicit, additional data to understand and process transmitted complex information more effectively. Aligning to a widely adopted definition [3], we assume context to be any information suited to characterize the situation of a person, a computing device, or a software agent. Instances of information covered by this definition are distinct locations, capabilities and services offered or sought, or activities and tasks in which services are utilized.

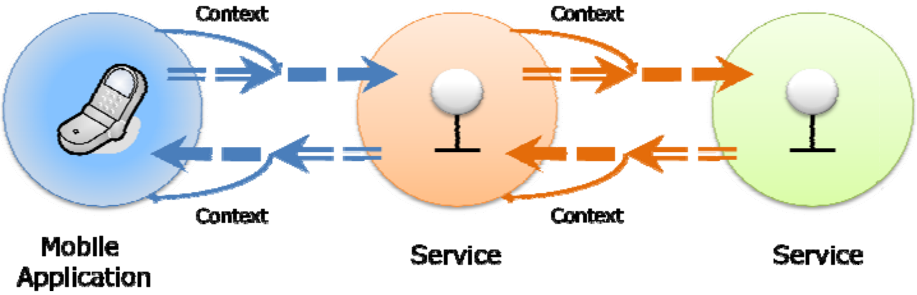


Fig. 1. Context enhancement for accessing services in mobile applications

The target of this work is to devise a mechanism which enhances service calls by context information as shown in Figure 1. The context can be delivered and extracted in a way fully transparent to the involved services. Thus explicit changes to the service interface can be avoided. The context enhancement can be applied to the service request and also to the response returned by the invoked service. A request can be initiated by an application or by an intermediate service in the request processing chain. Due to their importance we focus on web services and describe a mechanism for the automatic, at runtime adaptable and transparent provisioning of context data fulfilling the aforementioned requirement.

2 Sketch of Proposed Solution

Our solution consists of a generic mechanism for *implicit* context enhancement and extraction which makes customized context available to conventional web services. This means that the services do not have to take care of the actions necessary to attach context. Additionally, the target services do not have to process the contextual data themselves; context processing can in principal be delegated to other specific services.

We consider that the context-enhanced interaction has to be possible even if the services belong to different infra-structures. Most importantly, the contextualized services will stay usable and functional, no matter if they are capable of processing the context or not. Therefore, the solution will not limit the set of potential target services in a composition by requiring special properties like being consistent with a specific framework and its conventions.

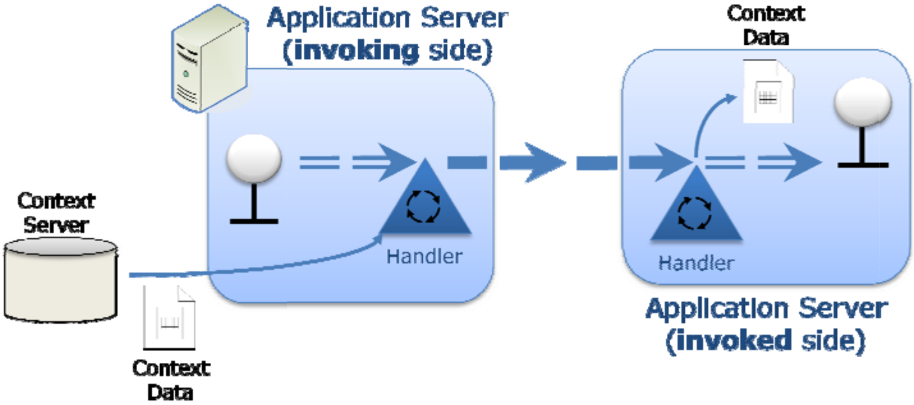

Application Server (invoking side)
Context Data
Context Server
Handler
Context Data
Handler
Application Server (invoked side)

Fig. 2. General principle of enhancing service calls with context data

In the subsequent description we will focus on the invoking side for reasons of conciseness. We assume that components representing the invoking and invoked side are deployed to some kind of container, e.g. an application server or service container. The handlers used for context enhancement have to be installed on these containers as shown in Fig. 2.

3 Related Work

The two research areas, context-aware computing and web service interaction, are rather active. A lot of work on how to apply e.g. spatio-temporal context exists. However, existing approaches are mostly limited to special scenarios and do not use a more capable overall framework for implementing context awareness. They also often consider a fixed set of context elements and thus lack extensibility. Especially, there have been several projects on location-aware systems [2]. [5] describes a system where users can access information depending on their geographical positioning. This idea has also been implemented in [11] where users can attach annotations to geographic positions such that the information is accessible by other users whenever they approach that position. A presence aware system is presented in [7].

In the COWSPOTS [9] project context-awareness is used to provide enriched Web services to health professionals. The system consists of a central server and several mobile devices that run so called SPOTlets. According to this centralized approach, the whole processing is completely done on the server side.

While several approaches to the problem stated above present good mechanisms to context management and/or processing, they fail to constitute a mechanism that considers traditional web services when it comes to context-aware service invocations. Solely [6] presents a framework that facilitates the development and deployment of context-aware adaptable Web services. The framework features context plug-ins that pre- and post-process service requests based on available context information. Context is iteratively appended to the exchanged messages whenever a service sends a message to another service. On the invoked side the appended context can be extracted, processed, and used either by the invoked service or through the context processing plug-ins. The authors assume one plug-in per type of context, i.e. if a Web service *A* invokes another service *B* with context information, this context is appended in addition to the context transmitted by *B*, if *B* invokes another Web service *C* during that invocation by *A*. The approach supports invocations of services which are not context-aware. However, the solution does not permit the sent context to be customized to an invocation (e.g., the target or the method parameter). Also, the entire context of a service is forwarded along cascaded service calls which might likely cause a significant message overhead. Since context plug-ins are designed to handle one type of context only, several context plug-ins have to be activated simultaneously during each invocation which complicates the use of context combinations.

4 Context-Enhanced Service Calls

4.1 General Principle

We start off by separating the different concerns of the envisaged solution. In general, a framework that separates tasks like context acquisition / retrieval and storage from context queries would be of benefit. Therefore, the general solution consists of two parts. The first part is given by the *context server*. Through defining such server concurrent and remote access to contextual data is granted. The server acts as the central point for context data retrieval requests, context processing and distribution. Such approach ensures the maximum flexibility concerning context handling.

The advancement of this work over the state of the art is given by the second part, a *handler mechanism* deployed to the application server that post- and pre-processes exchanged messages to acquire and adjust context from the context server during the invocation of business methods and provide it to the invoked web service.

4.2 Context Storage and Retrieval

The context server supports multiple remote data sources by a standardized interface and concurrent access. This is important in order to enable different kinds of sensors to send context information to the server. Part of this solution is an adapter interface to attach different types of *context producers* (sensors). The adapter hides the details of low-level data acquisition and retrieval. It receives context data from the proprietary sensor interface and uses the server interface to publish context to the connected *context consumers*. For the general use of the context server in different scenarios, the ability to define new context types is offered.

For implementing this functionality the context server comprises a powerful rule engine as shown in Fig. 3. In this regard context pre-processing rules can for instance specify how to aggregate and transform context data delivered by the producers. Clearly, the variety and precision of changes to the context data prior to its persistent storage are directly linked to the supported complexity of these rules. Though the data flow is always directed from the context producers to the context consumers, for both – producers and consumers – *push* and *pull mode* have to be supported. This is handled by the management component inside the context server which permits to define rules needed to query for context (pull mode consumer).

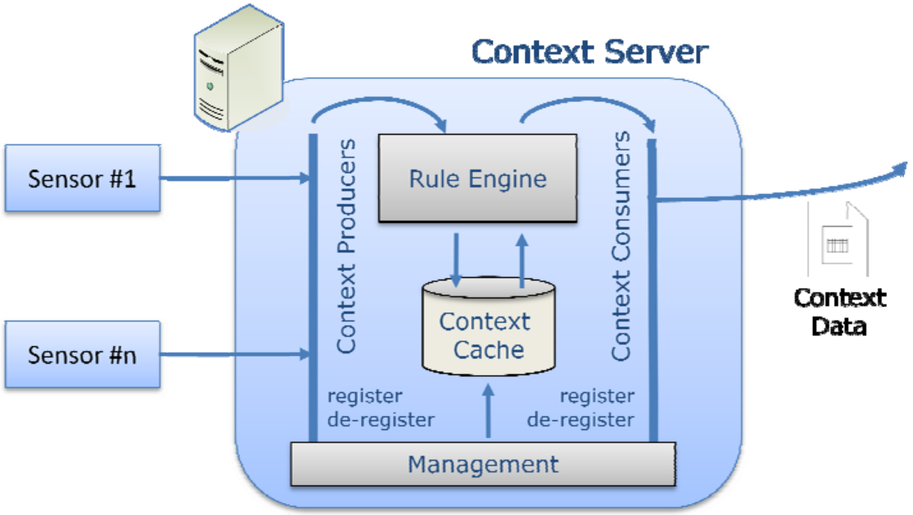


Fig. 3. Context storage and retrieval through the context server

An important task of the context server is to store the collected context information and make it available for subsequent use (*historic context data*). Hence, if no active context is available and requesting the respective context source / sensor is not possible, the most recent value can be obtained directly from the context cache.

It is worth mentioning that the conceptually centralized context server can be implemented using several physical servers for increased scalability or for providing the service to different application domains. This is easily achieved by making one context server a *context producer* to another one.

4.3 Context Transmission, Encoding and Provision

The transmission of context has to comply with web service standards. In addition, the mechanism has to be *implicit* such that context does not have to be transmitted *explicitly*, e.g., by adjusting WSDL files. Usually redirection, port and message filtering techniques are in place to secure the company network; only special point-to-point-connections are maintained for the communication with other (external) businesses. Setting up the network devices to allow a further communication channel can therefore be very expensive and ineffective, thus impossible. Hence, the context should be transmitted along with the SOAP message.

Different standards exist to populate the SOAP message header with meta-data in order to accomplish certain tasks. Our solution makes use of a header block which is inserted into the message sent by the invoking service. The header block contains the actual XML-encoded context information. This is in line with a W3C recommendation [12] stating that all information which could be of importance to nodes other than the actual communication endpoint (the target Web service) should be moved into the header, so that these nodes can provide value-added functions.

Instead of attaching context data on a dedicated middleware layer underneath the level of the service, a *handler* inside the application / service container is used to attach and detach the context. The handler extends the core functionalities of an application server by implementing functions that are invoked during the processing of incoming and outgoing messages. The handler used in the solution basically queries the context server for contextual information and creates the according header block in the SOAP message.

A handler is not only used on the sending side, but also on the receiving side, inside the container hosting the invoked service where it pre-processes the SOAP message. Before the web service is invoked with the according parameters contained in the SOAP body, the handler extracts the context information. If the web service is context-aware, it can intercept the context during the operation invocation and does not have to parse the SOAP message itself. In case the web service is not context-aware, it will simply ignore the context and will be not affected.

An important requirement constraining the context provisioning mechanism inside the service container is that the employed data structure has to be unique for every invocation. If for instance an operation of one service is invoked simultaneously by two different services *A* and *B*, inside every invocation access to the according context has to be provided. Otherwise a race condition could occur (see Fig. 4).

4.4 Context Selection and Handler Configuration

For deciding which context data should be appended for a specific invocation, *general* and *invocation-related* properties have to be distinguished. While the *invocation-related* properties (i.e., the invoked target service, the invoked target operation of the service, the current operation parameter values) are configured and detected on the handler side, the *general* properties are set up and evaluated in the context server.

Thus,

1. the *invocation-related* properties are set up in a handler configuration file
2. the *general properties* can be configured directly in the context server via according rules

To decide for the general properties which context information to include, different rule sets are defined directly inside the server. Depending on the set of rules used, different contextual information is collected and returned upon a query.

The decision is made in two steps.

1. First, the handler of the invoking service takes the SOAP message to determine the targeted service, the invoked operation and the parameter values. A *scenario identifier* (*SCID*) in the handler configuration file is provided for each service operation. This identifier along with the parameter values is used to query the context server for the required data.
2. In the second step, the context server uses the transmitted scenario identifier to select one of the rule sets. The rules define how the parameters have to be taken into account and which context should be collected in general. Rules are also used to transform the contextual information into an XML structure such that the handler can directly insert the response of the context server into the header block for the SOAP message.

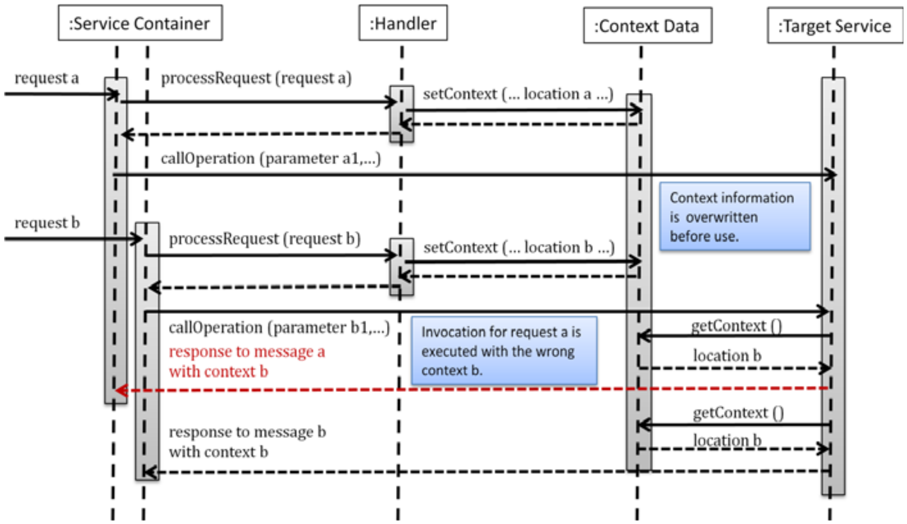

:Service Container
:Handler
:Context Data
:Target Service
request a
processRequest (request a)
setContext (... location a ...)
callOperation (parameter a1,...)
Context information is overwritten before use.
request b
processRequest (request b)
setContext (... location b ...)
callOperation (parameter b1,...)
response to message a with context b
Invocation for request a is executed with the wrong context b.
getContext ()
location b
getContext ()
location b
response to message b with context b

Fig. 4. Invocation of a service by two entities A and B causing race condition if the context data (here location of an object) is not handled per invocation

In general, the parameters of a service invocation can be complex, e.g., multi-dimensional types, for example composed objects or arrays. For use with the context server, scalar values are most suitable because usually one might not need all entries of a complex type for the context query, but only certain ones. In order to offer a general way to describe which scalar value inside a possibly multidimensional parameter should be chosen, XPath expressions are used.

With this approach, it is possible to keep the handler as simple (and thus effective) as possible and leave all context-related decisions to the context server.

4.5 Implementation Notes

The previously specified system was built on top of the context server described in [13] which is not specifically addressing contextualized services as context consumers and producers. The handler mechanism was initially developed for a stationary environment [14] and makes use of the according features in the Apache Axis2 as the service container.

5 Experiments

5.1 Purpose

The above approach can be judged concerning two criteria:

1. *Performance* is a key demand for applying the above described mechanism in business environments. This translates into the requirement that the benefits of context-enhanced services come with an as small as possible overhead in terms of message latency, throughput and processing time.

2. The overhead can be also measured in terms of the *message size* which has to relate the net size of a service invocation (non context-enhanced) with the gross size of it (context-enhanced).

The latter is very scenario specific which is why we have focused on evaluating the performance. One way to approach this is to compare the context handling mechanism by *explicitly* extending the service interface through the *implicit* mechanism using the proposed handler. If the latter is not slower and does not produce significantly larger messages than using the service interface to transfer context information, then the performance requirements are met. Hence we conducted an experiment test on a service invocation in which a certain amount of information (and context) is transferred using the explicit and the implicit transmission of context. First, the runtimes for executing the invocations are compared.

5.2 Environment

In order to factor out effects like packet loss or latency that occur in network environments, all applications are executed on the same computer using the local loopback device for communication. In such environment, the measuring of execution times can be simplified since the local system time can be used to determine execution time intervals between different invocations. The time between the end of the invocation of operation *A* and the start of the invocation of operation *B* is simply determined by measuring the interval between the time for the last instruction in *A* and respectively first instruction of *B*.

5.3 Results

Aside from tests measuring times spent at the different stages of the processing pipeline (request creation, context retrieval, context appending, request processing, sending response) we focused on the direct comparison of the execution times for the explicit and implicit method for transmitting contextual information.

Fig. 5 shows the results of executing these tests. We conducted a series of 6 scenario runs (*A*, *B*, *C, D, E, F*) each comprising 500 invocations; each invocation is handled by the explicit and by the implicit method. The execution times are recorded and the truncated averages (max / min values are dropped to factor out to temporal phenomena) are computed and shown for all three runs.

The tests indicate that the deviation concerning runtime between the handler-based and explicit interface-based mechanisms for context delivery is below 2% on average over the examined scenarios. This is because the additional software components (handlers for context attaching / detaching, context server) on the processing chain perform quite efficiently. As will be explained later, the benefits (separating the static business logic from the rather instance-specific dynamics of context management) more than compensate this overhead.

Similar observations were gained when evaluating the message size overhead: assuming the same encoding of the context to be transmitted in either approach, the message sizes are the same. The explanation is straight-forward: whereas in the explicit, interface-based approach the context data is contained in the SOAP body, the handler-based, implicit approach moves this data to the SOAP header. Thus the overall message size is the same.

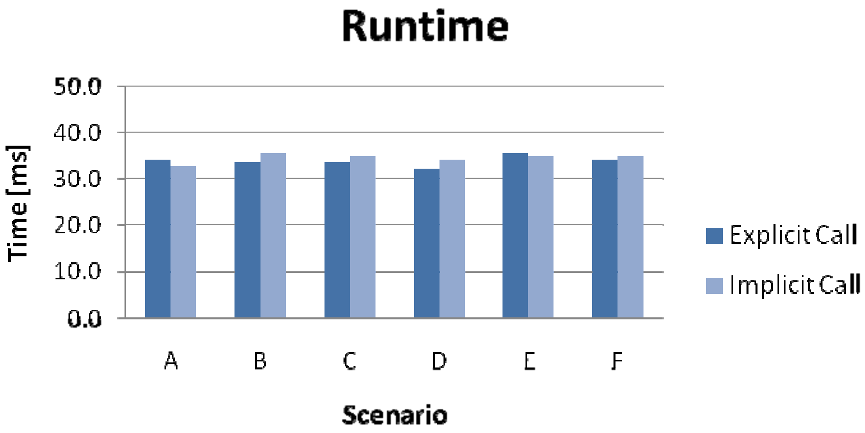


Fig. 5. Comparison explicit / interface and implicit / handler based approach for testing scenarios

6 Case Study: Field Service Automation

6.1 Field Service Automation in the Automotive Sector

Advances in technologies have lead to an increasing diversity in the automotive sector. The consequences are bigger differences between car manufacturers and their models, so that the collaboration between car manufacturers and business partners (e.g., field workers) handling repairs is gaining importance. In our scenario, we look at the interaction of the entities in case a client car is broken during the warranty period.

Among the actors is the car *dealer (A)* contracted to a *manufacturer* (*B*). *Field workers* (*C* and *D*) are associated with the manufacturer. In the scenario the customer contacts his dealer because an issue (e.g., malfunctioning of the car electronics) with his car occurred.

For fixing the problem the dealer contacts the manufacturer by sending a notification to the manufacturer's (step 1.) service via a straight-forward interface. The sent message (2.) contains the incident id (*I-ID*), a number the dealer uses to refer to the case, and a problem class id (*P-ID*), a number which describes the class of the problem (electronic problem, tank leakage, …), to *B*. Since the incident happens at the site of the dealer, the dealer's location might be of importance to field worker for deciding whether to take over the job or not. The manufacturer maintains a directory from which the contact data for *A* can be retrieved (3.).

With the *I-ID* and *P-ID* information plus the contact information of the dealer, the car manufacturer *B* tries to assign the problem to one of the field workers. He checks with one field work after the other, until the issue is successfully assigned which is indicated by an according response of the field worker.

After the issue was successfully assigned, the field worker accepting the issue (here *D*) contacts the dealer with the *I-ID* (the number the dealer assigned to refer to the case) which was obtained from the manufacturer and they negotiate the next steps to cooper up the car.

6.2 Service-Oriented Implementation of Scenario

Figure 6 shows the service-oriented implementation of the scenario. The web service of manufacturer *B* provides an operation *assignIssue()* which requires two arguments, an integer value for the *I-ID* and another integer *P-ID*. The interface to this operation is known by the service of dealer *A*. Similarly, the services of *C* and *D* published their operation *assignIssue()* that requires an additional parameter besides the *I-ID* and the *P-ID* for the contact information. A list of the associated field worker services is maintained in the service for *B*. Every time *A*'s service invokes the method *assignIssue()* of the manufacturer *B*, *A*'s application server generates and sends the according SOAP message.

At the target URI of *B* the SOAP engine in the service container of *B* accepts the message and invokes *assignIssue()* of the according service with the transmitted parameters *I-ID* and *P-ID*. The service *B* looks up the contact information of *A* by the IP address of the message. With the parameters *I-ID*, *P-ID* and the contact information of *A* (contact) the service *B* now loops through all known field worker's services. For one after another, it invokes their *assignIssue()* method with the three parameters. The application server of *B* therefore creates the according SOAP messages and delivers them to the according end-points.

This straight forward implementation reflects a very low usage of context information. Context information relevant to specific actors (dealer's location for field workers) is resolved through explicit service interfaces which complicates the business logic of the manufacturer unnecessarily.

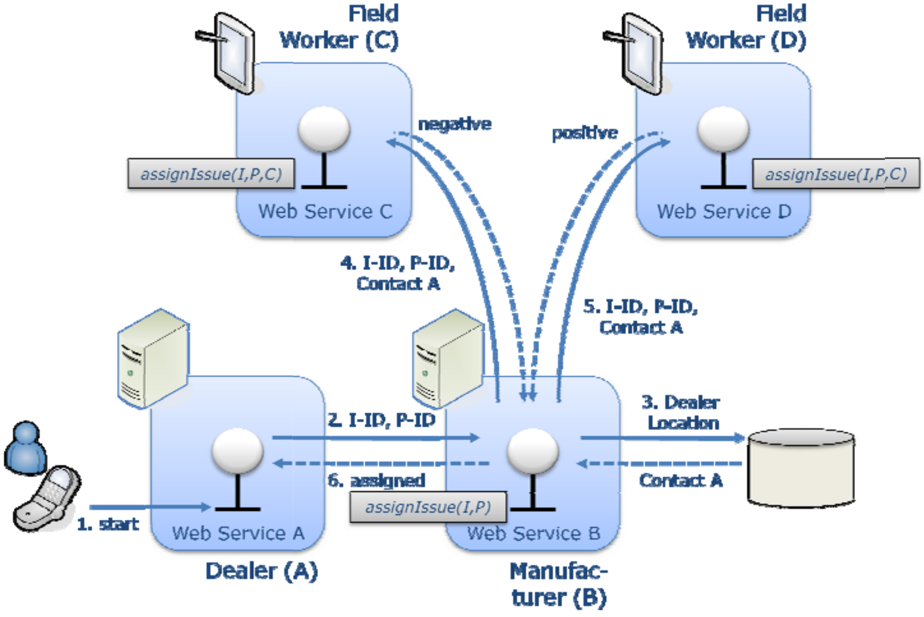


Fig. 6. Assigning an issue in a car maintenance scenario

6.3 Context-Enhanced Implementation of Scenario

For the context-enhancement of the service calls, we now consider the simple case of identifying the contact information (location) of the dealer and propagating it to the field workers. Figure 7 depicts the implementation.

Before the implicit context enhancement mechanism can be used, the sending handler (*ContextAppender*) and the context server have to be set up. For the context-enhanced invocations by the manufacturer, the handler configuration file of the manufacturer *B* contains entries for the field worker service operations *assignIssue()*. Attached to this entry are the value of a scenario identifier *SCID* and the XPath expression to determine the *I-ID* parameter of the SOAP message that is to be sent to *C*'s and D's web services. The context server of the manufacturer is configured with a rule set that returns the needed context when the server is queried by providing an *I-ID* and the *SCID*.

The *ContextAppender* handler determines the target of the SOAP message and looks up the according *SCID*. Afterwards it executes all XPath expressions that were set up to gather the parameters for the context server invocation.

There are different ways to identify the targeted location context of the dealer. A rather simple and fully implicit approach consists of analyzing the IP address of the originating service end-point (i.e., the dealer's service): the context server of *B* compares the IP address of the message to a database table of IP addresses of all dealers and retrieves the correct contact information. The handler inserts the context server response into the header block of the SOAP message. This message is handed back to the service container. *B*'s service container now sends the SOAP message to

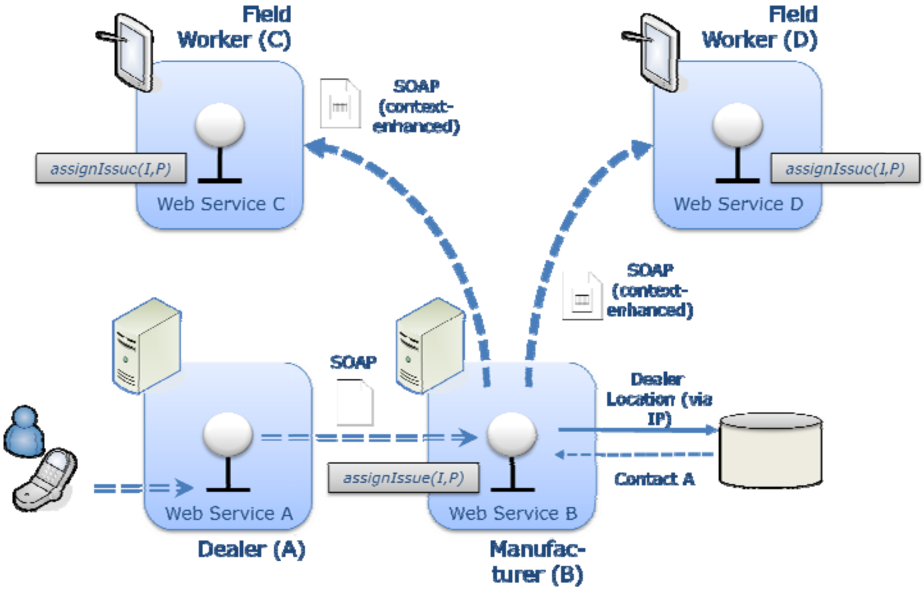


Fig. 7. Context-enhanced service calls in car maintenance scenario. The IP address of the dealer is used to determine the location of the dealer which is attached to the SOAP requests delivered to the repair shops.

the URIs of C and D. Here the *ContextExtractor* handlers preprocess the messages, before the actual operation is invoked. The *ContextExtractor* looks for the header block with the context data. In this case, the message was sent with the context enhancement mechanism, so the handler finds the according header block and extracts the context information.

To make the context accessible to the field worker service, it sets up the according data structure using the API of the application server. Afterwards it passes the control back to the application server.

On the field worker side, a *ContextExtractor* in the according application servers makes this additional context information available to the invoked services. The service of the field worker can then use the location context of the dealership to decide to either accept or reject the job.

6.4 Discussion

We demonstrated the use of context-enhanced service calls in a field force automation scenario. By nature, such scenarios are rather location-sensitive. The power of the approach presented here stems from the fact that the use of other types of context can be easily supplied, e.g.

- additional information concerning the issue to be fixed (beyond the problem class ID) like for instance a problem description could be delivered as context;
- for smaller businesses like field workers, business data on the status of the client (e.g., solvency) could influence the result of the *assignIssue()* method;
- information on the status of the field workers (e.g., work load) constitutes context information useful for the manufacturer during dispatching.

The advantage is that this information can be delivered without changing the core logic of the business process and the involved services.

7 Conclusions

Context information is to be perceived as an essential part of mobile processes. We have described a mechanism which implicitly enhances service calls with context information. This means that the context can be applied and extracted in a way which is transparent for the involved services. The focus is on automatic, customized and transparent provisioning of context and not on how the context is stored or acquired.

The proposed mechanism addresses particularly web services which nowadays are used in all sorts of business processes. While this is evident for services being delivered by stationary hosts, mobile hosts featured in mobile business processes like for instance field force automation scenarios become more appealing. Web service containers offer elegant and powerful mechanisms to deploy the according handlers. Future work will focus on generalising the approach to low foot-print containers running on more limited mobile devices.

References

1. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a Better Understanding of Context and Context-Awareness. In: Gellersen, H.-W. (ed.) HUC 1999. LNCS, vol. 1707, pp. 304–307. Springer, Heidelberg (1999)
2. Baldauf, M., Dustdar, S., Rosenberg, F.: A survey on context aware systems. IJAHUC 2(4), 263–277 (2007)
3. Chen, H.: An ontology for context-aware pervasive computing environments. The knowledge engineering review 18(3),197–208 (2003)
4. Dustdar, S., Schreiner, W.: A survey on web services composition. IJWGS 1(1), 1–30 (2005)
5. Espinoza, F., Persson, P., Sandin, A., Nyström, H., Cacciatore, E., Bylund, M.: GeoNotes: social and navigational aspects of location-based information systems. In: Proceedings of the 3rd International Conference on Ubiquitous Computing, Atlanta, Georgia, USA, pp. 2–17
6. Keidl, M., Kemper, A.: Towards context-aware adaptable web services. In: The Thirteenth International World Wide Web Conference. Alternate track papers & posters, pp. 55–65. Association for Computing Machinery, New York (2004)
7. Kerer, C., Schahram, D., Jazayeri, M., Szego, A., Gomes, D., Caja, J.A.B.: Presence-Aware Infrastructure using Web services and RFID technologies. In: Proceedings of the 2nd European Workshop on Object Orientation and Web Services, Oslo, Norway
8. Papazoglou, M.P.: Service-Oriented Computing: Concepts, Characteristics and Directions. In: Catarci, T. (ed.) Proceedings of the 4th International Conference On Web Information Systems Engineering, pp. 3–12. IEEE Computer Society, Los Alamitos (2003)
9. Pauty, J., Preuveneers, D., Rigole, P., Berbers, Y.: Research Challenges in Mobile and Context-Aware Service Development (2006)
10. Turner, M., Budgen, D., Brereton, P.: Turning Software into a Service. Computer 36(10), 38–44 (2003)
11. Ryan, N., Pascoe, J., Morse, D.: Enhanced Reality Fieldwork: the Context Aware Archaeological Assistant. In: Bar International Series, vol. 750, pp. 269–274 (1999)
12. W3C. (27.04.2007). SOAP Version 1.2 Part 0: Primer (2nd edn.), `http://www.w3.org/TR/soap12-part0/` (retrieved September 18, 2007)
13. Eikerling, H.-J., Benesch, M., Berger, F.: Integrating Analysis of User / Asset Spatiotemporal Relationships for Mobile Field Processes (Demo). In: 4th International Conference Networked Sensing Systems. Braunschweig, Germany (2007)
14. Fazal-Baqaie, M.: Design and Implementation of a Handler Mechanism for Context-Enhanced Service Calls, B.Sc. Thesis, Paderborn University (2008)

Session 6: Short Papers
(Chair: Paolo Bellavista)

Power Aware with the Survivable Routing Algorithm for Mobile Ad Hoc Networks

Golla Varaprasad

Computer Science Editorial, Department of Computer Science and Engineering, B.M.S. College of Engineering, Bangalore-560 019, India
varaprasad555555@yahoo.co.in

Abstract. The mobile ad hoc networks are gaining importance because of their versatility, mobility and ability to work with a limited infrastructure. In the mobile ad hoc network, each node works as a router as well as host. It is generally decentralized network, where all network activities include route discovery and message delivery must be executed by the nodes themselves. In this paper, we present a power-aware with survivable routing algorithm for the mobile ad hoc networks to route the data packets from the source to destination. It works based on the transmission-power and relay-capacity of the node. Both source and destination pair uses the route-selection-window mechanism to route the data packets. The model has simulated using C++ language. The proposed model has tested under various conditions and compared with the minimum total transmission power routing model and min-max battery cost routing model. The simulation results show that the proposed model has increased the route survivability and throughput. It decreases the number of path reconstructions over the network.

Keywords: MANET, route survival, throughput, relay-capacity node, power.

1 Introduction

A Mobile Ad Hoc Network(MANET) is an autonomous collection of mobile nodes, that communicate over the wireless links. Due to the node mobility, the topology will be changed rapidly and unpredictably over the period of time. The MANET does not require any fixed infrastructure and central administration for communications. In the MANET, each mobile node acts as a router and host. It means that all the mobile nodes participating in the network have to send and receive the data packets. Depending on transmission range and current location of the node, the mobile nodes can get in and out, forming a network in an arbitrary fashion. The network partition is an event in MANET environments and inconsistency can prove to be very costly in mobile computation scenarios. The mobile nodes interact with others over wide spaces, inconsistency can be propagated indefinitely. It can cause the unrecoverable damages in all the critical applications. High mobility of nodes has more link failures. It refers to communication failures/ disconnections caused by the nodes moving out of coverage[1].

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 265–275, 2010.

The MANET can be applied anywhere, where there is a little communication or no communication infrastructure or infrastructure is expensive to use. The MANET allows mobile nodes to maintain the multiple connections over the network. It is easily adding and removing the nodes into the network. The set of MANET applications are ranging from the large-scale, mobile, highly dynamic networks to small, static networks that are constrained by the power sources. Due to quick and economically less demanding deployment, we can use this in several areas such as military applications, collaborative and distributed computations, emergency operations, wireless mesh networks, wireless sensor networks and hybrid wireless network architectures.

There are two types of routing models in the MANET such as proactive and reactive used to route the data packets. The proactive routing models are Destination Sequenced Distance Vector Routing Protocol(DSDV) and Wireless Routing Protocol(WRP). The proactive routing models are good at low mobility. The reactive routing algorithms are Dynamic Source Routing(DSR) and Ad Hoc On-Demand Distance Vector(AODV). The reactive routing algorithms are good at low load. Both the proactive routing and reactive routing models do not reduce the power consumption of the node and do not increase the network lifetime[2].

The main challenge in the MANET is to increase the efficiency of the data transfer while handling harsh conditions such as power constraints and highly mobile devices. The advances in the wireless communication are required to overcome the limitations of the broadcast radio networks. In addition to the routing protocols and transport protocols, it must work an intelligent system while routing the data packets from one location to another[3].

In this paper, we present a power-aware with survivable routing algorithm for the mobile ad hoc networks to route the data packets. The proposed work is based on the transmission-power and relay-capacity of the node to increase the route survivability and throughput. Rest of the paper is organized as follows. In section 2, it describes some of the existing projects and their limitations. The proposed model is presented in section 3. Section 4 presents the simulation results. Conclusions are discussed in section 5.

2 Some of the Related Work

This section discusses about some of the existing research algorithms on power-aware routing in the MANET. In the MANET, mobile nodes are operated with a limited battery capacity and frequently recharging/replacement of the batteries may be undesirable or even impossible. The power failure of the node will affect the node-itself and also its ability to forward the data packets to others. For this reason, many researchers have been devoted to design an energy-aware routing protocol for the MANETs. Several recent studies have tried to increase the node lifetime and network lifetime by using the power-aware metrics at different layers[4].

2.1 Power Aware Multi Access Protocol with Signaling (PAMAS)

The PAMAS model is an energy efficient media-access-control protocol for the MANETs. Here, it uses the separate-signaling- channel protocol apart from the channel to transmit the data. The request-to-send message and clear-to-send message packers are used while transmitting the data packets. PAMAS model achieves the goal by making the nodes with power-off. PAMAS protocol is tested in a random network, a line topology and a fully connected network. It provides best results in dense networks, but in small network the power saving is low. The PAMAS protocol exhibits the best performance under light load[5].

2.2 Minimum Total Transmission Power Routing (MTPR)

The MTPR model tries to minimize the total transmission-power consumption for all the nodes participating in a route[6]. The total transmission-power for all the routes is calculated as follows:

$$P(L_d)=\sum_{i=0}^{D-1} T(n_i, n_{i+1}) \tag{1}$$

The optimal route L_o is one of the routes, which verifies the following conditions:

$$P(L_o)=\underset{L_k \in L_*}{Min}\, P(L_k) \tag{2}$$

Drawback of this model is that it selects a path with more number of hops. It accepts the possibility that the participation of more nodes in forwarding the data packets. It also increases the end-to-end delay. The MTPR model fails to consider the remaining-battery capacity of the nodes so that it may not succeed in extending the lifetime of each node in the network.

2.3 Min-Max Battery Cost Routing (MMBCR)

Here, it treats the nodes more fairly from the standpoints of their remaining-battery capacities. The smaller remaining-battery capacities of the nodes are avoided and the nodes with more residual-battery capacities are chosen in a route.

Let us assume that $B_i(t)$ is the battery capacity of the node i and battery cost of the node i calculates as follow:

$$C_i(t)=1/B_i(t) \tag{3}$$

The path cost is defined as follows:

$$R(L_e)=\underset{n_i \in Le}{Max}\, C_i(t) \tag{4}$$

$$R(L_o)=\underset{L_e \in L_*}{Min}\, R(L_e) \tag{5}$$

In equation(5), it selects a route with the minimum cost among all. However, there is no guarantee that the total transmission-power is minimized[7-9].

2.4 Conditional Max-Min Battery Capacity Routing (CMMBCR)

The CMMBCR model takes into account of the residual-battery capacity of the node and total transmission-power consumed by the route while selecting a path. When all the nodes in some possible routes have sufficient battery capacities, a route with the minimum total transmission-power among all is chosen. In order to maximize the lifetime of the network, the power-consumption rate of each node must be evenly distributed. However, if all the nodes in a given path have higher remaining-battery capacity (thr- eshold value(θ)), then chooses a path using MTPR model, otherwise selects a path L_o with the maximum remaining-battery capacity by using MMBCR model[10-13].

$$\mathrm{R}(L_e) \geq \theta, \text{ for any route } L_e Є L_* \qquad (6)$$

$$\mathrm{R}(L_o) = \underset{L_e Є\, L_*}{Min}\ \mathrm{R}(L_e) \qquad (7)$$

The drawback of this model is that it does not consider the network coverage and network partition.

2.5 Minimum Drain Rate (MDR)

The MDR model was proposed by *Kim et al.*[14]. It incorporates the drain rate metric in routing process. The MDR model behaves like a power-aware routing. It can be applied into one of the MANET routing protocols while finding a path from the source to destination. MDR model does not guarantee that the total transmission-power is minimized over a chosen route as in MTPR model.

3 Proposed Model

In order to route the data packets from the source to destination, it uses two metrics namely, minimum total transmission-power and relay-capacity of the node. Here, the source-destination pair chooses an efficient route by using the route-selection window mechanism.

3.1 Route Discovery

In this work, it uses the DSR protocol to route the data packets from the source to destination. The route-discovery process is initiated by the source node. The source node specifies the entire path in a packet header itself to the destination. The route-discovery process allows the mobile nodes to discover a path to the destination using Route Request(RREQ) packet. In the RREQ packet, the type field indicates the type of packet is sent over the network and the flag field is used to make synchronization. The reserved field with ‘0’ value is used to ignore the packet on the reception. The hop-count field is measured the number of hops from the source to destination. In order to identify a route, it uses the RREQ_ID field. The originator IP field indicates the IP address of the source, which originates the RREQ packet. The destination IP field indicates the address of the destination for which a route is desired. The originator-sequence number field provides the current sequence number for the route entries to the source. The destination sequence number field is used for the route entries pointing to the destination. The P_{XT} field indicates the transmission-power of the node. The P_{XR} field indicates the received power of the node and RN_{i} field represents the relay- capacity of the node as shown in table 1.

Table 1. RREQ packet

Type					Reserved	Hop-count
RREQ-ID						
Originator IP						
Originator-Sequence Number						
Destination IP						
Destination-Sequence Number						
P_{XT}						
P_{XR}						
RN_{i}						

3.2 Route Selection

Let us assume that S is the source and D is the destination. The nodes 2 and 3 are intermediate nodes with the transmit powers P_{XL2}, P_{XL3} and relay capacities with RN_1 and RN_2 respectively. Here, the node S uses the route-selection window for 3 ms to find a route. In the route-discovery phase, the S broadcasts the RREQ packet. The intermediate nodes 2 and 3 forward the RREQ packets over the network. The node D accepts the RREQ packets, reverses the route and sends Route Reply(RR) packet within the route-selection window for 2 ms. The intermediate nodes 2 and 3 receive the RR packets and calculate their transmission powers and relay-capacities of the nodes. The node S receives the RR packets from the nodes 2 and 3, then it selects a path with the minimum transmission-power and maximum relay-capacity of the nodes. In this work, each node maintains the power table as shown in the Table 2 with the entries of neighboring nodes such as estimated transmission-power and received power of the node and last-packet received time. The power table is result of all the RR packets received by the node. If a node S wants to forward a data packet, then it uses the power table. Once the route formation is completed, the node S sends the data packet to the node D. Every node records its P_{XT} in the data packet and sends it to the next hop. If the next-hop receives the data packet at P_{XR}, then it reads the P_{XT} and calculates *Total power* for previous node as follows:

$$Total\ Power = P_{XT} - P_{XR} \qquad (8)$$

Table 2. Power table of the node *i*

SN	Link	P_{XL}	P_{XR}	Last-packet-received time

The intermediate nodes 2 and 3 check the *Total Power* value in the RR packet. If the RR packet contains less value of *Total Power,* then the value is stored in its database. The data packets are transmitted through the node with less energy consumption over the network.

$$RN_i = NT_i * LT_i \tag{9}$$

In equation(9), NT_i is the current data rate of the node *i* and LT_i is the lifetime of the node *i*.

$$LT_i = \frac{RE_i}{E_i(t)} \tag{10}$$

In equation(10), RE_i is the residual-energy and $E_i(t)$ indicates how much energy is needed per second at the node *i*.

The proposed model allows the node *D* to select a path among the multiple-RRE packets based on the minimum total transmission-power and relay-capacity of the node within the route-selection-window time(2 ms). If a node *D* receives the REQ packet from the *S*, then it starts a timer(route-selection time window(3 ms)). The node *D* selects a route among all the viable routes according to the computation of a decision function. If any two nodes have equal transmission-power, then chooses a node with more relay-capacity.

4 Simulation

In this section, first we explain some of the parameters used in the simulation. Then, we present the simulation results. We compare the performance of all the three models.

4.1 Simulation Environment

The simulation environment has shown in the Table 3. The proposed model simulated in a 1000m×1000m area with 50 mobile nodes using random waypoint model. Here, we designed and implemented our test-bed using C++ language to test the performance of all the three routing algorithms. The mobile speed of each node was from 0-20 m/s and the transmission range was 250 m. Here, it used Constant Bit Rate(CBR) and the data packet size was 512 bytes. The data transmission rate was set to 2 Mbps. Total simulation time was conducted for 7 hours, the source and destination nodes were randomly chosen. Each node was randomly assigned an initial energy (9,000 Joules).

Table 3. Simulation parameters

Simulation parameters	**Value**
Traffic type	CBR
CBR packet size	512 bytes
Routing protocol	DSR
Hello_packet_interval	1 s
Node mobility	0-20 m/s
Frequency	2.4 Ghz
Channel capacity	2 Mbps
Transmission range	250 m
Transmit power	1.32 W
Receiver power	0.96 W
Idle power	0.82 W
Mobility model	Random waypoint
Voltage	5 V
Initial node energy	9000 J
Route-selection window time at source	3 ms
Route-selection window time at destination	2 ms

5 Results

In order to evaluate the network performance of all the three protocols, it uses following metrics such as route survivability, throughput, transmission-power and number of path reconstructions.

5.1 Route Survivability

Here, we deployed 100 mobile nodes within the defined area and the node mobility varied from 0-20 m/s. The experimental setup has executed for 25 runs with the different mobility speeds in a given topology and 5 mobile nodes transmit the data at the rate of 5 packets/s. From the results, we conclude that the route survivability is more in the proposed model as compared to other models. After 4 hours and 75 mints, the proposed model has dropped piercingly due to energy exhaustion of the nodes. Whereas MTPR and MMBCR models drop at 4 hours and 12 mints, 4 hours and 25 mints respectively as shown in the figure 1. It is also noticed that the proposed model has transmitted the data packets without having new route-discovery packets due to robustness of the network connectivity.

5.2 Throughput

In this scenario, it considered 100 mobile nodes spreading within the defined area and the node mobility differed from 0-20 m/s. The simulation has executed for 25 runs

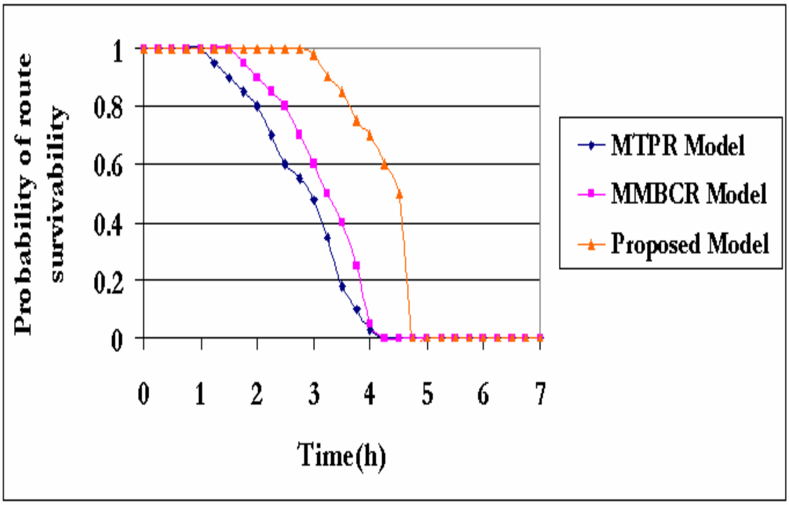


Fig. 1. Probability of route survivability

with the different mobility speeds in a given topology. Here, 5 nodes transmit the data at the rate of 5 packets/s. Figure 2 depicts the packet delivery ratio of all the three protocols under different motilities. The packet delivery ratio has decreased as the node mobility increases due to more number of the link breaks. The proposed model has delivered 95% of the data packets at 10 m/s due to relay-capacity of the nodes, whereas MMBCR and MTPR models have transmitted the data packets at 93% and 91.5% of the data packets respectively.

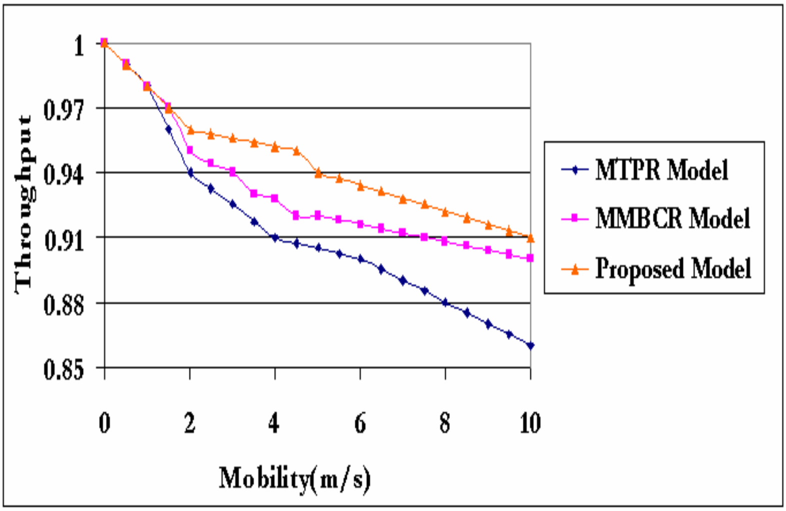


Fig. 2. Mobility versus throughput

5.3 Power Consumption

Here, it deployed 25 mobile nodes and the number of data packets sent between 0-80 packets/s and each node traveled constantly at 2 m/s. The experimental setup has executed for 20 times with the different arrival rate of the data packets. For investigation, the energy consumptions of all the nodes have their initial energy values, which are randomly selected. In the figure 3, it is clearly shown that MTPR model is designed to minimize the power consumption by selecting the routes, which are the most power efficient. In MTPR model, 15% to 20% of the paths are consumed with less power than the paths in MMBCR model. There is no much difference between the proposed model and MTPR model with respect to the power consumption from 1-3%.

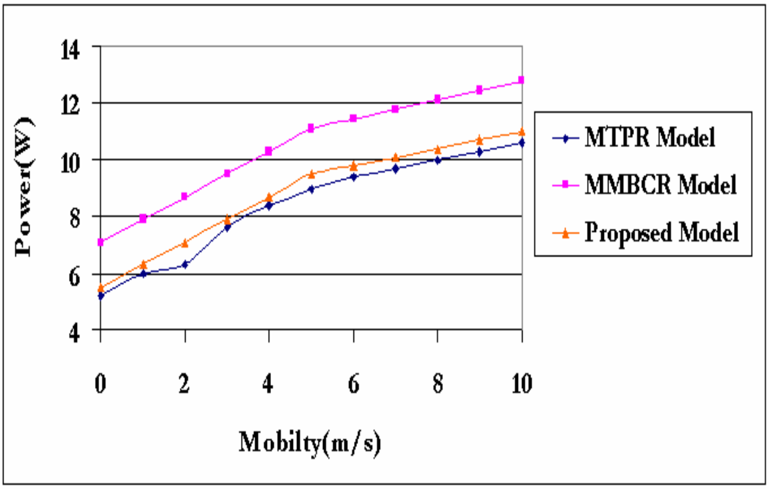


Fig. 3. Mobility against power

5.4 Number of Path Reconstructions

In the experimental setup, we deployed 50 mobile nodes within the defined area. The number of data packets sent between 5-20 packets/s and each node moved constantly with 0-25 m/s. As the number of nodes decreases, the number of path reconstructions has increased due to the node mobility. The path reconstruction is consistently low in the proposed model as compared to MMBCR and MTPR models as shown in the figure 4. The proposed model works well if the network has an adequate number of strong nodes in terms of relay-capacity. In fact, it has reduced the number of route reconstructions by 45% as compared to MMBCR model and 65% against to MTPR model at mobility 10 m/s. Since, the proposed model has routed the data packets through the route with strong relay-capacity nodes. The strong relay-capacity nodes are less vulnerable than the weak relay-capacity nodes.

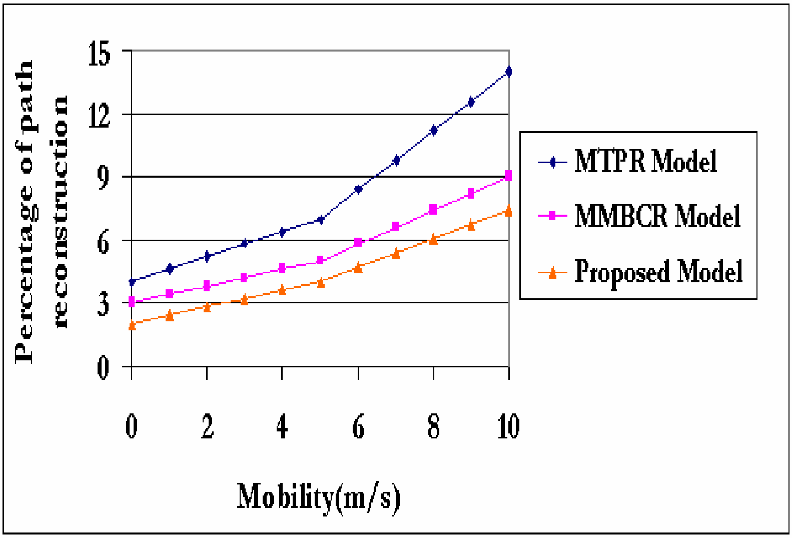


Fig. 4. Mobility versus percentage of path reconstructions

6 Conclusions

This paper presents a power-aware with survivable routing algorithm for the MANETs. It used the minimum transmission-power and high relay-capacity node to route the data packets from the source to destination. In this work, we used nearly 50 mobile nodes with area of 1,500 m X 1,500 m. Here, it is likely to have the unpredictable links, nodes, and variable mobility patterns. The proposed model forwarded the data packets based on more relay-capacity nodes and minimum transmission-power at physical layer. It is also helped us to switch new route before failure occurs and the proposed work makes QoS for end-users over the network. The simulation results are proved that the proposed model has reached at top position in terms of the route survivability, throughput, number of path reconstructions as compared to MMBCR and MTPR models. The drawback of this model is that it takes more number of hops to reach the destination because the data packets are routed with the more relay-capacity nodes.

References

1. de Moraes, R.M., Sadjadpour, H.R., Garcia-Luna-Aceves, J.J.: Mobility-Capacity-Delay Trade-Off In Wireless Ad Hoc Networks. Ad Hoc Networks 4(5), 607–620 (2006)
2. Chao, C.M., Sheu, J.P., Chou, I.C.: An Adaptive Quorum-Based Energy Conserving Protocol for IEEE 802.11 Ad hoc Networks. IEEE Transactions on Mobile Computing 5(5), 166–170 (2006)

3. Ma, C., Yang, Y.: A Prioritized Battery-Aware Routing Protocol for Wireless Ad Hoc Networks. In: Proceedings of The ACM International symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 45–52 (2005)
4. Sánchez, J.A., Ruiz, P.M.: Improving Delivery Ratio and Power Efficiency in Unicast Geographic Routing with a Realistic Physical Layer for Wireless Sensor Networks. In: Proceedings of The DSD, pp. 591–597 (2006)
5. Raghavendra, C.S., Singh, S.: PAMAS-Power Aware Multi-access Protocol with Signaling for Ad Hoc Networks. ACM Communications Review 28, 5–26 (1998)
6. Singh, S., Woo, M., Raghavendra, C.S.: Power-Aware with Routing in Mobile Ad Hoc Networks. In: Proceedings of The ACM MOBICOM, pp. 181–190 (1998)
7. Toh, C.-K.: Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks. IEEE Communications 39(6), 2–11 (2001)
8. Ahmad, S., Awan, I., Waqqas, A., Ahmad, B.: Performance Analysis of DSR and Extended DSR Protocols. In: Proceedings of The International Conference on Modeling and Simulation, pp. 191–196 (2008)
9. Balaswamy, C., Soundararajan, K.: An Efficient Route Discovery Mechanism for Mobile Ad Hoc Networks. International Journal of Computer Science and Network Security 8(10), 25–51 (2008)
10. Roux, N., Pegon, J.-S., Subbarao, M.W.: Cost Adaptive Mechanism to Provide Network Diversity for MANET Reactive Routing Protocols (2007), `http://w3.antd.nist.gov/pubs/roux_m2000.pdf`
11. Gobriel, S., Mosse, D., Melhem, R.: Mitigating the Flooding Waves Problem in Energy-Efficient Routing for MANETs. In: Proceedings of The IEEE Conference on Distributed Computing Systems, pp. 47–47 (2006)
12. Jaikaeo, C., Sridhara, V., Shen, C.C.: Energy Conserving Multicast for MANET with Swarm Intelligence. In: Proceeding of The International Symposium on a World of Wireless, Mobile and Multimedia networks, pp. 7–11 (2006)
13. Hanashi, A.M., Siddique, A., Awan, I., Michael, E.: Woodward: Dynamic Probabilistic Flooding Performance Evaluation of On-Demand Routing Protocols in MANETs. In: Proceedings of The CISIS, pp. 200–204 (2008)
14. Kim, D., Garcia-Luna-Aceves, J.J., Obraczka, K., Cano, J.-C., Manzoni, P.: Routing Mechanisms for Mobile Ad Hoc Networks Based on the Energy Drain Rate. IEEE Transactions on Mobile Computing 2(2), 161–173 (2003)

Testing and Evaluating of Predictive Data Push Technology Framework for Mobile Devices

Ondrej Krejcar

VSB Technical University of Ostrava, Center for Applied Cybernetics, Department of measurement and control, 17. Listopadu 15, 70833 Ostrava Poruba, Czech Republic
Ondrej.Krejcar@remoteworld.net

Abstract. Current mobile devices with a wireless connectivity are used increasingly as clients of online application servers. Users can use also a increasingly portfolio of hardware and software capabilities to support such connectivity. One problem however still exists in a slow wireless connection in such defined type of using. To allow a work with same comfort as on desktop devices, the prebuffering techniques can be used to reduce a problem of slow downlink. Managing of prebuffering of large data artifacts is made on intelligent decision core based on a current user position computation and future predicted movement computation. All large data artifacts are stored in database along with its position information. The accessing of prebuffered data artifacts on mobile device improve the download speed and reduce a response time needed to view large multimedia data. Testing and evaluating of developed PDPT Framework solution is also presented and discussed along with major tests results.

Keywords: Prebuffering; Response Time; Download Speed; Mobile Device.

1 Introduction

People use their mobile devices mainly to communicate with outer world. They are able to not only make calls or send a text messages, but they can and want to use very sophisticated software running on their mobile devices. Such software can play a role of remote client of some kind of server application. For such kind of connection with a remote world they need to use some type of implemented communication standard like GPRS or WiFi. The connection speed of these standards varies from hundreds of kilobits to several megabits per second. In case of online information systems (e.g. facility management, zoological or botanical gardens, libraries or museums), the WiFi infrastructure network is often used to connect mobile device clients to a server. Unfortunately, the theoretical maximum connection speed is only achievable on laptops where high-quality components are used. The limited connection speed presents a problem for online systems using large artifacts data files. It is not possible to preload these artifacts before the mobile device is used in remote access state.

Every information system with remote mobile clients needs to specify a response time for whole system – mainly for user side. Application cannot wait with response on user request so long, because users are not able to waste their time.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 276–283, 2010.

Nielsen [4] specified this time delay to 10 seconds [5]. During this time the user was focused on the application and was willing to wait for an answer. In newer literature a shorter time is resulted. We used 10 seconds to calculate the maximum possible data size of a file transferred from server to client. We executed a number of tests of real downlink for WiFi network with result of 160 kB/s for actual PDA devices [2], [3], [15]. The client application can download during the 10 second period from 2 to 3 artifacts (real artifacts of building plan were used with an average size of 470 kB). The goal of project is to prebuffer data artifacts to user device before need of them based on localization of user. Information about location is used to determine both an actual and future position of a user [6], [11], [14]. A number of experiments with the information system have been performed and their results suggest that determination of the location should be focused on. The following sections describe also the conceptual and technical details of Predictive Data Push Technology Framework (PDPT).

2 The PDPT Framework

A combination of a predicted user position with prebuffering of data associated with physical locations bears many advantages in increased throughput of mobile devices.

The key advantage of our PDPT solution in compare to [12], [13] is that the location processing, track prediction and cache content management components are situated at server side (Fig. 1). This fact allows for managing many important parameters (e.g. AP info changes, position determination mechanism tuning, artifacts selection evaluation tuning, etc.) online at a PDPT Server.

Position Oriented Database (Wireless Location Architecture - WLA)

If the mobile device knows the position of the stationary device (transmitter), it also knows that its own position is within a range of this location provider. In PDPT framework only the triangulation technique is used due to the sufficient granularity of user position information. Information about the user position are stored in *Position* table. *Locator* table contains info about wireless AP with signal strength which are needed to determine user position. *WiFi_AP, BT_AP* and *GSM_AP* tables contain all necessary info about used wireless base stations. *WLA_data* table contains data artifact along with their position, priority and other metadata.

PDPT Client - Mobile Database Server

The large data artifacts from PDPT Server (WLA_data table) are needed to be presented for user on mobile device. In case of classical online system the data artifacts are downloaded on demand. In case of PDPT solution, the artifacts are preloaded to mobile device cache before user requests (based on user location). Our mobile cache (SQL Server 2005 Mobile Edition was selected for it [7],[8], [16]) contain only one data table *Buffer.* An advantage of using a Microsoft solution is that it takes only small data amount for installation (2,5 MB).

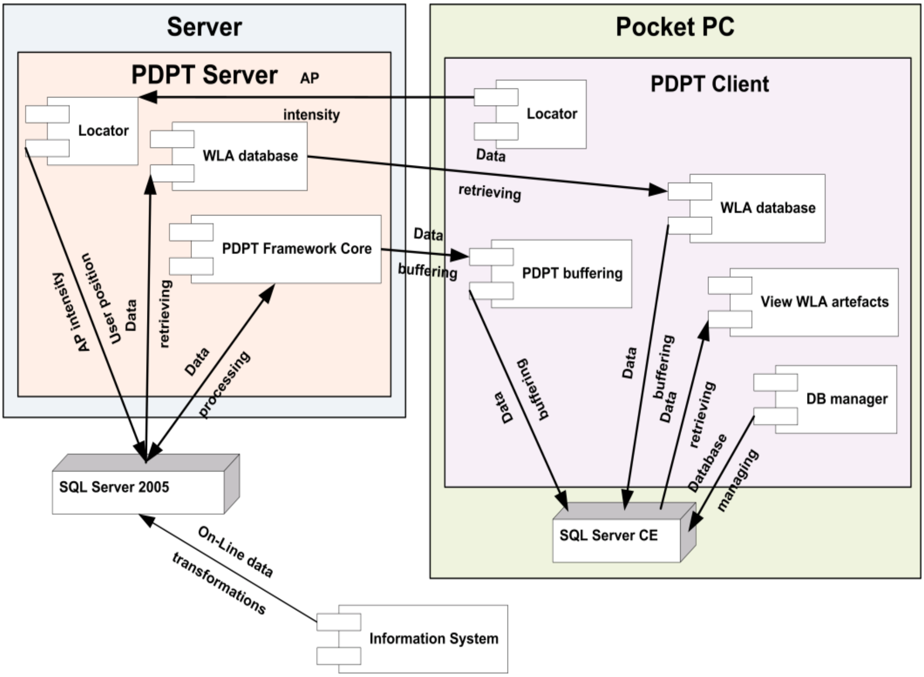


Fig. 1. PDPT architecture – UML design

Data Artifact Creating

Data artifact represents an object in WLA SQL server database with some kind of multimedia file types. Every artifact must have associated with position coordinates in 3D environment (S-JTSK format is used [18]). To manage and work with locations of artifacts, firstly the building floor map is needed to obtain. In most cases the scanned version is adequate. The obtained map needs to be converted to Tagged Image File Format (TIFF). Location coordinates (location, scale, and rotation of a map) for such file must be stored in TFW separate file. Artifacts with position coordinates are stored in WLA database by “WLA Database Artifact Manager” [15].

Data Artifact Managing

The WLA server database manages the artifacts in the context of their location in building environment. The PDPT Core selecting the data to be copied from PDPT server to PDA client by context information (position info).

Created software application called “WLA Data Artifacts Manager” is used to manage the artifacts in WLA database. User can set the priority, location, and other metadata of the artifact. The Manager allows creating a new artifact from multimedia file source, and work with existing artifacts [1], [2], [3], [15].

PDPT Core - Area Definition for Selecting Artifacts to Buffering

The PDPT buffering and predictive PDPT buffering principle consists of several following steps. Firstly the client must activate the PDPT on PDPT Client. This client creates a list of artifacts (PDA buffer image), which are contained in his mobile SQL Server CE database (from previous usage). Server create own list of artifacts (imaginary image of PDA buffer) based on area definition for actual user position and compare it with real PDA buffer image. The area is defined as an object where the user position is in the center of object. The cuboid form (area with a size of 10 x 10 x 3 (high) meters) is used in present time for initial PDPT buffering. The PDPT Core continues in next step with comparing of both images. In case of some difference, the rest artifacts are prebuffered to PDA buffer.

When all artifacts for current user position are in PDA buffer, there is no difference between images. In such case the PDPT Core is going to make a predicted user position. On base of this new predicted user position it makes a new predictive enlarged imaginary image of PDA buffer. The size of this new cuboid is predefined area of size 20 x 20 x 6 meters. The new cuboid has a center in direction of predicted user moving and includes a cuboid area for current position of user. The PDPT Core compares the both new images (imaginary and real PDA buffer) and it will continue with buffering of rest artifacts until they are same.

Creation of an algorithm for dynamic area definition is better in some types of real usage to adapt a system to user needs more flexible in real time.

Accessing the Artifacts in PDPT Client Application

The PDPT Client application realizes thick client and PDPT and Locator modules extension. The data artifact can be viewed from MS SQL CE database to user (immediately with only maximum delay of one second). The PDPT tab presents a way to tune the settings of PDPT Framework. This tab also shows the log info about the prebuffering process and the time of measurement of last artifact loading ("part time") and also a full time of whole prebuffering process in millisecond resolution.

3 Testing of PDPT Framework

Developed PDPT Framework is mentioned rather than as a whole new way how to work with artifacts, as an addition to classical user access to client application. It cannot be awaited to prebuffer all needed data artifacts at every time. While the proportion of both accesses cannot be measured, both accesses will be tested separately. The whole process of managing with PDPT framework is separated to two ways. In case of ideal PDPT Buffering:

1. PDPT prebuffering of artifacts
2. Selecting of artifacts from combo box menu of client application
3. Viewing of artifacts

In classical case without a use of PDPT Buffering:

1. Selecting of artifacts from combo box menu of client application
2. Downloading of artifacts to PDA Buffer
3. Viewing of artifacts

These several parts are needed to subject of tests, which show a proportion of delay in several mentioned parts. Testing methodology is possible to divide it to two directions: static and dynamic. Static testing mention a way of use a PDA device in one defined position, without any additional moving. While a dynamic testing is performed by a moving of PDA by user. Due to a very bad penetration of WiFi APs in our testing environment in our Campus of Technical University of Ostrava, only a static testing was executed.

3.1 Methodology of Static Testing

For static testing a proportion of quality of PDPT Buffering versus a downloading of artifacts without PDPT were measured. A three function parts was used for testing:

1. PDPT Buffering – self localization
2. Downloading of artifacts from server to PDA Buffer by user
3. Viewing of artifacts from PDA Buffer to display

The last part serves for a testing of viewing speed of artifacts, when the user needs to view them. This delay is a very important value for quality of user work with an application.

Successful of PDPT Buffering is possible to evaluate by a “quality” of prebuffered artifacts in PDA Buffer after PDPT Buffering process. If all artifacts for actual PDA position are after PDPT Buffering process prebuffered in PDA Buffer, we can evaluate such prebuffering as 100% of successful. If some of artifacts are missing, proportion of success is decreasing. The quality of prebuffering is measured as a ratio between real prebuffered artifacts versus awaited sum of artifacts.

As a testing position an expertly defined positions are selected based on number of APs (minimum signal of one AP is needed). For every such test position a sum of artifacts to be presented after prebuffering process is defined. As a self localization a process of determination of actual position of PDA is mentioned.

Quality of PDPT is a percentage value of a structure of artifacts in PDA Buffer, which rate a ration if awaited artifacts were prebuffered at the earliest in compare of all prebuffered artifacts:

$$\text{Quality of PDPT} = \frac{\Sigma\ \text{ideal sequence of artifacts}}{\Sigma\ \text{real sequence of artifacts}} \times 100 \tag{1}$$

Content of PDPT is a percentage value of really prebuffered awaited artifacts versus all prebuffered artifacts:

$$\text{Content of PDPT} = \frac{\text{number of prebuffered awaited artifacts}}{\text{number of all prebuffered artifacts}} \times 100 \tag{2}$$

Fruitfulness of PDPT is a ratio value in percentage which rate a number of prebuffered awaited artifact were prebuffered in compare to sum of all awaited artifacts:

$$\text{Fruitfulness of PDPT} = \frac{\text{number of prebuffered awaited artifacts}}{\text{sum of all awaited prartifacts}} \times 100 \qquad (3)$$

The testing process will consist of several consequence processes as follows: (1) Actual position determination, (2) PDPT Buffering activation, (3) summarization of prebuffered artifacts, time of prebuffering and a size of PDA Buffer database (after finish of prebuffering process), (4) renew of PDA Buffer database (to prevent some old artifacts in Buffer). The testing process will then repeated from (1).

For testing of classical access without a PDPT Framework was a localization process substituted by a selecting of artifacts relevant to selected position (taken from previus testing process). The testing process in this case was as follows: (1) Selecting of artifacts relevant for position, (2) Downloading of all artifacts to PDA Buffer, (3) summarization of downloading time.

3.2 Test Results of Static Testing – PDPT Buffering - Self Localization

Tables (Table 1), (Table 2) show the summary of PDPT Prebuffering test results. Two PDA devices were used. A colored highlight is used for better adjustment of results, where lighter hue present a worse values while darker mean better ones.

Table 1. PDPT Framework – testing results for HTC Universal device

Position	**A651**			**A2**			**NK2**		
time [s]	54,9	62,8	64,7	174,8	152,5	121,9	241,8	246,8	247,3
PDA DB size [kB]	2,06	2,31	2,31	6,81	5,81	4,56	6,75	8,68	8,5
speed [kB/s]	37,5	36,8	35,7	38,95	38,1	37,41	27,92	35,18	34,37
LOCATOR [ΣAP]	3	1	1	2	1	2	1	2	1
awaited artifacts [Σ]	6	6	6	12	12	12	13	13	13
prebuffered artifacts [Σ]	5	6	6	12	11	10	9	11	12
Quality of PDPT [%]	78,9	60	60	54,93	74,16	62,5	56,96	53,23	59,54
Content of PDPT [%]	62,5	66,7	66,7	57,14	64,71	66,67	50	50	60
Fruitfulness of PDPT [%]	**83,3**	**100**	**100**	**100**	**91,67**	**83,33**	**69,23**	**84,62**	**92,31**

In case of first tested device (Table 1), the results provide a very high level of fruitfulness of PDPT Buffering, because al artifacts were prebuffered in most cases. In worst case a 70% was achieved, which is still very good (only one WiFi AP was visible for position determination).

Content of PDA Bufferu after PDPT Buffering is however on lover level, where only 50% was achieved in case of NK2 room testing position.

Quality of PDPT Buffering varies from 53% to 79%, which represents a very successful sequence of prebuffered artifacts.

Fruitfulness of PDPT Buffering is in second case (Table 2) quite lower in compare to previous, but still very good. The worst content of PDA Buffer is same (50%) even in case when two WiFi APs were visible.

Table 2. PDPT Framework – testing results for HTC Roadster device

Position	**A651**			**A2**			**NK2**		
time [s]	53,6	38	49,6	114,4	154,2	174,7	265,9	258,7	260,5
PDA DB size [kB]	2,31	1,62	2,06	4,56	6,06	6,81	8,68	8,5	8,68
speed [kB/s]	43,1	42,7	41,5	39,86	39,31	38,98	32,64	32,86	33,32
LOCATOR [ΣAP]	1	3	2	3	2	1	2	1	2
awaited artifacts [Σ]	6	6	6	12	12	12	13	13	13
prebuffered artifacts [Σ]	6	5	5	12	12	10	11	11	11
Quality of PDPT [%]	60	88,2	83,3	78,79	65	77,46	53,23	56,9	53,23
Content of PDPT [%]	66,7	83,3	62,5	71,43	66,67	50	50	57,89	50
Fruitfulness of PDPT [%]	**100**	**83,3**	**83,3**	**83,33**	**100**	**83,33**	**84,62**	**84,62**	**84,62**

We executed a number of tests for every artifact when viewing them, where we measured a time delay from request to response (artifacts was displayed). An average artifact can be showed in 505 [ms], which present an excellent application response for native image format processing in client application.

3.3 PDPT Testing Evaluation

From PDA user point of view (PDPT Client), the most important parameter for comparing of contribution of PDPT solution is speed of application - response on user's requests. Based on this definition, we can state that a prebuffering though a one artifact which is awaited in PDA Buffer before user request is a huge contribution for application response. PDPT Framework solution is therefore very useful and suitable for stocking in any kind of information systems which manage a position based data artifacts.

4 Conclusions

We develop a PDPT Framework for data artifact prebuffering. Described solution is appropriate to apply in such online information systems, which are targeted to serve with a remote client of such systems. The tests were also evaluated with a suggestion to apply developed PDPT Framework solution in systems which manage with artifacts of size no smaller than 500 kB. In case when smaller artifacts are downloaded a cost of PDPT Framework solution is don't bring any significant improvements. PDPT framework is currently used in another projects of biotelemetrical system to make a patient's life safer and more comfort [9], [10], [17].

Acknowledgement. This work was supported by the Ministry of Education of the Czech Republic under Project 1M0567.

References

1. Krejcar, O.: Problem Solving of Low Data Throughput on Mobile Devices by Artefacts Prebuffering. EURASIP Journal on Wireless Communications and Networking, Article ID 802523, p. 8. (2009)
2. Krejcar, O., Janckulik, D., Motalova, L.: Accessing of Large Multimedia Content on Mobile Devices by Partial Prebuffering Techniques. In: Second Joint IFIP Wireless and Mobile Networking Conference, WMNC 2009. IFIP AICT, vol. 308, pp. 80–91 (2009)
3. Krejcar, O.: PDPT Framework - Building Information System with Wireless Connected Mobile Devices. In: 3rd International Conference on Informatics in Control, Automation and Robotics, ICINCO 2006, Setubal, Portugal, August 01-05, pp. 162–167 (2006)
4. Nielsen, J.: Usability Engineering. Morgan Kaufmann, San Francisco (1994)
5. Haklay, M., Zafiri, A.: Usability engineering for GIS: learning from a screenshot. The Cartographic Journal 45(2), 87–97 (2008)
6. Brida, P., Duha, J., Krasnovsky, M.: On the accuracy of weighted proximity based localization in wireless sensor networks. In: Personal Wireless Communications. IFIP, vol. 245, pp. 423–432 (2007)
7. Arikan, E., Jenq, J.: Microsoft SQL Server interface for mobile devices. In: Proceedings of the 4th International Conference on Cybernetics and Information Technologies, Systems and Applications/5th Int. Conf. on Computing, Communications and Control Technologies, Orlando, FL, USA, July 12-15 (2007)
8. Jewett, M., Lasker, S., Swigart, S.: SQL server everywhere: Just another database? Developer focused from start to finish. DR DOBBS Journal 31(12) (2006)
9. Krejcar, O., Janckulik, D., Motalova, L.: Complex Biomedical System with Biotelemetric Monitoring of Life Functions. In: Proceedings of the IEEE Eurocon 2009, St. Petersburg, Russia, May 18-23, pp. 138–141 (2009)
10. Cerny, M., Penhaker, M.: Biotelemetry. In: 14th Nordic-Baltic Conference an Biomedical Engineering and Medical Physics, IFMBE Proceedings, Riga, Latvia, June 16-20, vol. 20, pp. 405–408 (2008)
11. Liou, C., Cheng, W.: Manifold construction by local neighborhood preservation. In: Ishikawa, M., Doya, K., Miyamoto, H., Yamakawa, T. (eds.) ICONIP 2007, LNCS, vol. 4985, pp. 683–692. Springer, Heidelberg (2008)
12. Brasche, G.P., Fesl, R., Manousek, W., Salmre, I.W.: Location-based caching for mobile devices. In: United States Patent, Microsoft Corporation, Redmond, WA, US (2007)
13. Squibbs, R. F., Cache management in a mobile device. United States Patent, Hewlett-Packard Development Company, L.P., 20040030832 (2004)
14. Brida, P., Majer, N., Duha, J., Cepel, P.: A Novel AoA Positioning Solution for Wireless Ad Hoc Networks Based on Six-Port Technology. In: Wireless and Mobile Networking. IFIP, vol. 308, pp. 208–219 (2009)
15. Krejcar, O.: Localization by Wireless Technologies for Managing of Large Scale Data Artifacts on Mobile Devices. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS, vol. 5796, pp. 697–708. Springer, Heidelberg (2009)
16. Dondio, P., Longo, L., Barrett, S.: A translation mechanism for recommendations. In: IFIPTM 2008/Joint iTrust and PST Conference on Privacy, Trust Management and Security Trondheim, Nórway, June 18-20. IFIP, vol. 268, pp. 87–102 (2008)
17. Penhaker, M., Cerny, M., Martinak, L., Spisak, J., Valkova, A.: HomeCare - Smart embedded biotelemetry system. In: World Congress on Medical Physics and Biomedical Engineering, Seoul, South Korea, August 27-September 01, vol. 14, PTS 1-6, pp. 711–714 (2006)
18. Horak, J., Unucka, J., Stromsky, J., Marsik, V., Orlik, A.: TRANSCAT DSS architecture and modelling services. Journal: Control and Cybernetics 35, 47–71 (2006)

Service Space Portability Validation Modeling the Vehicular Context

Mihály Börzsei

Nokia Research Center,
P.O. Box 1000, 33721
Tampere, Finland
mihaly.borzsei@nokia.com

Abstract. Devices in our proximity getting sophisticated and they provide more and more services to their user and surroundings. Interconnecting solutions between devices and services being developed constantly to address the interoperability in a multi vendor ecosystem. This paper describes the set of service interconnect approaches and present study to validate of their portability. A prototype implementation of a music player service on an internet tablet controlled by an input service represented by a driving wheel and a mobile phone was created to evaluate the different architectural design portability.

Keywords: Service Oriented Architecture, Transport Independency, NoTA, Network on Terminal Architecture, M3, Keyboard service, Music service, UPnP, Vehicular networks.

1 Introduction

Consumer electronic interoperability requires novel and visionary approach of system design. The concept of smart space emerged to describe systems addressing the interactions in such environments. A smart space is a multi-user, multi-device, dynamic interaction environment that enhances a physical space by virtual services [1], These services form a collaborative software systems. Industry standardization allows interoperability by device certification. It is time taking process involve conformity check against the DLNA/UPnP standard set of services and devices [2]. An alternative solution to industrial standardization addresses the challenges arise from the dynamic nature of smart spaces [3]. This complementing approach to standardization is the M^3 concept [4] proposal over the Network on Terminal Architecture [5] (NoTA) as a smart space application development platform. NoTA service interconnect solution in itself has some capability to orchestrate a collaborative software space. As a result of improving onboard computers in cars the need aroused to apply these approaches to vehicular context.

Prototyping simple audio rendering control service assisted with an input service form the contribution which allowed evaluating the portability of the different smart space approaches.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 284–291, 2010.

2 Related Work

A first step towards smart space service interoperability is the description of the collaboration software approaches [6]. This work provides a general overview by explaining the collaborating-software design space without investigating a solution which takes account the special need of the vehicular networks. One of the challenges is that automotive. networks may randomly overlap each other; meanwhile some of their services should be kept separate and others designed to take advantage of the possible collaboration. Another challenge is that the product cycle of the car itself is much longer than the product cycles of the consumer electronic devices, which nevertheless are expected to be compatible with the embedded car systems. There is also certain constrains on the embedded car memory and networking capabilities when compared to the existing consumer electronic devices.

A simple audio rendering control service assisted with an input service can be implemented using various interconnect techniques and approaches. The major service collaboration designs were studied and explained in general level, before their application to the vehicular environment is shown.

2.1 Service Interconnect in Smart Spaces

Interconnected services in smart spaces depend on service discovery mechanisms. Smart spaces themselves can be discoverable for other smart spaces to join either enabled by proximity protocols by or out of band techniques such as a short range radio communication outside the wireless or wired transport layer of the space itself. The network transport layer and the underlying network topology are transparent for a smart space.

Collaborating services can be implemented as blackboard design pattern, multi agent system or adding various levels of these two properties together adapted to the performance requirements and the task that the system has to perform. This study focus on the portability aspect of an audio rendering control service of three distinct smart space interconnect design approach: device and protocol based interoperability (e.g. UPnP[7]), service based interoperability (e.g. NoTA [8]), or ontology based interoperability (e.g. M3).

2.1.1 UPnP Audio/Video (AV)

The UPnP protocol relies on standardization process which may take relatively long time, however certain areas the result are already available and accepted widely by the industrial players. The UPnP Forum has already specified protocols which enable electronics devices, to discover and use each other's services. The UPnP AV architecture has as a goal to solve the selection and controlled discovery of media content. It introduces service elements as shown in Figure 1.

Media Server is a device hosting and offering content for browse/download, while also accepting content uploads. Media Renderer, a device that can render (e.g. "play") content offered by a Media Server. Control Point is an entity that coordinates the communication between the Media Server and Media Renderer. UPnP uses Simple Service Discovery Protocol. The foundation for UPnP networking is established over IP addressing. From Network Simulation [9] UPnP protocol performance over Ethernet in Desktop environment for smart space scale well in terms of response times between services. The UPnP advertisements generated wireless radio wake ups

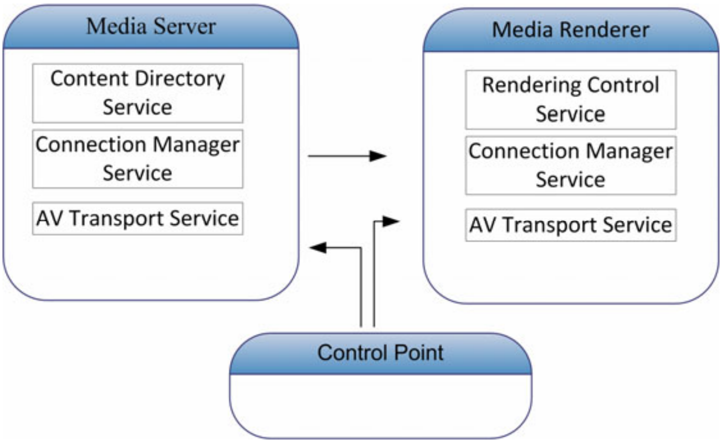

Media Server
Content Directory Service
Connection Manager Service
AV Transport Service
Media Renderer
Rendering Control Service
Connection Manager Service
AV Transport Service
Control Point

Fig. 1. UPnP A/V architecture

realized in form of penalty on power consumption. [10] UPnP low power architecture purpose to address this issue. [11] The remaining service interoperability issues such as digital media formats and control commands are resolved by guidelines and specifications of the Digital Living Network Alliance (DLNA). Unfortunately even DLNA certification may not means a fluent service interoperability in the home or mobile domain.

2.1.2 Network on Terminal Architecture

The NoTA Release 3 is modular services interconnect system architecture for mobile and embedded devices. NoTA device consists of Service Nodes (SN) and Application Nodes (AN) that communicate through logical Interconnect (IN). NoTA node

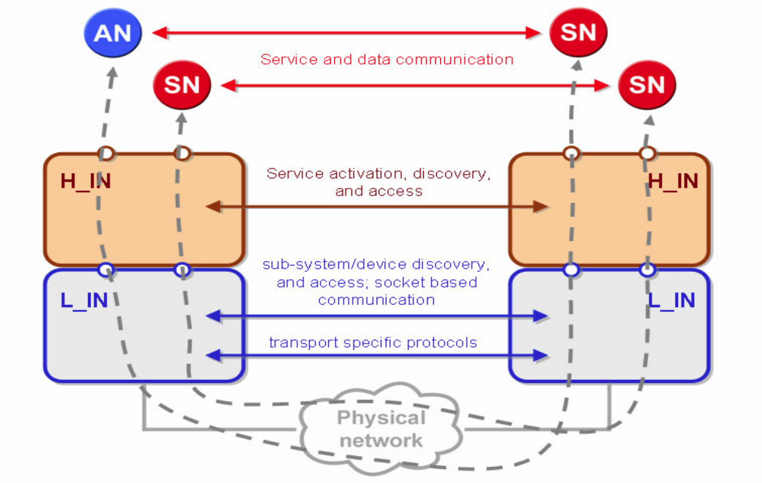

AN
SN
SN
SN
Service and data communication
H_IN
H_IN
Service activation, discovery, and access
L_IN
L_IN
sub-system/device discovery, and access; socket based communication
transport specific protocols
Physical network

Fig. 2. NoTA service interconnect architecture

interconnect address system is independent from IP protocol enabling those embedded devices to join the network which are not capable to contain the IP addressing protocol.

A NoTA layered architecture containing high level and low level interconnect as a result service implementations interconnected over different transports, including TCP/IP, Bluetooth or a hardware specific protocol for intra-device communications. The service level of NoTA is abstracted over the transport, so that the same service implementation can be used over several backend upon performance permitting. Nota Interconnect Architecture is shown on Figure 2.

NoTA use a dedicated Resource Manager Node where SNs register themselves to advertise their services. ANs and SNs query the resource manager for available services.

2.1.3 M^3 Concept

M^3 [12] makes it possible to mash-up and integrate information between all applications and domains spanning from embedded domains to the Web. M^3 is independent of transport mechanisms. M^3 is designed to provide information interoperability by means of ontology sharing as opposed to a standardization process. Nodes which communicate via M^3 using the same ontology are compatible and any later entity using the same ontology are compatible as well. Interoperability layers are shown on Figure 3.

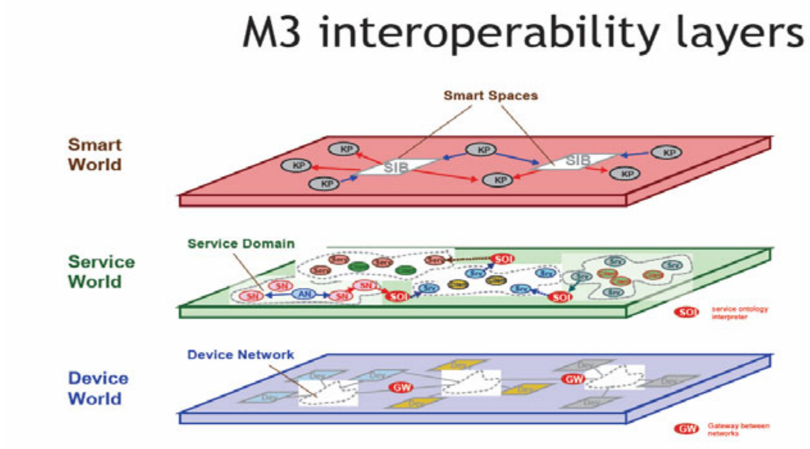


Fig. 3. M^3 Interoperability layers

The architecture of M^3 is that underlying service implementations can publish information to M^3 space and if needed operate completely independently. If the service implementations only publish information to M^3, they naturally are not able to benefit from future semantic level services. If the service implementations subscribe and react to information on the M^3, they can be benefit from future developments. For service discovery and smart space access it uses the Smart Space Access Control Protocol. Ontology could describe the Audio control between services. The control logic implemented by the reasoning engine called the Knowledge processor, meanwhile the information stored in the Semantic Information Broker according to the ontology model.

2.2 Comparison

Comparing these three given approach to smart spaces interconnection design, many common features are noticeable and a few key differences also exists. These observation were taken account when selected the prototype service interconnect scenario for portability validation. Features of service interconnect approaches are elaborated on Table 1.

Table 1. Brief comparisons of the smart space interconnect approaches

	UPnP	NoTA	M^3 concept
Underlying protocol	Http	BSD sockets	HTTP or NoTA
Addressing	Dynamic (requires discovery)	Interconnect Address (RM - SID) and depends on underlying transports	SSAP
Power Saving	UPnP low power –since 2007	By design principles	Depends on the underlying transport, priorities allows suspension
Input Service	Not standardized, Control point is a kind of input service	Custom keyboard service	Activity instance
Audio player service	Media Renderer	Custom audio player	Subscriber to an activity
User authentication	Only at radio access point level	With custom made authentication service	Via Local policies
Dependency on external components	TCP/IP, UDP, HTTP, XML, and SOAP	Bluez (SDP, HCI, L2CAP, RFCOMM), or DHCP, TCP/IP	Same as NoTA and DBus, Expat, Uuid, Avahi, Python

3 Portability Validation

Smart space portability validation using approaches compared earlier in general level assuming a desktop environment. Prototype evaluation required to implement six scenarios over two use cases and look into feasibility, performance and developer friendliness during remapping the services in the similar service space over different device platforms. This chapter focuses on to comparing the different smart spaces designs and their portability to vehicular environment.

3.1 Prototype Implementation

All of the approaches supports, simple communication over wireless radio connection between devices shown on Figure 4.. Upon UPnP network is established using the wireless radio connection available UPnP renderer and control point were available by default on the network and interconnected without any further setup. The service discovery took longer time compare with the others and initial setup of the UPnP media server service also requires a few steps as configuration from the end user. Implementation over Nota network requires definition of the input service and audio player service and a translator service between those. The end user involvement is limited to the selection of the services or starting application which utilities the available services. Connecting over M^3 requires the ontology definition from the system designer to establish service interoperability and also require deciding the transport for M^3. For this study the implemented is decided to use the transport over NoTA and TCP/IP.

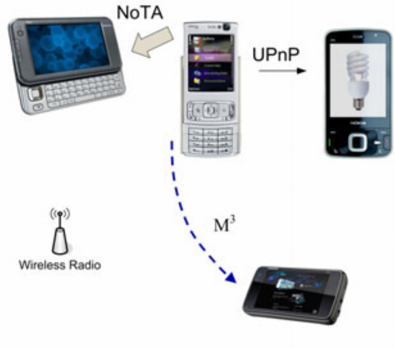


Fig. 4. Audio rendering services in an internet tablet and a Symbian OS Smartphone controlled by input services on a mobile phone established using the three distinct service interconnect approach

For simulating a vehicular environment a wheel controller attached to a laptop used for the setup instead of the Symbian OS mobile phone.

3.2 Lessons Learned

It required remapping of the service interconnection of the initial setup to enable the vehicular setup. Using UPnP protocol for connecting the services over the network initially though as the easiest service interconnect possibility as UPnP renderer still exist, however control point have to be hand adjusted. UPnP does not support the concept of a standard input device as its core and by implementing one as a device extension the portability of smart space is compromised. The loss of power as described earlier is more important in this setup - where a vehicle may be standstill and as a result battery operated - than in a home environment which would raise question for UPnP approach to vehicular smart spaces compare to other domains. Porting of the Nota network setup required a definition of another input service and a translator service along with it. The difference came from the fact that initial Symbian OS setup supported Graphical UI type of input service and understood the notion of key event with various statuses such as pressed or released. The translator service can understand the longpress from the input service and translate to audio renderer as fast forward within a song. The wheel keyboard attached to linux computer may treat input source as input pipe without the notion of input item states. The two way of input result two distinct input service. One named as the keyboard service which send the last event as pressed key and one named as the button service which send the latest event and the state related to it. These two services can share a common translator service, which assign a timeout for any non released keys however it results in a very complex design for control transitions. The M^3 ontology handle the best the system porting and at knowledge processor level elegantly resolves all the issues during the porting. However this setup involved dependencies on external libraries and overall developer friendliness of the M^3 system itself is a bit less than working with NoTA only setup due to its complexity to set up the working prototypes.

4 Discussions and Future Work

Portability validation experiment established requirements towards smart interconnect architectures for wider platforms deployment over vehicular Operating System [13].

The implementation of UPnP low power profile does eliminate some of the technology obstacles for vehicular context; however portability to vehicular networks would require standardization and design a vehicular specific extension.

End-user using a system built upon a vehicular network demands full control and coordination between all the available services. M^3 concept follows the blackboard design pattern which is well suited orchestrating the software and service collaboration. The conclusion based on these experiments is the recommendation of open source M^3 smart space in automotive context. The author believes that the development setup and portability of M^3 due to many of external component dependency is improving rapidly over the time.

Acknowledgments. Software implementation has been written as part of our ongoing research in the Network on Terminal Architecture Team and in the Ubiquitous Architectures in the Nokia Research Center. The authors want to acknowledge the contributions from Dirk-Jan Binnema, Arto Palin, Vesa Luukkala, Rod Walsh, Stirbu Vlad and Joonas Salo.

References

1. van Gurp, J., Tarkoma, S., Prehofer, C., di Flora, C.: A Web based platform for smart spaces, Demo (2008)
2. DLNA compliant devices, http://product.dlna.org/eng/browse_cat.aspx
3. Liuha, P.: Application development platforms for emerging smart environment. In: 2nd International Mobilware Conference, Berlin (2009)
4. Lappeteläinen, A., Tuupola, J.-M., Palin, A., Eriksson, T.: Networked systems, services and information The ultimate digital convergence. In: 1st International NoTA Conference, Helsinki (2008)
5. Network on Terminal Architecture, http://en.wikipedia.org/wiki/Network_on_Terminal_Architecture

6. Corkill, D.D.: Collaborating Software Blackboard and MultiAgent Systems & the Future, Department of Computer Science, University of Massachusetts. In: Proceedings of the International Lisp Conference (1991)
7. UPnP Forum, UPnP AV Architecture:0.83 (June 2002)
8. Eriksson, T.: NoTA 2nd International Conference. NoTA Tutorial, San Jose (2009)
9. Tamai, M., Shibata, N., Yasumoto, K., Ito, M.: Network Simulation Architecture for Smartspace, Citeseer (2006)
10. Liong, Y.-L., Ye, Y.: Effect of UPnP advertisements on User Experience and Power Consumption
11. UPnP Low Power, `http://www.upnp.org/specs/lp.asp`
12. Soininen, J.-P., Lappeteläinen, A.: M3 Smart Environment Infrastructure. In: NoTA Conference (2009)
13. Microsoft Corporation Automotive, `http://www.microsoft.com/auto/ma.mspx`

Experiences from Developing a Context Management System Applied to Mobility

Baptiste Gaultier and Jean-Marie Bonnin

Institut TELECOM, TELECOM Bretagne, France
Université européenne de Bretagne
{baptiste.gaultier,jm.bonnin}@telecom-bretagne.eu

Abstract. Recent advances in electronic and automotive industries as well as in wireless telecommunication technologies have drawn a new picture where each vehicle became "fully networked". In order to provide IP connectivity to on-board devices, the IETF has proposed the NEMO (NEtwork MObility) protocol. In this approach, a new device, the Mobile Router (MR), will take place in vehicles. It has to manage mobility and takes advantage of the surrounding wireless technology diversity to offer connectivity and reachability for all nodes in the mobile network as it moves.

To be efficient, the MR has to take into account various contextual parameters regarding the management of wireless network interfaces and the routing of the flows. Exchanging such a contextual information can be achieved easily through basic polling and broadcasting mechanisms. However, systems involving more than one MR and systems having hot sensor plugging capabilities will require more advanced techniques for exchanging context information. In [2], we proposed to use a CMS (Context Management System) in order to process and exchange contextual information in a vehicular network. This paper describes our experience with the design and the implementation of a new CMS applied to mobility.

Keywords: Middleware, Context-awareness, Mobile network, NEMO.

1 Introduction

Wireless communications for ITS (Intelligent Transport System) is an enabling technology to improve driving safety, reduce traffic congestion, and support information services to vehicles. In fact, continuous connectivity can make their transportation time more pleasant (browsing the web and watching online videos) and efficient (consulting emails on the way to work or read online newspaper during a train travel).

A major step in this evolution was the introduction of a new embarked component responsible for managing all communications of the vehicle: the Mobile Router (MR). The NEMO (NEtwork MObility) Basic Support protocol designed by the IETF manages mobility and offers continuous and seamless IPv6 connectivity to on-board Mobile Network Nodes (MNN in fig. 1). The CALM architecture[1]

[1] Continuous Air interface for Long and Medium range (ISO TC204 WG16).

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 292–298, 2010.

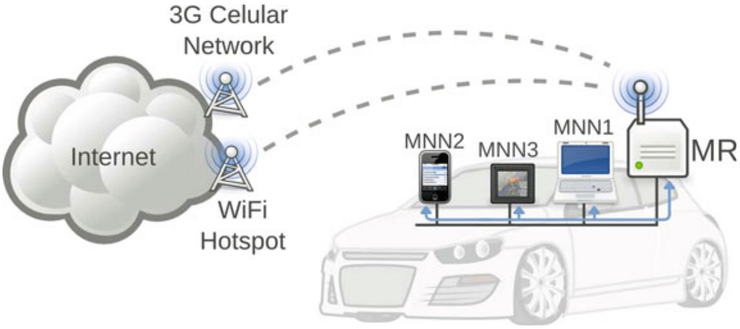


Fig. 1. Example of a Mobile Network

relies on IPv6 and NEMO to offer continuous communication between vehicles and the Internet infrastructure. This architecture introduces a distributed management plan which allows more advanced features such as simultaneous use of multiple wireless networks, per-flow routing, application adaptability to network conditions and advanced interface management.

In order to provide such services, CALM-compliant systems provide the ability to use the most appropriate access technology for information exchange. Selection rules are supported that include contextual information (e.g., current network environment, current position, current speed, etc.) which can be gathered from other devices or sensors available in the mobile network. User preferences and access technology capabilities can be taken into account in making decisions as to which access technology to use for a particular session, and when to handover between access technologies or between service providers on the same access technology. In [2], we came to a conclusion that a more convenient way to process and exchange these contextual information is to use a Context Management System (CMS).

The purpose of this paper is to share our experience in designing and implementing such a CMS applied to mobility.

The structure of the paper is as follows. Section 2 present the architecture we propose and analyzes a number of requirements for a distributed context management system suitable in a mobility context. Section 3 describes the implementation choices we made in order to provide a suitable context management system targeting adaptive applications in a mobility environment. Finally, Section 4 closes with our conclusions.

2 Requirements for a CMS Applied to Mobility

In a previous paper [2] we have shown that none of studied CMS based solutions is suitable for a mobility framework. [2] has also highlighted a set of important functionalities which are important in the design of a context management system applied to network mobility. The purpose of this section is to analyze these requirements in order to present and justify the design choices we have made.

The designed system aims at providing a convenient way to exchange and reason contextual information in a vehicular network.

2.1 General Design Requirements

In a vehicular network, we have many types of contextual information exchanged: network environment provided by the MR, current position, current speed,... which can be gathered from other devices or sensors available in the mobile network. Moreover, multiple mobile nodes can access to the shared context available. In order to deal with this kind of exchange we propose to use the *Context Server* Approach described in [1]. This distributed approach extends the *Middleware Infrastructure* [1] by introducing mechanisms to gather context from distributed sources. We propose to integrate the so-called *Context Repository* (see fig. 2) on the MR in order to facilitate concurrent multiple access to the context information. The usage of the *Context Server* approach has the advantage of relieving clients of resource intensive operations. This is important because embedded mobile nodes can be resource poor (GPS, smartphone ...).

Figure 2 shows the architecture and components of the solution we propose.

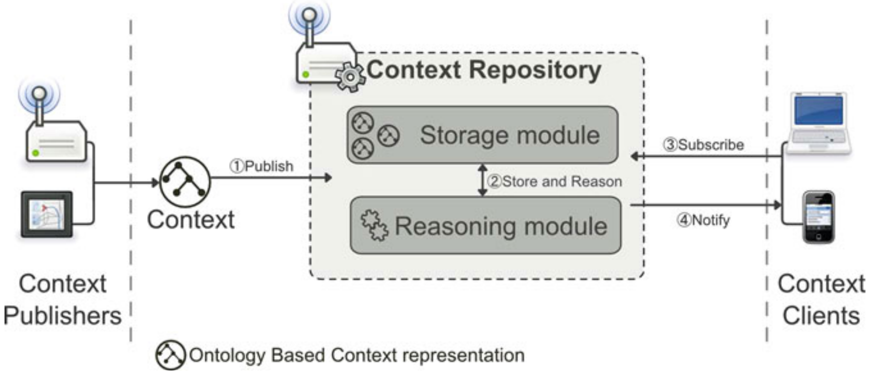


Fig. 2. Context Management System architecture and interactions

2.2 Context Discovery Requirements

In [2], we expressed the need to dynamically discover context providers and context repositories. Indeed, when a mobile node comes into the mobile network, he has to discover context information that match with the users requirements (e.g. network environment, localization, presence...). In our solution, we propose to use a centralized approach where context publishers have to publish their context to a repository called *Context Repository* (see fig. 2). In our context, we can imagine the following types of context advertised by producers in a vehicular network:

– Location data: a GPS or a smartphone with a localization capability can provide this kind of information and publishes these to *Context Repository*
– Network environment information: delay, bandwidth available, jitter can be computed by the mobile(s) router(s) and added to the context representation.

Moreover, rather than using a CMS exclusively for mobility management, it can be interesting to widen its purpose to offer context management for all the in-vehicle applications.

2.3 Distribution of Context Requirements

Context management systems adhere generally to the publish/subscribe paradigm where producers publishes information, and an event notification is sent to all authorized subscribers. All mobile nodes can be both context publisher and consumer (e.g. a smartphone can act as a context client wanting to know the network environment and a context publisher for transmitting location information). In general, the relationship between the publisher and subscriber is mediated by a service that receives publication requests, broadcasts event notifications to subscribers, and enables privileged entities to manage lists of applications that are authorized to publish or subscribe. The focal point for publication and subscription is a "node" to which publishers send data and from which subscribers receive event notifications. In our architecture this focal point is situated on the Mobile Router and its called *Context Repository* (see fig. 2).

2.4 Context Modeling Requirements

The context modeling refers to the requirement for formatting the context information. Information modeling offers a convenient way to define, store and reason this data in order to guarantee compatibility among the possibly heterogeneous devices (i.e. GPS, smartphones, sensors, computers...). The context modeling is particularly important in a vehicular network where mobile nodes are not a priori aware of each other. We propose to use an ontology based model in our solution to take advantage of their high and formal expressiveness. Moreover, ontologies enable reasoning and decision-making mechanisms which are very useful in deducing entailed context information from different sources of context (i.e. Vertical reasoning and/or Horizontal reasoning).

3 Implementation Choices

In addition to the general requirements described above, we have to address the following specific requirements we find out in our previous paper [2]:

3.1 Support for Discovery

When a mobile node is introduced into a vehicular network, its first task is to dynamically discover the network. A common approach in a managed network is to use DHCP (Dynamic Host Control Protocol) but the complexity to manage such a server based solution in a vehicular context led us to look for a solution without configuration or server. Therefore, we studied the IETF Zeroconf set of mechanisms which propose to solve the network discovery problems:

1. Allocation of IP network addresses for networked devices (link-local address auto-configuration)
2. Automatic resolution and allocation of computer hostname
3. Service discovery

In this paper, we use the last capability to discover the *Context Repository* (see fig. 2) offered by each MR as a service. This service can be advertised thanks to Zeroconf dynamic announcement mechanisms (DNS-SD [6]).

3.2 Context Exchange Mechanisms

As we said before, our architecture meets the publish-subscribe model. To implement this feature, we propose to use XMPP (Extensible Messaging and Presence Protocol [4]) and its publish-subscribe extension [3] which provides a convenient way to exchange context in the vehicular network. This protocol addresses problems we highlighted in [2]: First, publish-subscribe extension deals with the privacy and security issues. XMPP publish-subscribe extension contains a hierarchy of affiliations for the purpose of authorization and access control to the data (i.e. context information). Another interesting point is the ability of XMPP to offer fail-safe mechanisms (e.g. context repository database can be replicated and stored on multiple nodes).

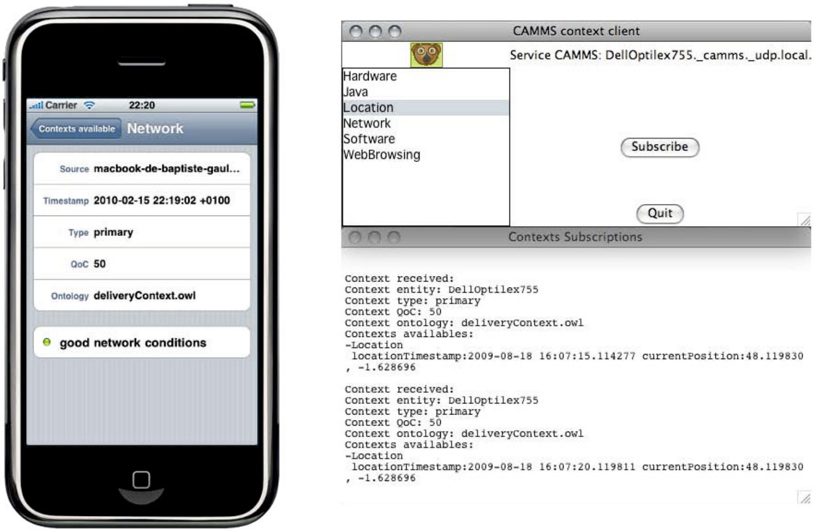
3.3 Context Modeling and Reasoning

By introducing context-awareness in vehicular network, applications become increasingly complex and interconnected. This raises the need for context modeling. The conclusion of the evaluation presented in [5] show that ontologies are the most suitable model because of its high and formal expressiveness and the possibilities for applying ontology reasoning mechanisms. As we mentioned before, we want to express the concept of physical location which can be represented as 'location', 'place', 'position' etc. To be able to interpret and reason such a context information, a context model is needed to capture concept unambiguously.

We propose to use the Web Ontology Language OWL [8] which is a flexible, extensible, expressive and common language to describe a ontology. To represent the types of contexts described in Section 2, we are interested in the Delivery Context Ontology [7] which provides a formal model of the characteristics of the environment in which devices interact with the Web or other services. The Delivery Context includes the characteristics of the Device, the software used to access the service and the Network providing the connection among others.

3.4 Implementing the Architecture

The Architecture described in Figure 1 was designed and implemented in order to test the performance and the viability of the system. The *Context Repository* was developed in Python and tested on a desktop computer. The *Context Client* software that can both publish and subscribe context information was implemented in two programming language: Python and Objective-C. This client was then tested on a wide range of devices spanning from resourceful laptop (running Mac OS X) to smartphone (running iPhone OS). The *Context Client* provides also a context visualizer with a graphical user interface which allows a user or a developer to dynamically monitor and produce (publish) context information (shown in fig. 3, when deployed on both a smartphone and a laptop).



Source macbook-de-baptiste-gaul...
Timestamp 2010-02-15 22:19:02 +0100
Type primary
QoC 50
Ontology deliveryContext.owl
good network conditions

Service CAMMS: DellOptilex755._camms._udp.local.

```
Context received:
Context entity: DellOptilex755
Context type: primary
Context QoC: 50
Context ontology: deliveryContext.owl
Contexts availables:
-Location
 locationTimestamp:2009-08-18 16:07:15.114277 currentPosition:48.119830
, -1.628696

Context received:
Context entity: DellOptilex755
Context type: primary
Context QoC: 50
Context ontology: deliveryContext.owl
Contexts availables:
-Location
 locationTimestamp:2009-08-18 16:07:20.119811 currentPosition:48.119830
, -1.628696
```

Fig. 3. The left picture depicts a network context viewed on an smartphone, while the right picture depicts a localization context on a laptop

4 Conclusions

In conclusion, this paper proposes a distributed context management system which provides exchanging and reasoning mechanisms. We have detected a number of both general and more specific requirements imposed by the mobility aspect. In this respect, we have proposed an approach which follow the publish-subscribe paradigm for forming loosely coupled entities and for advertising their context information. We argue that this approach fulfills the requirements we find out to a great extend. Furthermore, this architecture has been implemented, tested, and evaluated in real world applications, on both resourceful (laptops) and resource poor (smartphone) computer with significant success.

References

1. Baldauf, M., Dustdar, S.: A survey on context-aware systems. International Journal Of Ad Hoc And Ubiquitous Computing (2004)
2. Gaultier, B., Rayana, R.B., Bonnin, J.-M.: Context management systems applied to mobility. In: 9th International Conference on ITS Telecommunications RSM - Dépt. Réseaux, Sécurité et Multimédia (Institut Télécom-Télécom Bretagne, ITST (2009)
3. Millard, R.M.P., Saint-Andre, P.: Xmpp publish-subscribe extension. XMPP Draft XEP-0060 (2009), http://xmpp.org/extensions/xep-0060.html
4. Saint-Andre, P.: Extensible messaging and presence protocol (xmpp). RFC 3920 (October 2004), http://www.ietf.org/rfc/rfc3920.txt
5. Strang, T., Linnhoff-Popien, C.: A context modeling survey. In: Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England (2004)
6. Cheshire, M.K.S.: Dns-based service discovery. IETF working draft (September 2008), http://files.dns-sd.org/draft-cheshire-dnsext-dns-sd.txt
7. W3C. Delivery context ontology. W3C Recommendation (2009), http://www.w3.org/TR/dcontology/
8. W3C. Owl 2 web ontology language. W3C Recommendation (2009), http://www.w3.org/TR/owl-overview/

International Workshop on Mobile and Location-Based Business Applications (MAPPS 2010)

The Relationship between User Location History and Interests in Products and Services

Joshua Hurwitz, David Wheatley, and Young Lee

Motorola Applied Research Center,
1295 E. Algonquin Road, Schaumburg, Ilinois, USA
{joshua.hurwitz,david.j.wheatley,younglee}@motorola.com

Abstract. This study evaluated the use of location history as a predictor of user interests in products and services. Over a 1-month time period, subjects used a voicemail or email diary to report their visits to various establishments, such as shops and restaurants. At the end of the study, they completed questionnaires asking about their demographic characteristics, as well as their use of advanced mobile services and involvement in making decisions about the purchase or use of various products and services. A series of stepwise linear regressions showed that parameters derived from the diary data, when combined with demographic and mobile usage parameters, significantly improved predictions of product/service involvement, when compared to using the demographic and mobile usage predictors alone. These results suggest that location history measures could potentially be valuable components of algorithms for targeting commercial content to end users.

Keywords: Location history, targeting algorithms, consumer behavior, mobile commerce.

1 Introduction

While location-based services have been useful for providing context-sensitive functionality to users, other applications have focused on using context history to categorize user behaviors, interests and other characteristics [1]. These have included, for example, identifying users' social networks [2], determining their daily routines [3,4], categorizing their in-home activities [5] and inferring the purpose of their travels [6].

Another use of context history is to infer user interests in products and services, which could then help in targeting ads and other commercial content to users [7]. Historically, such targeting has been based on users' content preferences (e.g., favorite TV Shows), their prior product or service purchases using credit or loyalty cards, and their searching and browsing behaviors on the Internet. However, with context history, such interests could be estimated based on prior visits to retail shops, restaurants and other establishments.

Using context history in this way has some advantages over traditional approaches. It can provide information about users' interests even when there is no computer record identifying them with their purchases. This could happen if they do not make purchases using a credit or loyalty card, either because they pay with cash or because someone else pays for their purchases.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 301–312, 2010.

Location history can also provide information about more general lifestyle interests, based on prior visits to schools, parks and other non-commercial establishments. Thus, if a user is a member of a retailer's loyalty program, such lifestyle data could potentially identify product or service interests that are not evident in the purchase history data that the retailer has for that user.

Finally, information about regularities in the user's travel and visit times (e.g., dining out every Saturday evening) can be used to anticipate purchases. This could enable more timely presentations of commercial content (e.g., every Saturday afternoon, presenting mobile ads that offer restaurant discounts), thereby increasing the relevance of the content to the user.

Aside from discovering users' product or service interests, location history could also be used to discover typical patterns of visits that large groups of users have in common. Then, through market research, it may be possible to find associations between such patterns and the predominant product or service interests for those groups. This approach could then promote collaborative filtering, where ads or special offers for such products or services can be presented to new users who exhibit similar patterns. This is analogous to approaches used in online shopping recommender systems, except that those systems rely on user Internet-based behavioral patterns, such as browsing patterns, for targeting ads or recommending products [8].

Product/Service Interests. In order to evaluate the use of location history as a predictor of product/service interests, the current study evaluated the relationship between patterns of visits to certain categories of establishments and one important measure of product/service interest: involvement in making decisions about the purchase or use of certain categories of products or services (Product/Service Involvement, or PSI). Following Zaichkowsky [9], involvement includes doing research and comparing brands, versions etc. of products or services. However, the current study also includes the frequency with which users make these decisions.

In evaluating location history as a predictor of PSI, the approach taken here is to assess whether it incrementally adds predictive capabilities above other traditional predictors. For example, there are well-known demographic differences in involvement for different types of products and services, including differences in gender, age and income. A strong test of the predictive capabilities of location history is whether it predicts variability in involvement that is not accounted for by these other variables.

Another potential predictor of involvement for certain products and services is the use of advanced mobile services, such as location-based and mobile Internet services. This variable is accessible to analytics systems that collect user mobile data, and Pagani [10] has shown that use of such services is associated with user segments, such as Innovators and Early Adopters, who tend to more readily adapt new technologies. Thus, use of advanced mobile services should be associated with greater involvement in making purchase and usage decisions regarding technology products and services.

Measuring Location History. The results reported here come from a study evaluating the use of location history to estimate the times of visits to establishments, and the validity of location history as a predictor of PSI. In this study, establishment visits were recorded using both GPS data and subjects' diary reports. However, the analyses in the current paper will focus only on the diary reports, because more visits could be identified from the diary entries than from the GPS results. This was due to both subject errors (e.g., subjects forgetting to bring the GPS data logger with them on a shopping trip) and technical problems (e.g., lack of a GPS signal in multi-level shopping malls). Furthermore, the diary data was considered to be a good substitute for the location data, since there was a significant relationship between 1) the rankings of subjects' preferred establishment categories derived from the diary reports of their establishment visits and 2) those same rankings derived from the GPS-based estimates of such visits [7].

In order to produce, from the diary data, a set of variables representing visit patterns, the study used Principal Components Analysis (PCA), a form of Factor Analysis. This approach extracts the common variance out of a set of correlated measures to produce a smaller number of more stable variables. Thus, if there is a tendency for users who visit one category of establishments (e.g., clothing shops) to visit others as well (e.g., accessories and beauty shops), there would be a relatively high correlation between visit frequencies across these categories. PCA would then derive one factor that represents this tendency, along with a factor score for each user representing the degree to which he or she prefers visiting these establishments. The factor analytic approach helps produce fewer and more stable variables for use as predictors in the models of PSI. Similarly, PCA was used to reduce the number of PSI variables, thus producing fewer, more stable dependent measures as well.

2 Method

A total 24 subjects (11 male and 13 female) participated in the study, ranging in age from 23 to 66 years. Individuals were selected to participate in the study if they 1) resided in Schaumburg, Illinois and adjacent towns and villages, and 2) reported being frequent shoppers having an annual household income exceeding $50,000. These criteria were used to increase the chances that they would engage in a relatively large amount of shopping during the course of the study.

The study itself consisted of two parts, a field study and a lab study. The field study involved collecting diary data regarding the users' visits to establishments during the 1-month time period of the study. The lab study included questionnaires on the subjects' basic demographics, their use of mobile services, and their involvement in commercially-relevant activities.

For their diary entries, the subjects were instructed to report on their visits to establishments each day by calling a toll-free number and leaving a voicemail message for the experimenter, or by sending an email message to the experimenter. They were instructed to include, in each report, their name, the name of each establishment they visited, the type of establishment, its location, the date and time they entered and exited, and the reasons for visiting that establishment.

2.1 Lab Study

After subjects had completed the field study, they were brought into the lab to complete some questionnaires. Two questionnaires that are relevant to the current report were the Demographics Questionnaire (DQ) and the Product/Service Questionnaire (PSQ). The DQ asked basic questions, such as subjects' age, gender and household income. It also asked about how frequently they use a mobile device to talk, access web sites, send or receive email and text messages, get GPS navigation instructions, and play games.

The PSQ measured involvement in making decisions regarding the purchase or use of certain products and services. For each product or service, subjects were asked two questions, an “Experience” question and an “Involvement” question. The Experience question asked about the frequency with which they purchased or consumed that item over a given time period. The Involvement question asked how involved they were in purchase or consumption decisions, which entailed doing research to learn more about the product or service, determining the best brand to buy or use, and deciding how much to pay for it. For consumption, involvement referred to making decisions about, for example, what TV programs to watch, meals to prepare, etc.

Product/Service Involvement. To analyze the PSQ results, the “Experience” and “Involvement” scores were combined together into one measure. However, the problem with the PSQ “Experience” items was that the response scales differed across items. The items asked how frequently subjects performed activities across various time spans, from weeks to years, depending on the activity. In order to create a standard scale for all PSQ items, the response of each subject, *j*, on each “Experience” item, *i*, was rescaled by computing a standard score, E_{ij}, for that item, using the following formula:

$$E_{ij} = \frac{x_{ij} - \overline{x}_i}{s_i} \qquad (1)$$

where x_{ij} was the original response of that subject for this item converted to an integer scale, $\overline{x}_i$ was the average rating for the item, and s_i was the standard deviation of the ratings for that item.

Unlike the “Experience” items, responses to the “Involvement” items were already on the same 5-point scale, from “I have not been involved at all” to “I have been very involved”. However, the problem with these items was in how subjects interpreted “being involved”. Some subjects, for example, might have had a more liberal interpretation of this than others, so their overall responses could have been toward the top of this scale (i.e., “C”, “D” or “E”), whereas others might have been more conservative, giving responses of mainly “A” or “B”.

Thus, to assure that all subjects were on the same scale, each subject’s responses to these items were standardized with respect to that individual’s overall distribution of responses to the Involvement items. Thus, the standard score, V_{ij}, for Involvement item *i* and user *j*, was defined as

$$V_{ij} = \frac{x_{ij} - \overline{x}_j}{s_j} \qquad (2)$$

where x_{ij} is the subject’s response for this item converted to an integer scale, $\overline{x}_j$ is the average rating that the subject gave for all of the “Involvement” items, and s_j is the standard deviation of that subject’s ratings for these items.

Once the responses to the “Experience” and “Involvement” items were standardized, they were combined together to produce the Product/Service Interest Index (PSII), using the following logistic function:

$$I_{ij} = \frac{1}{1+e^{-\left(E_{ij}+V_{ij}\right)}} \tag{3}$$

As shown in Table 1, there were 2 product or service items for each of 15 categories: Autos (AUT), Entertainment (ENT), Food and Drink (FAD), Finance (FIN), Health (HEA), Household (HSD), Internet Services (ISP), Leisure (LEI), Print Media (PRM), Public Service/Nonprofit (PSN), Retail (RTL), Sports (SPO), Toiletries and Cosmetics (TAC), Technology (TCH), and Telecommunications (TEL).

Table 1. Products and services used in the Product/Service Questionnaire

CAT	ITEM 1	ITEM 2	CAT	ITEM 1	ITEM 2
AUT	Cars	Automobile Magazines	PRM	Magazines	Newspapers
ENT	Movies	Television Programs	PSN	Charities	Food Banks
FAD	Groceries	Meals prepared	RTL	Clothing	Jewelry
FIN	Stocks Traded	Checking Financial Markets on the Web	SPO	TV Sports Programs	Live Professional Sporting Events
HEA	Non-prescription remedies	Health Products	TAC	Hair Products	Creams and Moisturizers
HSD	Lawn/ Garden Maintenance	Home Repair/ Maintenance	TCH	Computers	Mobile Phones
ISP	Email Addresses	Web Hosting Services	TEL	Call Forwarding	Caller ID
LEI	Vacation/ Holiday Travel	Amusement Parks			

3 Results

Diary Data. There were 401 voicemail diary reports and 100 email diary reports during the course of the study, and a total of 1500 reported visits to establishments, or an average of 62.5 visits per subject (SD 39.8). Three subjects reported fewer than 20 visits, whereas 3 reported 120 or more.

For the analyses of the diary data, a visit was included if there was sufficient information about the name, location and entry and exit times for that visit. However, subjects sometimes left out important information or gave incorrect information about their visits. Out of the original 1500 diary entries, 1286 or 85.7% included sufficient information about the name and location, as well as entry and exit times. In some cases, it was possible to fill in missing or incorrect information by searching for establishments on the web using the information that was provided in the diary.

Categorizing the Diary Data. A list of 485 unique brands of commercial establishments were derived from the diary data. These brands were then divided into 21 categories. Among these categories, restaurants comprised the highest number of brands (124), followed by retail (e.g., department stores; 94 brands), food & drink (e.g., groceries; 35), health-related establishments (e.g., doctors office; 35), auto (e.g., auto service stations; 29), recreation (e.g., health clubs; 21), and toiletries and cosmetics (21). However, snack shops (e.g., donut shops) had the highest number of visits per brand (6.3), followed by food and drink (5.8 visits per brand), technology establishments (e.g., home electronics; 4), educational (e.g., schools; 3.4), retail (3.2), finance (e.g., banks; 3.2), household (e.g., hardware stores; 3.1), and print media (e.g., book stores; 3.1).

In order to reduce the number of categories, the diary results were subjected to a Principal Components Analysis, where the unit of analysis was the number of reported visits by each subject in each category. The version of PCA used here employed Varimax rotation and Kaiser Normalization. This transformation assured that the factors were maximally differentiated from each other, and that they accounted for a significant percentage of the variance.

Given the large number of variables, two analyses were performed in order to produce more stable models. The decision about which variables to include in which analyses was made based upon the Pearson Product Moment correlations among the variables.

The results of these analyses are shown in Table 2, which displays the factor loadings for each of the diary categories in each factor, and the percentage of variance accounted by each factor in the model. As shown in the Table, there were two analyses performed, each limited to 3 factors. The first analysis accounted for 76.6% of the variance and the second accounted for 79.6%.

In the first analysis, the first factor, "RecHealth", combined the recreation and health categories together. The second factor, "AutoPrint", combined Auto and Print Media, and included also Retail, which had a negative loading for in this factor. The last factor, "Leisure", combined Travel and Dining, as well as "Food & Drink", which had a negative loading on this factor.

In the second analysis, the first factor, "SnackHouse", combined the Snack category (which included coffee and donut shops) with the Household category (including home repair, lawn/garden maintenance, etc.). The second factor, "Entertain", revolved around entertainment activities and hobbies. The final factor, "TechToys", incorporated toys, games and technology.

Note that the categories and factor solutions presented here do not necessarily constitute the only way of dividing the space of establishments. Furthermore, other approaches, such as cluster analysis, are likely to yield different results. However, the solution here produced sensible factors that accounted for a large-enough percentage of variance in the data to justify using the factor scores derived from these analyses to predict product/service involvement.

Table 2. Results of factor analyses of the diary data

Item	**Component**			
	1	**2**	**3**	
	RecHealth	**AutoPrint**	**Leisure**	
Recreation	**0.91**	0.12	-0.10	
Health	**0.84**	-0.14	-0.23	
Auto	-0.42	**0.83**	0.19	
Print Media		**0.80**	-0.17	
Retail	-0.39	**-0.77**		
Travel			**0.87**	
Dining	-0.21	0.23	**0.79**	
Food & Drink		0.35	**-0.72**	
% of Variance	30.4	23.7	22.5	Total = 76.6

Item	**Component**			
	1	**2**	**3**	
	SnackHouse	**Entertain**	**TechToys**	
Snack	**0.89**			
Household	**0.86**	0.25	-0.12	
Entertainment		**0.90**		
Hobbies	0.29	**0.82**	-0.14	
Toys & Games	-0.23		**0.86**	
Technology		-0.25	**0.85**	
% of Variance	27.8	26.7	25.1	Total = 79.6

PSQ Results. In order to reduce the number of variables in the PSQ, the PSII scores were submitted to three Principal Components Analyses, each with Varimax Rotation and Kaiser Normalization. As shown in Table 3, four orthogonal factors were identified in the first analysis, 4 in the second, and 3 in the third. The first analysis accounted for 78% of the variance, the second accounted for 76.3% and the third accounted for 79.4%. Four of the PSQ items, Amusement Parks, Automobile Magazines, Books and Mobile Phones did not load significantly on any factor.

The factors in the first analysis revolved around the themes of Homemaker, Sundries, Entertainment and Technology. The "Homemaker" factor incorporated items relating to family care and self-grooming, while the "Sundries" factor contained items relevant to accessories and self-grooming. The "Entertainment" factor included movies and vacation/holiday items, as well as food shopping and preparation. Finally, the Technology factor ("Tech") incorporated computers and web hosting services.

The first factor in the second analysis, "EmailCar", had items relevant to autos and email addresses. The "Telecom" factor items related to telecommunications services, and the "Financial" factor included Internet financial services and investment in stocks.

In the third analysis, the "HomeCare" factor incorporated items relating to home and lawn/garden maintenance. The "SportsTV" factor included spectator sports and TV watching items. The third factor, "Charity", included donating to charities and food banks.

Table 3. Results of factor analyses of the PSQ.
(Top: Analysis 1, Middle: Analysis 2, Bottom: Analysis 3)

Item	**Component**				
	Homemaker	**Sundries**	**Entertain**	**Tech**	
non-prescription remedies	**0.80**			0.23	
visited an amusement park	**0.73**		0.38	0.41	
gone shopping for groceries	**0.65**	0.21	**0.51**	-0.24	
hair products	**0.64**	**0.48**		-0.15	
shopped for jewelry		**0.87**		0.32	
Clothing		**0.79**	0.23		
face creams, body creams . . .	**0.48**	**0.73**		-0.24	
Movies			**0.87**	0.11	
traveled on vacation	0.13	**0.47**	**0.77**		
prepared a meal	**0.52**		**0.66**	0.25	
web hosting services	0.11	-0.13	-0.12	**0.91**	
Computers		0.24	0.35	**0.74**	
Percentage of Variance	37.6	17.4	11.9	11.2	Total = 78.0

Item	**Component**				
	EmailCar	**Telecom**	**PrintNews**	**Financial**	
cars have you owned	**0.88**	0.29	-0.11		
email addresses	**0.88**	-0.31	0.12	0.17	
caller ID		**0.88**	-0.15		
call forwarding		**0.72**	0.46	0.20	
news magazines	-0.20	-0.24	**0.78**		
Newspapers	0.19	0.26	**0.75**		
check markets on the Internet		0.18	0.21	**0.88**	
traded stocks	0.34	-0.18	-0.22	**0.76**	
Percentage of Variance	21.7	20.7	18.9	18.0	Total = 79.4

Item	**Component**			
	HomeCare	**SportsTV**	**Charity**	
home repair/maintenance	**0.86**	0.26		
maintaining lawn & garden	**0.85**	-0.23		
donating to charities	**0.68**		**0.59**	
sport(s)-related programming	-0.13	**0.86**	0.13	
live professional sporting events	0.33	**0.79**	-0.25	
hours of television	-0.11	**0.73**	0.39	
vitamins, supplements, etc.		0.20	**0.86**	
donating to food banks			**0.81**	
Percentage of Variance	25.8	25.7	24.8	Total = 76.3

Regression Analyses. The final analyses focused on whether diary factors provided significant incremental improvements in predictions of PSQ factors over using the demographic and mobile usage variables alone. The first step was to fit linear regression models using Gender, Age Range, and Income as predictors, and the PSQ factors as dependent measures. Then the next step was to add the mobile context variables to the models. For these analyses, the Gender variable was coded 1 for male and 2 for female, and Age-Range was coded according to the 5-point scale used in the study: "1" for under 20 years, "2" for 21 to 30, "3" for 31 to 40, "4" for 41 to 50, and "5" for over 50.

Tables 4 and 5 shows the results of modeling the PSQ factors using the demographic predictors alone, and then adding the mobile context variables. For the first three models, the dependent variable was the Sundries factor. The first of these models included, as the predictor, only the Leisure factor derived from the diary data. Subjects with greater involvement in Leisure-relevant products and services more frequently

Table 4. Results of regression analyses using factor scores derived from the diary data as predictors and factor scores derived from the PSQ data as the outcome measures [1]

Prod. Invmt. Factor		Demographic Variable			Context Variable 1				Context Variable 2			
		Name	Coef.	Adj. R^2	Label	Coef.	Adj. R^2	Incr. R^2	Label	Coef.	Adj. R^2	Incr. R^2
Sundries	1	[none]			Leisure	0.51	0.23	0.23				
	2	Gender**	0.90	0.18	Leisure	0.50	0.42	0.24				
	3	Income**	0.02	0.22	Leisure	0.49	0.44	0.22				
Sports-TV	1	[none]			Mobile	-0.39	0.17	0.17	Tech-Toys	0.42	0.32	0.15
	2	Gender*	-0.64	0.07	Mobile	-0.41	0.24	0.17				
Home-Care	1	[none]			Snack-House	0.49	0.30	0.30	Rec-Health	0.45	0.48	0.18
Email-Car	1	[none]			Leisure	-0.43	0.15	0.15	Auto-Print	0.43	0.30	0.15
	2	Gender*	-0.68	0.08	Leisure	-0.43	0.24	0.15				
Print-News	1	[none]			Mobile	-0.42	0.14	0.14	Auto-Print	0.43	0.29	0.16
	2	Age Range**	-0.35	0.09	Auto-Print	0.43	0.25	0.15				

[1] Note that, except where "[none]" is indicated, each regression included one demographic variable. The context variables were added in via stepwise regression, with a variable added as long as $p < .05$, and removed when $p > 0.1$. All regressions are statistically significant at $p < .05$, all context predictor variables are significant at $p < .05$, all demographic predictors with an asterisk (*) are nearly significant at p < 0.1, and all demographic predictors with a double asterisk (**) are significant at $p < .05$.

reported visiting restaurants and travel-related establishments, but less frequently reported visiting grocery stores and similar establishments. This model produced a statistically significant fit, but accounted for only 23% of the variance.

As Sundries model 2 shows, when Gender was the lone predictor, the model accounted for 18% of the variance, whereas when both Gender and Leisure were the predictors, the model accounted for 42% of the variance. Similarly, when Income and Leisure were the predictors, the model accounted for 44% of the variance. Thus, the best predictors for the Sundries factor was 1) being female, 2) having a larger income, and 3) visiting establishments that provide Leisure-relevant services (i.e., travel and dining), but not visiting grocery stores and related establishments.

When "SportsTV" was the dependent variable, the best-fitting model, accounting for 32% of the variance, included Mobile and "TechToys" as predictors. Subjects who watched more sports and TV were less likely to use mobile services and more likely to visit toy and technology stores. In the second "SportsTV" model, accounting for 24% of the variance, the key predictors were being male and not using mobile services.

With "HomeCare" as the dependent variable, the significant predictors were "SnackHouse" and "RecHealth". For this model, which accounted for 48% of the variance, those subjects who more often reported visiting snack shops (e.g., donut and coffee shops), household-relevant establishments (e.g., stores selling supplies for home and garden care and maintenance), and recreation- and health-related establishments (e.g., health clubs) were more likely to be involved in making decisions about purchasing and using home, lawn and garden maintenance products and services, and were also more likely to be involved in deciding what charities to contribute to.

Table 5. Summary of regression results

Subjects who…..	were more likely to….	were less likely to….
watched more TV and watch more live/televised sports ("SportsTV")	…visit toy & games stores and technology stores …be male	…use advanced mobile services
had more involvement in home, lawn, garden maintenance and donations to charity ("HomeCare")	…visit snack shops, and household supplies stores (including home repair, lawn & garden care & maintenance) …visit recreation & health related establishments, e.g., health clubs …be involved in deciding what charities to donate to	
had more involvement in purchase decisions about email services, auto products & services	…visit bookstores, auto products and services establishments …be male	…visit restaurants, travel establishments, clothing and grocery stores.
had more involvement in purchasing decisions about newspapers & news magazines	…visit bookstores, auto products and services establishments …be younger	…use advanced mobile services

The significant predictors of “EmailCar” were Leisure and “AutoPrint”. Thus, the subjects who were more involved in decisions regarding email services and automotive products and services were 1) less often visiting restaurants, travel-related establish- ments, and clothing and grocery stores, and 2) more often visiting book-stores and establishments selling automotive-relevant products and services. Furthermore, “EmailCar” model 2 shows that males were more involved in making these decisions.

The final set of models showed that the use of fewer mobile services and more frequent visits to bookstores and automotive-relevant establishments were both significantly associated with greater involvement in making purchase decisions about newspapers and newsmagazines. Furthermore, as “PrintNews” model 2 shows, younger individuals were more likely to be involved in making these decisions as well.

4 Discussion

The results of this study demonstrate the potential for using location history as an indicator of user interest in products and services. These results also suggest that location history could add significant value to more traditional measures (e.g., demographic measures) as predictors of user interests. Adding the location history factor scores as predictors significantly improved the predictive capabilities of the regression models, when compared to using demographic variables and a measure of mobile usage alone. The implication is that visit patterns are a significant component in models for targeting ads and other commercial communications.

One limitation, however, was the small sample size. It is likely that a larger sample would have produced more significant prediction models. Furthermore, a longer data collection period would have produced more accurate estimates of visit patterns. There was likely some bias since, in the 1-month time period of the study, the observed patterns were probably influenced by the season during which the study was conducted.

Another limitation was the lack of purchase history data for the subjects. Many targeting algorithms rely on such data to make inferences about user product interests. Incorporating purchase history would also have improved the model predictions.

Finally, it was clear from the data that some subjects were better than others at reporting their establishment visits. Despite this, there were more visits identified in the diaries than in the GPS data [7], due to the technical and human-error issues discussed above. Perhaps some of the technical problems might be overcome by using both GPS and Wi-Fi to estimate location [11].

Overall, the results presented here demonstrate that use of location history for targeting could potentially support location-based and m-commerce services that rely on analytics, advertising and sales as sources of revenue. If future research supports the validity of location history as a predictor of PSI, then the next steps should include evaluating whether targeting based on location history can significantly increase consumer basket size and other outcome variables that are important for retailers.

References

1. Matsuo, Y., et al.: Inferring Long-term User Properties based on Users' Location History. In: IJCAI 2007, Hyderabad, India (2007)
2. Pentland, A.: Automatic Mapping and Modeling of Human Networks. Physica A: Statistical Mechanics and its Applications 378(41), 59–67 (2006)
3. Eagle, N., Pentland, A.: Eigenbehaviors: Identifying Structure in Routine. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, Springer, Heidelberg (2006)
4. Farrahi, K., Gatica-Perez, D.: Daily Routine Classification from Mobile Phone Data. In: 5th Joint Workshop on Machine Learning and Multimodal Interaction. Utrecht, The Netherlands (2008)
5. Zimmermann, A., Loren, A.: LISTEN: Contextualized Presentation for Audio-Augmented Environments. In: 11th Workshop on Adaptivity and User modelling in Interactive Systems. Karlsruhe, Germany (2003)
6. Wolf, J., Guensler, R., Bachmann, W.: Elimination of the Travel Diary: An Experiment to Derive Trip Purpose From GPS Travel Data. In: Transportation Research Board 80th Annual Meeting. Washington, DC (2001)
7. Hurwitz, J.B., et al.: Using Location History to Identify Patterns in Mobile Users' Visits to Establishments. In: Human Factrors and Ergonomics Society, San Francisco (in press)
8. Prassas, G., et al.: A Recommender System for Online Shopping Based on Past Customer Behaviour. In: Bled Electronic Commerce Conference, Bled, Slovenia (2001)
9. Zaichkowsky, J.L.: Measuring the Involvement Construct. Journal of Consumer Research 12(3), 341–352 (1985)
10. Pagani, M.: Determinants of Adaption of Third Generation Mobile Multimedia Services. Journal of Interactive Marketing 18(3), 46–59 (2004)
11. Kang, J., et al.: Extracting places from traces of locations. Paper presented at the WMASH, Philadelphia, PA, USA (2004)

Predicted and Corrected Location Estimation of Mobile Nodes Based on the Combination of Kalman Filter and the Bayesian Decision Theory

Muhammad Alam[1,2], Mazliham Muhammad Suud[1], Patrice Boursier[2], Shahrulniza Musa[1], and Jawahir Che Mustapha Yusuf[1,2]

[1] Centre for Research and Postgraduate Studies (CRPGS) & UniKL MIIT
Jln Sultan Ismail, 50250, Kuala Lumpur
muhammad.unikl@gmail.com, mazliham@unikl.edu.my,
{shahrulniza,jawahir}@miit.unikl.edu.my
[2] Laboratoire L3i, Université de La Rochelle, 17000 La Rochelle, France
patrice.boursier@univ-lr.fr

Abstract. The main objective of this research is to apply statistical location estimation techniques in cellular networks in order to calculate the precise location of the mobile node. Current research is focusing on the combination of Kalman filter and the Bayesian decision theory based location estimation. In this research basic four steps of Kalman filter are followed which are Estimation, Filtering, Prediction and Fusion. Estimation is done by using Receive Signal Strength (RSS), Available Signal Strength (ASS) and the Angle of Arrival (AOA). Filtering is done by calculating the average location and variation in values of location. Prediction is done by using the Bayesian decision theory. Fusion is done by combining the variances calculated in filtering step. Finally by combining the prediction and fusion results PCLEA (Predicted and Corrected Location Estimation Algorithm) is established. Timestamp is used for recursive step in kalman filter. The aim of this research is to minimize the dependence on the satellite based location estimation and increase its accuracy, efficiency and reliability.

Keywords: Kalman filter, Bayesian decision theory, location estimation.

1 Introduction

Location estimation of a mobile user is a very popular research area from past few years. Due to the growth of cellular architecture the mobile users originating calls are also increasing at the same time. It is estimated that more than 50% emergency calls are originated by the mobile phones [1]. Techniques which are used for location estimation are satellite based techniques, geometric techniques, statistical techniques and the mapping techniques [2], [3]. All techniques have different accuracy level, processing time, coverage and the cost. The location of the mobile node can be estimated by the mobile node itself which is known as self positioning. Otherwise it can be calculated by the server with the help of the reference points, which is known as remote positioning or network centric positioning [4]. Two different approaches are used by the researchers, the direct positioning approach and the two step-step

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 313–325, 2010.

positioning approach. In direct positioning approach position is estimated directly from the signal travel between two nodes [5]. In two steps positioning approach first different signals parameters are calculated and in the second part position of the mobile node is estimated by using these parameters. The accuracy level of two step approach is higher as compare to direct approach [5], [6]. Figure 1 is explaining the direct and the two step positioning approaches [3], [5], [6].

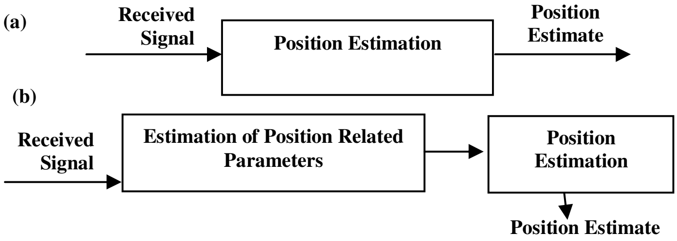


Fig. 1. (a) Direct Positioning, (b) Two-step Positioning [3], [5], [6]

Current research is falling under two step position technique which is focusing on the Kalman filtering combine with the Bayesian decision theory results. The cycle of Kalman is based on prediction and correction. It is very powerful as it supports past, present and future predictions [7]. Kalman filter also use four step procedure for accuracy in measurement updates [7] estimation, filtering, prediction and fusion.

In this research estimation is done by using RSS, ASS and the AOA as a parameters. Filtering is performed by the mean and the variant values of location at different timestamps to calculate the Region of Interest (RoI). Whereas prediction is done by using Bayesian decision theory, overlapping area (**Ω**) is used as a-priori probability in Bayes law. Finally fusion is performed by combining the variances of different timestamps and fuse results with the predicted value calculated by Bayesian decision theory. Based on the Kalman filter cycle [7], an algorithm PCLEA is developed which predicts and corrects the location values based on recursive timestamps.

This paper is divided into six sections; in section II literature survey of previous research work is done. Discussion review is included in section III. Section IV is defining the problem statement. Section V presents proposed architecture which includes estimation, filtering, prediction, fusion and PCLEA. Section VI is dedicated for results and discussion and finally conclusion is added in section VII.

2 Relevant Studies

In [8] authors used Extended Kalman filter approach in WSN for location estimation. Parameter considered was RSSI, where is RSSI=K/R2.

K is considered as the measure of confidence level and R is representing the radius of the sensor network communication range. Authors claimed that their method is showing better results when compare with existing methods.

In AVG [9], authors used Kalman filter very efficiently to control noise and error control. Based on experiments they claimed that derived approach offers better suppression of vision measurement noise and a better performance in the absence of vision measurements.

In [10] authors proposed Kalman filter based Location estimation with NLOS mitigation. They fused results of geometric approach with the Kalman for location estimation. They suggested that geographical information can be added for better accuracy of mobile location.

In [2], author proposed and implemented GPS free global positioning method for mobile units for indoor wireless environment. He used Bayesian filtering approach for initial measurement, cell-ID of the serving base station, and the predetermined route radio maps. In step one author derived generic recursive Bayesian filter algorithm for prediction and update. In step two global positioning algorithm is derived which is used to track the probable position of mobile unit. Author repeated the experiment for 100 times and claimed the estimated position tracking error is between 15m and 20m.

Sinan Gezici et al [3] investigated and presented various positioning algorithms. Author pointed out that two approaches used for the position estimation

(i) the "Direct approach" in which position of the mobile node is estimated based on the signal travel between two nodes [5],

(ii) and the two-step positioning that first calculate the different position parameters like AOA, TOA, TDOA, ASS, RSS etc and based on these parameters position of the cell phone is estimated. Author also pointed out the geometric and statistical techniques can be used for the accuracy.

In [11] authors introduced a Bayesian hierarchical modeling approach for location estimation. Instead of locating a single node they simultaneously located a set of wireless nodes. Their work is based on prior knowledge and they constructed the network as used in Boltzmann learning. They demonstrate that their model achieved similar accuracy as previously published models and algorithm.

In Selective Fusion Location Estimation (SELFLOC) [12] authors assigned weight to each location and weighted sum gave the SELFLOC estimation. They calibrated the branch weights during the offline stage using the error feedback. Authors adopted the Minimum Mean Square Error (MME) [13] for SELFLOC weight calibration. Authors also applied SELFLOC algorithm with other classical location algorithms to improve accuracy. Classical algorithms they used in their experiments are Triangulation (TN), K-Nearest Neighbour averaging (KNN) and the Smallest M-vertex Polygon (SMP).

In Region of Confidence (RoC) [12] authors attempted to counter aliasing in the signal domain. By using the probabilistic techniques the algorithm first forms a region of confidence within which the true location of a user lies. Through implementation and experiments authors experienced best mean distance error of 1.6m while using SELFLOC weighted localization algorithm. By using RoC with eKNN they experienced that the error can be improved from 6m to 4.5m.

Vinay Seshadri and others et al [14] used Bayesian sampling approach for the location estimation of indoor wireless devices by using RSSI as main sensing parameter. The proposed architecture used posterior probability of the target location using sequential Monte-Carlo sampling, which is capable of using arbitrary a-priori distribution to compute a posterior probability [14]. Based on the simulation results authors believe that the method is less computationally intensive and it is also suited to an indoor wireless environment where other standards may not work.

In [15] author presented a statistical location estimation technique based on a propagation prediction model. Author took signal parameters such as Receive Signal Strength and Angle of Arrival. Propagation delay is considered as random variable which is statistically dependent on the location of receiver, transmitter and the propagation environment. Author comments that statistical approaches include certain type of flexibility.

If the signal propagation environment differs significantly from ideal condition the distance or angle measurements will be unreliable [15].

Different statistical tools are used by researchers for the precise location of wireless nodes specially Bayesian based location estimation is not new but the combination of the kalman filtering with the Bayes theorem for location estimation is a research area which is not extensively touched by researchers.

3 Problem Statement

Location estimation of a mobile node is not new area of research. Global Positioning System (GPS), Cell Identifier (CI), Location Area Identifier (LAI), GSM and WLAN positioning all falls under the location estimation. On the other hand Bayesian decision theory is commonly referred by researchers. But there is a need of mechanism which can reduce error rates from few meters to few centimeters. Current research is trying to cater the same problem with the help of Bayesian decision theory using a-priori condition of overlapping coverage area (**Ω**) and the fusion results of combine variance ($1/\sigma^2$). Kalman filter recursive model of prediction and correction is followed to achieve accuracy.

4 Propose Architecture

The propose architecture is divide into four steps. In step I we use geometric approach by using RSS, ASS and the AOA to calculate the estimate value of mobile node [16], [17]. In step II we calculate average and variance values in order to filter the estimate position. Step III calculates the Bayesian decision theory based prediction by using overlapping coverage area (**Ω**) as a-priori probability. In Step IV we fuse results of Step II by using kalman filter combine variance approach. Finally we propose PCLEA by fusing the results of Step III and IV in order to get the most accurate position of a cellular node.

4.1 Step I: Estimation (W)

At the first step of the proposed architecture geometric position estimation techniques are used. ASS, RSS and the AOA are used as parameters. Our assumption is based on,

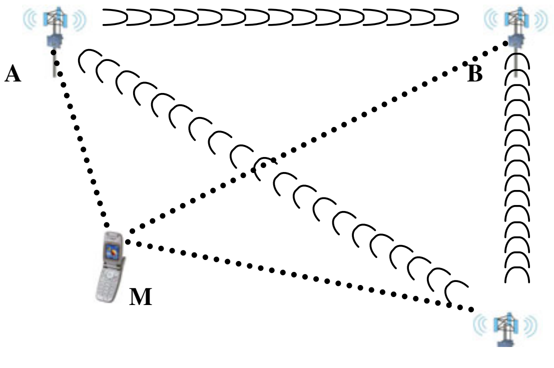


Fig. 2. Mobile node (M) is receiving signals from antennas

that the cell phone is receiving signals from 3 BTS (Base Transceiver Station). In this condition 3 triangles will be constructed i.e. ΔABM, ΔACM and ΔBCM as shown in the figure 2 [16], [17].

By using the ASS and the RSS, the distance between points AB, AM and AC is calculated.

$$D_{(AB)t0} = \frac{ASS_{(A)t0+}ASS_{(B)t0}}{2} - \frac{RSS_{(A)t0} + RSS_{(B)t0}}{2} \qquad (1)$$

$$D_{(AM)t0} = ASS_{(A)t0} - RSS_{(M)t0} \qquad (2)$$

$$D_{(BM)t0} = ASS_{(B)t0} - RSS_{(M)t0} \qquad (3)$$

where
ASS is Actual Signal Strength at A and B at t0.
RSS is Receive Signal Strength from A, B and M at time t0.
$D_{(AB)}$ is distance between point A and B.
$D_{(AM)}$ is distance between point A and M.
$D_{(BM)}$ is distance between point B and M.

As the location of points A, B and C are known and the distance between A, B and M is calculated. By using the simple trigonometry formula angles α and β are calculated at the next step.

$$\text{Cos}\alpha = \frac{\{D_{(AM)t0}\}^2 + \{D_{(AB)t0}\}^2 - \{D_{(BM)t0}\}^2}{2\,D_{(AM)t0}\,D_{(AB)t0}} \qquad (4)$$

By using basic trigonometric f*ormula for angle calculation with three known sides,*
Cosa = $(b^2 + c^2 - a^2)/\,2bc$

$$\text{Cos}\beta = \frac{\{D_{(BM)t0}\}^2 + \{D_{(AB)t0}\}^2 - \{D_{(AM)t0}\}^2}{2\,D_{(BM)t0}\,D_{(AB)t0}} \qquad (5)$$

By using the distance between AB, AM and BM and the angles α and β a triangle is plotted to estimate the location of M (Loc M) at time t_0 by using ΔABM as shown in figure 3.

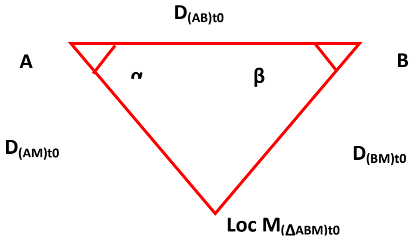


Fig. 3. Mapping of M by using distances AB, AM and BM and the angles α and β

Similarly by using triangles ΔACM and ΔBCM two other locations of M are calculated as shown in figure 4.

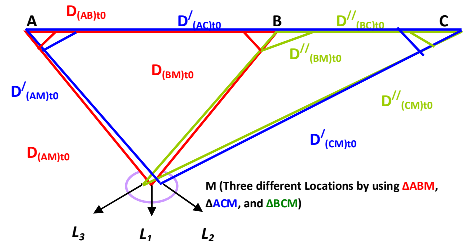


Fig. 4. Location estimation of Mobile by using three triangles, where D is the distance calculated by ΔABM, $D^{/}$ is calculated by ΔACM and $D^{//}$ is calculated ΔBCM

Theoretically calculated locations that we mention above are not accurate because radio waves contain noise. Kalman filter prediction and correction is use with the combination of Bayesian decision theory to minimize errors. As we have three different locations at t_0 we average them to calculate the location of mobile node at this timestamp.

4.2 Step II: Filtering (X)

Filtering of the estimated location is done by calculating average and variance by using different timestamps. In current scenario we are considering four timestamp.

$$\boldsymbol{Loc\ M_{to} = \frac{Loc\ M_{(\Delta ABM)t0}+Loc\ M_{(\Delta ACM)t0}+Loc\ M_{(\Delta BCM)t0}}{3}} \quad (6)$$

Similarly the average at t_1,t_2 and t_3 are calculated .

$$\boldsymbol{Loc\ M_{t1} = \frac{Loc\ M_{(\Delta ABM)t1}+Loc\ M_{(\Delta ACM)t1}+Loc\ M_{(\Delta BCM)t1}}{3}} \quad (7)$$

$$\boldsymbol{Loc\ M_{t2} = \frac{Loc\ M_{(\Delta ABM)t2}+Loc\ M_{(\Delta ACM)t2}+Loc\ M_{(\Delta BCM)t2}}{3}} \quad (8)$$

$$\boldsymbol{Loc\ M_{t3} = \frac{Loc\ M_{(\Delta ABM)t3}+Loc\ M_{(\Delta ACM)t3}+Loc\ M_{(\Delta BCM)t3}}{3}} \quad (9)$$

Refer to Kalman filter [7], to calculate variation in location at different timestamps the variance computing formula is used.

$$\boxed{\sigma^{2} = \frac{\sum (X - \overline{X})^{2}}{N} \quad [7]}$$

$$\sigma^{2}_{\mathbf{(Loc\ M)\ t_0}} = \frac{\mathbf{\Sigma\ (L_{n(t_0)} - Loc\ M_{t_0})^2}}{\mathbf{N}} \quad (10)$$

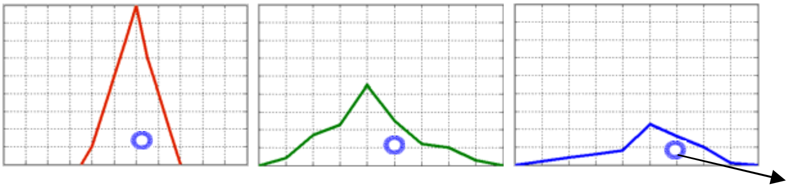
Where n= 1, 2, 3 and N= 3

Similarly variation of location can be recorded by calculating variance at $t_1, t_2 \ldots t_k$.

$$\sigma^2_{(\mathrm{Loc\,M})\,t_1} = \frac{\Sigma\ (L_{n(t_1)} - \mathrm{Loc\,M}_{t_1})^2}{N} \tag{11}$$

$$\sigma^2_{(\mathrm{Loc\,M})\,t_2} = \frac{\Sigma\ (L_{n(t_2)} - \mathrm{Loc\,M}_{t_2})^2}{N} \tag{12}$$

$$\sigma^2_{(\mathrm{Loc\,M})\,t_3} = \frac{\Sigma\ (L_{n(t_3)} - \mathrm{Loc\,M}_{t_3})^2}{N} \tag{13}$$



Variation recorded from $\sigma^2_{(\mathrm{LocM})t0}$ to $\sigma^2_{(\mathrm{LocM})\,t1}$

Variation recorded from $\sigma^2_{(\mathrm{LocM})t1}$ to $\sigma^2_{(\mathrm{LocM})\,t2}$

Variation recorded from $\sigma^2_{(\mathrm{LocM})t2}$ to $\sigma^2_{(\mathrm{LocM})\,t3}$

Fig. 5. Variation in location of M at t_0 to t_1, t_1 to t_2 and t_2 to t_3

Figure 5 is representing the variations at three different timestamps. Although the location of mobile node is falling inside the Region of Interest (RoI) but still it is only pointing out only the region of high availability and unable to predict the actual position (each box in a region is representing 2 square meters).

4.3 Step III: Prediction (Y)

Overlapping Coverage Area (Ω). It is also possible that at any time t_n mobile node receive signals from less than three BTS (Base Transceiver Station). Overlapping coverage area is considered as condition in Bayesian decision theory for the location estimation. Ω is representing the overlapping coverage area of three BTS.

where $\Omega = (A \cap B \cap C)$, $\Omega_1 = (A \cap B)$, $\Omega_2 = (A \cap C)$, $\Omega_3 = (B \cap C)$

If the location of mobile node is confirm from all of the three BTS then the probability of precision will be higher whereas in case of Ω_1, Ω_2 and Ω_3 probability of precision will be lesser. Figure 6 representing the signal coverage and the overlapping coverage areas.

The probability of selecting the location with the condition of posterior probability Ω will be higher than the posterior probability of Ω_1 and Ω_2. By applying the Bayesian theorem we will get

$$\mathrm{P}\,(\boldsymbol{L_1}|\,\Omega) = \frac{\mathrm{P}\,(\Omega\mid \boldsymbol{L_1})\ \mathrm{x}\ \mathrm{P}\,(\boldsymbol{L_1})}{\mathrm{P}\,(\Omega\mid \mathrm{L1})\ \mathrm{x}\ \mathrm{P}\,(\mathrm{L1}) + \mathrm{P}\,(\Omega\mid\sim \mathrm{L1})\ \mathrm{x}\ \mathrm{P}\,(\sim\boldsymbol{L_1})} \qquad (14)$$

$$\mathrm{P}\,(\boldsymbol{L_2}|\,\Omega) = \frac{\mathrm{P}\,(\Omega\mid \boldsymbol{L_2})\ \mathrm{x}\ \mathrm{P}\,(\boldsymbol{L_2})}{\mathrm{P}\,(\Omega\mid \boldsymbol{L_2})\ \mathrm{x}\ \mathrm{P}\,(\boldsymbol{L_2}) + \mathrm{P}\,(\Omega\mid\sim \boldsymbol{L_2})\ \mathrm{x}\ \mathrm{P}\,(\sim\boldsymbol{L_2})} \qquad (15)$$

$$\mathrm{P}\,(\boldsymbol{L_3}|\,\Omega) = \frac{\mathrm{P}\,(\Omega\mid \boldsymbol{L_3})\ \mathrm{x}\ \mathrm{P}\,(\boldsymbol{L_3})}{\mathrm{P}\,(\Omega\mid \mathrm{L3})\ \mathrm{x}\ \mathrm{P}\,(\mathrm{L3}) + \mathrm{P}\,(\Omega\mid\sim \mathrm{L3})\ \mathrm{x}\ \mathrm{P}\,(\sim\boldsymbol{L_3})} \qquad (16)$$

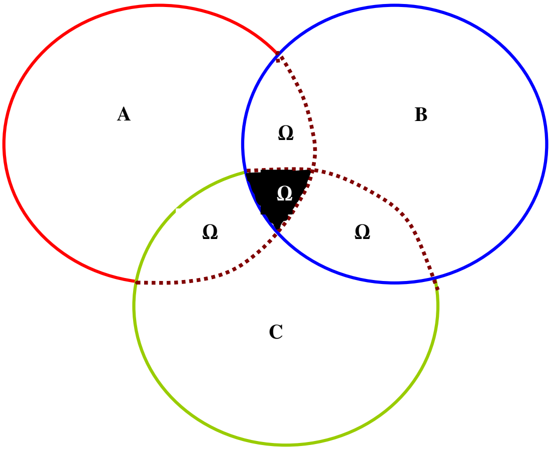


Fig. 6. Footprint of BTS A, B and C, showing the overlapping area

For a given Ω if any estimated location is falling outside the Ω then the chances of precision will be lesser. It will be considered only if all the locations falling outside the Ω. This rule will minimize the average probability of error. By applying the Bayesian decision theory on equation 14, 15 and 16 we will get the following equation.

$$\mathrm{P}\,(\boldsymbol{L}|\,\Omega) = \max\,[\mathrm{P}\,(\boldsymbol{L_1}|\,\Omega),\ \mathrm{P}\,(\boldsymbol{L_2}|\,\Omega),\ \mathrm{P}\,(\boldsymbol{L_3}|\,\Omega)] \qquad (17)$$

4.4 Step IV: Fusion (Z)

By combining the variances [7] of step II we will get combine variation area for the location estimation. This overlapping variant area is considered as a most powerful candidate for the location of mobile node. Simulation results in figure 7 are explaining and justifying the scenario.

$$1/\sigma^{2}=1/\sigma^{2(\mathrm{LocM})\ \mathrm{t0}}+1/\sigma^{2(\mathrm{Loc\ M})\ \mathrm{t1}}+1/\sigma^{2(\mathrm{Loc\ M})\ \mathrm{t2}}+1/\sigma^{2(\mathrm{Loc\ M})\ \mathrm{t3}} \qquad (18)$$

$$\text{Let } \varsigma = 1/\sigma^{2}$$

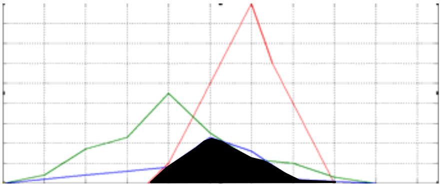


Fig. 7. Overlapping variant area by combining variences

Figure 7 is the combine variance representation in Matlab, which is combining the overlapping variance area shown in figure 5. The shaded portion is representing the combine variances. We use fusion here to minimize the region of interest of step II. Although fusion is helpful to minimize the Region of Interest (RoI), but does not support to pinpoint actual position. Based on the above four steps (estimation:W, filtering:X, prediction:Y and fusion:Z) predicted and corrected location estimation algorithm is proposed base on kalman filter recursive approach of prediction and correction. In our algorithm prediction is based on W, X, Y and Z whereas correction is obtained by combining their results.

PCLEA (Predicted and Corrected Location Estimation Algorithm)

```
1.  estimation: W                         ┐
2.  filtering: X                          ├ Prediction
3.  prediction: Y                         ┘
4.  fusion: Z
5.  if  P ($L_1$| $\Omega$) →S              ┐
6.                  select: $L_1$           │
7.  goto 15                               │
8.  else if P ($L_2$| $\Omega$)→ S          │
9.                  select: $L_2$           ├ Correction
10. goto 15                               │
11. else if P ($L_3$| $\Omega$)→ S          │
12.                 select: $L_3$           │
13. goto 15                               ┘
14. else goto 1
15.             timestamp:                  Recursion
16. estimation: W                         ┐
17. filtering: X                          │
18. prediction: Y                         ├ Prediction
19. fusion: Z                             │
20. goto 5                                ┘
```

Above algorithm is based on the prediction and the correction rule of the Kalman filter. As shown in figure 7 the overlapping area of the most variant values of locations L_1, L_2, L_3 and in figure 6 "Ω" is the receive signal overlapping area for same L_1, L_2, L_3 locations. Line 5, 8, 11 of algorithm are analyzing either L_1, L_2 and L_3 are falling in the variant area, If selected then that location will be the most precise value otherwise as mention in line 16, 17, 18 and 19 after a described timestamp it will start the kalman filter cycle of prediction and correction. Figure 8 is representing the tracking architecture of above algorithm.

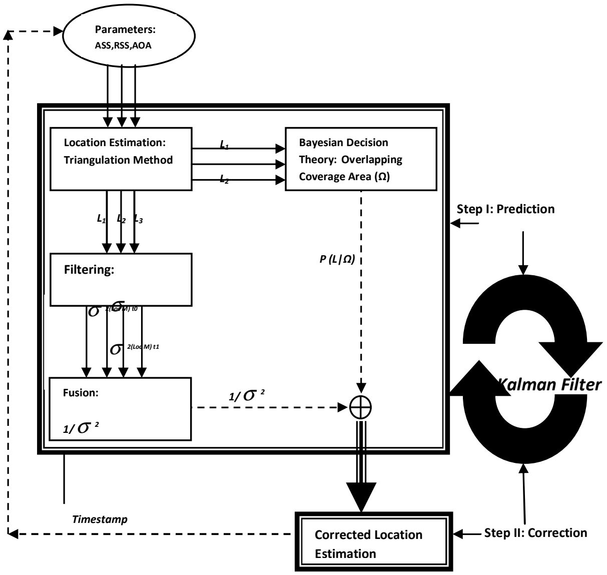


Fig. 8. Tracking architecture of PCLEA

5 Results and Discussion

Figure 9 is base on estimation which is using triangulation method for location estimation at time t_0, t_1, t_2 and t_3. Circle is representing a mobile node. Recorded variation is representing in figure 10, while prediction step is simulated by using P (L| Ω) P (L| Ω_1) P (L| Ω_2) and P (L| Ω_3), as shown in results in figure 11 that estimated location by using Ω is more precise as compare to Ω_1, Ω_2 and Ω_3. Figure 12 is representing fusion with the combine variance approach. Shaded area is the overlapping variance area which is the most appropriate are for location. Figure 13 is representing the comparison between each step (estimation, filtering, prediction, fusion) results with PCLEA. As shown in figure 13, estimation has maximum error rate which is 9.5 to 10 M. Average of the calculated variance area showing error in distance at the maximum of 7M. Based on estimation results prediction is done by using Bayesian decision theory, which is showing huge improvement in location estimation which is with the error rate of 2M. As also shown in figure 11 Ω_1, Ω_2 and Ω_3 results provide less precision in location estimation which is 3.5 to 3M (figure 13). Figure 13 is representing overall comparison of estimation, filtering, prediction and fusion with the PCLEA. Note that the error of fusion and Ω_1, Ω_2 and Ω_3 is almost same (2.6 M and 3 M - 3.5M respectively) but if we combine the fusion results with prediction (as in PCLEA) then the estimated location of mobile node is almost approaching the actual position, which is with the error rate of 0.6M. PCLEA is actually combining benefits of Kalman filter and the Bayesian decision theory for location estimation.

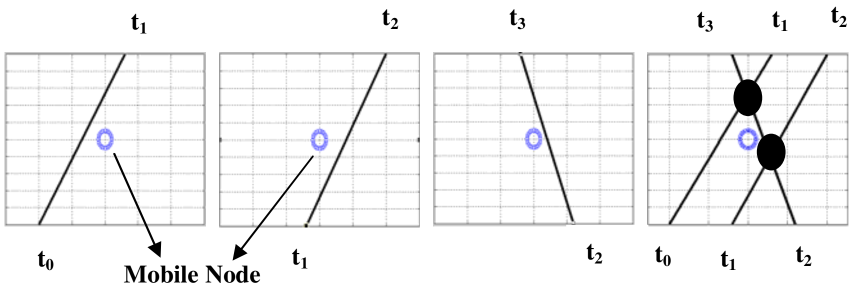


Fig. 9. Estimation

Mobile node is represented by circle, whereas t_0, t_1, t_2 and t_3 are representing the estimated location by using triangulation in figure 9. The error rate is almost 10M in this case. By combining trend lines of all four points we get intersection which is still unable to achieve precision (each box is representing 2 square meters).

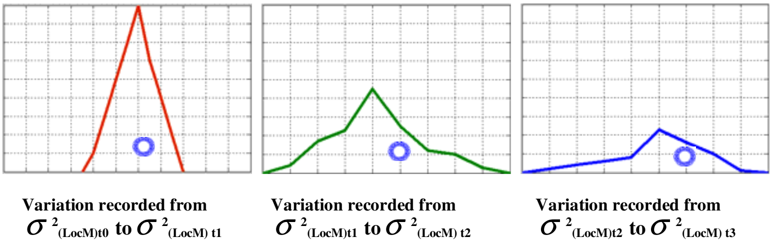


Fig. 10. Filtering

By using the variance we are able to minimize the area in figure 10. By averaging the selected area we still face the error rate of almost 7M (figure 13).

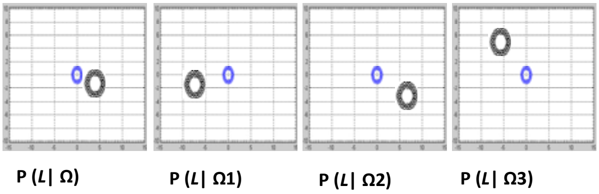


Fig. 11. Prediction

We apply Bayesian decision theory with the condition of overlapping area Ω for prediction. Simulation results shows if the mobile node is receiving signals from all three antennas (i.e. constructing three triangles), then the error rate will be almost 2M. In case of Ω_1, Ω_2 and Ω_3 it may increase by 3 to 3.5M.

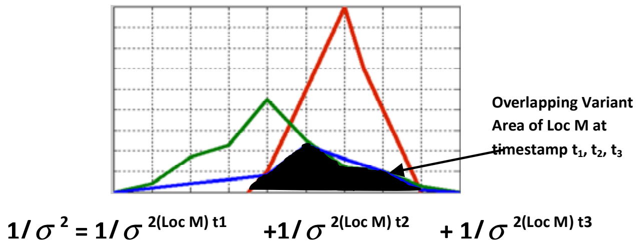


Fig. 12. Fusion

Figure 12 is representing fusion. The shaded portion is representing the combine variances. Fusion is use to minimize the Region of Interest of filtering step.

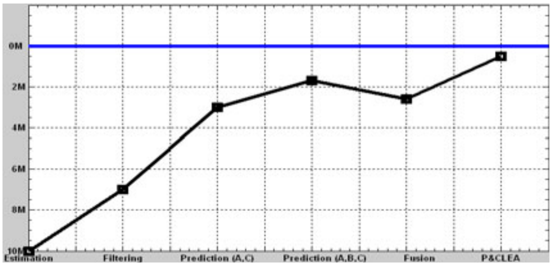


Fig. 13. Comparison of Predicted & Corrected Location Estimation Algorithm (PCLEA) with single step calculation

Figure 13 is representing the distance error recoded at estimation, filtering, prediction at Ω_1, Ω_2 and Ω_3, prediction at Ω, fusion and PCLEA. The error of PCLEA is up to 0.6M. If we compare our region of Interest (RoI) with Region of Confidence (RoC) [12], it shows that RoI has more distance error (2.6M), whereas RoC claimed 1.6M error. The distance error of PCLEA is less with RoC [12] and Selective Fusion Location Estimation (SELFLOC) [12].

In terms of computational complexity the PCLEA is heavier as it is combining four different algorithms; also it is using recursive approach to reach the minimum distance error. It might not produce better results where computational cost is more important like WSN.

6 Conclusion

This research is focusing the prediction and correction rules of kalman filtering in a recursive way to estimate the precise location of a cellular node. In prediction we combine the triangulation, means, variances, Bayesian decision theory and kalman filter fusion whereas for correction we combine results of Bayesian with Kalman filter fusion. Timestamp is use before the recursive iterations. Our results shows that PCLEA is producing less distance error as compare to old location estimation techniques. Infect PCLEA is combining benefits of triangulation, RoI, Bayesian decision theory and the fusion with the kalman filtering.

References

1. EU Institutions Press Release. Commission Pushes for Rapid Deployment of Location Enhanced 112 Emergency Services, DN: IP/03/1122, Brussels (2003)
2. Khalaf-Allah, M.: A Novel GPS-free Method for Mobile Unit Global Positioning in Outdoor Wireless Environments. Wireless Personal Communications Journal 44(3) (February 2008)
3. Gezici, S.: A Survey on Wireless Position Estimation. Wireless Personal Communications: An International Journal 44(3) (February 2008) ISSN: 0929-6212
4. Gustafsson, F., Gunnarsson, F.: Mobile positioning using wireless networks. IEEE Signal Processing Magazine 22(4), 41–53 (2005)
5. Weiss, A.J.: Direct position determination of narrowband radio frequency transmitters. IEEE Signal Processing Letters 11(5), 513–516 (2004)
6. Qi, Y., Kobayashi, H., Suda, H.: Analysis of wireless geolocation in a non-line-of-sight environment. IEEE Transactions on Wireless Communications 5(3), 672–681 (2006)
7. Bishop, G., Welch, G.: An Introduction to the Kalman Filter. in NMQQ4!9O)RSST, Course 8 (2001), http://www.cs.unc.edu/~tracker/ref/s2001/kalman/
8. Karthick, N., Prashanth, K., Venkatraman, K., Nanmaran, A., Naren, J.: Location Estimation Using RSSI and Application of Extended Kalman Filter in Wireless Sensor Networks. In: Proceedings of the 2009 International Conference on Advanced Computer Control 2009, January 22 - 24 (2009) ISBN: 978-0-7695-3516-6
9. Larsen, T.D., Bak, M., Andersen, N.A., Ravn, O.: Location Estimation for an Autonomously Guided Vehicle using an Augmented Kalman Filter to Autocalibrate the Odometry. In: First International Conference on Multisource-Multisensor Information Fusion (1998)
10. Le, B.L., Ahmed, K., Tsuji, H.: Mobile location estimator with NLOS mitigation using Kalman filtering. In: Proc. IEEE Wireless Communications and Networking (WCNC 2003), New Orleans, LA, vol. 3, pp. 1969–1973 (March 2003)
11. Madigan, D.E., Martin, E., Ju, R.P., Krishnan, W.-H., Krishnakumar, P., A.S.: Bayesian indoor positioning systems. In: 24th Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2005, vol. 2, pp. 1217–1227 (March 2005)
12. Youngjune Gwon Jain, R., Kawahara, T.: Robust indoor location estimation of stationary and mobile users. In: Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM 2004, March 7-11, vol. 2, pp. 1032–1043 (March 2004) ISSN: 0743-166X ISBN: 0-7803-8355-9
13. Widrow, B., Steams, S.: Adaptive Signal Processing. Prentice Hall, Upper Saddle River (1985)
14. Seshadri, V., Zaruba, G.V., Huber, M.: A Bayesian sampling approach to in-door localization of wireless devices using received signal strength indication. In: Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005, March 8-12, pp. 75–84 (2005) ISBN: 0-7695-2299-8
15. Tonteri, T.: M.Sc Thesis A Statistical Modeling Approach to Location Estimation Department of Computer Science University of Helsinki (May 25, 2001)
16. Muhammad, A., Mazliham, M.S., Shahrulniza, M., Amir, M.: Posterior Probabilities based Location Estimation (P^2LE) Algorithm for Locating a Mobile Node in a Disaster Area M. In: MULTICONF 2009, July 13–16, American Mathematical Society, Orlando (2009)
17. Muhammad, A., Mazliham, M.S., Shahrulniza, M.: Power Management of Portable Devices by Using Clutter Based Information. IJCSNS, International Journal of Computer Science and Network Security 9(4), 237–244 (2009) ISSN: 1738-7906

Enhancing Customer Privacy for Commercial Continuous Location-Based Services

Jens Bertram, Carsten Kleiner, and David Zhang*

Fachhochschule Hannover, Fakultät IV
Ricklinger Stadtweg 120
30459 Hannover
Germany
jens.bertram1@stud.fh-hannover.de, ckleiner@acm.org,
david.zhang@fh-hannover.de

Abstract. The likelihood of consumers to use commercial location-based services significantly depends on their perception of privacy protection by the service provider. In this paper we discuss existing privacy-enhancing architectures for LBS and argue that they are either not applicable or insufficient for services requiring continuous location queries. In order to offer such services providers often prefer to refrain from storing fine-grained location information of their customers. Instead some form of data aggregation on the mobile device is used and only aggregated information is released to the service provider upon approval of the customer. This leads to a rather loose integration of the mobile device into the backend process. We explain our concept for such an enhanced architecture and discuss some implementation aspects. The work has been motivated by a specific application scenario in an insurance context for which we are currently developing a prototype.

Keywords: Enterprise Location-based Service, Privacy, Mobile Device, Continuous Query, Navigation, Data Aggregation.

1 Introduction

Location-Based Services (LBS) have been considered a very important application setting in an increasingly mobile society for a couple of years. In recent years most of those services are rather simple consumer-oriented services being offered for free or at a small charge (e.g. find the nearest restaurant of a specific type from a given location). Such services typically do not require a previous registration of the consumer with the provider. This will most probably change in the near future because there are also very interesting and commercially promising business applications based on LBS. These can be pure business applications such as automated fleet management which are already in place in some companies. But it

* The participants in the research project *DaSimoD* leading to the results presented in this paper have been funded by a grant of the *German Federal Ministry of Education and Research* under grant no. FKZ17N5309.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 326–337, 2010.

might also be business to consumer applications where both parties gain advantages by the service. An example is the Pay-As-You-Drive scenario for a car insurance company which has been used as motivation for this paper. It is described in more detail in section 2.

LBS are of particular interest in the context of devices such as mobile phones or PDAs since modern devices typically provide some kind of position technology on the device. As many LBS are used based on the current position of the client and the mobile device is carried by the consumer most of the time, it is very simple to use LBS from a mobile device.

Unfortunately in this setting large privacy concerns arise; most people would not want a service provider (maybe even unknown to them) to be able to record their position at any time when they consume the service. By combining or aggregating such information potentially from different services used by the same device detailed motion profiles could be assembled. Note that even without a registration the identity of the client might be revealed to the service when submitting a request (e.g. by IMEI, phone number, MAC address). In addition even when LBS are used anonymously or with pseudonym it is possible to determine the identity of the user through the indirect location privacy problem. This is even more true when using passive position technologies.

There has already been some research on how to offer LBS privacy-friendly. We will review the existing suggestions in section 3. Each has its individual strengths and weaknesses. In this paper we introduce an example business application that relies on continuous location information. For this application scenario which will be described in section 2 we will argue that none of the existing approaches is both applicable and sufficient. Consequently we present an extended privacy-friendly architecture for our and similar application scenarios in section 4. Neither our nor any of the existing approaches offers perfect privacy at an acceptable computational cost; we explain why in sections 3 and 4. In section 5 we briefly discuss some implementation aspects for our architecture, before we conclude in section 6 with a summary and some ideas for future work.

2 Application Scenario “Pay-As-You-Drive”

Our research has been motivated by a specific application scenario for mobile LBS. This scenario leverages mobile devices, location information and LBS to allow for a “Pay-As-You-Drive” car insurance model. It refers to a specific tariff option for car insurance. The insurance company offers a reduced tariff for drivers which comply with certain rules. An example might be young drivers that do not drive at night where the risk of accidents for them is particularly high. Or drivers that claim to always stick to the posted speed limits might be reimbursed.

More generally speaking “Pay-As-You-Drive” is a car insurance model, where the insurance premium depends on the driving behavior of the policyholder (e.g. type of road, time, speed, break of speed limits, driven distance, etc.). Based on this information, the insurance company may calculate the risk more accurately, which could result in flexible and potentially lower costs for the policyholder.

Obviously this application has potential to put the clients' privacy at risk. Our goal is to find a solution that protects the user's privacy, so that he will not reveal more about himself than he has agreed to by contract. This means that only data needed for the risk calculation is collected and aggregated in a way that is sufficient to fulfill the requirements as provided in the contract. Additionally the system architecture should prevent the ability to create profiles of movement of the different users. For example it is not necessary to know where and when exactly a user/car was, it might be sufficient to know the type of road, distance driven and average speed. Note that we consider online access to road information mandatory due to space constraints on the mobile device combined with the requirement for a high degree of actuality of data and metadata.

On the other hand the information assembled in such services is so extremely sensitive, that not even insurance companies want to have detailed information about the client on their servers in order to prevent being forced to reveal it (e.g. to governmental parties).

We have designed a privacy-friendly system architecture for offering Pay-As-You-Drive insurance contracts. Note that this task is not restricted to use the classical point-based LBS services internally. But it is also an option to use trajectories or similar extended and/or aggregated geographic data as parameters for the services in the architecture. Thus the architecture may be used for services that go far beyond classical LBS as well. Some different possible options will be discussed in section 4.

3 Related Work

The increasing use of LBS has created new privacy risks for users. Using LBS involves confiding personal data like current location to the location-based service provider (LBSP). This data may convey personal details about the client. Even when using anonymous LBSs (a service that does not require users to convey their identity) the identity of the client may be revealed from the location data by the indirect location privacy problem. This has been nicely explained in [9]. A lot of research has been done in this area to handle privacy problems. Among the methods to protect the privacy are location k-anonymity [5], false dummies [1], false locations [2] and private information retrieval (PIR) [3, 4].

The k-anonymity technique which is described in [5] protects privacy by providing anonymity for clients based on trusted third party architecture. This approach implies that a client uses a service in an anonymous way i.e. he does not send his real identity and may also hide his network address by technologies like onion routing. Additionally the trusted third party provides k-anonymity (i.e. an individual request may not be distinguished from at least k-1 other requests) and so ensures that the client cannot be practically identified based on position data. To achieve this, precision of location information is reduced both in spatial and temporal dimensions. The degree of reduction is based on statistics and is chosen in a way that the location information sent to the service may have come from k different entities, where k may be chosen so that it is not practical to identify the actual client. This approach works well for snapshot queries. But if a client continuously uses a LBS, anonymity may be reduced by maximum movement bounds. Moreover in our use case we require precise location information in order to be able to use the service in a meaningful manner. Approximate information is not sufficient for navigation services.

Temporal cloaking in contrast is possible to a certain degree. For example the time of the day does not have to be precise to minutes or maybe even hours. Because only a weekly or monthly report is required, the time of the day may be important but not the exact date. Details depend on the particular service and contract.

The idea how to protect location privacy with false dummies is to hide the true location among a number of false locations. Whenever a client uses a LBS he will not only request this service for his true location but also for some dummy locations. Because the service cannot distinguish the true from the false locations, privacy is protected. This works well for snapshot queries. If on the other hand a LBS is used frequently, true locations will form a route and dummies could be easily identified. Therefore a smart algorithm has to be used to generate dummies in this case. In [1] it is suggested to remember the last dummies sent to a service and to generate new dummies in the neighborhood. This will make it more difficult to identify dummy requests, because they will also form a route. In our use case this will not be sufficient because we track cars driving on roads. The true route will be easily identified because it is the only one following a road. The dummy generating algorithm would need to be improved to generate dummies in the neighborhood that are on a road as well. This will require the client to have a complete map available. To further improve the quality of dummy requests in this use case the algorithm may also take constraints like one way streets and speed limits into account. In our use case clients do not have road maps available and so dummy requests would not be generated in a smart way. That is why dummy requests cannot be applied in this case.

Another privacy protecting technique is SpaceTwist [6]. It uses false locations for nearest neighbor queries sent to the LBS. The client specifies the false location called an anchor. Then he queries the service for the nearest neighbors of this anchor. The service will return points of interest in ascending distance of the anchor. The client terminates the request when the answer covers the area around its position sufficiently. In our use case the service would have to return fragments of roads with consistent metadata until the client identifies a fragment which includes its position. SpaceTwist has only been studied for snapshot queries, not for continuous queries. For continuous queries the anchors may form an approximate route for the movement of the client. This again would reduce location privacy.

In [3] a framework based on private information retrieval (PIR) is presented. PIR is based on the Quadratic Residuosity Assumption which states that it is computationally very expensive to find the quadric residues of a large number N = q1 * q2 where q1 and q2 are prime. This framework requires that the database is indexed appropriately. A request to a LBS using PIR does not contain spatial information. The points of interest are retrieved based on an object index. Therefore location privacy can be guaranteed here. Unfortunately a PIR request is a quite costly operation both in terms of computational complexity on the server side as well as regarding message sizes. Our use case implies a large number of clients accessing the LBS in high frequency. In [4] a scalable approach leveraging PIR called SPIRAL is described. This framework is based on trusted third party architecture and blinds the LBS so it will not know which objects have been sent to the client. While this might remedy the complexity issues on server side it does not reduce messages sizes significantly. The latter is not acceptable for services on mobile devices.

In summary, for LBS used continually, it becomes even more difficult to protect privacy. All of the single locations form a route. False dummies may then be easily identified. Also false locations will form an approximate route. Because of a high frequency of requests PIR may not be applicable. And k-anonymity achieved by cloaking locations to regions will reveal maximum movement bounds. Apparently there is no perfect solution to protect privacy for a continuous LBS application like Pay-As-You-Drive. In the following chapter we will propose a system architecture for this application which respects the users' privacy to a high degree and is a good tradeoff between privacy and performance.

4 Privacy-Friendly System Architecture

In this chapter we present and discuss the architecture we designed for a Pay-As-You-Drive application. Because until now there is no ultimate solution to protect location privacy for continuous queries we propose a somewhat pragmatic solution. In the first section we present and discuss the general architecture of commercial applications for which we propose privacy enhancements. The Pay-As-You-Drive application fits well in this architecture. The second section focuses on measures to provide practical location privacy by means of the Pay-As-You-Drive example. These measures can be transferred to other applications that fit in the general architecture.

4.1 General Architecture

The architecture described in this section mainly consists of three different and independent systems (cf. Fig.1). A mobile device (e.g. a smart phone) provides raw GPS position data like latitude, longitude, altitude and speed. It integrates position data with additional meta information and aggregates it to create a report for a business partner. This might be for example a report about driving behavior which is sent to an insurance company. The required meta information related to the position like road-type and speed-limit are provided by an external service provider (LBS in Fig. 1). They are periodically requested by the mobile device. The meta information is typically stored together with geometries in a GIS/Spatial Database, which allows fast detection of a specific road from a given position based on road geometries. The third part of this architecture is the business partner system, which receives the reports. For the Pay-As-You-Drive example this is the insurance system which stores the user-reports and calculates the risk and insurance premium accordingly. Note that location privacy against the business partner is considered sufficient by providing only aggregated information in weekly or monthly reports. This assumption is reasonable in all practical cases. Thus the remaining privacy issues are against the LBS provider. Since the most relevant part concerning the users' privacy in the Pay-As-You-Drive concept is related to the interaction between mobile device and LBS, the following part focuses on this issue.

To be able to create an accurate report, the mobile device will continuously request the LBS in short time periods (about seconds). This behavior is similar to a tracking

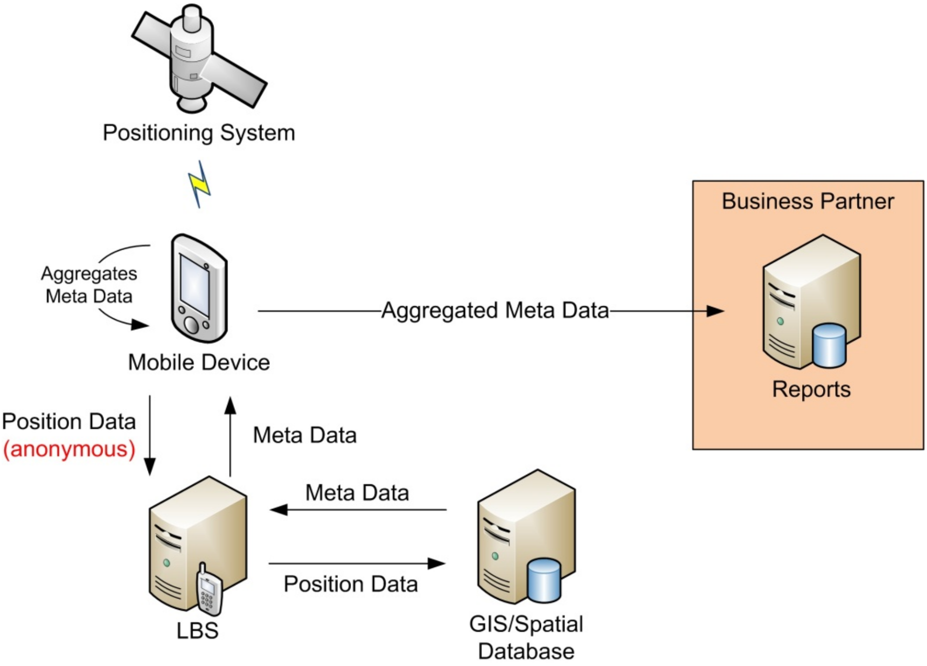


Fig. 1. System Architecture

system, where a car's position is periodically reported to a monitoring system to be able to locate a specific vehicle. It is this side effect we want to prevent; the LBS shall not be able to locate or track a specific vehicle/user. Please note, that the results of the requests are not needed immediately; they are aggregated to a report which is only evaluated in regular time intervals.

By using an anonymous connection to the LBS, which implies that no authentication is required and the service provided by the LBS is free of charge, an incoming request gives no information about the users' identity. Otherwise, if the service is not free of charge, authentication is necessary but it would be sufficient if an authentication mechanism is used that authenticates a user as member of a group and not providing any further information about his identity. But using an anonymous connection to the LBS is not enough. The user may still be identified by the indirect privacy problem. Because of the high frequency of client requests they will form a route which might help to identify the user, for example the regular route from home to work place. Once a client is identified, the route can be used to find out where else that person went.

After receiving the meta information according to the previously sent position the client (mobile device) aggregates the data. As for the Pay-As-You-Drive application the report describes the users driving behavior by providing information about several parameters such as distance driven on a specific road type, violation of speed limits and average accident rate. Details depend on the specific contract. The reports are created for a certain time period and are periodically submitted to the insurance system.

While the connection to the LBS is made anonymously, there still exists an indirect location privacy problem. As the LBS receives all position data of the clients it could store the data, analyze it and search for certain patterns in the data set to identify individuals. In the next section we present suggestions for improving on this situation.

4.2 Considerations to Improve User Privacy

The architecture presented in chapter 4.1 has one major drawback: it continually sends its position to the LBS and thus it is vulnerable to the indirect location privacy problem. In this section we present an improved concept protecting location privacy based on the same simple client-server architecture: a client communicates with an anonymous location-based service. The improvements proposed in this section are made by means of the Pay-As-You-Drive example, but can be transferred to other applications which fit in the general architecture described in the previous section.

The Pay-As-You-Drive application depends on accurate location information to measure the driving profile of the policy holder. But the report will only be assembled periodically i.e. at the end of the month. This fact can be used to increase location privacy. The basic idea is to remove temporal correlation of the requests. The location data send to the LBS is required not to contain any temporal attributes of the specified position. To achieve this, the request will not specify the time at which the client was at the location nor the current speed or direction and it will also not be sent immediately. Because the report only needs to be assembled e.g. monthly sending can be deferred for hours, days or even weeks. Also the meta data for locations will not be retrieved in the sequence they were recorded but in random sequence.

The result of this method is that it becomes difficult to impossible to identify routes from the locations sent to the server and thus it is difficult to identify the person that sent the request. If there was only one client using this anonymous LBS in a city, this method would not protect his anonymity against the indirect privacy problem. If the LBSP would display all the requests he received on a map, he would see where the person went and even how often. From this knowledge he could derive the identity of that client. But it would not be possible to determine at what time the client was at a specific place or in which direction he moved from there. If there are two or even more clients in that city, it will become more difficult to identify them. Also it becomes difficult to determine which client went to a specific place that is not already known to be related with one of the clients. The degree of anonymity increases with the number of clients moving around in a common area.

Until the LBS is queried for the meta data for a specific location the client device will have to store the location data. The local storage must retain the sequence of locations and may also contain a timestamp for each entry. The client will choose randomly from stored location entries to issue a request. After it received the response from the LBS for some location entries that have been recorded in sequence, it can aggregate metrics into the corresponding report. Now the inner entries of the location sequence are not needed anymore and are removed from local storage. The first and last entries of a sequence will still be needed to compute metrics for the neighboring location sequences.

Because locations may be stored for a longer period of time this allows for some optimizations to reduce network traffic and to increase privacy even further. When the client asks the LBS for metadata for a specific location the service could include the geometry of the area associated with this metadata. Then the client does not have to query for all the neighboring locations which would yield the same result. They can now all be answered by a single request. These may include locations that have been recorded in the area over a couple of days. Thus every request will resolve coherent sequences of location entries that can be aggregated and removed from local storage.

The next step is to improve the algorithm choosing the next location entry from the local storage to be evaluated by the LBS. To minimize the overall number of requests the algorithm could take the location data of the entries into account. It could favor to choose locations that are in a less frequently used area. Thus location entries for frequently visited areas would accumulate in local storage and be answered with a single request. The effect is that frequently visited areas are not frequently sent to the LBS. Obviously these measures reduce the number of requests sent to the service and therefore also reduce the data that LBS could collect about the client.

Our concept uses temporal cloaking and is applicable to Pay-As-You-Drive, because the client application does not need short-term answers. In contrast to the k-anonymity concept (e.g. in [5]) it is not based on a trusted third party architecture. On one hand this simplifies the overall architecture. On the other hand it is not possible to ensure a given degree of anonymity. The degree of anonymity depends on the number of clients using the service in a common area. However, it is possible to derive this number from the number of policyholders within a common area.

Note that in order to ensure a certain degree of anonymity a trusted third party may also be employed in our architecture. In this sense the approaches are orthogonal and may be combined if desired. But the practical improvement of privacy might be rather small, especially in urban areas, when adding k-anonymity to our architecture. Thus the tradeoff between simplicity of the architecture and privacy increase would favor using our approach purely. The optimizations described above have the potential to significantly reduce the number of requests and so cloaking the movement of the client. Altogether this is a practical approach to implement Pay-As-You-Drive and other similarly structured commercial services with a reasonable degree of privacy.

5 Implementation Considerations

The previously described architecture relies on a continuous communication of the mobile device and the LBS. This is necessary to receive additional information based on the current location and to be able to record the policyholders driving behavior.

Timing of Messages

The accuracy of the aggregated data depends on different aspects. As the mobile device itself does not store location related information like the geometry of a road, the driven distance between two different positions has to be interpolated, which might not represent the actual driven distance, if the two positions are far apart. This means the higher the resolution of the route (positions/time) is, the more accurate is the aggregated data. As positioning itself and data aggregation run in constant time, the resolution of the route mainly depends on the time between position data is sent and the service response is received. This in turn depends on the structure of data, the network quality and bandwidth and the protocol used for message exchange.

The data sent by the mobile device to the LBS mainly consists of the values of latitude and longitude. Optionally the altitude and velocity could be included to distinguish nearby roads like parallel running roads or bridges. Based on this position data the service responds with related meta data. The report sent from the mobile device to the insurance system contains characteristics about the driving behavior.

Today's smart phones and mobile devices provide fast network connections and are designed for an "always-online" usage. But there are still areas and situations where only low bandwidth or even no network service is available. As a fallback procedure in this case all position data combined with a timestamp could be stored in local storage on the device and send later (without timestamp), when the network is available again or the bandwidth increases. In case of the improved concept described in section 4.2 this procedure is an implicit part of the concept already.

Message Protocol

The processing time and so the resolution of the route also depends on the protocol used for the message exchange between the different systems. There are several requirements affecting the choice of a protocol like data structures, implementation effort, reusability and security. In case of Pay-As-You-Drive the previously described data is structured in a static and simple way without complex and varying elements and attributes, which is well suited for the use with different protocols. We will now describe different protocols for the data exchange and their advantages and disadvantages in the Pay-As-You-Drive or similar scenario.

Based on the statically structured data a binary format, e.g. an ordered set of key-value pairs, or a specifically structured XML format could be used to describe data. The main advantage of the binary format is the reduced data volume and the faster processing speed for parsing the message, but it requires a specific and matching implementation on both client and server side. Both, the binary and an XML based format could be sent through a socket based connection, without the need of establishing a new connection for each piece of position data, which decreases the transmission time. As a drawback this approach requires a specific implementation and further network configuration, e.g. firewalls, which reduces the reusability of the service for future applications. Also security aspects like encryption and authorization are not provided automatically and would need to be implemented within both the client and service application.

Another approach for messaging and data exchange is the use of Web Service technologies like REpresentational State Transfer (REST) [11] and SOAP. A conceptual comparison of REST and SOAP is e.g. given in [7].

REST uses a HTTP connection for data transmission and the service and service method is identified by an URI in the HTTP header. As an improvement the use of a HTTP connection does neither require a specific interface implementation nor a specific network configuration since HTTP traffic is typically allowed in most networks. Most mobile device platforms will provide an API for HTTP connections and thus support RESTful Web Services directly. The message payload format can be chosen as desired in this case, for example the previously described binary format or an XML based format could be used. An advantage of the use of XML is that message elements are reusable and customizable. Thus services can be reused for different purposes, for example by providing a list of points of interest (POI) for a given location or other location based services. As with sockets the advantage of a binary format is its smaller data volume which might help for mobile devices with rather thin bandwidth. It may be used for REST with the same disadvantage of reduced reusability as before. Regardless of the format the data volume is increased by the HTTP header representing a message overhead when compared with sockets. From a security point of view end-to-end encryption is available for REST by using HTTPS. Any further security mechanisms such as authentication have to be implemented within the application.

Similarly SOAP can be used based on HTTP as transport protocol, but several others like FTP and SMTP are also possible. As the main difference SOAP itself is based on XML and defines a message format containing a header and a body part with the payload described in XML. The binary format could still be used with SOAP as a value of an XML element, but this would lead to the same disadvantage as before. The advantages of the use of XML are the same as described for REST, mainly the reusability of the service for different areas of application. Differing to REST not only the HTTP header but also the whole SOAP message implies large overhead and thus increases the transmission time. The message overhead of SOAP and REST is compared in [8]. As an advantage the WS-Security Specification [12] for SOAP specifies security aspects like encryption and authentication. Unfortunately the support for SOAP is not wide-spread on mobile device platforms nowadays, especially the support of WS-Security is far from perfect.

As an additional improvement an OGC OpenLS Standard [10] compliant service would increase the ability to reuse the service in a large area of applications and also increases the ability to choose a service from a set of equivalent services (Provider Change [9]). This may be combined with either REST or SOAP but would incur an additional significant message overhead.

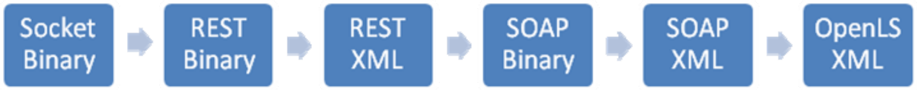


Fig. 2. Increased flexibility and message overhead for service implementation protocols

A remedy to the message overhead issue would be to send a set of positions within a single request and to receive a set of related meta-information in a single response. This results in fewer but bigger messages within a given time. The overhead of HTTP and SOAP header is significantly reduced as for n positions sent at once only a single header is included instead of n headers in n messages. For the privacy aspect n should be chosen small enough to only represent a short part of the route and to decrease the risk of the indirect location privacy problem.

As shown in Fig. 2 the message protocol to be chosen depends on the particular application. There is a classical tradeoff between the desired degree of flexibility of the service on one hand and the message overhead on the other side. In the current situation we would suggest to choose an XML-based REST protocol. This observes current bandwidth restrictions on one hand. But on the other hand it is rather easy to extend it to more flexible protocols in the near future when the bandwidth restrictions are no longer relevant.

6 Conclusion and Future Work

In this paper we have presented a promising commercial application for mobile LBS in the Pay-As-You-Drive application for car insurance. We have explained why privacy issues are of particular interest in this as well as other mobile LBSs. Thereafter it has been shown that none of the existing solutions for observing privacy in LBS is applicable and sufficient in this context (see [13] also). Consequently we have proposed a new architecture which is a good solution from both a user privacy point of view as well as from a pragmatic perspective. Location privacy is enhanced in two stages: on one hand we introduce a party independent from the original service provider; this LBS provider may offer its services in the context of several different commercial applications. In order to improve on this classical trusted-third-party architecture we introduced the second stage: data sent to the LBS provider is anonymous and temporally blurred (cf. section 4.2). Thus the LBS provider does not have to be trusted as it only receives less sensitive data than in the original scenario.

We have suggested several improvements to the basic architecture which may be added to increase privacy as well as efficiency. Finally we have discussed some important implementation issues regarding the timing of messages as well as protocols to be used. These issues are specific to mobile devices and are thus subject to continuous change as technology evolves. Therefore we have presented a general discussion with recommendations in the current situation.

Currently we are working on a prototypical implementation of the basic architecture as presented in sections 4 and 5. The client components are developed for Android based smartphones. We decided to use multi-purpose client devices as opposed to proprietary hardware as in many currently available tracking applications. This is due to the fact that they contain all necessary technology and will soon have a sufficient market share. Thus we don't expect proprietary devices to survive for a long time anymore. In the near future we plan to extend the prototype to include the improvements explained in section 4.2 as well.

Apart from Pay-As-You-Drive there are many other similarly structured commercial LBS (namely ones involving continuous queries) for which the findings in this paper are relevant. Therefore it would be interesting to extend the prototype to different services in this area and evaluate the advantages of flexible protocols further.

From a commercial point of view there has to be a business model for the LBS provider in our architecture. Apart from a classical ad-based business model we also see an open source offering e.g. based on Open Street Map as well as a provider (e.g. the insurance company) sponsored model for LBS providers. In the latter case a LBS provider could offer basic services for many different applications thus remedying the potential influence of a single application provider.

Our approach only works for client based positioning which is not really a restriction nowadays. Nevertheless an important improvement would be to extend the architecture to include the use of a tracking platform which is still widely used and important for certain kinds of enterprise applications. On those platforms positioning of devices is initiated by the tracking platform.

References

1. Kido, H., Yanagisawa, Y., Satoh, T.: An anonymous communication technique using dummies for location-based services. In: Proceedings of International Conference on Pervasive Services, ICPS 2005, pp. 88–97. IEEE, Los Alamitos (2005)
2. Hong, J.I., Landay, J.A.: An architecture for privacy-sensitive ubiquitous computing. In: Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services, MobiSYS 2004, p. 177. ACM Press, New York (2004)
3. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.: Private queries in location based services. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, p. 121. ACM Press, New York (2008)
4. Khoshgozaran, A., Shirani-Mehr, H., Shahabi, C.: SPIRAL: A Scalable Private Information Retrieval Approach to Location Privacy. In: 2008 Ninth International Conference on Mobile Data Management Workshops, MDMW, pp. 55–62. IEEE, Beijing (2008)
5. Gruteser, M., Grunwald, D.: Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In: Proceedings of the 1st International Conference on Mobile systems, Applications and Services, MobiSys 2003. ACM, New York (2003)
6. Yiu, M.L., Jensen, C.S., Huang, X., Lu, H.: SpaceTwist: Managing the Trade-Offs Among Location Privacy, Query Performance, and Query Accuracy in Mobile Services. In: IEEE 24th International Conference on Data Engineering 2008, pp. 366–375. IEEE, Los Alamitos (2008)
7. Pautasso, C., Zimmermann, O., Leymann, F.: Restful web services vs. ”big” web services. In: Proceeding of the 17th International Conference on World Wide Web, WWW 2008, p. 805. ACM Press, New York (2008)
8. Aijaz, F., Ali, S.Z., Chaudhary, M.A., Walke, B.: Enabling High Performance Mobile Web Services Provisioning, pp. 1–6. IEEE, Los Alamitos (2009)
9. Decker, M.: Location Privacy-An Overview. In: 2008 7th International Conference on Mobile Business, pp. 221–230. IEEE, Los Alamitos (2008)
10. Mabrouk, M.: OpenGIS Location Services (OpenLS): Core Services. Open Geospatial Consortium Inc. (2008)
11. Fielding, R.: Architectural Styles and the Design of Network-based Software Architectures. Ph.D:180 Building (2000)
12. WS-Security specification (March 2004), `http://www.oasis-open.org/specs/index.php#wssv1.0`
13. Kulik, L.: Privacy for Real-time Location-based Services. The SIGSPATIAL Special 1(2), 9–14 (2009)

SociCare: Towards a Context Aware Mobile Community Emergency System

Mark Bilandzic, Christian Menkens, Julian Sussmann, Daniel Kleine-Albers, Eva Bittner, Armand Golpaygani, Bernhard Mehl, Jonas Huckestein, and Othmane Khelil

Center for Digital Technology and Management
Technische Universitaet Muenchen
80333 Munich, Germany
{bilandzic,menkens,sussmann,daniel.kleine-albers,eva.bittner, armand.golpaygani,bernhard.mehl,jonas.huckestein,othmane.khelil}@cdtm.de

Abstract. Demographic change and the increase in life expectancy continuously increase the average age in our society. Depending on the physical shape and mental constitution, the elderly need assistance to master regular activities in their everyday lives, as well as urgent help in emergency situations (e.g. in case of a collapse or heart attack). Thereby, time is the most crucial factor. In some emergency cases such as heart attack, people might die if there is no immediate help available. This paper presents the design, architecture and prototype implementation of SociCare, a ubiquitous context aware mobile community emergency system. SociCare is designed to that help emergency call centers leverage and coordinate random voluntary helpers nearby the emergency location. It's user interface enables human call center agents to quickly and easily identify, verify and select voluntary emergency helpers based on their context information. Such information can be for example current availability, distance to the emergency location and general skills and ability to provide first aid in the specific emergency case. Based on a prototype implementation of the developed concept an evaluation including for example field and usability studies will be conducted.

Keywords: Ubiquitous Emergency Case System, Mobile Communities, Mobile Web 2.0, Context Awaren, Location Based Services.

1 Introduction

Due to demographic change and increase in life expectancy the average age in our society increases continuously. In Germany for example, people aged 65 years and older will represent around 30% of the population by 2050, which is a 50% increase as of today [6]. This shift will lead to an increasing number of emergency situations and emergency calls in the future [6]. As a consequence, more workforce and money will be required by government and emergency institutions to serve and provide help in emergency situations. A second important aspect

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 338–352, 2010.

regarding emergency services is the first aid response time. For instance, people who experience a heart attack have a much higher chance to survive if first treatment arrives within 90 minutes [7]. Valuable time to the arrival of first aid is lost in detecting and/or reporting the emergency case as well as emergency vehicles losing crucial time due to long access routes and potential traffic jams.

Emergency service providers are in a very challenging position. They need to provide the quickest and best possible emergency service. As mentioned previously, the process of providing first aid is often too slow. In addition, due to the current development of demographic change, health services will face more pressure and costs in serving the increased number of emergency calls. Therefore existing emergency service providers need to be supported and further ways to provide quicker, more immediate and more efficient first aid need to be found.

Our idea to contribute to this development is a service that tracks and gathers context information (e.g. location or availability) of a large community of voluntary helpers and provides it to existing emergency call centers. Such call centers can then use the system to recall and leverage this community's general willingness to provide first aid in emergency situations where they are nearby. Registered voluntary helpers can be selected based on the relation between their situational context and the context of the emergency situation. Thus, the system allows emergency call centers to request their help if their knowledge and current location fits to the context of the emergency situation. If helper's accept the request, they can reach the emergency location within a minimized time frame and provide first aid bridging the time before the ambulance arrives. If the reported emergency case requires only minor or no help at all, they can solve the situation on their own or report a false alarm respectively.

1.1 The Idea: A Ubiquitous Mobile Community Emergency System

The idea of a ubiquitous mobile community emergency system in the previous section bases on two factors: First, the general willingness of people to provide help to others is very high. In Germany for example, according to [8, p. 165] almost one third of the population is committed to some type of volunteer work. Another third of the population is generally interested in providing volunteer work. Their top motives are to help other people and make a useful commitment to the social community [8, p. 176]. We aim to leverage this untapped potential of voluntary helpers that want to support others if they had the opportunity to. Second, in most developed countries mobile phones are widespread throughout the population, and new embedded technology such as high-speed-internet and global positioning technology (GPS) is becoming more and more available for the mobile phone mass market.

Using a combination of such embedded mobile technologies and existing emergency service providers' infrastructures, we aim to close the gap between urgent emergency help seekers and the willingness of nearby potential voluntary help givers. As a first approach to this vision, we have designed the architecture and implemented SociCare, a ubiquitous, context aware mobile community emergency system that manages and provides realtime context information of an arbitrary

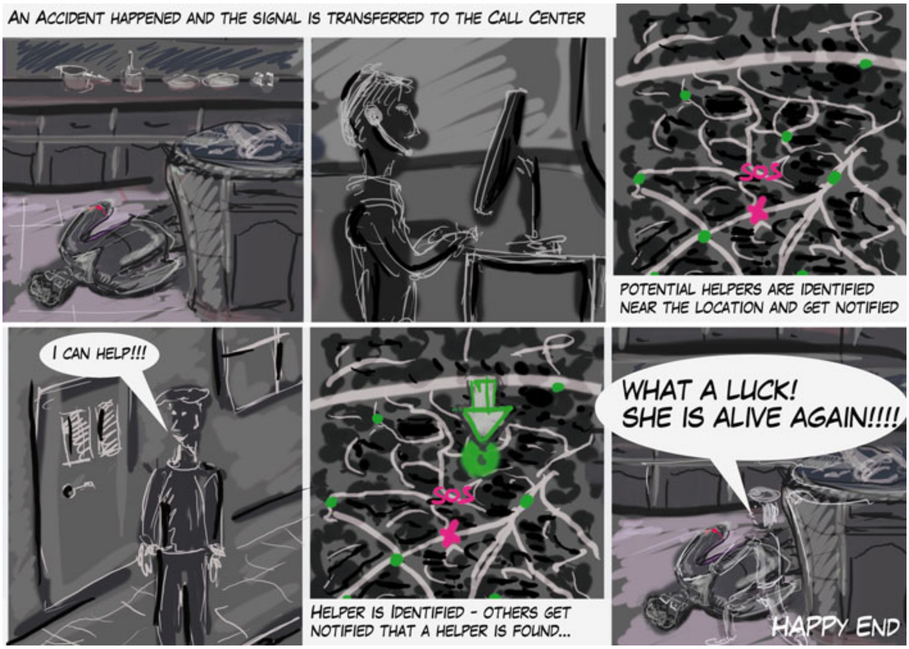

AN ACCIDENT HAPPENED AND THE SIGNAL IS TRANSFERRED TO THE CALL CENTER
SOS
POTENTIAL HELPERS ARE IDENTIFIED NEAR THE LOCATION AND GET NOTIFIED
I CAN HELP!!!
SOS
HELPER IS IDENTIFIED - OTHERS GET NOTIFIED THAT A HELPER IS FOUND...
WHAT A LUCK! SHE IS ALIVE AGAIN!!!!
HAPPY END

Fig. 1. Use Scenario: SociCare addresses voluntary helpers to provide first aid in emergency cases before the ambulance arrives (own illustration)

number of certified voluntary community emergency helpers. The context information managed by SociCare includes location and distance to an emergency situation as well as the availability and ability (i.e. skills) of a voluntary helper to provide first aid.

By doing so, SociCare enables emergency call centers to quickly and easily identify, verify and select voluntary emergency helpers that are able to provide aid in an emergency situation. As a result, SociCare contributes to the above mentioned problem in the following ways:

- Response time: By the integration of voluntary helpers and random passers-by in emergency situations, SociCare enables emergency services to reduce the response time from receiving the emergency call to having someone to help at the location of emergency. Thereby, the emergency call can be triggered by the help seeker himself, or an emergency detection sensor attached to the person's body.
- False alarms: It helps emergency services to reduce costs by integrating a cost free community of voluntary helpers that can detect costly false alarms early at the point of interest, thus avoid e.g. costly emergency vehicles to be sent.

– Coordination: The system integrates a Web 2.0 enabled interactive map interface that enables emergency call centers to easily identify, communicate with and coordinate voluntary helpers in real time (see Figure 1). With its adaptable database infrastructure and web based interface, this solution can be easily built upon the existing infrastructure of widespread existing call centers.

2 Related Work

Providing help to elderly people and serving emergency situations has been targeted by previous work. These proposed solutions focus on emergency detection, such as industrial products that provide body sensors measuring and detecting anomalous vital signs or other emergency situations (e.g. a sudden collapse). In a similar context, previous research studies have focused on integrating various sensor types and emergency detectors in an extensible middleware architecture [4] [5] [3] towards a solution for extensive in-house safety. However, such approaches do not cover the issue of late first aid arrival in emergency cases. Up to our knowledge, the only previous work that is based on the principle of integrating a community of voluntary helpers in the emergency process is presented in [3]. The proposed system AGAPE proposes a similar concept for realizing a ubiquitous mobile community emergency system to SociCare. Same as SociCare, AGAPE leverages mobile clients as well as location- and proximity detection technology in order to identify potential nearby helpers. However, AGAPE focuses on a purely automatic group formation, coordination and management of outdoor voluntary helpers. In contrary, we believe that this complex task can be better performed by a human operator. Therefore, with SociCare we put our focus on providing an efficient tool and user interaction to enable a human emergency call center agent to quickly identify, select and notify the best suitable and available voluntary emergency helper depending on the context of himself and the emergency situation.

3 SociCare Design

The design of the SociCare system is twofold. It covers (1) a web front end to be used by a human agent at the emergency call center and (2) mobile clients that run as mobile applications on the voluntary helpers' devices. Therefore, the overall architecture is based on a client-server architecture. The SociCare server communicates with the call centers' servers and the voluntary helpers' mobile clients. This way the helpers can be sure that their position is only visible to the call centers in case of an emergency.

As the application is time-critical by use case, the emergency notifications should be transmitted quickly and reliably to the helpers. This suggests some kind of push mechanism. The call center agent should be able to immediately view the location of potential helpers around an emergency. This means that the SociCare server has to be aware of all helpers' locations at any time.

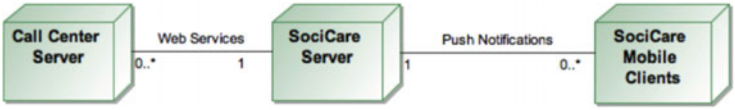


Fig. 2. SociCare connects existing emergency call centers and the community of voluntary helpers (own illustration)

3.1 Web 2.0 Map Client for the Emergency Call Center

In emergency call centers advanced IT support is already widely established. Operators have e.g. the possibility to detect the position of an emergency quickly by the information they receive from the caller or emergency sensor, thus they can send an emergency vehicle closest to the place of interest. Therefore we decided not to provide a stand-alone emergency platform, but rather integrate the ubiquitous mobile community of voluntary helpers as an add-on to the existing services. Integration within the existing systems is supposed to save costs and increase the likelihood of adoption. The value added by SociCare to emergency call centers lies in the database of registered voluntary helpers that can be accessed from the standard platform interface. Features include a real time map-display of the emergency location and nearby voluntary helpers. As helpers are registered in the SociCare database, additional information on e.g. their profession, their skills and abilities to provide first help or their mobility can be shown, if available to allow the choice of the best helper suitable for the respective emergency case. Call center agents can view helper's details, select the closest by location and inform them by push notification on their mobile phones. Visualization and usability is critical. Therefore the map on the agent's side shows in clear symbols where tracked people are and if they agreed to help or declined a query. In case one helper agrees to take the task, one more key feature is the communication between the call center and the helper. The agent is then able to call the helper directly from the platform to guide him to the emergency location and give instructions how to provide first aid.

3.2 Mobile Phone Application for Voluntary Helpers

Helpers who register for SociCare provide their contact information and willingness to help in emergency situations. Thus, it is crucial for them that the sign up process is simple and user friendly and their data is being held by a trustworthy organization. Therefore we intent the SociCare system to be run by the call centers themselves or a trusted organization that acts as a mediator between the voluntary helpers community and a symposium of different call centers. In the latter case, one would also benefit from network effects and scalability of the service.

The mobile client is based on the Google Android mobile phone platform. Voluntary helpers can join the network by simply downloading the SociCare application to their phones and providing some additional information about their profession and skills in providing first aid. In case an emergency in their surrounding happens, the call center agents can track their current location and will inform them conveniently by push notification. If the helper feels able to provide help, he can agree to accept the request. As helpers might want to help but fear to do something wrong in the help giving process, they can request a voice call and be guided by the call center agent while providing first aid. Furthermore, data security is crucial; so all helper's context information, location data and emergency specific details have to be kept save and protected from misuse.

4 SociCare System Architecture

SociCare is a distributed system with two different types of clients, different communication channels and protocols for each type and a central application server. This section illustrates all architecture decisions and ideas for the SociCare deployment and communication architecture as well as the server and client side architectures.

4.1 Deployment and Communication Architecture

The general deployment and communication architecture of the system is based on a standard Client-Server architecture concept which defines one central server that manages one or many secure communication channels with an arbitrary number of different types of clients. Figure 3 shows an overview about the SociCare Client-Server architecture.

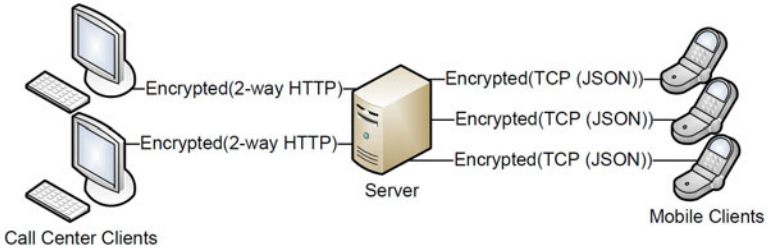


Fig. 3. SociCare Client-Server Architecture and Communication Channels (own illustration)

The server hosts the SociCare application logic, all SociCare related data and handles all data communication with all clients using two different types of SSL (or similar) encrypted communication channels:

- **Call Center Channel -** Represents a steady data connection between the SociCare server and multiple clients on multiple call center workstations. The channel uses the Hyper Text Transfer Protocol (HTTP) protocol [9] to enable call center clients to render the web browser based GUI of the

SociCare application. Since HTTP is a protocol that allows clients to request information from servers but does not allow servers to push information to clients, a HTTP-Push [10] concept is used to enable a two way data communication between the SociCare server and the web browser based call center client.

- **Mobile Channel** - Represents a steady data connection between SociCare server and multiple mobile clients of voluntary helpers. Transmission Control Protocol (TCP) is used as two way data communication protocol and the JavaScript Object Notation (JSON) [11] is used as payload data format. This format was chosen for two reasons: i) it is lightweight and therefore has a comparably low data volume, which might be important for the users on expensive mobile contracts, ii) it is easy to encode and decode from any object.

The communication architecture of SociCare defines a stable and extendable application logic process and communication protocol between all types of clients and the central server. Figure 4 illustrates a simplified standard SociCare emergency process and communication sequence without error message and component failure handling.

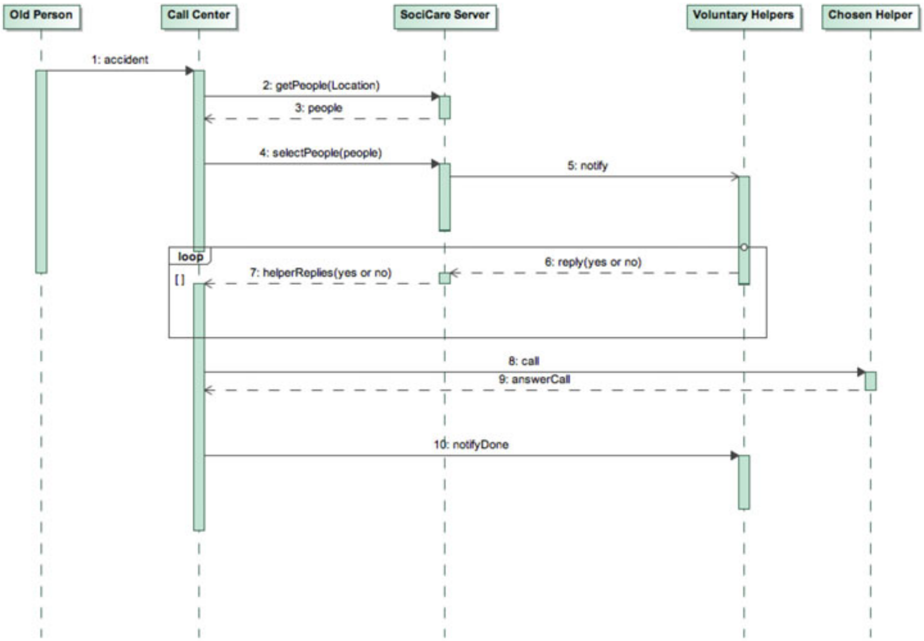


Fig. 4. SociCare Communication Architecture (own illustration)

In this sequence example an "Old Person" indicates an emergency situation (1) which the emergency call center receives. The call center uses the SociCare call center client application to trigger a request (2) to find all voluntary helpers in the area of the emergency. The server responds (3) with a list of voluntary helpers including data about their certification details and ability to help immediately. The call center agent selects the most appropriate voluntary helpers from that list (4) and notifies (5) them on their mobile devices. The SociCare server collects (6) the responses (yes - available; no - not available) of all notified voluntary helpers and forwards (7) them to the call center client immediately. The call center agent selects one or a group of available helpers and informs (8) them about the emergency situation details. These helpers rush to the emergency situation location and help the person in need. After the situation was solved, they inform (9) the call center. The call center notifies (10) all other voluntary helpers that the situation was solved.

4.2 Server Side Architecture

The server side architecture of the SociCare application is a Model 2 web application architecture that defines a Model View Controller (MVC) [1] concept for web based applications. By applying the MVC architecture to a web and mobile based application, data model details are separated from the presentation and the application logic and processes that use this model. Such separation allows multiple views to share the same model, which makes supporting multiple clients easier to implement, test, and maintain [1]. This is essential for the SociCare system since the different types of clients have to work on the same data model and the overall SociCare system has be extendable with new services, applications or different types of clients in the future.

Figure 5 illustrates the dependencies between the SociCare MVC components:

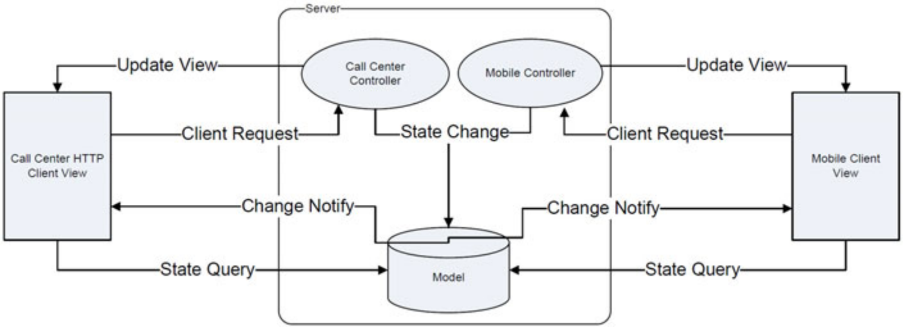


Fig. 5. SociCare Server Side MVC Architecture (own illustration)

- **Model** - The model represents SociCare application specific data and rules that govern access to and updates of this data. Often the model serves as a software approximation to a real-world process, so simple real-world modeling techniques apply when defining the model. The SociCare model stores data and access rules for SociCare call center user accounts, all registered voluntary helper accounts including details about their mobile devices. Further more, it manages a real time data set of emergency situations, their current status, communication details and up-to-date context data of all voluntary helper mobile clients that are connected to SociCare. In addition, the model stores all logging and event history information of the SociCare system, so every action, event, interaction and task is logged and can be tracked.

- **Views** - SociCare defines two different types of views, workstation based call center client views and mobile client views. The views access the data within the model, define and specify how the data needs to be presented and render the data on the clients. The views are responsible for maintaining consistency in their presentation when the model changes. This can be achieved by using a push model, where the view registers itself at the model for change notifications to query the model. Or a pull model, where the view is responsible for querying the model regularly to present the most current data. Due to the fact that SociCare works with real time context data of mobile clients and emergency situations, a pull model is not efficient enough and only a push model is able to update the views right at the moment when updated data is available. In addition to rendering model data, the views present the GUI of the application and initiates user interaction requests (including entered user data) to the controller of the application.
- **Controller** - The controller translates interactions with the GUI on the view into actions to be performed on the model. The actions performed on the model include activating application logic processes or changing the state of the model. Based on the user interactions and the outcome of the actions, the controller responds by updating the current or changing to an appropriate view. Thus, the controller centralizes functions such as view selection, security, and templating, and applies them consistently across all views. Consequently, a major advantage of this architecture is, when the behavior of these actions needs to change, only a small part of the application needs to be changed: the controller and its helper components [2].

  SociCare defines two completely different types of clients that require different corresponding actions, processes, permissions and configurations. So two controllers, one for call center clients and one for mobile clients have to be defined. Each of these controllers only implements the application logic and processes that are needed to fulfill all tasks for the client it serves.

4.3 Client Side Architecture

The mobile client represents the user interface for voluntary helpers and needs to have a stable two way data connection to the SociCare server as well as access to up-to-date context information such as the current geographic location. Figure 6 shows an overview of the mobile client architecture.

The mobile client application is a native application on the mobile device and consist of four different components:

- **Application Logic and GUI** - Implements all application logic code and the mobile platform specific GUI for the voluntary helper application.

- **Data Store** - Implements a central data storage component on the mobile client for all configuration and application data such as user account settings, location information history, etc.
- **Location Service** - Implements a parallel running process that accesses the mobile devices operating system (OS) in order to retrieve up-to-date geographical location information from the location hardware built into the device.
- **Connection Service** - Implements a parallel running process that manages a stable two way connection between mobile client and server. It takes care of session negotiation, sending and receiving messages as well as dealing with connection losses and reconnecting etc.

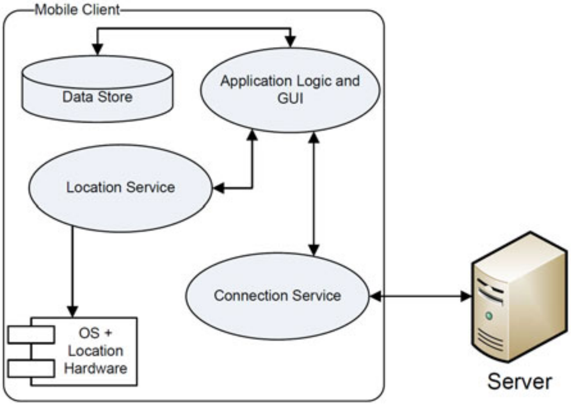


Fig. 6. SociCare Mobile Client Side Architecture (own illustration)

5 SociCare Prototype Implementation

In order to show the advantages of the architecture decisions above and to being able to use a running system in future studies and evaluations, a proof of concept prototype implementation of the SociCare system was implemented. The following section outlines some details about implementation and development platform decisions and illustrates the client GUIs.

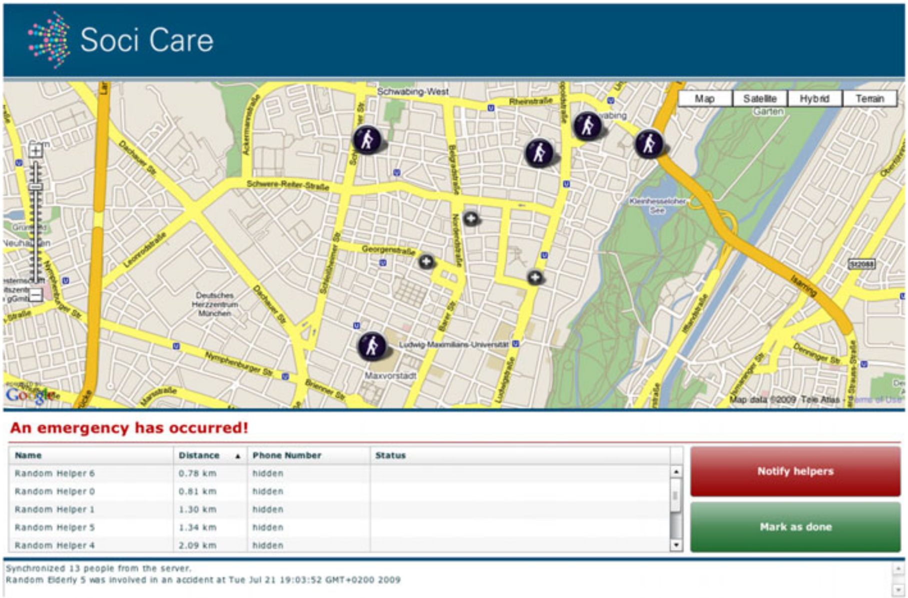
5.1 Server

As server platform Apache Tomcat 6.x [12] was selected and both controllers as well as the model were implemented as Java J2EE [14] application components. The mobile client TCP communication channel was implemented in a standard Java component using third party JSON encoding and decoding libraries. Since the call center communication channel requires a non standard HTTP-Push concept to function properly, several options were considered and the open source technology BlazeDS was used.

BlazeDS is a server-based Java remoting and web messaging technology that allows an application to connect to the back-end and push data in real-time to Adobe Flex and Adobe AIR Rich Internet applications (RIA). The Message Service provides a complete publish/subscribe infrastructure allowing Flex clients and the server to exchange messages in real time. Remoting allows a Flex application to directly invoke methods of Java objects deployed in an application server. [13]

5.2 Call Center Client

The SociCare call center client was implemented using the Adobe Flash based Adobe Flex technology which perfectly integrates with the HTTP-Push server technology BlazeDS. Flex enables the implementation of stateful rich web browser based applications where significant changes to the view or sending and loading data to and from a server do not require reloading the current view. Flex comes with a set of user interface components including buttons, list boxes, trees, data grids, several text controls, and various layout containers. Further more, advanced features like web services, drag and drop, modal dialogs, animation effects, application states, form validation, and other interactions go far beyond HTML web application possibilities and make Flex the perfect choice for the SociCare call center client prototype implementation. [15] Figure 7 shows the main screen of the Flex based SociCare call center client.



An emergency has occurred!

Name	Distance ▲	Phone Number	Status
Random Helper 6	0.78 km	hidden	
Random Helper 0	0.81 km	hidden	
Random Helper 1	1.30 km	hidden	
Random Helper 5	1.34 km	hidden	
Random Helper 4	2.09 km	hidden	

```
Synchronized 13 people from the server.
Random Elderly 5 was involved in an accident at Tue Jul 21 19:03:52 GMT+0200 2009
```

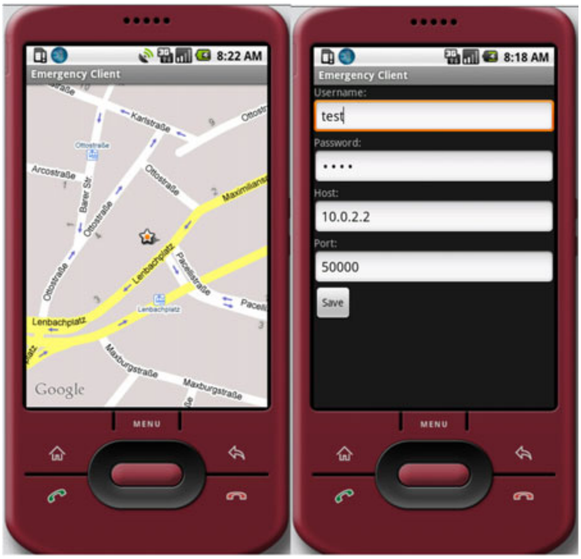
Fig. 7. SociCare Flex based call center client (own illustration)

The call center client uses a map with up-to-date mobile client location information as main user interface component. The agent is able to select helpers and view detailed information about them as well as to notify them about an emergency situation. By interacting with a mobile user's symbol on the map, the agent is able to communicate with the user and so coordinate the group of helpers. Finally, the agent is also able to mark an emergency situation as completed which informs all helpers on their mobile devices.

5.3 Mobile Client

The SociCare mobile client for voluntary helpers was implemented using mobile devices based on the Google Android platform [16]. Android provides a mature support for accessing the geographic location information of the mobile device, for parallel processes for the location service and connection service as well as for two way TCP network connections between client and server. Encoding and decoding of the JSON payload of the messages was implemented using third party libraries.

Figure 8 illustrates two screens of the SociCare mobile client. Each screen renders a view and handles all interactions with the user.



Username:
test

Password:
••••

Host:
10.0.2.2

Port:
50000

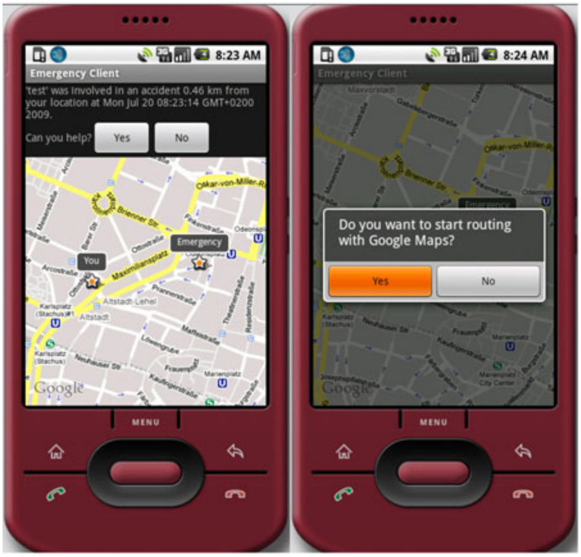
Save

Fig. 8. SociCare Mobile Client - Position screen and Settings Screen (own illustration)

The Position screen shows the own most up-to-date geographical location on a map. The Settings screen renders an input form for modifying the mobile client communication channel settings Host Internet Protocol (IP) address, Host-Port, Username and Password.

Figure 9 illustrates the two most important screens that are rendered when an emergency situation occurred and the mobile client is notified. The Emergency screen renders an overview map that shows the own most up-to-date geographical as well as the geographical location of the emergency. Also, the screen includes a dialog that allows the helper to state his availability to help.

If the helper is not available and selects "No", the screen disappears and the Position screen is rendered. If the helper is available and selects "Yes", a Emergency Routing Dialog is rendered on top of the map and offers to connect to the Google Maps Routing service in order to route the shortest and/or fastest path to the emergency.



'test' was involved in an accident 0.46 km from your location at Mon Jul 20 08:23:14 GMT+0200 2009.

Can you help?

Do you want to start routing with Google Maps?

Fig. 9. SociCare Mobile Client - Emergency Screen, Emergency Routing Dialog (own illustration

6 Conclusion and Future Work

This work aims to close the gap between urgent emergency help seekers and the willingness of nearby potential voluntary help givers. From this background, we have designed the architecture and implemented SociCare, a mobile community emergency system that tracks real-time context information of certified voluntary helpers, and visualizes this information on an interactive map interface for emergency call centers. After receiving an emergency call, the human agent in the call center can identify and request nearby voluntary helpers who are equipped with the SociCare mobile client to help the person in need. The context information managed by SociCare includes the distance to an emergency situation as well as relevant context information, such as availability and skills of the voluntary helpers to provide first aid. We have designed SociCare that it can be easily integrated with the existing emergency management software used in the call centers. As to our knowledge this approach of a ubiquitous mobile community emergency system attached to existing emergency call centers is the first of its kind, we plan to undertake further research on how well it performs in real use cases. Therefore, we aim to pursue usability studies on both provided interfaces: (1) the interactive map interface needs to be tested with real call center employees and evaluated how quick and efficient they perform in selecting and interacting with the identified voluntary helpers. (2) On the other side the mobile client's interface needs to be evaluated on usability factors in real context. We believe that such a user study under real emergency conditions, i.e. the user experiencing a stress situation, is crucial, since their performances will differ from usual labor setting tests that we have performed throughout the development phase.

Acknowledgment

The authors would like to thank Deutsche Telekom Laboratories for providing the prototyping equipment for SociCare as well as their technical support and feedback. Furthermore, we thank the anonymous reviewers for their feedback and input on earlier versions of this paper.

References

1. Singh, I., Stearns, B., Johnson, M.: Designing Enterprise Applications with the J2EE Platform. Sun Microsystems, Inc. (2003)
2. Bien, A.: J2EE Patterns. Addison Wesley, Reading (2003)
3. Bottazzi, D., Corradi, A., et al.: Context-Aware Middleware Solutions for Anytime and Anywhere Emergency Assistance to Elderly People. IEEE Communications Magazine 44(4), 82–90 (2006)
4. Pung, H.K., Gu, T., et al.: Context-Aware Middleware for Pervasive Elderly Homecare. IEEE Journal on Selected Areas in Communications 27(4) (2009)
5. Taleb, T., Bottazzi, D., et al.: ANGELAH: A Framework for Assisting Elders at Home. IEEE Journal on Selected Areas in Communications 27(4), 480–494 (2009)
6. Statistisches Bundesamt Wiesbaden, Bevoelkerung Deutschlands bis 2050. Ergebnisse der 10. koordinierten Bevölkerungsvorausberechnung (2003)
7. Wire, H.: Philips seeks to reduce time from heart attack to treatment (2007), `http://www.healthtechwire.com/Pressrelease.146+M5a1042be3f1.0.html` (accessed 01.02.2010)
8. BMFSFJ, Altenhilfestrukturen der Zukunft - Abschlussbericht der wissenschaftlichen Begleitforschung zum Bundesmodellprogramm, Bundesministerium für Familie, Senioren, Frauen und Jugend (2004)
9. Fielding R.: Hypertext Transfer Protocol - HTTP/1.1 (1999),`http://www.w3.org/Protocols/rfc2616/rfc2616.html` (accessed 01.02.2010)
10. Google, ServerPush (2007), `http://code.google.com/p/google-web-toolkit-incubator/wiki/ServerPushFAQ` (accessed 01.02.2010)

11. Crockford D.: The application/json Media Type for JavaScript Object Notation (JSON) (2006), http://www.ietf.org/rfc/rfc4627.txt (accessed 01.02.2010).
12. The Apache Software Foundation, Apache Tomcat (2009), http://tomcat.apache.org/ (accessed 01.02.2010)
13. Adobe Open Source, BlazeDS (2009), http://opensource.adobe.com/wiki/display/blazeds/(accessed 01.02.2010)
14. Sun Microsystems, Java 2 Platform - Enterprise edn.(2009), http://java.sun.com/j2ee/overview.html (accessed 01.02.2010)
15. Adobe Labs, Adobe Flex Framework Technologies (2009), http://labs.adobe.com/technologies/flex/ (accessed 01.02.2010)
16. Google, Android (2009), http://developer.android.com/index.html (accessed 01.02.2010)

A Workflow Definition Language for Business Integration of Mobile Devices

Martin Werner[1], Stephan A.W. Verclas[2], and Claudia Linnhoff-Popien[1]

[1] Mobile and Distributed Systems Group
Ludwig Maximilian University Munich, Germany
martin.werner,linnhoff@ifi.lmu.de,
http://www.mobile.ifi.lmu.de/
[2] T-Systems International GmbH, Germany
Stephan.Verclas@t-systems.com,
http://www.t-systems.de/

Abstract. The integration of mobile devices into business processes is a challenging but promising task. We designed a system that shows the possibility of constructing software systems that accomplish the difficult achievement of a system that simplifies tasks for businesses as well as for their customers. We designed a very simple and straight-forward workflow definition language that has the advantage of allowing source-code generation for arbitrary mobile platforms. With this language we assembled a system that is able to support insurance case documentation with context-aware input wizards and cross-platform software. It is possible to take different actions in different surroundings and to integrate mobile sensor data (GPS, cell-location, audio, compass, ...) into workflow management.

Keywords: Workflow management, Workflow management systems.

1 Introduction

Workflow modelling and management has become a very important tool for business management and organization. With this paper we show that workflow definition can even be used as a tool for structured communication schemes with customers. We assembled a system with which basic workflow can be specified and automatically translated into source code for a mobile application. In this way it is possible to integrate the process of documenting some event into the surrounding workflow in a natural and automated way. We solved the problems of diversity arising from mobile devices by specifying a language that can be translated into different programming paradigms without difficulties. By using workflow constructs it is even possible to design advanced input forms that make use of context information. An example might be to use audio input only in cases where the surrounding noise is not too high and to give screen input possibilities as an alternative in the other case. The pay-off of such a system is the correctness and completeness of the document as well as the adaptivity of the application.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 353–362, 2010.

The scenario for which we provide a solution with this paper is based on a company and an individual which have seldom but important and complex communication demands. As a main example we chose an insurance company. Insurance companies usually do not have regular communication with their customers. In an insurance case however the customer (insurance holder) and the business (insurance company) have to follow pretty complicated workflows which at least the customer is not familiar with. The complexity of the workflows arises from the need to check the insuree's information (address, phone number, ...), to exchange complex information (what, when, where, why) and the need for decisions such as whether expert assessment are needed or how to provide an appropriate solution (e.g. immediate reparation vs. taking over costs for car rental and transportation).

For our application let us assume that the insurance case happens in a substantial level of provision of infrastructure for mobile Internet applications to work and that the insurance holder is equipped with a mobile phone providing Internet access. The idea is now to support the classical workflow for an insurance case - which usually starts with an insurance holder calling the insurance company's call center - with adaptive and context-aware software for the mobile device of the insurance holder.

The insurance company's call center agent will ask for the type of mobile phone, the mobile phone number and the level of familiarity with installing and using mobile applications. In cases where the insurance holder is capable of installing specialized software and using applications which he is not familiar with, the call center agent can automatically create a mobile application tailored to the customers device and situation which will support the complete documentation workflow of the insurance case. He will then send a download link for the software via SMS, MMS or email directly onto the insurance holder's device.

In this way the insurance company can make use of the full power of the mobile device and support the insurance case documentation with sensor data, video, audio and photo as well as GPS position information. In this way the insurance gets a very clear and complete record of the insurance case and the insurance holder gets a simple and clear way to explain what has happened and what he would expect from the insurance company.

One of the major aims of such a software supported workflow is the reduction of the number of expert assessments which are made because of incomplete or incorrect insurance case documentation.

Figure 1 gives an example of an insurance case which is documented using our specialized mobile software. The insurance holder has had a car accident. He calls the insurance company for help. The call center agent of the insurance company finds out that the insuree has a suitable mobile phone and the needed background on using mobile applications. They discuss the insurance case such that the call center agent can decide what has to be done for documenting the case. The call center agent constructs an application which the insuree can use to document the insurance case. This software package is then provided to the customer over the air. The insurance holder then uses the application to

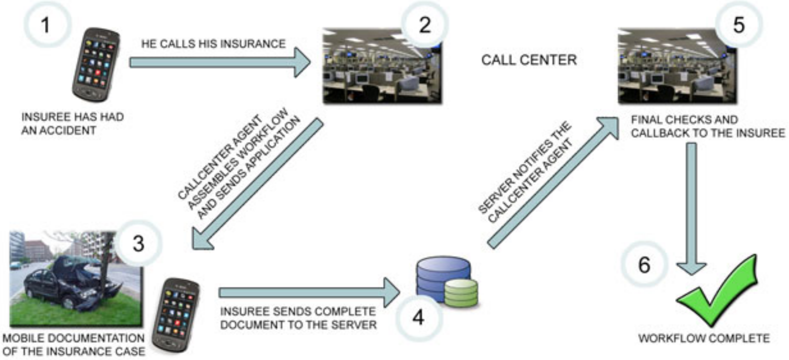

1
HE CALLS HIS INSURANCE
2
CALL CENTER
5
INSUREE HAS HAD AN ACCIDENT
CALLCENTER AGENT ASSEMBLES WORKFLOW AND SENDS APPLICATION
SERVER NOTIFIES THE CALLCENTER AGENT
FINAL CHECKS AND CALLBACK TO THE INSUREE
3
6
INSUREE SENDS COMPLETE DOCUMENT TO THE SERVER
4
MOBILE DOCUMENTATION OF THE INSURANCE CASE
WORKFLOW COMPLETE

Fig. 1. An example case

compile a complete documentation of the insurance case. In cases where he has a question, he uses a help button which directly connects him to the same call center agent. The data he accumulates is sent to the call center agent who can immediately see the photos and videos the insurance holder collected. Once the information has been uploaded, the call center agent gets notified and will call the the insurance holder for further steps.

2 Related Work

Recently workflow management has become one of the most valuable information technological tools for business management. Most systems in this area try to support business by modelling the complex relations between business tasks (parallelity, concurrency), business demands (pre- and postconditions of tasks) and business resources (manpower, computing resources, deadlines) into one global language. The complexity of such a system is of course the biggest pitfall in this area. It is not easy to show that a dynamically changing and distributed business process model keeps complete and well-defined for all input and free of dead-locks and even contradiction.

To remedy these restrictions, a business process is seldom modelled in its full complexity. Though there can be areas where a workflow definition is not easy to model and a description of data and data requirements might be better, we want to concentrate on the subpart of workflow specification. The term workflow can be understood from various different perspectives [8]. If we are talking about workflow in the following, we always mean the *control-flow* perspective of workflow. To make this explicit: A workflow description in our sense describes the tasks and their execution ordering and logic where a task is an atomic unit of work. There have been several languages in discussion ranging from simple languages with few features to complex languages with many features but the difficulties arising from faults.

Process definition languages have been there for a long period of time. Examples are the Process Interchange Format [1] which was later incorporated

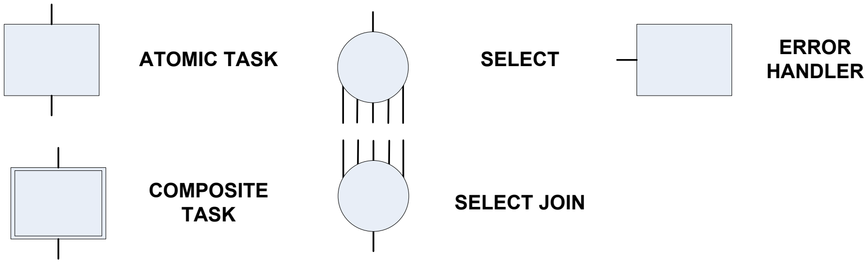


Fig. 2. Symbols used for Workflow Definition

to NIST's PSL [2] which also has an XML mapping [3]. Business modelling languages always contain mechanisms to define processes. The Business Process Management Initiative (BPMI) tried to establish a standard language BPML for design and specification of business processes since 2001. The BPMI joined the Object Management Group (OMG) in 2005. The Workflow Management Coalition proposed another XML-based language XPDL [4] which aims at business interoperability. A systematic analysis of this language and of possible workflow patterns is [5]. A newer open initiative is YaWL (2005), a language that tries to solve issues which pop up due to synchronization problems and non-local withdrawals that arise in complex business models organized as Petri nets.

The overall aim of these languages is the adoption of automation and controlling to general business management. One tries to put all business logic and side-effects into a global model which can be understood and optimized with automated systems.

All these languages have in common their high level of complexity which makes them difficult to use in mobile and distributed ([7]) environments. Many of the enhanced features of current process definition languages are not needed for our scenario. Hence we decided to define a suitable subset of such languages which can be used to model the workflow of an intelligent and self-adjusting context-aware application helping the users filling out complex forms.

3 A Workflow Definition Language for Mobile Environments

From the discussion of the previous section we now derive requirements and limitations for a language tailored to defining complex workflow without concurrency, mainly supporting user input. It is common to use a set of symbols to define workflows. Symbols are the atomic parts of the language and can be semantically arranged as a graph where each symbol can have a restriction on the number of incoming and outgoing edges. Figure 2 gives an overview of the atomic symbols of our workflow language.

These symbols can be arranged into a tree-structure. The application will then walk through the tree and take appropriate action for a specific task. The conditions that we need range from a set of device properties which are only known at runtime (e.g. camera resolution, GPS-accuracy) to some process events (failed to get a position).

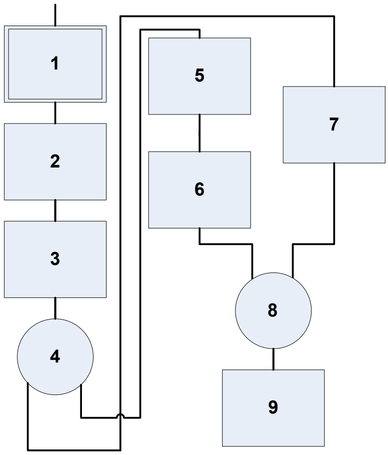
In favor of maximal simplicity we have collected the following symbols for our language. Atomic tasks are parts of work that can not be split up. Examples are taking pictures, video, input forms, web browser session, phone calls etc.). These atomic tasks can be arranged into lists which have a defined ordering (e.g. wizards). For context-awareness and adoption we decided to only allow those splits where exactly one of the possible choice is taken (XOR-split) and none of the connected atomic tasks has been executed before. For error management each atomic task can set a pointer to another task which shall be executed on error. In this way we are able to model the following two basic error handling mechanisms: A "On Error Resume Next"- strategy, where a task which has reported some error condition is stopped and the workflow continues with the next atomic task or an Error-Handling task which can be set for a given atomic task. These error handling strategies make it possible to have a default strategy of e.g. calling the call center-agent to discuss the problematic situation but also the flexibility to react on a problem with a specific alternative (e.g. if there is no GPS fix we can use a network location service and a map).

Most process definition languages also contain constructs for circuits and concurrent task execution such as AND-splits, XOR-splits, AND-joins and XOR-joins. We decided not to allow any of these complex constructs because of the difficulties with synchronization points (see [6]). The most important problem with these constructs is that it might be unclear at design time as well as at running time whether a synchronization is needed (i.e. waiting for several branches to complete) or whether synchronization is not needed or intended. This will make the application behavior unpredictable and hence has implications on usability. A concrete example would be the case where the application on the mobile phone of the user does not work correctly or the user is not able to use it correctly and then calls back the call-center agent through a help button. The call-center agent should have a complete understanding of the workflow to be able to help in this situation.

There are of course workflow patterns which are not possible or very difficult to model without the constructs we removed. But the language shall only be used for a small part of some bigger workflow where the non-mobile part of the workflow can have all these complex constructs.

The workflow which can be defined with our proposed language is free of circuits except for the case of error handlers pointing back and forth between two atomic tasks that fail. To remove this case, we insist that the depth of error handler execution has a fixed maximum and that a global error handler will terminate such situations.

We decided to describe the workflow in an XML-file format. We are aware of the fact, that the usage of XML will lead to very much overhead, but the reason for using XML is, that for the prototype the benefit of using XML lies in the fact that most mobile platforms support XML-parsing and that XML has a



1) Composite Task: Form
Check and update your address record.
2) Atomic Task: Audio Input
Give name and address of the opposite party
3) Atomic Task: GPS
Record GPS information
4) Switch: cam.resolution
Decide whether the camera resolution is high
5) Atomic Task: Photo
Take an overview picture
6) Atomic Task: Photo
Take a detail photo
7) Atomic Task: Photo
Take one picture
8) End-Of-Switch

9) Atomic Task: Upload
Upload results and request call.

Fig. 3. An Example Workflow

tree structure and is extendable. For a product, the complete language should of course be packed into a more efficient data format.

For each symbol we introduce an XML-element. Each element must have an attribute **id**. Each symbol can have an attribute **errorhandler** which contains either the **id** of the error-handler or the special value **next**. If the error-handler is not specified for a task but some parent element has set it, then it is inherited. An atomic task must not have any child-elements. A composite task can have any workflow object as a child element. A switch has several **case** elements which define conditions on the execution of a case. These cases are checked in the order they are specified and the first one whose condition attribute evaluates to true is used. The error-handler of the switch statement is used if none of the case conditions evaluates to true. A switch element has to be closed in the same composite task (this is automatically enforced by XML).

Figure 3 shows an example workflow. The error handlers have been omitted in the graphical representation. The application being defined will first ask the user for some personal information for which presets are given. Then the contact data of the opposite party shall be recorded in audio. Once this basic information is available the workflow tries to locate the phone using GPS. The following switch decides whether the camera of the phone has enough resolution (e.g. 5 megapixels) to document the case using one photo or whether a detail picture is needed. Recall, that switches need not be complete and may have several cases which evaluate to true. Only the first one of those cases which have a condition that evaluates to true is being executed. If there is no such task, then the error-handler of the task is invoked.

The XML description of this example workflow is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<workflow errorhandler="9">
```

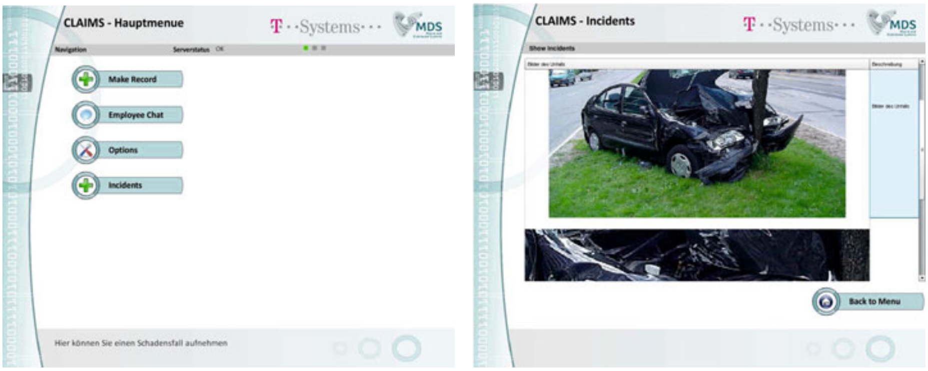
```
  <compositetask id="1">
    <task id="10" type="inputtext"
          preset="Martin Werner">Name:</task>
    <task id="11" type="inputtext"
          preset="Some Street 20">Street:</task>
    <task id="12" type="inputtext"
          preset="D-81549 Munich">ZIP / City:</task>
  </compositetask>
  <task id="2" type="inputaudio">Say name, address and
                   phone number of the opposite party!
  </task>
  <task id="3" type="inputgps"/>
  <switch id="4">
    <case condition="phone.camera.resolution >= 5">
      <task id="7" type="inputphoto">Please take a
                            photo of the situation!
      </task>
    </case>
    <case condition="true">
      <task id="5" type="inputphoto">Please take an
                    overview photo of the situation!
      </task>
      <task id="6" type="inputphoto">Please take a
                 detailed photo of the main damage!
      </task>
    </case>
  </switch>
  <task id="9" type="uploadandcall"/>
</workflow>
```

4 Prototype

To show the feasibility of our approach we have assembled a prototypical implementation. This implementation consists of three parts: An agent application, a server infrastructure and a mobile application.

For the call-center agent we developed an application with which he can define properties of the mobile phone as well as define a specific workflow using drag-and-drop with predefined tasks (either atomic or composite). We used a car accident example as our show-case.

The server infrastructure implements a HTTP interface to facilitate the communication between agent application, database and applications. The main format for data exchange is XML or HTTP file uploads. The decision for this comes from the fact that nearly all devices with Internet access have possibilities to perform file uploads and have XML support built in.



(a) Start screen

(b) Complete incident

Fig. 4. Screenshots from the agent application

4.1 Agent Application

The agent application is implemented in Flash and can be used to create new database records for an incident, to create the mobile application, to send the mobile application to the customers mobile phone and to read out the results of the workflow. A basic ticketing structure has been introduced to organize these records. The state of such a ticket is

- new, if it has been created
- open, if the mobile application has been downloaded
- complete, if the mobile application has uploaded a result
- closed, if the call center agent has checked the data

This application creates the XML definition of the workflow and uploads it to the server. It is also capable of sending a download link via SMS or email directly to the mobile phone of the insuree. The phone number of the calling person is extracted automatically where possible.

Figure 4(a) shows a typical view of the agent application and 4(b) shows the user interface for a complete incident.

4.2 Server Infrastructure

The server component is implemented in PHP and MySQL. It provides an interface to add an attachment file to some given incident as well as an interface to upload the results of basic questions as an XML file. The applications will create the answer and upload an XML description of the results of the workflow which it then extends with attachments (photos, videos, audio, etc.) using HTTP file upload.

4.3 The Android Application

For the example case we assembled a generic Android application which can be compiled with an incident id and an session id used for authentication. Supported atomic tasks are

- Input text
- Take a photo
- Take a video
- GPS position
- Coarse network location
- Location selection on a map
- Call a phone number
- Upload and request call-back

A composite task which only contains input text atomic tasks is accumulated to one scrollable form. Due to the structure of an Android application we were able to directly map an atomic task to an Android activity and to implement the workflow logic in the main activity which posts intents for each atomic task. In each activity we show a help button which directly calls the call center agent.

This Android application shows that it is possible to integrate modern mobile platforms in a very intuitive way into business processes.

4.4 Integration of Other Mobile Platforms

Other mobile platforms can be integrated very easily. Once the agent has published the workflow description to the server, the server can automatically create source-code from templates for each platform and compile and sign them. This is possible for all mobile platforms which have support for downloading and installing software over the air. It is even possible to implement very basic workflow as a web application and hence provide help and access for all users. Though we do not have access to sensor data, photo, etc. in this case, our application still helps to organize and explain the workflow for the specific user.

5 Outlook

With this paper we have presented a new approach and a case study on integration of mobile phones of customers into the workflow of a business. We have solved the problem of the huge differences between the individual mobile platforms by introducing an abstract and simple language to describe the functionality in a way that can easily be mapped to source-code for different platforms. In this way it is possible to use different mobile platforms with the same logic. Most of the automatic source-code generation can be done with XSLT and some script that assembles the sources, compiles the program, signs it and distributes it via a web server.

We showed that the usage of mobile phones can help to follow complex workflows. The benefits are clear. The company can assure that a specific workflow is completed immediately and satisfactorily and the customer does not suffer from misunderstandings and time-consuming correspondence. Interesting features for such an application could be to have a life-chat with the call center agent while the application is active. This feature is however difficult to implement across all major mobile platforms. We plan to investigate this question further.

An application as described before is not limited to the documentation workflow we aimed at. We think that it could be an interesting question whether it is possible to support business management by mobile workflow management using mobile phones. We have shown that the diversity of platforms is not that a big problem in this area.

References

1. Lee, J., Grunninger, M., Jin, Y., Malone, T., Tate, A., Yost, G., et al.: The PIF Process Interchange Format and Framework Version 1.2. The Knowledge Engineering Review 13(1) (March 1998)
2. Schlenoff C., Gruninger M.,Tissot F.,Valois J.,Lubell J.,Lee J.: The Process Specification Language (PSL) Overview and Version 1.0 Specification (1999)
3. Lubell, J., Schlenoff C.: Process Representation Using Architectural Forms: Accentuating the Positive. In: Proceedings of the Markup Technologies Conference 1999 (1999)
4. Workflow Management Coalition: Workflow Process Definition Interface XML Process Definition Language (XPDL), WfMC Standards (2001), http://www.wfmc.org
5. van der Aalst, W.: Patterns and XPDL: A Critical Evaluation of the XML Process Definition Language (2003)
6. van der Aalst, W., ter Hofstede, A.H.M.: YAWL: yet another workflow language (2004)
7. Dong, G., Hull, R., Kumar, B., Su, J., Zhou, G.: A framework for optimizing distributed workflow executions. In: Connor, R.C.H., Mendelzon, A.O. (eds.) DBPL 1999. LNCS, vol. 1949, pp. 152–167. Springer, Heidelberg (2000)
8. Jablonski, S., Bussler, C.: Workflow Management: Modeling Concepts, Architecture, and Implementation. International Thomson Computer Press (1996)

Adaptivity Types in Mobile User Adaptive System Framework

Ondrej Krejcar

VSB Technical University of Ostrava, Center for Applied Cybernetics, Department of measurement and control, 17. Listopadu 15, 70833 Ostrava Poruba, Czech Republic
Ondrej.Krejcar@remoteworld.net

Abstract. User Adaptive Systems are nowadays widely used in modern information systems for their better reaction on user declared or nondeclared requests. Paper describes a concept of User Adaptive System (UAS) as a complex UAS framework. Main focus is in contribution of UAS to user or patient, his life quality and improvements of it. Several interesting examples of UAS are discussed and described as mobile application for sleep state detection as well as several developed user interface components for use at mobile devices.

Keywords: User Adaptive System; Mobile Device; Biotelemetry; UAS Components.

1 Introduction

The idea of User Adaptive Systems (UAS) grown from interaction between user and system (e.g. throws his mobile device). Such interaction can behold in the reaction on user's non declared requests. These requests are based on current user environment and biological or emotional state (e.g. where I am?, what I feel?, am I ok?, etc.). Such user questions can be answered by sensors on user body or inside the user devices. By the help of user mobile device, we can get a user location (e.g. user current position, user future-predicted position, his movement and tracking, etc.). Biomedical sensors on user body can detect several important biomedical data, which can be used for determination of user emotional state in the environment around.

By the combination of user's requests (known or predicted) in conjunction with other sources of user's knowledge and behaviors, the sophisticated information system can be developed based on presented UAS Framework.

The impact of UAS can be seen in the increased user comfort when accessing these mobile UAS. In ideal case, everything what user can imagine to have in his mobile UAS is there.

A one specific kind of problems is based in increased data amount in new mobile systems. In current cases, the user need to specify a data to be downloaded to his mobile device and he need to wait for data downloading and displaying. Due to a several limitations in hardware of current mobile devices, the use of such large amount data has result in lower user comfort. The needs of any techniques to reduce such large data amount or to preload them before user's needs, is still growing up. We created a Predictive Data Push Technology (PDPT) Framework to solve these

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 363–374, 2010.

problems by data prebuffering. Our idea can be applied on a variety of current and future wireless network systems. More usability of PDPT grows from definition of area to be prebuffered as well as from evaluation of artifacts or other user's behavior sources.

The second area of problems which we would like to solve is based on a users biomedical data inputs and a wide area of their possible utility. Current body sensors allow a monitoring of a huge number of biomedical data information (e.g. use a special t-shirt equipped with an ECG, temperature, pressure or pulse sensors). Current hi-tech mobile devices are equipped with a large scale display, provide a large memory capabilities and a wide spectrum of network standards plus embedded GPS module (e.g. HTC Touch HD, HD2). These devices have built-in also a special accelerometer which can be used to determine a user's body situation (user is staying or lying). Last but not least equipment is a light sensor which can be used not only to brightness regulation.

2 Architectural Design for Ubiquitous Computing Systems

Ubiquitous Computing (UbiCom) is used to describe ICT (Information and Communication Technology) systems that enable information and tasks to be made available everywhere, and to support intuitive human usage, appearing invisible to the user [1].

Three basic architectural design models for UbiCom system can be divided to smart devices, smart environment and smart interaction. The concept of "smart" means that the object is active, digital, networked, can operate autonomously, is reconfigurable and has a local control of the resources which it needs such as energy, data storage, etc [1]. These three main types of system design may also contain sub-systems, sub-parts or components at a lower level of granularity that may also be considered as a smart (e.g., a smart environment device may contain smart sensors and a smart controller, etc).

Many sub-types of smarts for each of the three main types of smarts can be recognized. These main types of smart design also overlap between. Smart device can also support some type of smart interaction. Smart mobile device can be used for control of static embedded environment devices. Smart device can be used to support the virtual view points of smart personal spaces (physical environment) in a personal space which surrounding the user anywhere.

Satyanarayanan [3] has presented different architectures for developing UbiCom systems in way of which angle it is focused on a design:

1. Mobile distributed systems are evolved from distributed systems into ubiquitous computing.
2. UbiCom systems are developed from smart spaces characterized by invisibility, localized scalability and uneven conditioning.

Poslad [1] has extended a Satyanarayanan model to Smart DEI model (Device Environment and Interactions). Poslads model also incorporates smart interaction. Smart DEI model also reverses to hybrid models. It is widely assuming by users that the general purpose of end-user equipment will endure but also it will evolve into a more modular form.

2.1 Adaptive Systems for Ubiquitous Computing

Ubiquitous computing provides a vision of computing systems which are located everywhere around us, embedded in the things of our everyday life. They provide an easy access to information and communications bases dedicated to our current location. People are able to interact with any ubiquitous computing environment which they attend. This is a reason why the ubiquitous computing environments must respond dynamically to specific user needs, resources dedicated to their owner's rights or to the current usage context. These require a high level of adaptivity which must be provided by ubiquitous computing systems and related connecting networks [2]. Described project deals with several of issues related to providing such adaptivity for ubiquitous computing environments which will be described more in the following sections.

3 Reaction on a Change of Location – Location-Aware Adaptation

We can imagine the usage of such described UAS in the information systems area of botanical or zoological gardens. In such areas there has been a big potential of usage of a continual localization by use of GPS or wireless networks (in case the GPS has not a sufficient signal – e.g. in urban centers or neighborhoods with high buildings, forest parks or in deep valleys). There is also a possibility to compute a current and predicted user track, so we can predict a position of user in near future (e.g. 25 meters north in one minute). Usability of these information sources is uncountable.

One of possible use of user predicted position is for a determination of a data, which will be needed by user of mobile UAS in near future. Such data (data artifacts) can be preloaded to user's device memory for future requests. The need of preloaded artifacts grown from a need of up to date data context of dynamic online system. Of course when static offline system is used, there is a possibility to load a needed data before usage (e.g store artifacts at SD Card with a size limit to several GB). When user request info about his location in context of zoo or garden (turn-on the device is only needed by user), the client application will respond with a map of near surroundings and a prebuffered data artifacts. User can select a documentary about animals or vegetation around him which can be viewed or played. User can act with direct requests to selected kinds of these. These preferred kinds will be taken into account to evaluate future objects/artifacts and preloaded only the most important ones for a user. The type of artifact is also evaluable as well as his size because the user may not want to look at too long or micro presentation.

As client devices of online UAS, the mobile wireless devices like PDA or Smart phones are commonly used equipped with internet connectivity. The connection speed of the two most common standards GPRS and WiFi varies from hundreds of kilobits to several megabits per second. In case of online UAS or some other types of facility management, zoological or botanical gardens, libraries or museums information systems, the WiFi infrastructure network is often used to interconnect mobile device clients with a server. Unfortunately, the low performance hardware components are used in PDAs or SmartPhones due to a very limited space. Due this a theoretical maximum connection speed is not reachable on such devices. The limited connection speed represents a problem for clients of online system using large artifacts (data files). In some specific cases it is not possible to preload these artifacts before the use of mobile device in a remote access state due any reason.

4 Reaction on a Change of Biomedical Data – Active Context-Aware Adaptation

A key problem of context-aware systems design is to balance the degree of user control and awareness of their environment. We can recognize two extreme borders as active and passive context-aware. In active context-aware system, the UAS is aware of the environment context on behalf of the user, automatically adjusting the system to the context without the user being aware of it [1]. This is a useful in our application where a strict time constraints exists, because the user-patient cannot due to immobility, or would not otherwise be able to adapt to the context quickly enough.

We are using principles of UAS in area of biomedical data processing, where we try to predict some kind of problems by patient data analysis. We developed a context-aware Biotelemetrical Monitoring System (BMS) [11] as a part of the UAS and PDPT Framework project facilitates the following:

- Real-time collection of the patient vital signs (e.g. ECG, EEG) by means of a Body Area Network (BAN) or direct wireless connection to PDA device monitoring station.
- Real-time transmission of the vital signs using the wireless connectivity to the healthcare professionals through a complete architecture including a server database, web services, doctors web access to patients collected and preprocessed data.
- Seamless handover over different wireless communication technologies such as BlueTooth, WiFi, GPRS or UMTS.
- Context-aware infrastructure to sense the context (e.g. location, availability, activity, role) of the patients and Emergency Response Services (ERSs) to provide assistance to the patient in case of an emergency. An ERS could be fixed (e.g. hospital) or mobile (e.g. caregiver). A mobile ERS is published in the BMS.

Classical access to patients request are made by reactive flowchart (Fig. 5.a), where a patient is equipped with a classical offline measuring devices with some type of alarms. Every violated alarm need to be a carried out by doctor decision. Such access is very time-consuming. Second proposed access is based on a proactive principle (Fig. 5.b), where the patient is equipped with an online measuring devices with an online connection to some kind of superior system (in our case the BMS is presented). In this case, a patient's measured data are processed on mobile monitoring station or at server. An alert will invoke when the anomaly data are founded in patient's records. Consequently the doctor is responsible to make a decision to invoke other ERSs or to remove Alarm (in case of false detection of anomaly). Such kind of behavior is based on UAS. In many of events a predicted and solved problems can save a life. The predicted patient's problems are in most cases minor in compare to a major problems detected in time where occurred.

4.1 Biomedical Data Acquisition, Processing and Proactive Reaction

Our developed BMS can currently handle two types of biomedical data:

- 12 channels wireless ECG – BlueECG (CorScience company)
- bipolar wireless ECG – corbel (CorScience company)

These data are measured, preprocessed on mobile monitoring station (PDA, embedded device, notebook), visualized on monitoring station's display (in available), sent by wireless connection to web service and stored on server for consequential access by doctors or medical personal. Used data acquisition devices provide a successful result in case of testing a developed solution. In near future we plan to use a t-shirt with equipped biosensors network (e.g. ECG, pulse, oxy, pressure).

The biomedical ECG data are continually processed (in Real Time) through a complete infrastructure of developed UAS. First false artifact recognition is made on mobile measurement stations near the patient to allow an immediate action from ERSs. The more sophisticated data analysis is made at server level. This data processing is made on the base of neuron network and fuzzy logic behavior. Unfortunately, we reach only a small level of successful false detection (patient problem detection) up to date. In this area we are expected a future impact of our solution. The low detection rate is caused by several facts. Of course the better algorithms are needed at the first, but this problem cannot be solved satisfactory in near future. Another problem is caused by a slow connection by WiFi network, because some biomedical data contain a huge amount of data. This problem is possible to solve by our PDPT framework as a part of our UAS solution. By this solving, we improve the quality of detection by a 40 % (median value of 12 channels ECG). All the same, the real time transfer rate is now still fail to reach.

4.2 wakeNsmile Application – Proactive User Adaptive System

Proactive principle can be used not only in large distributed solution for medical centers, but it can be found usable in a many other solutions. One of them we found in an application to allow for people have a happy wake up. A mobile device application was developed to solve a problem of bad wake up at morning for all the people.

Sleep is a complex process regulated with our brain and as such is driven by 24 hour biological rhythm. Our biological clocks are controlled by chemical substances that are mostly known to us [12].

Approximately two hours after we fall asleep our eyes starts to move back and forth irregularly. Based on this fact scientists divided sleep stages into two main stages REM sleep with (Rapid Eye Movement) and NREM sleep stage (Non Rapid Eye Movement). NREM sleep is divided into another four sub-stages, when with increasing number the sleep is more and more deeper.

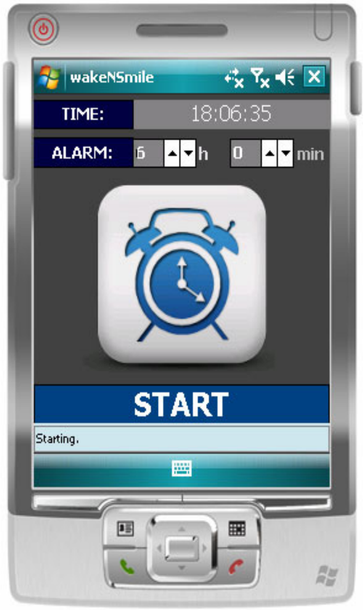
During healthy individual sleep, REM and NREM stages changes a few times. Most of the dreams are happening in REM stage. Body muscles are completely loosened and thanks' to this fact one is awaken refreshed.

During deep (NREM 3 and 4) sleep stages blood pressure is decreasing which lowers chance of cardiovascular danger. Also growth hormone is produced in its maximum in adolescent age [9]. Sleep stages are possible to divide into several: (a)

Wake (Awake), (b) REM – we dream in this stage, (c) NREM1 – falling asleep, (d) NREM2 – light sleep, (e) NREM3 – deep sleep, (f) NREM4 – deepest sleep.

wakeNsmile application (Fig. 1) was developed in C# programming language and uses .NET compact framework version 3.5, which is a special derivative of .NET framework for mobile devices [8]. Application was developed in Visual Studio 2008 Team Edition on Windows Mobile 6.5 emulator and tested on a Hewlett-Packard mobile device (originally HTC Roadster) with Windows Mobile 6.5 operating system. Minimal requirements for application running are Mobile device with Windows Mobile 6 and higher and .NET compact 3.5 or higher.

wakeNsmile application uses user control called Alarm, that has been created as a part of this project. Application is using Math.NET [10] neodym library for FIR (Finite Impulse Response) filter design and WaveIn and WaveOut libraries [11] for mobile device sound interface communication.



TIME: 18:06:35

ALARM: 6 h 0 min

Starting.

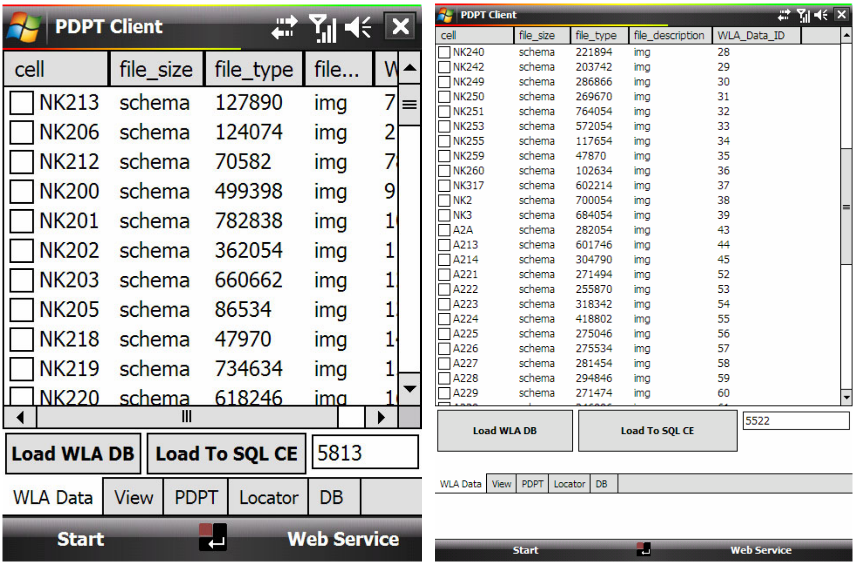
Fig. 1. wakeNsmile application example in Visual Studio 2008 Windows Mobile 6 emulator

wakeNsmile application is developed to react on users declared request in form of happy wake up at predefined time (Fig. 1). The time defined for alarm is however the latest possible time to wake up of user. We are trying to detect a body state in which the user is most able to wake up with a smile. Time period for detection analysis of state phases is declared to 30 minutes. A Fast Fourier Transformation (FFT) and some other sophisticated methods are used for it. Created application is an ideal example of user adaptive solution for mobile devices. Currently a single application is developed, but a distributed architecture version with a neural network analysis and people database is planned for future steps to be a completely embedded solution at Mobile UAS Framework. First test provide very promising results with more than 50 % of successful happy wake up of test persons at morning.

Developed application act as a proactive solution is sense of wake up of users in most suitable time.

5 Reaction on a Change of Logged User – Personal-Aware Adaptation

Next possible way to react on user needs is in classical user input processing. Based on user login a personal-aware adaptation of UAS can be defined. Well known is a model of screen resolution adaptation based on a used mobile device. Classical way is in user setting module located in used application. This however requests a user action at each time a different user is logged in.



cell	file_size	file_type	file...	W
□ NK213	schema	127890	img	7
□ NK206	schema	124074	img	2
□ NK212	schema	70582	img	7
□ NK200	schema	499398	img	9
□ NK201	schema	782838	img	1
□ NK202	schema	362054	img	1
□ NK203	schema	660662	img	1
□ NK205	schema	86534	img	1
□ NK218	schema	47970	img	1
□ NK219	schema	734634	img	1
□ NK220	schema	618246	img	1

5813

cell	file_size	file_type	file_description	WLA_Data_ID
□ NK240	schema	221894	img	28
□ NK242	schema	203742	img	29
□ NK249	schema	286866	img	30
□ NK250	schema	269670	img	31
□ NK251	schema	764054	img	32
□ NK253	schema	572054	img	33
□ NK255	schema	117654	img	34
□ NK259	schema	47870	img	35
□ NK260	schema	102634	img	36
□ NK317	schema	602214	img	37
□ NK2	schema	700054	img	38
□ NK3	schema	684054	img	39
□ A2A	schema	282054	img	43
□ A213	schema	601746	img	44
□ A214	schema	304790	img	45
□ A221	schema	271494	img	52
□ A222	schema	255870	img	53
□ A223	schema	318342	img	54
□ A224	schema	418802	img	55
□ A225	schema	275046	img	56
□ A226	schema	275534	img	57
□ A227	schema	281454	img	58
□ A228	schema	294846	img	59
□ A229	schema	271474	img	60

5522

Fig. 2. User interface layout initiated based on UAS server data. QVGA layout on a VGA display (Left – Fig. 2.a) and VGA layout on same device (Right – Fig. 2.b).

5.1 Adaptive User Interface for Mobile User Adaptive System

To prevent such waste user time, user interface adaptivity can be developed and used based only on user login information. UAS server can collect a user data such as a request of special user interface layout (font size, buttons size and locations, wide of scrollbars, etc.). After user login application is initiated in used best fitting scheme. Example of such user defined user interface is shown at (Fig. 2).

Depending on a user ability to view smaller fonts an indispensable number of other rows are viewable by user a higher resolution (Fig. 2).

5.2 Adaptive Components for Mobile User Adaptive Systems

However not every user is able to access small fonts so user interface with a large elements of user interface are welcome. Examples of such elements are described in (Fig. 3, 4). A first example presents switchers (Fig. 3.a., 3.b.). They provide a sizable intuitive way to support an adaptation on user ability. Every described element is developed as components of UAS framework. Use in any other projects is therefore very easy and comfortable.

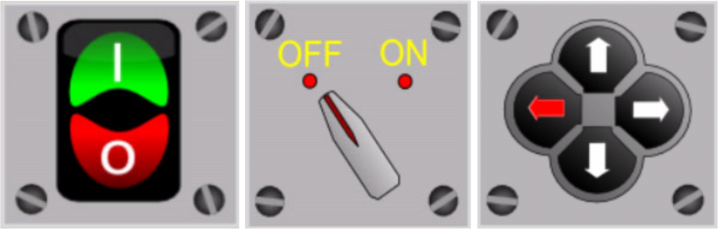


Fig. 3. User interface components: 0/1 switch (Left – Fig. 3.a), On/Off switch (Middle – Fig. 3.b) and navigation arrows where a left direction is selected (Right – Fig. 3.c)

Another component of UAS framework is navigation arrows (Fig. 3.c.), which is a sizeable component with one enumeration type of direction which can be used to easily navigate in some outdoor use cases.

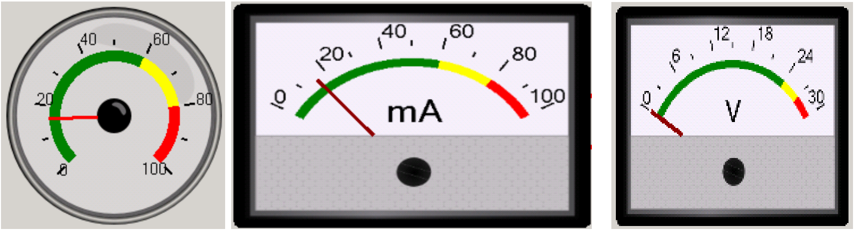


Fig. 4. User interface components of measurement visualization: value of 17 at circle visualizer (Left – Fig. 4.a), milliammeter (Middle – Fig. 4.b) and voltmeter (Right – Fig. 4.c)

Next component of UAS framework is circle visualizer (Fig. 4.a.), which is a sizeable component with two properties: color areas definition and min-max values. This component can be used to inform user about valued state of some controlled properties in the context of their boundary values. By use of this context a user can get more complex information instead of classical value information (e.g. in text/numerical form).

The last example of component is based on previous circle visualizer component, which is parent of a new component is sense of object programming model. The component can represent e.g. voltmeter (Fig. 4.b.) or milliammeter (Fig. 4.c.) as a two examples of measurement visualization component. From parent it inherits all properties and it adds a text properties for type of meter which it is represent in real case. Of course the shape is not a circle type, but it is rectangle.

5.3 Visual Indicator Components for Mobile User Adaptive Systems

The last examples of user interface components for UAS Framework are Battery and Signal Icons components [13]. Components are developed for .NET Compact Framework as well as previously described. Components visualize several cases of battery level state and wireless signal strength levels (Table 1). They all are stored in imageList. Components are designed to allow size adjusting as well as self adaptation on state of battery or signal level changes. Visual components (icons) have a more predicative value than use of classical text fields. The development resp. real states are show in (Fig. 5).

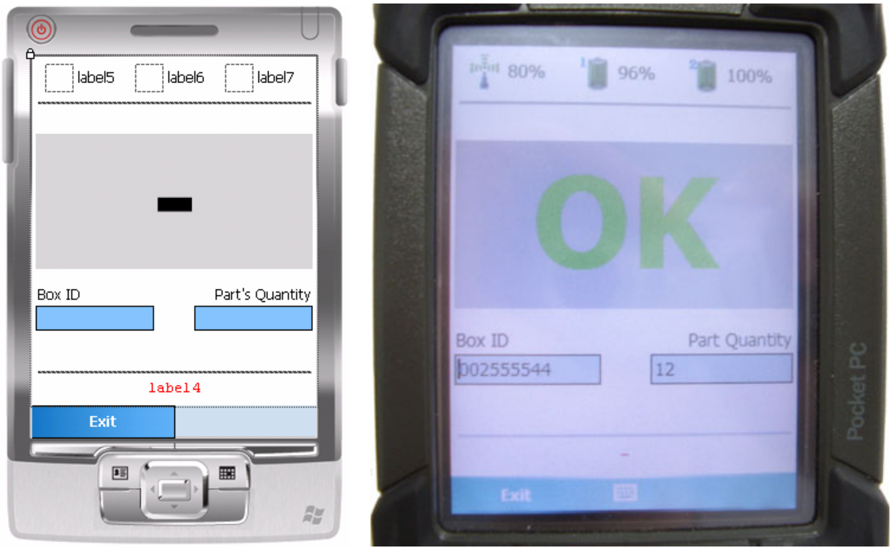


Fig. 5. Visual indicator components at development state in Visual Studion 2008 (Left – Fig. 5.a) and at real test (Right – Fig. 5.b)

Table 1. Battery and wireless signal icon components

Icon description	level [%]	Icon	level [%]	Icon
Charging – main battery	0%	1	20%	1
Battery source – main battery	30%	1	50%	1
Charging – secondary battery	10%	2	90%	2
Battery source – secondary battery	20%	2	40%	2
Wireless signal strength level (0, 10, 50, 100%)	📶	📶	📶	📶

6 The User Adaptive System Framework

A combination of a predicted user position with prebuffering of data, which are associated with physical location bears many advantages in increasing throughput of mobile devices. The key advantage of PDPT solution in compare to existing solutions [14] is that the location processing, track prediction and cache content management are situated at server side. The solution allows for managing many important parameters (e.g. AP info changes, position determination mechanism tuning, artifacts selection evaluation tuning, etc.) online at a server. By adding a *Biomedical Data Processing* solution, the Complex User Adaptive System (UAS) Framework is growing from (Fig. 6). While the whole PDPT Framework concept allow to manage a artifacts in context-awareness and time-awareness, the UAS Framework shift these possibilities to manage artifacts in biomedical context-awareness allowing a response for example to user´s non declared needs.

Biomedical Data Processing sensor at Mobile Device side of architecture (Fig. 6) collect information from user's body through a Bluetooth connection to any kind of wearable biotelemetrical devices. These data are transferred to UAS Server along with locator module data, which is processing these knowledge to act with adequate reaction in sense of user comfort improvement as a response time reducing for requested information by data prebuffering or any other reaction (e.g. screen resolution improvement, display brightness etc.).

Artifact data object can be defined as a multimedia file type in complex-awareness, which represent an object in Position Oriented Database – table *WLA_data* with time, position and biomedical-awareness. To manage locations of artifacts, firstly the

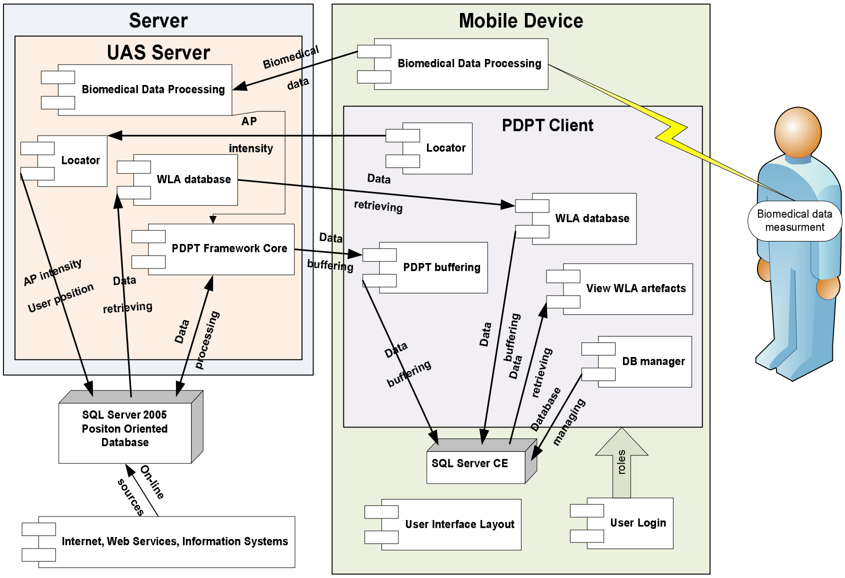


Fig. 6. User Adaptive System Framework architecture. Mobile device user is equipped with biotelemetry sensors to get info about user's body.

building map is needed [15]. The position of corporate APs is also needed to determine a user position based on a distance from each visible APs [7]. All obtained positions info need to be stored in UAS server database through a PDPT Core web service. Artifacts with position coordinates are stored in *WLA_data* table (Table 1) by use of “WLA Database Artifact Manager”. This software application was created to manage the artifacts in Position Oriented Database [8]. The PDPT prebuffering principle is well described in [16].

7 Conclusions

A concept of UAS Framework was described with main focus dealing with adaptive types of UAS on mobile devices. Coexistence of proposed solutions is in unnumbered areas and the results of complex solution are better than expected. The developed UAS Framework can be stocked on a wide range of wireless mobile devices. The localization part of PDPT framework is currently used in another project of biotelemetrical system for home care agencies to make a patient’s life safer [14]. Several areas for PDPT stocking was founded in projects of Biotelemetry Homecare. In these selected areas the use of PDPT framework is not only partial, but complete include the use of wide spectrum of wireless communication networks and GPS for tracking people and urgent need of a high data throughput on mobile wireless connected monitoring devices. Several of UAS principles can be used there also like first one described developing wakeNsmile application or several adaptive user interface components. These possibilities will also be investigated in future.

Acknowledgement. This work was supported by the Ministry of Education of the Czech Republic under Project 1M0567.

References

1. Poslad, S.: Ubiquitous Computing: Smart Devices, Environments and Interactions. John Wiley & Sons, Ltd., London (2009) ISBN 978-0-470-03560-3
2. Lewis, D., O’Sullivan, D.: Adaptive Systems for Ubiquitous Computing. In: Proceedings of the 1st International Symposium on Information and Communication Technologies. ACM International Conference Proceeding Series, vol. 49, p. 156 (2003)
3. Satyanarayanan, M.: Pervasive computing: vision and challenges. IEEE Personal Communications 8, 10–17 (2001)
4. Coen, M.H.: Design principles for inteligent environments. In: Proceedings of 15 th National / 10 th Conference on Artificial Intelligence / Innovative Applications of Artificial Intelligence, pp. 547–554 (1998)
5. Cook, D.J., Das, S.K.: How smart are our environments? An updated look at the state of the art. Pervasive and Mobile Computing 3(2), 53–73 (2007)
6. Krejcar, O.: Prebuffering as a way to exceed the data transfer speed limits in mobile control systems. In: 5th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2008, Funchal, Portugal, May 11-15, pp. 111–114 (2008)

7. Krejcar, O., Cernohorsky, J.: New Possibilities of Intelligent Crisis Management by Large Multimedia Artifacts Prebuffering. In: I.T. Revolutions 2008, Venice, Italy, December 17-19. LNICST, vol. 11, pp. 44–59. Springer, Heidelberg (2008)
8. Krejcar, O.: Problem Solving of Low Data Throughput on Mobile Devices by Artefacts Prebuffering. EURASIP Journal on Wireless Communications and Networking, Article ID 802523, 8 (2009)
9. Rechtschaffen, A.: Current perspectives on the function of sleep. Perspectives in Biological Medicine 41, 359–390 (1998)
10. Math.NET documentation, `http://mathnet.opensourcedotnet.info/doc/` (OpenSource NET Projects 2002 - 20010)
11. Recording and Playing Sound with the Waveform Audio Interface (January 2004), `http://msdn.microsoft.com/en-us/library/aa446573.aspx` (Seth Demsey – Microsoft)
12. Krejcar, O., Jirka, J., Janckulik, D.: wakeNsmile - Mobile Phone Application for Sound Input Analysis and Sleep State Detection. In: Proceedings of 2nd International Conference on Mechanical and Electronics Engineering, ICMEE 2010, Kyoto, Japan, August 01-03, vol. 2. IEEE Conference Publishing Services, NJ (2010)
13. Krejcar, O., Macecek, M.: Development of Bar Code Scanner Application for Windows Mobile Devices Using a Symbol Mobility Developer Kit for.NET Compact Framework. In: Proceedings of 2nd International Conference on Mechanical and Electronics Engineering, ICMEE 2010, Kyoto, Japan, August 01-03, vol. 2. IEEE Conference Publishing Services, NJ (2010)
14. Krejcar, O., Janckulik, D., Motalova, L., Kufel, J.: Mobile Monitoring Stations and Web Visualization of Biotelemetric System - Guardian II. In: Mehmood, R., et al. (eds.) EuropeComm 2009. LNICST, vol. 16, pp. 284–291. Springer, Heidelberg (2009)
15. Krejcar, O.: Full Scale Software Support on Mobile Lightweight Devices by Utilization of all Types of Wireless Technologies. In: Granelli, F., Skianis, C., Chatzimisios, P., Xiao, Y., Redana, S. (eds.) Mobilight 2009, May 18-20. LNICST, vol. 13, pp. 173–184. Springer, Heidelberg (2009)
16. Krejcar, O.: Complex Mobile User Adaptive System Framework for Mobile Wireless Devices. In: Mobilight Workshops 2010, Barcelona,Spain, June 30 – July 02. Social-Informatics and Telecommunications Engineering, LNICST, Springer, Heidelberg (2010)

International Workshop on Mobile Multimedia Networking (IWMMN 2010)

Bootstrapped Low Complexity Iterative Decoding Algorithm for Low Density Parity Check (LDPC) Codes

Albashir Mohamed[1], Maha Elsabrouty[1], and Salwa El-Ramly[2]

[1] Arab Academy for Science, Technology and Maritime Transport, Electronics and Comunications Engineering, 2033 Al Horriyah, Heliopolis, Cairo, Egypt
albashir.mohamed@staff.aast.edu, maha2000_eg@yahoo.com
[2] Ain Shams University, Faculty of Engineering, Electronics and Communications Engineering Dept., 11517 Abbassia, Cairo, Egypt
sramlye@netscape.net

Abstract. RRWBF (Reliability ratio based weighted bit-flipping) algorithm is one of the best hard decision decoding algorithms in performance. Recently several modifications are done to this technique either to improve performance or to lower the complexity. The IERRWBF (Implementation efficient reliability ratio based weighted bit-flipping) is developed targeting decreasing processing time of the decoding process. Low Complex IERRWBF (Low complex implementation efficient reliability ratio based weighted bit-flipping) algorithm is one of the latest algorithms targeting lowering decoding complexity, by decreasing number of iterations required to decode received code words and to solve the problem faced by IERRWBF which is the exponential increase in complexity as maximum number of iterations increases. In this paper we are targeting improving the performance of recent developed algorithm named Low Complex IERRWBF by adding a bootstrap step to the decoding technique which leads to increase in reliability of received bits then the number of decoded bits will be increased leading to improvement in performance.

Keywords: Bootstrapped Low Complex IERRWBF (Bootstrapped low complex implementation efficient reliability ratio based weighted bit-flipping); low density parity check (LDPC) codes; reliability ratio based weighted bit-flipping (RRWBF).

1 Introduction

Low density parity check (LDPC) coding is a prominent channel coding technique that has excellent performance approaching Shannon limit. It was proposed by Gallager in 1963 [1]. It has received increased attention and became a strong competitor to turbo codes as many digital communication systems adopted it as error control coding technique such as DVD-S2 [11].

LDPC codes can be decoded using various types of decoding techniques. There are three categories of decoding techniques for LDPC codes which are hard decision, soft

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 377–389, 2010.

decision and hybrid decision decoding techniques. The focus of this paper is on hard decision decoding techniques. Another naming of the hard-decision decoding is the bit-flipping family which was first presented in [1]. Modifications to the basic bit-flipping began by the WBF (Weighted Bit-Flipping algorithm) [2], which targeted to increase the performance of the basic bit-flipping algorithm, through a measure of the reliability of received symbols in their decoding algorithm.

Another modification was done for further improvement in the performance which is the MWBF (Modified Weighted Bit-Flipping algorithm) [3], which includes a weighting factor α in the weighted check sum equation of the WBF (Weighted Bit Flipping algorithm) to include the intrinsic message for each bit in this equation, the value of α is determined by simulation for every SNR and column weight of each matrix. To solve the problem of predetermining α for each SNR and column weight of each matrix, RRWBF (reliability ratio based weighted bit-flipping algorithm) was proposed in [4] to solve this problem and also to increase the performance. The main drawback in RRWBF is the long processing time, so IERRWBF (implementation efficient reliability ratio based weighted bit-flipping algorithm) was proposed in [5] to solve this problem as optimization was applied to the technique and a reduction in the processing time was accomplished. Another modification done to IERRWBF to reduce the complexity without affecting the performance which led to the Low Complex IERRWBF proposed in [6]. The Low Complex IERRWBF is based on observing the check node step, and for certain number of repeated syndrome words the decoding is ended declaring failed decoding rather than wasting more computational power if decoding is continued.

In this paper, a modification is done on Low Complex IERRWBF using a bootstrap step following the work done in [7] targeting increasing the performance of Low Complex IERRWBF to be one of the best hard decision decoding techniques which will be named bootstrapped low complex implementation efficient reliability ratio based weighted bit-flipping.

The rest of the paper is organized as follows. Section (2) presents the background of bit flipping algorithms especially the Low Complex IERRWBF. Section (3) presents the modification done to the Low Complex IERRWBF. Section (4) presents the results obtained compared to the original work in [6]. Section (5) illustrates conclusion and future work.

2 A Brief Review on Bit-Flipping Algorithms

The bit-flipping algorithms started with the work of Gallager in his PhD [1] which contains the basic bit-flipping algorithm. A lot of modifications done to the basic bit-flipping algorithm targeting lowering complexity or contributing performance, these algorithms will be illustrated in details in the following sections starting with the basic bit-flipping algorithm.

The following table summarizes the basic variables used in the bit flipping algorithm.

Table 1. Symbols used in algorithms

Symbol	Definition
$\mathbf{H}_{mn}$	mth row, nth column of parity-check matrix **H**
r_n	nth bit received from the channel
z_m	Hard decision of r_m
s_m	Syndrome of hard decision bit z_m
E_n	weighted check sum that is orthogonal on the code bit *n*
R_{mn}	Reliability Ratio
N (m)	The set of variable nodes that participate in mth check node
M (n)	The set of check nodes in which nth variable node participates

2.1 Basic Bit-Flipping Algorithm

The Basic bit-flipping algorithm [1] is the lowest in complexity compared to its variants and the simplest to be implemented, the algorithm procedures will be illustrated in the following steps:

Step 1: Compute the parity-check sums using eq. (1). If all parity-check equations are satisfied, stop the decoding.

$$s_m = \sum_{n=1}^{N} z_m \mathrm{H}_{mn} \ . \tag{1}$$

Step 2: Find the number of unsatisfied parity check equations for each code bit position denoted as $\boldsymbol{f}_i$ where $\boldsymbol{i} = 0,1,.......\boldsymbol{n} - 1$.

Step 3: Identify the set Ω of bits where f_i is the largest.

Step 4: Flip the bits in Ω.

Step 5: Repeat steps1 to 4 until all parity-check equations are satisfied or a maximum number of iterations is reached.

2.2 Weighted Bit-Flipping Algorithm

The simple bit-flipping can be improved to achieve better error performance by including some kind of reliability information of the received symbols in their decoding algorithm [2]. However, more additional decoding complexity is required to achieve such improvement in the performance.

The algorithm is mainly based on computing weighted check sum using eq. (2) to determine which bit will be flipped.

$$E_n = \sum_{s_m^{(l)} \in S_n} (2s_m^{(l)} - 1) \mid y_m \mid_{\min}^{(l)} . \tag{2}$$

The algorithm is briefly discussed through the following steps:

Step 1: Compute the check sums. If all the parity-check equations are satisfied, stop the decoding.
Step 2: Compute E_n based on eq. 2, for $0 \leq l \leq n - 1$.
Step 3: Find the bit position l for which E_n is the largest.
Step 4: Flip the bit z_n.
Step 5: Repeat steps1 to 4 until all parity-check equations are satisfied or a maximum number of iterations is reached.

2.3 Modified Weighted Bit-Flipping Algorithm

After observing the weighted bit-flipping algorithm and its limited performance, the modified weighted bit-flipping algorithm improves performance better than the weighted bit-flipping algorithm [3]. The algorithm is based on considering the check constraint messages and the intrinsic message for each bit.

$$E_n = \sum_{m \in M(n)} (2s_m - 1) \mid y \mid_{\min} - \alpha \mid y_n \mid \tag{3}$$

The weighting factor α is a real number and α ≥ 0. When α = 0, the modified weighted bit-flipping is converted into standard weighted bit-flipping algorithm. For a given LDPC code, the optimal α at a given SNR may be defined as the value for which the modified weighted bit-flipping algorithm generates the smallest BER for decoding this LDPC code at that SNR. The optimal value of α at each SNR for a given LDPC code can be determined through simulation.

The decoding procedures for this algorithm are the same as for WBF (weighted bit-flipping algorithm) except the weighted check sum equation used in determining which bit will be flipped.

2.4 Reliability Ratio Based Weighted Bit-Flipping Algorithm

The reliability ratio based weighted bit-flipping decoding for LDPC [4] performs very efficiently among the existing bit-flipping algorithms that have appeared. As it was shown before that weighted bit-flipping and modified weighted bit-flipping algorithms have some drawbacks. As for the weighted bit-flipping, it considers only the parity-node based information during the evaluation of the error term (*En*). The modified bit-flipping decoding algorithm performs better than the weighted bit-flipping algorithm since it considers both the check-node based and the message-node based information during calculation of error term (*En*). However, a drawback of the modified bit-flipping algorithm is its dependence on α, so it is required to find optimum α for each individual SNR. Both the weighted bit-flipping and modified weighted bit-flipping considers the specific check-node based information. However, all message nodes participating in the *m*th parity check are contributing.

Hence, a new quantity is defined named as "Reliability Ratio" and is given by:

$$R_{mn} = \beta \frac{|y_n|}{|y_m^{\max}|}. \tag{4}$$

where $|y_m^{\max}|$ represents the highest soft magnitude of all message nodes participating in the m^{th} parity check. The factor β is the normalization factor to ensure that:

$$\sum_{n:n\in N(m)} R_{mn} = 1. \tag{5}$$

The error- term is calculated from:

$$E_n = \sum_{m\in M(n)} \frac{(2s_m - 1)}{R_{mn}}. \tag{6}$$

The decoding procedures is the same as modified bit-flipping except for non-computing of α due to using of reliability ratio instead of it to determine which bit is reliable and which one is non-reliable for the flipping process.

2.5 Implementation Efficient Reliability Ratio Based Weighted Bit-Flipping Algorithm

Reliability ratio based weighted bit-flipping algorithm is an efficient hard decision decoding algorithm but it has a main drawback which is the long decoding time taken by the algorithm. Implementation efficient reliability ratio based weighted bit-flipping was proposed in [5] to solve this problem by optimizing the algorithm to reduce the decoding time to be suitable for simulation and hardware implementation especially when the maximum number of iterations assigned for the algorithm is small.

The algorithm is divided into four steps: initialization step, variable node step, decision step and check node step. The operation done in each step is shown as follows:

Initialization step:

$$T_m = \sum_{n\in N(m)} |r_n|. \tag{7}$$

Variable node step:

$$E_n = \frac{1}{|r_n|} \sum_{m\in M(n)} (2s_m - 1)T_m. \tag{8}$$

Decision step: Flip the bit z_n for **n** = arg max$_n$ $\mathbf{E}_n$.

Check node step:

$$s_m = \sum_{n=1}^{N} z_m \mathrm{H}_{mn}. \tag{9}$$

2.6 Low Complex Implementation Efficient Reliability Ratio Based Weighted Bit-Flipping Algorithm

One of the drawbacks of the IERRWBF algorithm as stated in [5], which the author failed to solve is that the algorithm spends a larger percentage of the time at the variable node step and the check node step. As the maximum iteration number assigned for decoding increases, delay increases without any significant improvement in the performance.

In case of channels with low Signal-to-Noise Ratio (SNR) having AWGN, the hard-decision based bit-flipping algorithms sometimes fails to decode the received codeword correctly, as the decoding in such a case causes the syndrome vector to be a non-zero vector, which leads to failed decoding of the received codeword. Even with large iteration number assigned for the algorithm (500 iterations and more), there will be no significant improvement in the error performance. It has been observed that in such low SNR the syndrome vector will continue to be a non-zero vector and the decoding will keep flipping endlessly consuming computational power with no improvement throughput. The low complex implementation efficient reliability ratio based weighted bit-flipping decoding algorithm is based on the original scheme described as in [5]. The initialization, decision and variable node steps are the same as the original algorithm. However, the modification is in the check node step. Such a modification has a great effect on significant reduction of the decoding time.

The check node step used in all bit-flipping algorithms is a mere syndrome check condition that checks if the decoded codeword is a valid codeword or not. If the syndrome vector is all-zero vector, then the decoded codeword is a valid codeword and if the syndrome vector is not all-zero vector, then the decoded codeword is not decoded correctly and the algorithm continues till the syndrome becomes all-zero vector or the maximum number of iterations is achieved without fulfilling the syndrome condition. As explained, such a condition is rendered in case of low SNR. However, what is required in such cases is to try decreasing the number of iterations required at each SNR for decoding any received codeword to a limited number of iterations as extending the algorithm to more number of iterations will not achieve any observable or enhanced performance. This is done by adding an extra conditional step to examine such situations and control the iteration loop by deciding whether to continue decoding or exit the iteration loop and stop the algorithm to have a final decoded codeword. So a mechanism of the added condition will be started by obtaining syndrome vector at the end of each iteration and be stored in 3-entry register. The register is chosen to be of minimum size equal to 3 precisely because we need at least 2 iterations to get the same decoded codeword and the same syndrome vector, starting from initial vector, if the same bit in the decoded codeword is being flipped two times to return to the initial state, so the 3 entries correspond to the initial syndrome (initial vector) in the register, the syndrome vector after first iteration and the syndrome vector after second iteration. So, size 3 is the minimum size of register that could be used for storing syndrome vector for comparing between the first and the third (last one) where each new entry is stored at the top of the register and the remaining entries are shifted down to remove the last entry. So when the point of oscillation is reached which means that no contribution to performance will occur while continuing to the maximum number of iterations, so decoding failure is declared and decoding is halted, this added condition significantly reduces the complexity without any effect on performance compared to IERRWBF algorithm.

3 Bootstrapped Low Complex Implementation Efficient Reliability Ratio Based Weighted Bit-Flipping Algorithm

Low complex implementation efficient reliability ratio based weighted bit-flipping is the last modification done to reliability ratio based weighted bit-flipping targeting lowering complexity. Our goal is to increase the performance of low complex implementation efficient reliability ratio based weighted bit-flipping which will be in this case a very efficient hard decision decoding technique.

The idea of the algorithm comes from the work done in [7]. In [7] a modification to the weighted bit-flipping algorithm is done by adding a bootstrap step to the decoding algorithm which leads to significant contribution in performance. Bootstrap step will firstly be discussed for further explanation of the idea of using bootstrap step to contribute to performance.

Before explaining the bootstrap step, we need to make a few definitions: we call a received value and its corresponding variable node "unreliable" if $| y_n |< \alpha$, for some predetermined value α, and "reliable" otherwise. A check node is referred to as "reliable" with respect to an unreliable variable node if all the other variable nodes connected to that check node are reliable.

We initiate the decoding process by identifying and erasing all the unreliable variable nodes. We then assign improved values and reliabilities to the erased bits by passing them messages from the reliable variable nodes through the reliable check nodes. The new value y'_n that substitutes y_n for an erased variable is computed as

$$y'_n = y_n + \sum_{m \in M_r(n)} \Big(\prod_{n' \in N(m)\backslash n} \operatorname{sgn}(y_{n'}) \Big) \min_{n' \in N(m)\backslash n} \left| y'_n \right| . \tag{10}$$

where $M_r(n)$ denotes the set of the reliable check nodes connected to the unreliable bit node *n,* all this operation is illustrated using fig. 1. If there is no reliable check node connected to an unreliable bit node, that node keeps the original received value y_n. The algorithm then proceeds with the conventional algorithm using the improved value y'_n.

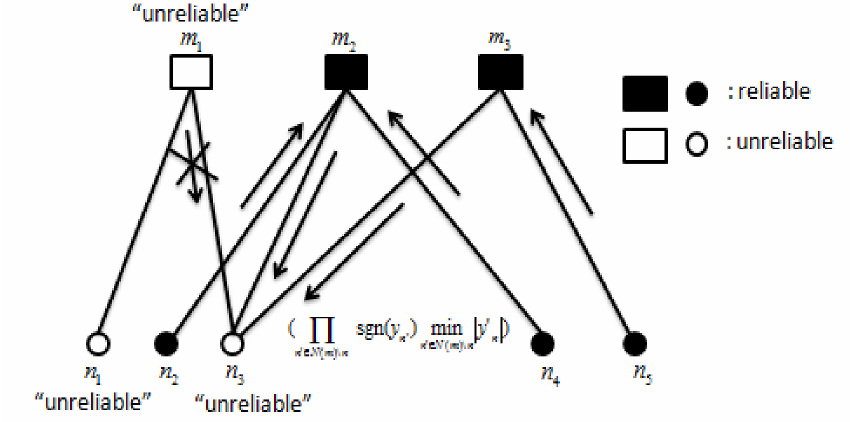


Fig. 1. Bootstrap decoding procedure over tanner graph

After studying and observing the bootstrap step in the bootstrapped low complex implementation efficient reliability ratio based weighted bit-flipping it was found that it combines horizontal step with vertical step of min-sum algorithm in one equation to be processed individually on the unreliable bits predetermined using certain threshold (α).

The embedding of bootstrap step in the weighted bit-flipping decoding algorithm leads to significant increase in reliability of received codeword that will lead to increase in performance. Our idea is to do the same for low complex implementation efficient reliability ratio based weighted bit-flipping algorithm to increase its performance, resulting a very good hard decision decoding technique that have two main properties; low complexity and high performance, then new algorithm is created

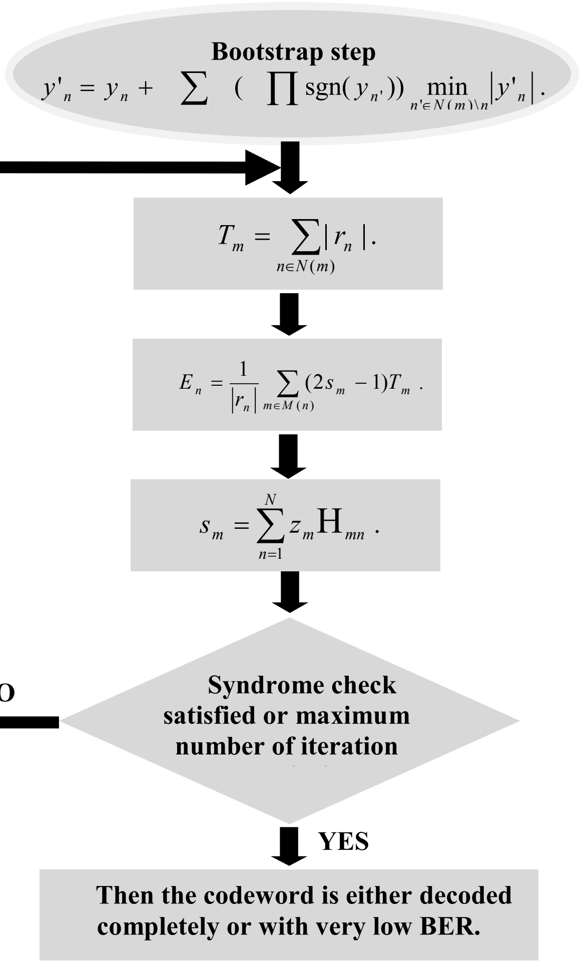

Bootstrap step
$y'_n = y_n + \sum (\prod \mathrm{sgn}(y_{n'})) \min_{n'\in N(m)\backslash n} |y'_n| .$
$T_m = \sum_{n\in N(m)} | r_n | .$
$E_n = \frac{1}{|r_n|} \sum_{m\in M(n)} (2s_m - 1)T_m .$
$s_m = \sum_{n=1}^{N} z_m \mathrm{H}_{mn} .$
NO
Syndrome check satisfied or maximum number of iteration
YES
Then the codeword is either decoded completely or with very low BER.

Fig. 2. Flow chart of Bootstrapped Low Complex IERRWBF decoding algorithm

which is called bootstrapped low complex implementation efficient reliability ratio based weighted bit-flipping. After processing on the bootstrap step the Low Complex IEERRWBF algorithm begins decoding the improved received codeword as mentioned in the last section, Also the whole decoding process is described in fig. 2 that contains flow chart of the new algorithm.

4 Simulation Results

The simulations are performed using *(n,k,v)* regular LDPC codes where *n* stands for the codeword length, *k* is the message length, and *v* is the number of '1's per column in the parity check matrix **H.** Progressive edge growth (PEG) and Gallager LDPC codes were used as the construction methods for generating the parity check matrix **H** of these regular LDPC codes. The regular LDPC codes used in the simulation were PEG (504,252,3), Gallager (504,252,3) and Gallager (204,102,3) by decoding each LDPC code using WBF, LLR-BP, IERRWBF algorithm presented in [5], Low Complex IERRWBF algorithm presented in [6] and the proposed algorithm in this paper. All codes used are of rate ½. The coded data are then modulated using a BPSK modulation scheme and sent over a channel having AWGN to simulate the effect of real wireless channel. The simulations were run on MATLAB.

The figures show the observable contribution in error performance. Fig. 3 shows the performance of decoding PEG (504, 252, 3) LDPC codes with 25 maximum iterations using WBF, Low Complex IERRWBF and Bootstrapped Low Complex IERRWBF. Also 15 maximum iterations assigned for Bootstrapped Low Complex IERRWBF for further proof of superiority of the new algorithm. Only 5 maximum iterations are assigned for LLR-BP. It is clear that the performances of the LLR-BP and the Bootstrapped Low Complex IERRWBF have the best performance. The high performance of the new technique is due to the insertion of bootstrap step, which combines vertical step with horizontal step of min-sum algorithm, which increases reliability of received code word leading to high performance as shown in the fig. 3.

Fig. 4 shows the performance of Gallager (504,252,3) LDPC codes decoded with 25 maximum iterations using WBF, Low Complex IERRWBF and Bootstrapped Low Complex IERRWBF. Also 15 maximum iterations assigned for Bootstrapped Low Complex IERRWBF for further proof of superiority of the new algorithm. Only 5 maximum iterations are assigned for LLR-BP. From the results the performance of Gallager code is inferior to that of the PEG code, but still the new algorithm is superior over the original work which is low complex implementation efficient reliability ratio based bit-flipping.

Fig. 5 shows the performance of Gallager (204,102,3) regular LDPC code when decoded with Low Complex IERRWBF and Bootstrapped Low Complex IERRWBF for 5 maximum iterations. It is clear that the performance of Bootstrapped Low Complex IERRWBF is better than that of the Low Complex IERRWBF at the same number of iterations. With increasing number of iterations for Bootstrapped Low Complex IERRWBF decoding to 25 iterations, the difference in performances between the Low Complex IERRWBF and Bootstrapped Low Complex IERRWBF is significantly increased. This shows that the Bootstrapped Low Complex IERRWBF decoding is better than that of the Low Complex IERRWBF decoding in terms of performance.

Table 2 shows the decoding time of IERRWBF, Low Complex IERRWBF and Bootstrapped Low Complex IERRWBF using machine with Intel® core™ 2 Duo T7200 @ 2 GHz with 2000 MB memory and 32-bit operating system is used. The results show that the proposed algorithm has extremely reduced complexity compared with original algorithm IERRWBF. As shown the complexity of Low Complex IERRWBF has the lowest algorithm but is inferior in performance compared by proposed algorithm as shown in fig. 2, fig. 3 and fig. 4. So the proposed algorithm is superior in performance with complexity close to Low Complex IERRWBF which is massively reduced compared with IERRWBF algorithm.

According to the obtained results the new technique proves that it is superior over all existing hard decision decoding techniques. Also the new technique combines between low complexity as it is created from low complex technique which is low complex implementation efficient reliability ratio based weighted bit-flipping and high performance, due to using of bootstrap step that enhanced the received codeword, so at same Eb/No and same number of iterations performance increased as shown in the obtained results.

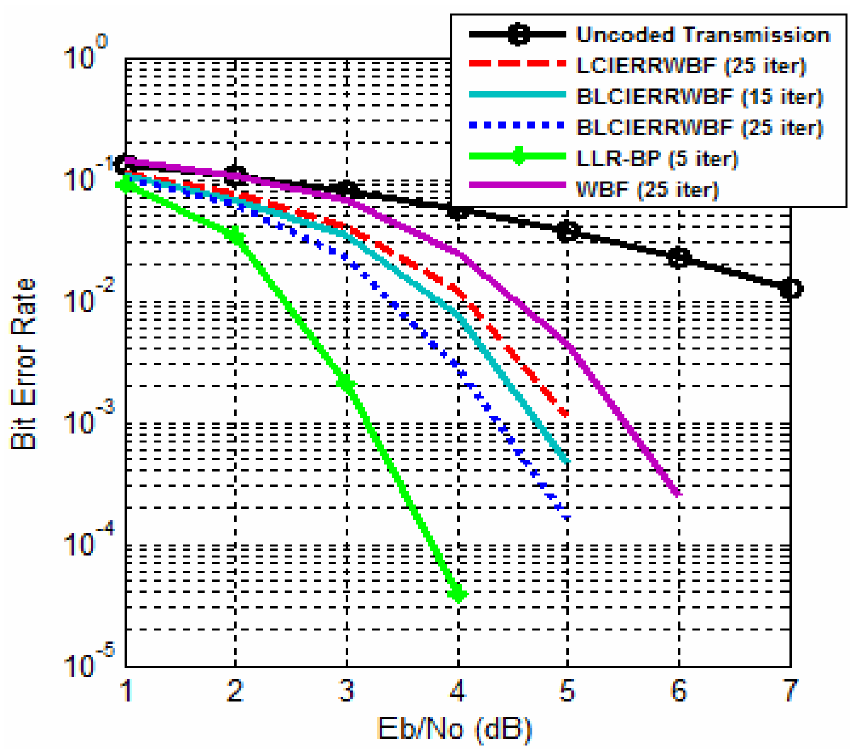


Fig. 3. BER for (504,252,3) PEG-LDPC code decoded by WBF, Low Complex IERRWBF, Bootstrapped Low Complex IERRWBF, and LLR-BP

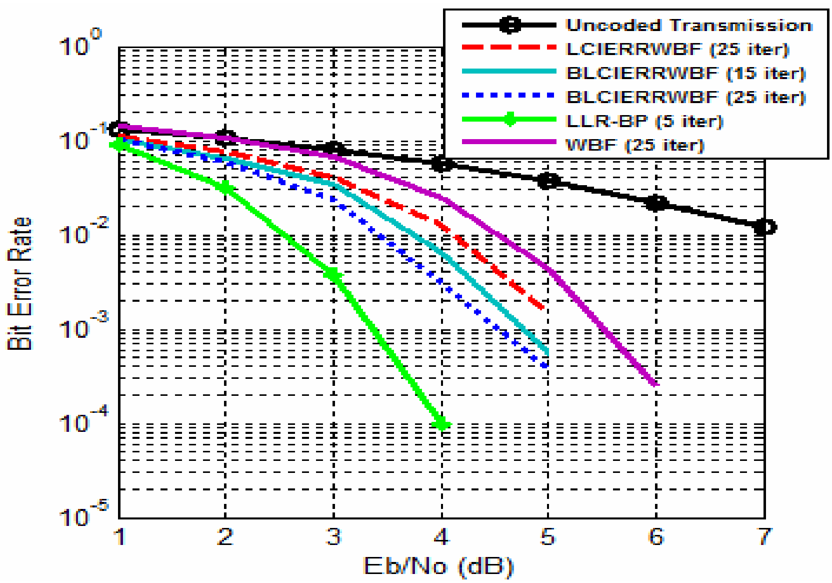


Fig. 4. BER for (504,252,3) Gallager LDPC code decoded by WBF, Low Complex IERRWBF, Bootstrapped Low Complex IERRWBF, and LLR-BP

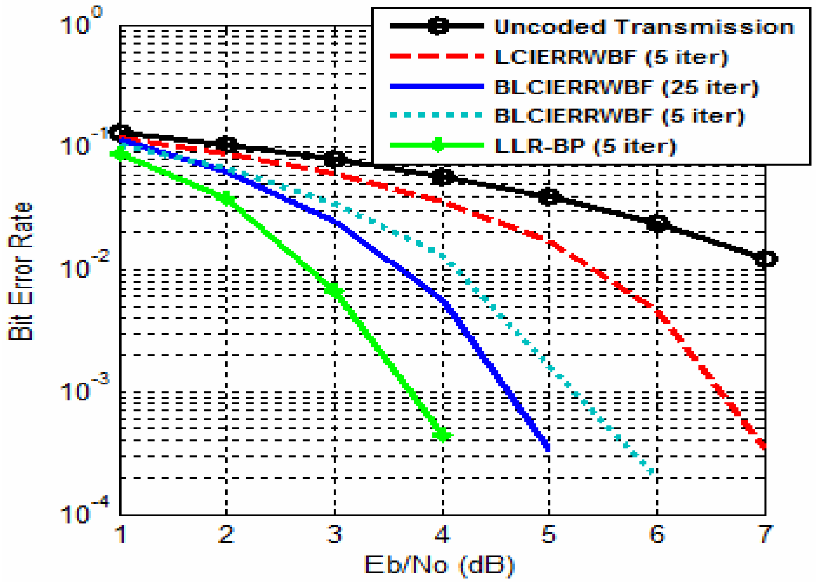


Fig. 5. BER for (204,102,3) Gallager LDPC code decoded by Low Complex IERRWBF, Bootstrapped Low Complex IERRWBF and LLR-BP

Table 2. Decoding time of presented decoding techniques compared with IERRWBF

SNR(dB)	1	2	3	4	5	6
IERRWBF	19.879	17.855	17.691	17.564	10.247	9.567
LCIERRWBF	0.0352	0.0256	0.0253	0.0252	0.0231	0.0212
BLCIERRWBF	0.2945	0.2643	0.2295	0.2063	0.1593	0.1375

5 Conclusion and Future Work

In this paper a new algorithm is created by modifying low complex implementation efficient reliability ratio based weighted bit-flipping by adding bootstrap step which leads to very good performance and very low complexity. The new algorithm is named bootstrapped low complex implementation efficient reliability ratio based weighted bit-flipping algorithm, which has very low complexity compared to IERRWBF and comparable complexity compared to low complex IERRWBF, also its performance is superior to IERRWBF and low complex IERRWBF. So the new algorithm has very low complexity and superior performance over hard decision decoding algorithms. As the performance of the proposed algorithm is superior over all hard decision decoding algorithms it is still inferior to the soft decision decoding algorithm represented by LLR-BP but it has very high complexity if it is compared to the proposed algorithm.

As for future work, while the new algorithms have very good performance and very low complexity it need more adjustments, where the value of α is obtained through simulation for each column weight and SNR, so before proceeding in decoding, α must be computed first. Then the new algorithms need a new mechanism instead of α to define the unreliable bits and the reliable ones to solve the problem of pre-computing of α before proceeding in decoding the received codeword. Also the complexity of the algorithms needs to be minimized, as the complexity of the algorithms is increased due to the insertion of bootstrap step which leads to more processing time as received codeword reliability increased. Then more adjustments to the algorithm are needed to minimize complexity.

Acknowledgment

The authors wish to thank D. J. MacKay for providing them with the parity-check matrices of the LDPC codes.

References

1. Gallager, R.G.: Low-Density Parity-Check Codes. MIT Press, Cambridge (1963)
2. Kou, Y., Lin, S., Fossorier, M.: Low density parity check codes based on finite geometries: A rediscovery and more. IEEE Trans. Inform. Theory 47, 2711–2736 (2001)

3. Zhang, J., Fossorier, M.P.C.: A modified weighted bit-flipping decoding of low-density parity-check codes. IEEE Communication Letters 8(3), 165–167 (2004)
4. Guo, F., Hanzo, L.: Reliability ratio based weighted bit-flipping decoding for low-density parity-check codes. Electronics Letters 40(21), 1356–1358 (2004)
5. Lee, C.-H., Wolf, W.: Implementation-efficient reliability ratio based weighted bit-flipping decoding for LDPC codes. Electronics Letters 41(13) (June 23, 2005)
6. Zeidan, H.R., Elsabrouty, M.M.: Low Complexity Iterative Decoding Algorithm for Low-Density Parity-Check (LDPC) codes. In: Wireless Days. WD 2008. 1st IFIP, pp. 1–5 (2008)
7. Nouh, A., Banihashemi, A.H.: Bootstrap decoding of low-density parity-check codes. IEEE Comrrtun. Len. 6(9), 391–393 (2002)
8. Kou, Y., Lin, S., Fossorier, M.P.C.: Low-density parity-check codes based on finite geometries: A rediscovery and new results. IEEE Trans. Inform. Theory 47, 2711–2736 (2001)
9. MacKay, D.J.: Encyclopedia of Sparse Graph Codes, `http://www.inference.phy.cam.ac.uk/mackay/codes/data.htm`
10. Inaba, Y., Ohtsuki, T.: Performance of Low Density Parity Check (LDPC) Bootstrap Decoding Algorithm on a Fast Fading Channel. In: IEEE Vehicular Technology Conference, pp. 333–337 (2004)
11. Digital video broadcasting (DVB); User guidelines for the second generation system for broadcasting, interactive services, news gathering and other broad-band satellite applications (DVB-S2). European Telecommunications Standards Institute (ETSI), TR 102 376

Quick Prototying of Multifacete Interface for Healthcare Wireless Sensor Network

Rahul Dubey, Kalyani Divi, and Hong Liu

University of Massashusetts Dartmouth
Department of Electrical And Computer Enginering
285 Old Westport Road, North Dartmouth, MA 02740, USA
{rdubey,KDivi,hliu}@UmassD.edu

Abstract. This paper presents a quick prototyping of a mobile wireless sensor network (WSN) in healthcare applications that provides multiple interfaces with various access rights to different personnel involved in medical systems such as patients, healthcare providers, and system administrators. The prototype demonstrates a seamless access from constantly sensing patients' vital signs to long term medical records stored in central database. A quick prototyping approach facilitates communications and understanding of the system requirements among the stakeholders and is a cost-effective way to build a functioning massive ubiquitous healthcare infrastructure.

Keywords: Human-computer interface; Mobile Wireless Sensor Network (WSN); Healthcare Sensor Network (HSN); security and privacy of mobile and wireless systems.

1 Introduction

The National Vital Statistics of death rate in United States of America states that the two major causes of death are heart disease and stroke. According to it, the majority of the deaths due to heart disease were 26% and the Cerebrovascular diseases (stroke) were 23.1% [1]. To trim down the number of deaths due to heart disease, many measures were adopted. To comprehend about the ways and to improvise the quality of health care, Electronic Health Care system is introduced. Primarily, the system focuses on the elderly and physically disabled patients [2]. In the system, the doctors can follow the records of the patient and patient does not need to see a doctor whenever he/she requires doing so. In this kind of system, there is a very vast use of wireless sensor nodes. Wireless sensor network itself have very wide range of application. Sensor network, nowadays are very prominent in Surveillance, Health care, Traffic monitoring and Military. Sensor network has immense prospective in Medical Industry.

Many applications on wireless sensor networks (WSN) have been proposed, and medical healthcare is in particular interest of the nation. The first sensor network application designed for medical and healthcare industry is Codeblue [3]. Codeblue is for emergency medical care and is used to monitor a patient's heart rate, blood oxygen saturation as well as ECG through the use of Berkley MICA2 motes, a pulse oxygen meter, and an ECG mote. The vital signs from the patient can be either

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 390–399, 2010.

transmitted via multi-hop communication to a wired base-station or to PDA devices carried by hospital staff or EMTs. The infrastructure incorporates routing, node naming and discovery. Codeblue uses an ad-hoc network to collect data from patients and then deliver the data to an information panel. However, security is not yet integrated. The researchers suggest using an ECC-based security protocol. The major drawback of Codeblue is its lack of security in its original architectural design.

Alarmnet [4] is another notable project that monitors assisted-living and residential patients. In terms of hardware, the infrastructure consists of several sensors (infrared, dust, integrated temperature, light, pulse, and blood oxygen), Star gateways, PDAs as well as computers. One of the major components of Alarmnet is the Star gateway, holding the code for the Alarm Gate module that handles most of the security functionality. The security services include the Secure Remote Password (SRP) protocol, authentication, and secure communication. It also takes care of message handling, query and report parsing, and database access. The main issue with Alarmnet is its platform-dependency.

Medical sensor network architecture called SNAP (Sensor Network for Assessment of Patients) [5] is proposed to address the security challenges of sensor networks. The infrastructure does not address routing, mobility or congestion issues. It deploys security mechanisms consisting of ECC-based secure key exchange protocol, symmetric encryption and decryption to protect data integrity, and two-tier authentication scheme using patient biometrics. SNAP uses two types of nodes: limited power node and unlimited power node. An unlimited power node would be active most of the times, and a limited power node goes into sleep state when inactive. Unfortunately, SNAP becomes ineffective due to its over-conservation of energy.

It is urgent to systematically address security issues in Healthcare Sensor Network (HSN) applications [6]. However, before a potential security mechanism can be integrated into a HSN, we must understand the measurement to assess and evaluate the security requirements and goals for the healthcare application and develop the architecture to house the security schemes/protocols. Divi et al have proposed a secure architecture for healthcare wireless sensor networks that put security in the design rather than patch work [7]. This paper shows a quick implementation that prototypes the architecture to demonstrate its feasibility and to study its efficiency.

2 Divi's Secure Architecture

In healthcare applications, sensor nodes are deployed to monitor patients and assist disabled. Our research is focused on designing a wireless sensor network that collects, transmits, and processes sensitive patient information for medical personals to monitor patients in real time. Since security is of significant challenge in transmitting data wirelessly and timely, we propose security architecture to support mobile healthcare infrastructure. Our approach is unique in that we place security in the center of the architectural design. The goals of our research are to

- Develop an architecture that positions security as a core component targeted to healthcare applications,
- Implement a security structure that provides low latency encryption and decryption, and
- Design security algorithms that are not resource intensive, permitting its deployment on sensor nodes.

2.1 Sensor Node Architecture

Our sensor node has five blocks: Controller, Sensors, Communication Device, Power Supply, and Memory. The sensors collect the data from the outside environment. The center component is the controller which processes the data (transfer data between a sensor and the memory as well as compute the data if necessary). The communication device transfers the data among the network. The Memory stores any short results or some important configuration information. The Power supply supports all these components. It is shown in Figure 1 below:

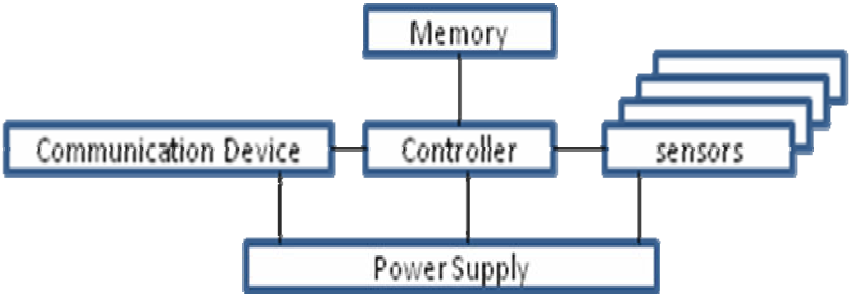


Fig. 1. Wireless Sensor Node Architecture

A wireless sensor network (WSN) has sensor nodes deployed in the environment strategically and communicating remotely as well as wirelessly to base station. Base station can be a router, a desktop, or another sensor node that communicates over the Internet to transfer data to different devices like PDA (Personal Data Acquisition) and laptops for data comprehension. For design considerations, the choice of base station effects cost and range of sensor node which in turn decides the structure of the network.

2.2 Patient Monitoring System (PMS)

The main design of the HSN architecture lies in the Patient Monitoring Network (PMN). The separate design of the PMN provides patient mobility. In the situations of emergency, the PMN can be set up in the ambulance and the aggregation node will be talking to the base station at the hospital. As the ambulance would be moving in and around the locations nearby to the hospital, the aggregation node can be maintained the same as for the body range. In the situations, like where a patient has to be monitored continuously, he can wear this PMN and keep contact with the base station even though he is moving around.

Our further research work has to be carried out on the choice of this aggregation node and the optimized design of a protocol so that this aggregation node utilizes less power. It makes the proposed architecture versatile in various conditions such as emergency or assisted living monitoring.

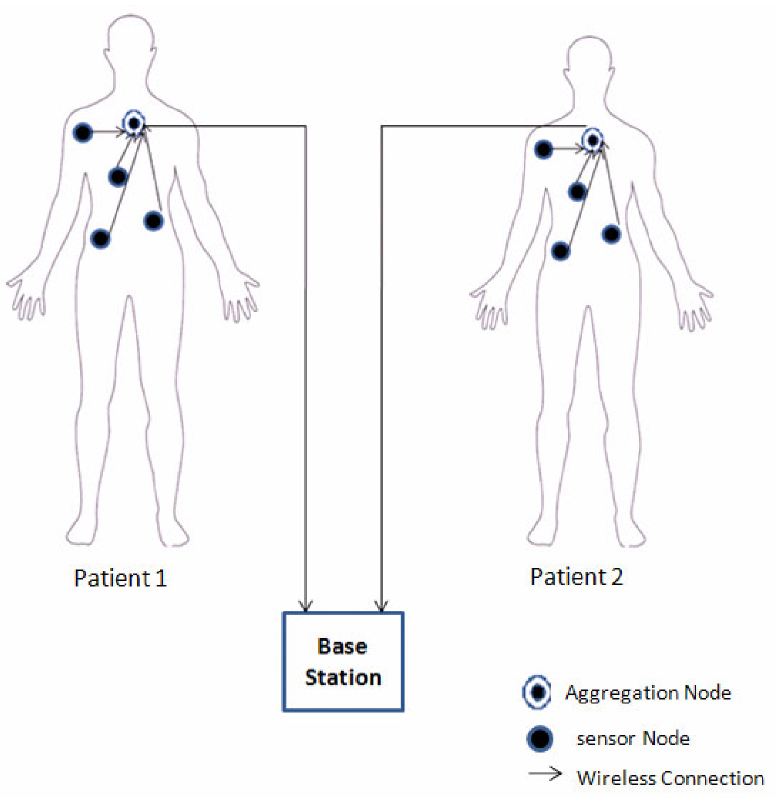


Fig. 2. Patient Monitoring Network (PMN)

2.3 Secure Architecture for HSN

In the secure architecture of a HSN, the PMN can be accessed in a variety of locations and each PMN has its own aggregation node identification. If the PMN is in the ambulance then the aggregation node uses the Internet or any WAN to transmit the patient data to the base station, which is (in most of the cases) located in the hospital. If the base station is in the hospital, then the aggregation node uses a LAN or a VPN to transmit the data. The base station collects information and stores in a database. In the database, each record might be stored with the aggregation node ID and patient ID for more confidentiality. Whenever a doctor or a care taker would like to go through the records of the patient, the doctor can access the data base directly using the local area network or VPN or directly access the database through the Internet.

The architecture shown is a three-tier architecture: Senor Nodes, Aggregation Nodes, and Base Station The communicating range of the sensor nodes is limited to body range, i.e., all the sensor nodes on the patient are always in the range of aggregation node by the patient body. The range of the aggregation nodes would be varying because it is the aggregation node that transmits the data to the base station.

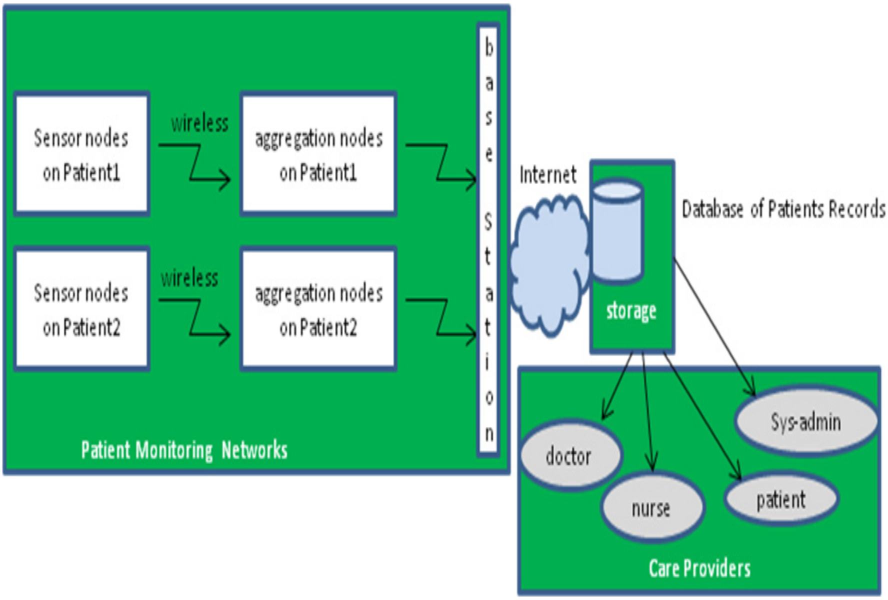


Fig. 3. Secure HSN architecture

Therefore, it is sufficient to have the aggregation node with varying range. The range of the aggregation node to the base station is local area for Region 1. For Region 2, the range for care providers would be wide area or via the Internet while Region 3 would be the databases. The secure architecture of HSN is shown in Figure 3.

The proposed architecture has many advantages when compared to the existing wireless sensor networks in healthcare application. As we have discussed in the state of art section, Codeblue, SNAP, and ALARMNET are the major existing wireless sensor architectures in healthcare sensor networks. Each of them does not include security in their architectural design; security is like an added-on feature. The proposed architecture has security built in it, and this comes with the features that the addition of aggregation node provides to us. The advantages of the proposed architecture over the existing architectures are:

- The architecture is designed for any type of application like for emergency purposes or assisted living and so on.
- The architecture has the security as a built-in feature rather than an additional feature.

3 A Quick Prototype

The main users of such system would be Care-Giver, patients and Administrators of the whole system. In the following approach the main concern is to make the health care system essentially mobile. Proceeding forward, system tries to focus on giving privileges to the patients as well. Since foremost care seeker would be elderly and physically disabled person, so the system would be expected to have some kind of user interface where bedridden patient can see day to day reports of their health. Also, the system should allow them to make an appointment to the specialized doctors. At the same time, the proposed schema of the system is likely to provide the doctors to see the minute to minute updates of the critically unwell patients. According to the proposed health care system, the doctors can prescribe the patient if a doctor wishes to do so. To maintain functioning of the whole system, the administrator should be able to see the technical reports and the minute to minute update of each cluster. Administrator would be able to enable or disable any sensor node and that too wirelessly. To reduce the health casualties, the administrator should be able to track the sensor network traffic and can limit it. Chief issues that the administrator would deal with are interoperability of the network along with the data like personal information of the patient. The personal data would flow through the network. For this, the system should provide some kind of the security aspect so that the data can be encrypted and can travel through the network. The proposed system schema would be used by ordinary patient. Therefore, the primary concern of the design would be to make the user interface as uncomplicated as possible but, at the same time proficient.

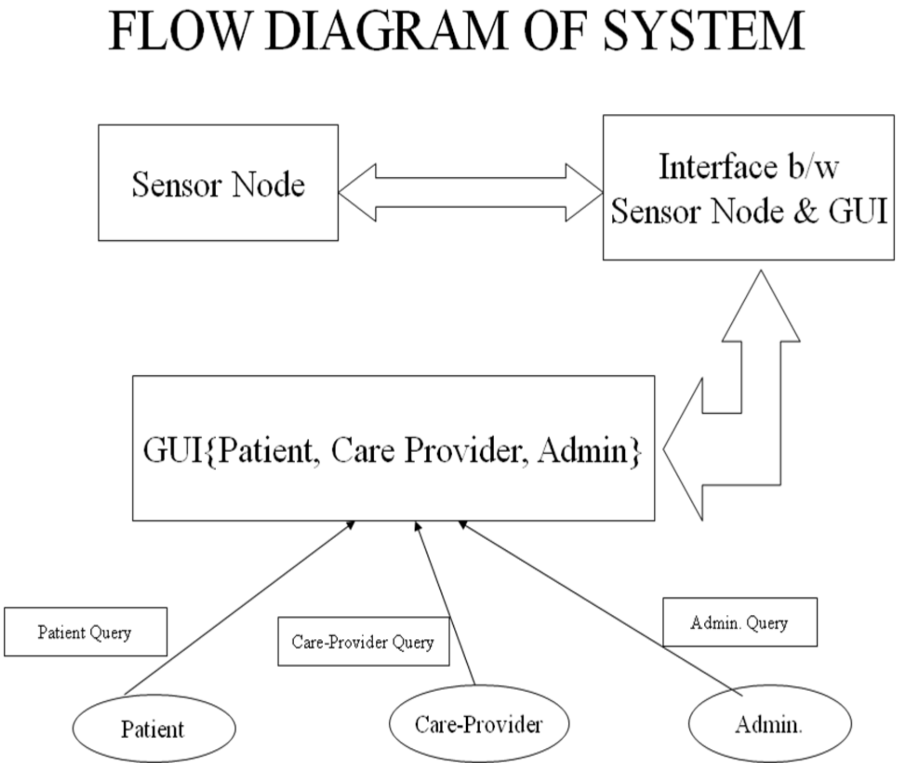


Fig. 4. Prototype Components

The anticipated scheme of the initial research-work allows clustering down the architecture. With the help of primary study and observation, we can think of three major sub-blocks- mainly the sensors (installed on the patient's body), GUI (user interface for the users of the system) and an interface (between the user interface and the sensors). The first task would be the collection of real time data from the sensor node and processing it at some base station. Then making the real time processed data readable to the user interface so that the users can easily read it. At transmitter end sensor node, the data has to follow some encryption method that allows secure data to stream across the network. At the user end, each patient and doctor would have their own login and password so that they can access their records only. To maintain the login, password details and to monitor the sensor nodes working, the administrator of the system would have access to the records.

- Data collection from Sensor nodes.
- Process entries at MOTE.
- Interfacing of MOTE and SQL server.

4 Our Proposed Workflow

In advancement of the initial studies, the workflow of the architecture would have a base station in between the user interface and the sensor nodes. The base station would be a programmable device which interacts with the sensors and the database server.

- Data collection from Sensor nodes.
- Process entries at a Base station; MOTE.
- Processing data at "TinyDB".
- Interfacing of "TinyDB" and SQL server.

The entire model provides an approach to make the health care system more secured and mobile keeping elderly and physically patient into consideration. The final outcome should provide a simple and operable user interface to the users and maintain the patients and doctor's data privacy and integrity. The proposed system should provide all the security requirements keeping all the data properties maintained. The model should provide the authentication at the time logging in and should be clear to read and understand by the patient and doctors. Model should be able to interact with the sensor nodes and should be able to communicate with the interfacing device, in this case MOTE. Along with this, the data streaming from the sensors should be able to follow the network protocols, as it would be flowing from one protocol to another.

Different users are presented with different interfaces. At the Log-In window as shown in Figure 6, user's credentials are checked, if user enters the correct login id provided manually/automatically to the user, a window will pop-up for the different type of function. Then the system gives different feature to the user; system would have different fields like foe patient, doctors, prescription and admin as shown in Figure 7. Here each tab would lead to a database where a record for each user is maintained. Figures 8 and 9 give the different user interfaces for patients verse doctors.

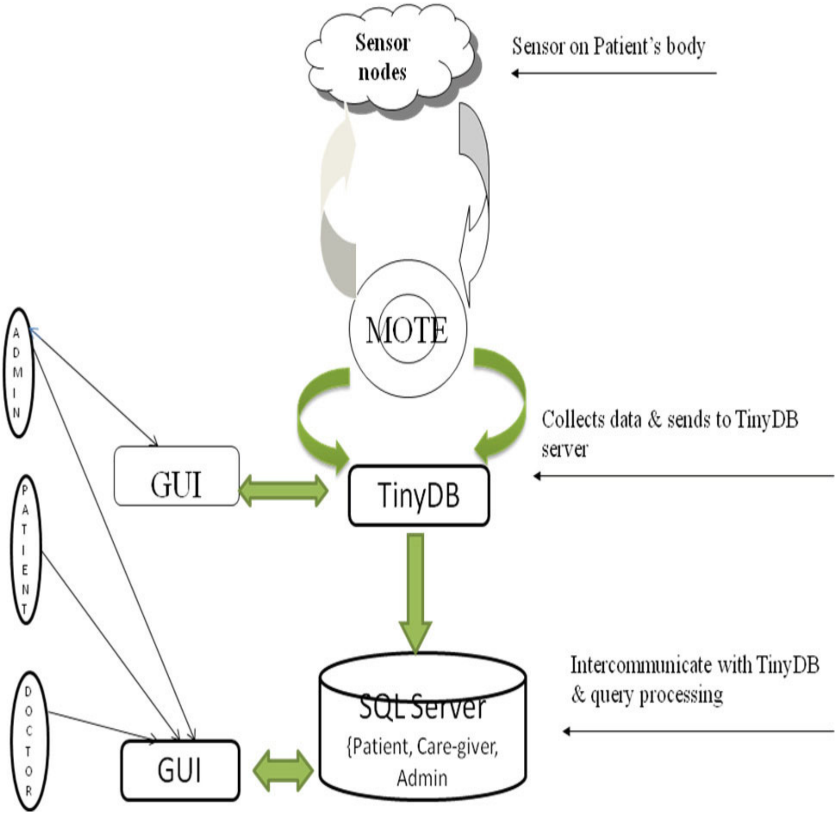

Sensor nodes
Sensor on Patient's body
MOTE
Collects data & sends to TinyDB server
GUI
TinyDB
ADMIN
PATIENT
DOCTOR
SQL Server
{Patient, Care-giver, Admin
Intercommunicate with TinyDB & query processing
GUI

Fig. 5. Workflow Model

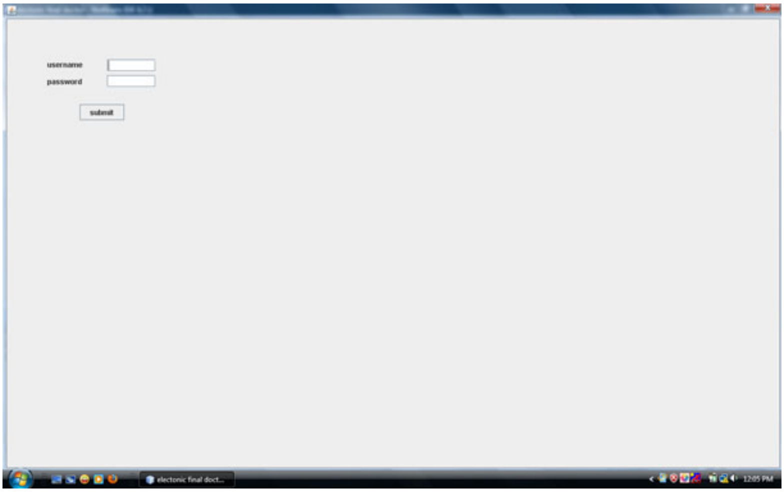


Fig. 6. Login In

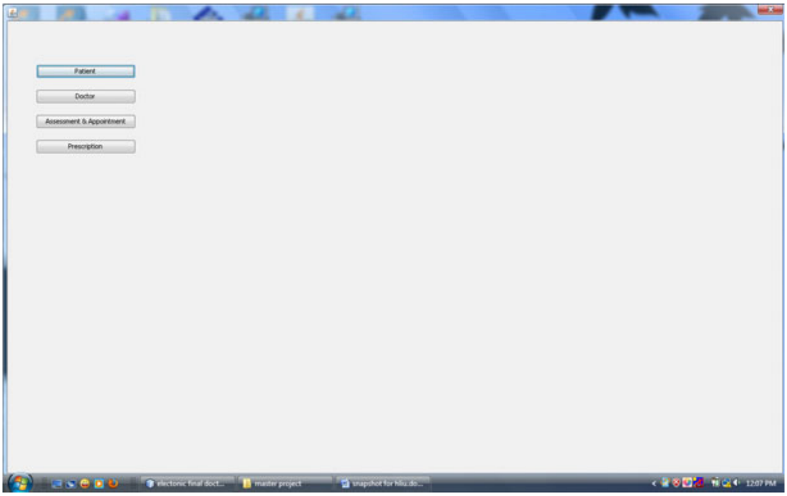


Fig. 7. Various User Interfaces

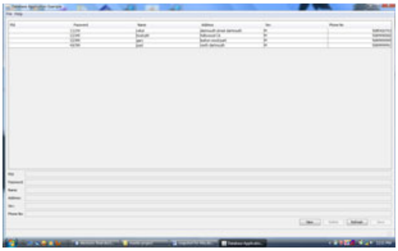


Fig. 8. Patient Interface

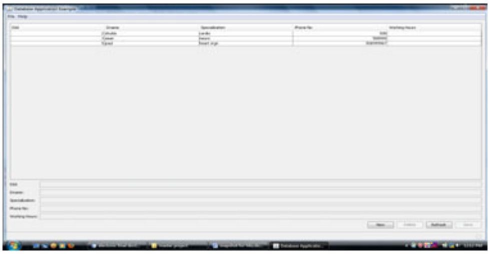


Fig. 9. Doctor Interface

5 Conclusion

This work demonstrates the feasibility of Divi's Secure Architecture for Healthcare Systems. Our further work includes validates the system's functionality and studies the efficiency issues.

References

[1] Lin, C.-C., Chiu, M.-J., Hsiao, C.-C., Lee, R.-G., Tsai, Y.- S.: Wireless Health Care Service System for Elderly With Dementia

[2] Gouaux, F., Simon-Chautemps, L., Adami, S., Arzi, M., Assanelli, D., Fayn, J., Forlini, M.C., Malossi, C., Martinez, A., Placide, J., Ziliani, G.L., Rubel, P.: Smart Devices for the Early Detection and Interpretation of Cardiological Syndromes. In: Proc. of the 4th Annual IEEE Conf. on Information Technology Applications in Biomedicine, UK

[3] Malan, D., Fulford-Jones, T., Welsh, M., Moulton, S.: Codeblue: An ad hoc sensor network infrastructure for emergency medical care. In: International Workshop on Wearable and Implantable Body Sensor Networks (2004)

[4] Wood, A., Virone, G., Doan, T., Cao, Q., Selavo, L., Wu, Y., Fang, L., He, Z., Lin, S., Stankovic, J.: ALARM-NET: Wireless sensor networks for assisted-living and health monitoring, Technical Report CS-2006–01, University of Virginia (2006)

[5] Malasri, K., Wang, L.: Addressing security in medical sensor networks. In: Proceedings of the 1st ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments, San Juan, Puerto Rico, June 11-13 (2007)

[6] Wang, Y., Attebury, G., Ramamurthy, B.: A survey of security issues in wireless sensor networks. IEEE Communications Surveys & Tutorials 8(2), 2–23 (2006)

[7] Divi, K., Kanjee, M.R., Liu, H.: Secure Architecture for Healthcare Wireless Sensor Networks. In: Proceedings of the IEEE & ITSS Sixth International Conference on Information Assurance and Security (IEEE & ITSS IAS 2010), Atlanta, USA, August 23-25 (2010) (to appear in)

[8] Malan, D.J., Welsh, M., Smith, M.D.: Implementing public-key infrastructure for sensor networks. ACM Transactions on Sensor Networks 4(4), 22–45 (2008)

[9] Karlof, C., Sastry, N., Wagner, D.: TinySec: A link layer security architecture for wireless sensor networks. In: Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems, Baltimore, Maryland, USA (2004)

[10] Liu, A., Ning, P.: TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks. In: Proceedings of the 7th International Conference on Information Processing in Sensor Networks, April 22-24, pp. 245–256 (2008)

[11] Kurian, J., Sarac, K.: A security framework for service overlay networks: access control. In: BroadNets 2008, Internet Track 3: Overlays and Traffic Estimation, London, UK, September 8-11 (2008)

[12] Wang, Y., Ramamurthy, B., Xue, Y., Zou, X.: A key management framework for wireless sensor networks utilizing a unique session key. In: BroadNets 2008,Wireless Track 6: MAC and Key Management, London, UK, September 8-11 (2008)

[13] Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An architecture for differentiated services. RFC2475 (December 1998)

[14] Rajan, R., Verma, D., kamat, S., Felstaine, E., Herzog, S.: A policy framework for integrated and differentiated services in the Internet. IEEE Network, 36–41 (September 1999)

[15] Liu, H., Dempsey, H.H.: Multi-facet Internet resource management system. In: Proceedings of IFIP/IEEE International Symposium on Integrated Network Management (IM), Boston, MA, USA, May 24-28 (1999)

[16] Yu, Q., (Liu, H., advisor): Denial-of-Service Countermeasure with Immunization and Regulation: Ph.D. Dissertation Dartmouth, MA: University of Massachusetts Dartmouth (2005)

[17] Ameen, M., Jingwei, L., Kyungsup, K.: Security and Privacy Issues in Wireless Sensor Networks for Healthcare Applications. Journal of Medical Systems (March 2010)

[18] Dishman, E.: Inventing Wellness Systems for Aging in Place. IEEE Computer (May 2004)

[19] Tan, C.C., Wang, H., Zhong, S., Li, Q.: Body sensor network security: an identity-based cryptography approach. In: Proceedings of the 1st ACM Conference on Wireless Network Security, Alexandria, VA, USA, March 31-April 02, pp. 148–153 (2008)

[20] Karl, H., Willig, A.: Protocols and architecture for wireless sensor networks. Wiley, Boston (2007)

An Evaluation of Mobile End Devices in Multimedia Streaming Scenarios

Michael Ransburg, Mario Jonke, and Hermann Hellwagner

Multimedia Communication (MMC) Research Group, Institute of Information Technology (ITEC), Klagenfurt University, Klagenfurt, Austria
{mransbur,hellwagn}@itec.uni-klu.ac.at,mjonke@edu.uni-klu.ac.at

Abstract. This paper compares handhelds based on the iPhone and Android operating systems in multimedia streaming scenarios. We simulate typical Internet network impairments, i.e. packet delay and packet loss, and evaluate their effects on the end devices. Additional evaluations include bandwidth overhead inflicted by the different streaming approaches and traffic shape and fairness when both handhelds consume media simultaneously. Based on the quantitative evaluation, both approaches show weaknesses and strengths. A final qualitative discussion points out additional advantages for the streaming approach implemented in the iPhone operating system.

Keywords: multimedia, streaming, mobile, RTP, HTTP, comparison.

1 Introduction

In this work we compare the iPhone 3.0 to the Android 1.6 operating system (OS) in multimedia streaming scenarios. For the iPhone OS a second generation Apple iPod Touch was chosen as a platform and for the Android OS the HTC Magic served as a platform. Both operating systems support the codecs H.264/AVC (AVC) [1] for video and MPEG-4 AAC (AAC) [2] for audio and are able to receive streamed multimedia content but they use different approaches to enable this. While the Android OS relies on the well known RTP protocol for AVC [3] and AAC [4], the iPhone OS facilitates a new approach called HTTP Live streaming. Other than RTP, HTTP Live streaming is a pull-based protocol which relies on breaking the overall stream into a sequence of small HTTP-based file downloads, which are referenced by an extended M3U playlist. Further information on HTTP Live streaming can be found in [5]. Our goal in this work is to evaluate which of the two streaming mechanisms is more suitable for streaming multimedia content to mobile end devices.

The remainder of this paper is organized as follows. In Section 2 we present our evaluation environment, i.e. the test data, test methodology and an overview of our testbed. Subsequently, we first evaluate the influences of packet delay and packet loss on the startup delay and playback in Sections 3 and 4, respectively. The bandwidth overhead which is caused by the two alternative streaming approaches is evaluated in Section 5. Next, we look at traffic shape and fairness in

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 400–412, 2010.

Section 6, by serving both handhelds simultaneously over the same channel and analyzing the traffic. Finally, Section 7 concludes our evaluation and provides an outlook to future work.

2 Evaluation Environment

2.1 Test Data and Methodology

As source content 100 seconds from a teaser from the movie Ice Age 3 were chosen and encoded in AVC for video and AAC for audio, which are the standard codecs supported by both operating systems. Since both operating systems only support the AVC baseline profile, B-frames, the CABAC filter and the Trellis algorithm were disabled during encoding. Audio was encoded with 96 kbps and two channels. For video we created variations differing in the temporal, spatial and quality domain. Table 1 shows the different encoding characteristics for the test sequences, based on common usage and on the end device capabilities, i.e. the highest resolution which is supported by both handhelds is 480x320 pixels and the highest bit rate is 500 kbps.

Table 1. Encoding characteristics for test sequences

Resolution [px]	480x320	320x240	240x160	160x120	
Frame rate [fps]	30	25	24	20	12.5
Bit rate [kbps]	500	400	200	100	

In the case of the iPhone OS two extra steps need to be performed, in order to prepare the content for HTTP Live streaming. First, the content is multiplexed into a MPEG-2 transport stream and second, it needs to be fragmented into segments. We used a minimum length of 10 s for each segment, as suggested in [5]. However, not all segments are of equal length, since the segmentation can only be performed at Instantaneous Decoder Refresh (IDR) frames within the video sequence. Therefore, the segmenter has to wait for the next IDR-frame to appear, before a new segment can be started. In our case, we used an IDR-frame interval of 250. This is the default for *MEncoder*[1], which we used for encoding the contents. In the worst case this means that an IDR-frame may occur shortly before the specified segment duration in which case the specific segments duration may almost double. Additionally, this encoder may decide to introduce additional IDR-frames in case of scene cuts.

We did not modify the handhelds in any way and used the native media players in order to get representative results. This means that our startup delay measurements were done manually with an external timer which leads to higher errors in measurement. To compensate for this factor and for unexpected behavior of the handhelds 10 repetitions were performed for each measurement.

[1] http://www.mplayerhq.hu, SVN-r29411

Considering all the entries in Table 1, a total of 80 different variations of the original sequence were created by incorporating every possible permutation. Because of space limitations we will focus on a relevant subset of our results in the following.

2.2 Testbed

Figure 1 shows our testbed, which consists of a combination of several different Linux computers with kernel version 2.6.27, running the Ubuntu distribution. The computers `pl05` and `pl06` act as content providers and are the locations where the HTTP Web server (in our case Apache HTTP Web server[2]) and the RTP streaming server (in our case Live 555 Media Server[3]) are located. The machine `mm08` is used to introduce additional network delay by increasing the round trip time (RTT) of packets, whereas `mm04` is used to apply traffic shaping and packet loss. For this, the Linux traffic control mechanism and its extension *Netem*[4] was used. *Tcpdump*[5] and *wireshark*[6] were used to record the traffic at the client side as well as the server side.

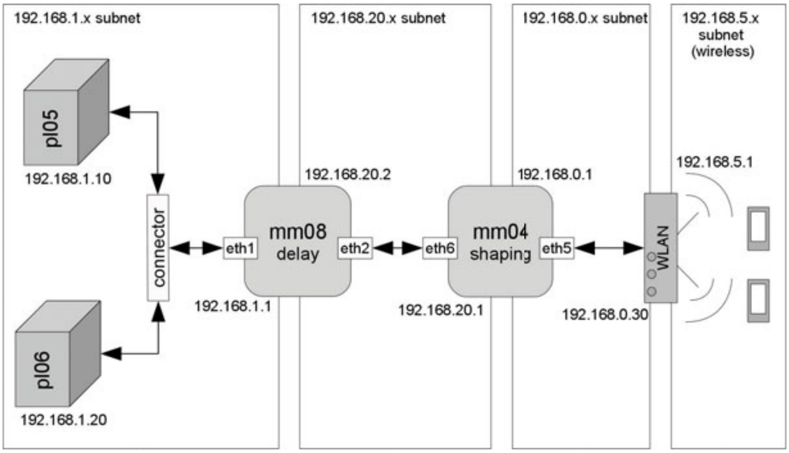


Fig. 1. Testbed

With the help of the test environment several measurements were performed, using the presented test set. These measurements included the startup delay, the impacts on playback, as well as the observation of traffic fairness, shape and data overhead. In the following sections the results are presented.

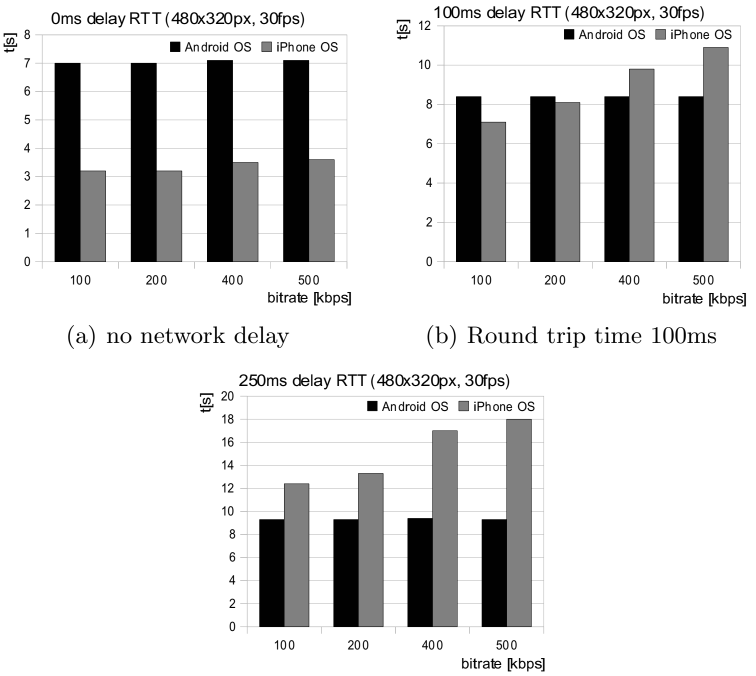
[2] http://www.apache.org, v2.28

[3] http://www.live555.com, v0.3

[4] http://www.linuxfoundation.org/en/Net:Netem, Kernel v2.6.27

[5] http://www.tcpdump.org, v3.9.8 (libcap 0.9.8)

[6] http://www.wireshark.org, v1.2.7



(a) no network delay

(b) Round trip time 100ms

(c) Round trip time 250ms

Fig. 2. Comparison of different round trip times

3 Evaluation of Startup Delay

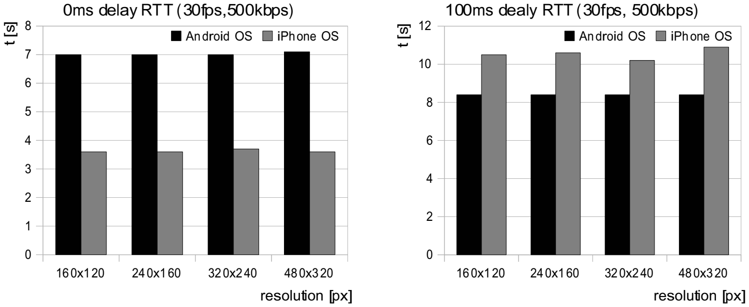
The term startup delay describes the time between the request of a sequence and the actual start of the playback.

3.1 Effect of Packet Delay on Startup Delay

In the Internet, the RTT depends on the infrastructure, its utilization and the geographical localization of hosts in the network. The study about round trip time conducted by Acharya et al. [6] shows that the wide majority of investigated hosts have shown an inherent RTT lower than 250 milliseconds. Therefore, RTT values of 0, 100 and 250 milliseconds where chosen for the evaluation. The delay distribution of forward and backward channel was set symmetrically.

The bar charts in Figure 2 show the startup delay for the test sequences with a resolution of 480x320 pixels and a frame rate of 30 fps. As depicted in Figure 2(a), the startup delay is about seven seconds for Android OS and three seconds for the iPhone OS. Without introducing packet delay, the startup times are relatively constant. However, once packet delay is introduced, the characteristics of the operating systems start to differ as can be seen in figures 2(b) and 2(c). The startup delay on the Android OS increases linearly when increasing the RTT, on the iPhone OS however, we can observe an exponential increase of startup delay depending on both the RTT and the bit rate.

The explanation for this difference is twofold. First, with RTP only a few frames of a sequence need to be received (i.e. until the playback buffer is full) before playback can be started, which is contrary to HTTP Live streaming where the playback does not start until a whole segment is received. Second, since the HTTP Live streaming approach is based on TCP, the amount of data it can send out at once is constrained by the contention window, limiting the number of packets that can be sent before receiving an ACK. Thus, the server is only allowed to send out a small amount of a packets at once. Due to this condition the overall delay multiplies with the amount of data and the increasing delay in the network. This restriction does not apply to the Android OS, since UDP, which RTP relies on, does not involve acknowledgement packets.



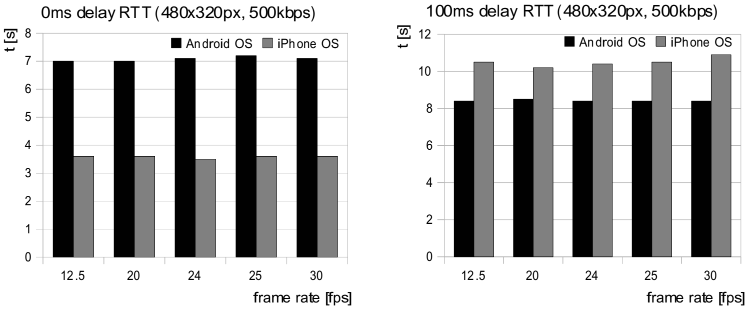
(a) no network delay (b) Round trip time 100ms

Fig. 3. Impacts on startup delay considering different video resolutions

So far, the evaluation focused on variations in bit rate, leaving the parameters for frame rate and resolution selected in the test set unchanged. Figures 3 and 4 show the impact of various resolutions and frame rates on the startup delay. As can be seen, in both cases and for both operating systems the startup delay increases linearly when increasing the resolution or frame rate (while keeping the bit rate constant at 500 kbps). This can be explained by the fact that for the transmission of a segment in HTTP Live streaming only the bit rate (and thus segment size) is a relevant factor. In case of RTP, we assume that the startup delay mostly depends on the size of the playback buffer. We cannot verify this, since the playback buffer is not user-configurable on the Android OS.

3.2 Effect of Packet Loss on Startup Delay

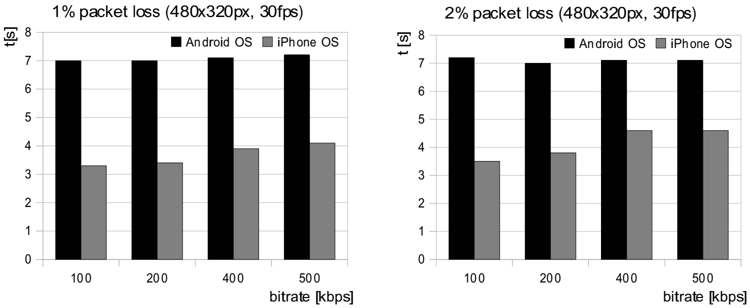
Packet loss is typically caused by congested networks, since routers can only buffer a finite amount of packets at a time, thus certain selected packets have to be dropped. Investigations performed by Wang et al. [7] have shown that about 95 % of tested Internet links have a packet loss rate lower than 2 %. Therefore, a maximum packet loss rate of 2 % was chosen to be considered for the evaluation.



(a) no network delay

(b) Round trip time 100ms

Fig. 4. Impacts on startup delay considering different frame rates



(a) 1% packet loss

(b) 2% packet loss

Fig. 5. Impact of packet loss on startup delay

The Figures 5(a) and 5(b) show that packet loss does not influence startup delay for the Android OS. For the iPhone OS however, packet loss results in an increased startup delay, since TCP is a reliable protocol which retransmits lost data packets. Although packet loss increases the startup delay, the impacts are not as high as they are in the case of introduced network delay.

4 Impacts on Playback

After evaluating the startup delay in the previous section, we now focus on the playback, i.e., how will factors such as packet delay and packet loss impact the continuity of the playback. Figure 6 shows the traffic distribution for both operating systems under ideal network conditions. The solid line shows the traffic occurred during streaming to the Android OS handheld, whereas the dashed line shows the traffic for the iPhone OS handheld. For HTTP Live streaming, the markers facing downwards from the x-axis display the times when a segment is requested. The markers facing upwards from the x-axis on the other hand show the deadline for each segment until its download has to be completed in order

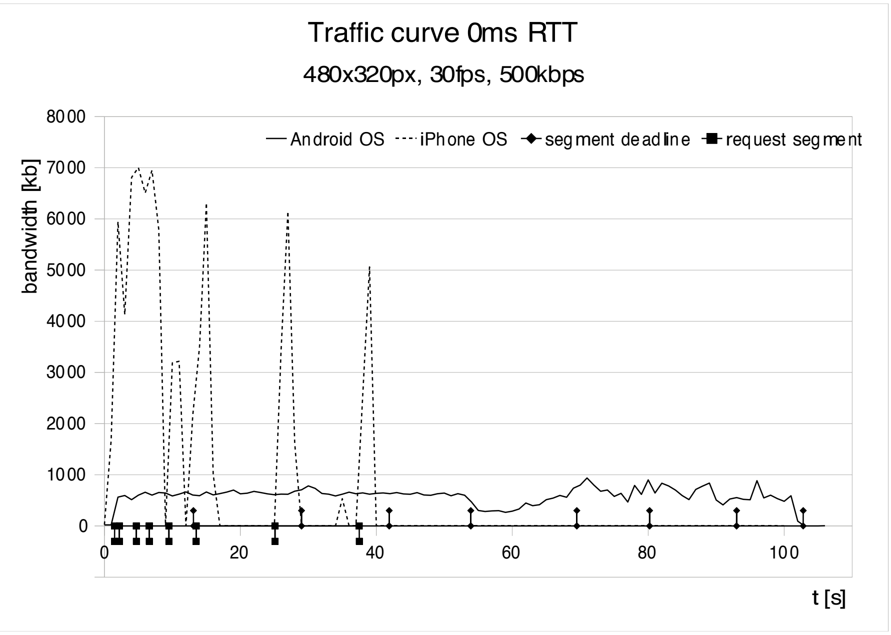


Fig. 6. Traffic under ideal network conditions

to guarantee continuous playback. The moment when the first segment is completely received, i.e. the start of playback, is also the start of the duration until the first segment deadline expires. The times for the deadlines correspond to the length of the segments. These segment deadlines are not applicable for RTP streaming, since the content is not split into segments. Additionally it has to be noted that the measurements of both handhelds were not performed simultaneously, rather the results where merged into one diagram in post-processing. As can be seen from the figure, HTTP Live streaming uses the maximum amount of bandwidth to gather the first few segments, in this case the initial four. After downloading enough segments the request of further segments is paused until the playback advances in time. This behavior can for example be observed at the 20th second. On the other hand, looking at the bandwidth of the Android OS it can be seen that the traffic is continuous.

4.1 Effects of Packet Delay on Playback

As shown by the results in Section 3.1, the network latency has a much higher impact on HTTP Live streaming than it has on RTP streaming. In order to evaluate how network latency influences the playback the same network delays were chosen. Figure 7 shows the results of these measurements. The plot includes three traffic lines. Two of them describe HTTP Live streaming at latencies of 100 ms and 250 ms respectively and one shows RTP streaming at a delay of 250 ms. Furthermore, the figure also shows the segment request segment deadline

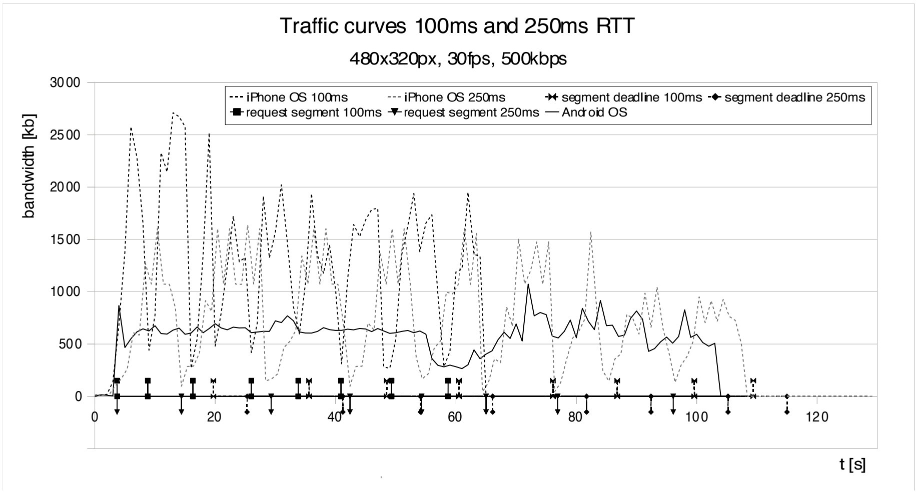


Fig. 7. Traffic considering different packet delays

markers, which hold the same semantics as the ones in Figure 6. As can be seen from the markers, all the segment deadlines can be kept in the case of 100 ms latency. For a latency of 250 ms, however, the download of the second segment (third marker) does not finish until the first segment deadline, thus resulting in non-continuous playback. Considering the other segment deadlines, only the 4th and the 5th can be kept. The cause of this problem is the same as in the case of startup delay. The high network latency forces TCP to wait much longer for acknowledgment packets, thus the utilized bandwidth becomes much lower and it takes much longer to fully retrieve a segment. For comparison to the Android OS, the solid line shows the RTP stream at 250 ms network delay. In contrast to the optimal case, duration of the playback is only slightly longer, caused by the initial RTSP negotiation, which builds on TCP. However, the traffic characteristic do not show any significant changes compared to Figure 6. The conclusion that can be drawn from this observation is that high network delay causes HTTP Live streaming to result in non-continuous playback, whereas RTP streaming is nearly unaffected.

4.2 Effects of Packet Loss on Playback

Packet loss is a common problem concerning RTP-based multimedia streaming, since lost packets directly affect the quality of the played content, because the decoder is missing data to restore the uncompressed state. Thus, different concealing techniques exist to deal with the problem of lost packets [8]. On the other hand, streaming content with the help of HTTP does not degrade the quality of the content in case of packet loss, because of the reliability of TCP. However, this does not come for free, since it involves retransmissions. In this section we therefore investigate the impacts of several packet loss rates on the playback.

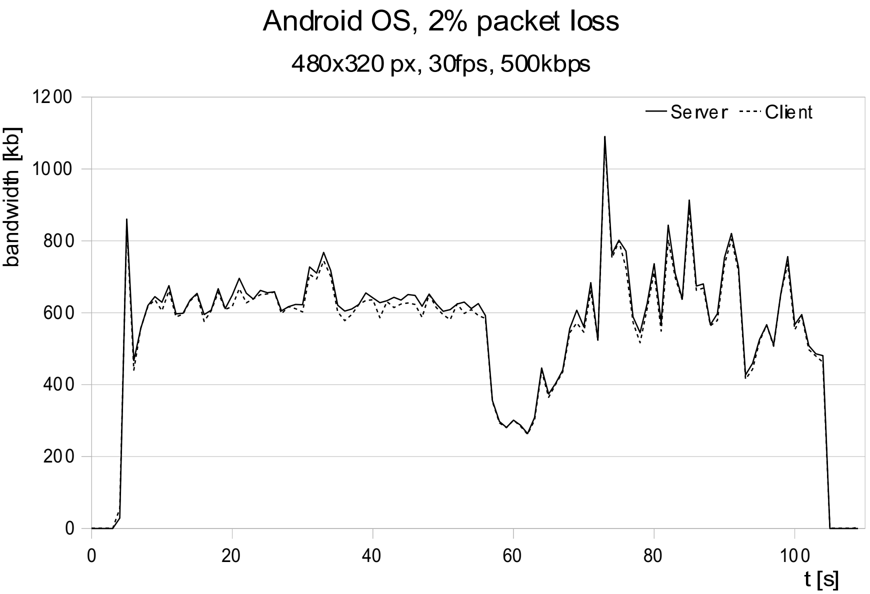


Fig. 8. Playback impacts during 2% packet loss for Android OS

The lines in Figure 8 show the traffic characteristics for the Android OS under the influence of 2 % of packet loss. Measurements were performed at the server and the client simultaneously. Thus, the solid line shows the traffic at the server, whereas the dashed line describes the client side, which is measured at the outgoing interface of machine `mm04` as described in Section 2.2. For the Android OS packet loss results in the creation of artifacts during the playback. In addition to packet loss rates of 1 % and 2 %, we also tested 5 % packet loss, in which case the Android OS was not able to finish playback.

The measurements for the iPhone OS can be seen in Figure 9. Only one line is shown, since the server and the client side did not show any noticeable difference. When comparing the traffic characteristics to those in the optimal case in Figure 6, one can see that in case of packet loss the utilized bandwidth is much lower and consequently the segment request times are more distributed. The explanation for this is the behavior of TCP in the case of packet loss. As TCP detects a lost packet, it halves the maximum allowed bandwidth according to the Additive Increase Multiple Decrease (AIMD) algorithm [9], by reducing the size of the contention window. After receiving further packets without loss, the contention window is increased again, but only in a linear way. Nonetheless, the tested packet loss rates did not influence the quality of the playback in terms of continuity. Furthermore, the content is displayed at full quality without the creation of artifacts, due to the reliability of TCP.

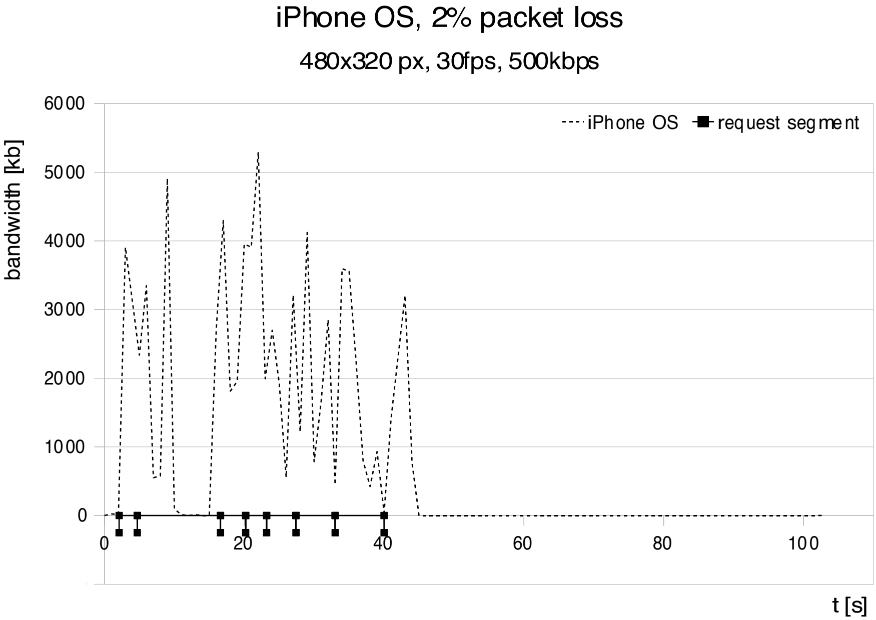


Fig. 9. Playback impacts during 2 % packet loss for iPhone OS

5 Bandwidth Overhead

When transferring content across a network additional information is needed. The amount of additional bytes spent depends on the protocols involved during the transmission. Therefore, the purpose of this section is to compare the amount of overhead caused by RTP streaming and HTTP Live streaming, respectively. As test sequence for the measurements the highest available quality was chosen, i.e. 480x320 pixels at 30 fps and 500 kbps. Actually, before considering the overhead introduced through streaming, the additional amount of data caused by container format multiplexing needs to be considered. Since RTP enables to stream AVC and AAC in their raw bit stream format, there is no additional overhead for the Android OS. On the other hand, the raw bit streams need to be multiplexed into the MPEG-2 transport stream format to enable HTTP Live Streaming for the iPhone OS. The selected test sequence requires 596 kbps including audio, increasing to 706 kbps after multiplexing and segmentation, i.e. the final bit stream includes an overhead of about 18.5 %. The reason lies in the stuffing of transport stream packets [10]. Overhead introduced during streaming, due to network protocols, was measured under optimal network conditions. In the case of HTTP Live streaming about 9000 TCP packets were observed. The additional overhead introduced by the TCP header is 32 byte, resulting in a total amount of about 23 kbps including the additional data required for requesting the playlist file and the single HTTP requests for each segment.

In the case of RTP streaming 10300 packets were encountered. With a packet header of 8 byte for UDP and 16 byte for RTP, this sums up to about 19.3 kbps overhead. Note that the 16 byte for RTP is an average number, since the RTP payload format headers for AAC and AVC use a different number of bytes. Streaming with RTP involves additional traffic caused by RTCP. Actually, about 230 RTCP packets were measured, resulting in about 13 kB of data, considering the 8 byte of UDP header and 48 bytes of data of an RTCP packet. Furthermore, also the traffic caused by RTSP needs to be considered, which was about 4 kB. That is, both RTCP and RTSP cause only insignificant overhead. At this point it has to be mentioned that the streaming server only used fragmentation units A and B (FU-A and FU-B)for packetizing AVC packets [3]. An single time packet aggregation packet (STAP) implementation would have resulted in a minor decrease of the overhead.

To conclude, without considering multiplexing, the overhead of both approaches is similar. Thus, the main amount of additional overhead on the iPhone OS is the result of the necessary multiplexing.

6 Traffic Shape and Fairness

In this section we evaluate the behavior of both devices in case of scarcity and rivaling traffic by limiting the available bandwidth to 1 Mbps and serving both end devices simultaneously. The traffic characteristics under these circumstances are depicted in Figure 10. The solid line describes RTP streaming, whereas the dashed line shows the characteristics of HTTP Live streaming. Additionally the total amount of traffic is depicted by the bold solid line. Furthermore, also the request segment and segment deadline markers for HTTP Live streaming are displayed. The streaming session for RTP is started first. After about 23 seconds HTTP Live streaming is also started. Comparing both lines to the optimal case in Figure 6, it can be seen that the traffic characteristics for HTTP Live streaming are completely different, while RTP almost equals the optimal case. After the end of the RTP session, HTTP Live streaming utilizes the full bandwidth to gather the next segments which already fell behind the deadline. From the observation of these traffic lines it can be seen, that the RTP traffic treats the HTTP Live streaming in an unfair manner. Although the playback by the Android OS has shown artifacts, caused by delayed packets due to network congestion, the traffic shape did not change in a noticeable extent. In contrast, the available bit rate for the iPhone OS adjusts to the bit rate required by the Android OS. This is due to the congestion control mechanisms implemented in TCP but not in UDP. A remedy to this problem could be the usage of the Datagram Congestion Control Protocol (DCCP) [11] instead of UDP, which provides congestion control mechanisms for unreliable traffic. In fact, measurements in wireless network environments performed by de Sales et al. [12] have shown that DCCP and TCP behave fair to each other, as long as no UDP flow is involved.

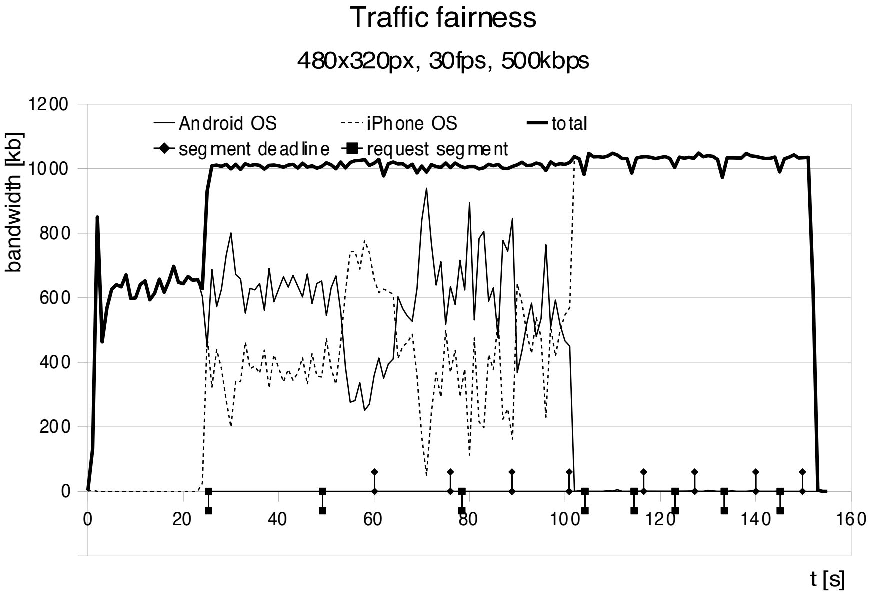


Fig. 10. Traffic fairness under bandwidth limitation of 1Mb/s

7 Conclusions and Future Work

Our evaluation shows the differences of the two streaming approaches used in iPhone OS and Android OS. We evaluated the startup delay in case of increasing packet delay and packet loss. This showed that the startup delay increased linearly for Android OS and exponentially for iPhone OS in case of increasing packet delay. By injecting packet loss we noticed a minor increase in startup delay for the iPhone OS, while the Android OS was not affected at all. Next, we evaluated the playback characteristics under the same network impairments. This showed that for high packet delay of 250ms the playback on the iPhone OS is non-continuous, while the Android OS was nearly unaffected and showed no playback disruptions. Packet loss, on the other hand, caused disruptions in the video on the Android OS, while having no impact on the video quality on the iPhone OS. We then analyzed bandwidth overhead, where the MPEG-2 transport stream format required by HTTP Live streaming mechanism of the iPhone OS caused a substantial overhead. Finally, we evaluated traffic shape and fairness, which showed the typical greedy behavior of RTP vs. HTTP.

Both approaches have their strengths and weaknesses. However, the HTTP Live streaming mechanism comes with the advantage that there is no dedicated streaming server needed. Additionally, the problems related to NAT traversal [13] are avoided. Finally, the HTTP-based approach comes at the advantage that existing Content Delivery Networks (CDNs) can be used to distribute the content in a very scalable way. It therefore promises to be a more lightweight and scalable solution, which will certainly help its adoption by industry.

In our future work we plan to evaluate the special characteristics of the wireless network more closely. Additionally, we would like to extend our evaluations based on the just mentioned advantages of the HTTP Live streaming approach, i.e. NAT traversal and scalability.

Acknowledgments. This work is supported by the Österreichische Forschungsförderungsgesellschaft mbH (FFG) in the context of the Celtic SCALNET (CP5-022) project.

References

1. Wiegand, T., Sullivan, G., Bjntegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transactions on Circuits and Systems for Video Technology 13(7) (July 2003)
2. Purnhagen, H.: An Overview of MPEG-4 Audio Version 2. In: Proc. 17th International Conference: High-Quality Audio Coding (August 1999)
3. Wenger, S., Hannuksela, M.M., Stockhammer, T., Westerlund, M., Singer, D.: RTP Payload Format for H.264 Video. RFC 3984 (February 2005)
4. van der Meer, J., Mackie, D., Swaminathan, V., Singer, D., Gentric, P.: RTP Payload Format for Transport of MPEG-4 Elementary Streams. Technical report, Internet Engineering Task Force, Proposed Standard, RFC 3640 (November 2003)
5. Pantos, R.: HTTP Live Streaming. In: Internet-Draft (work in progress), draft-pantos-http-live-streaming-03 (expries October 4, 2010)
6. Acharya, A., Saltz, J.: A study of internet round-trip delay. Technical report, University of Maryland (December 1996)CS-TR-3736
7. Angela Wang, Y., Huang, C., Li, J., Ross, K.W.: Queen: Estimating packet loss rate between arbitrary internet hosts. In: Moon, S.B., Teixeira, R., Uhlig, S. (eds.) PAM 2009. LNCS, vol. 5448, pp. 57–66. Springer, Heidelberg (2009)
8. Wang, Y., Zhu, Q.-F.: Error Control and Concealment for Video Communication: A Review. Proceedings of the IEEE 86, 974–997 (1998)
9. Allman, M., Paxson, V., Stevens, W.: TCP Congestion Control. RFC 2581 (April 1999)
10. ISO/IEC 13818-1:2000 Information Technology-Generic Coding of Moving Pictures and Associated Audio, Part 1: Systems. Recommendation ITU H.222.0 (December 2000)
11. Kohler, E., Handley, M., Floyd, S.: Datagram Congestion Control Protocol (DCCP). RFC 4340 (March 2006)
12. de Sales, L.M., Almeida, H.O., Perkusich, A.: On the performance of TCP, UDP and DCCP over 802.11 g networks. In: SAC 2008, pp. 2074–2078. ACM, New York (2008)
13. Paulsamy, V., Chatterjee, S.: Network Convergence and the NAT/Firewall Problems. In: Proc. 36th Hawaii International Conference on System Sciences (January 2003)

An Extended IGMP Protocol for Mobile IPTV Services in Mobile WiMAX

Eunjo Lee, Sungkwon Park*, Joohan Lee, and Phooi Yee Lau

Department of Electronics and Computer Engineering,
Hanyang University, Seoul, Republic of Korea
{leeej,sp2996,hitch100,laupy}@hanyang.ac.kr

Abstract. Mobile WiMAX access network is being developed to support various multimedia services such as mobile Internet Protocol Television (IPTV), mobile Video-on-Demand (VoD), and mobile Internet services. This mobile network is a shared radio medium which utilizes a point-to-multipoint method, where one base station (BS) can be connected to many mobile stations (MS). This environment enables mobile IPTV viewers join a specific multicast group over mobile WiMAX access network while others, at the same time, receive the same program channel even though they do not belong to the same multicast group. This, however, is different from the traditional Internet Group Management Protocol (IGMP) version used for IPTV services in network layer which does not allow immediate program channel sharing. This is because the Connection ID (CID) is required before the Multicast Broadcast Service (MBS) can transmit its service flows in mobile WiMAX. Therefore, in this case, viewers always need to perform two processes before they are able to view the program channels, i.e. performing the IGMP join/leave at network layer and obtaining the CID at Medium Access Control (MAC) layer. This paper propose a new extended IGMP protocol which can be used in mobile WiMAX radio access network especially for mobile IPTV services to reduce the channel change response time on the mobile network.

Keywords: Internet Group Management Protocol, Mobile WiMAX, program channel change.

1 Introduction

Lately, mobile access networks are being developed to support various multimedia services such as mobile Internet Protocol Television (IPTV), mobile Video on Demand (VoD), and mobile high-speed internet services. In particular, Mobile WiMAX defines an IP end-to-end network architecture, which is an integrated telecommunications network architecture that uses IP for the end-to-end transport of all user data and signaling data [1]. This paper extend the application scope to IP multicast services for multimedia program delivery using radio access resources. IP

* Corresponding author.

This work was supported by the Seoul R&BD Program (No. PA090720).

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 413–426, 2010.

multicast delivers a multimedia program to many hosts that belong to the same group and at the same time accompanied by unicast. Therefore, using IP multicast in mobile access networks is highly efficient for multimedia services because it uses a Point-to-multipoint (PMP) method in which one Base Station (BS) could connect to many Mobile Station/Subscriber Stations (MS/SS). In case of providing multimedia services through IP multicast method using radio access resources, some viewers located in the same multicast transmission zone can start receiving both the multicast program channel they requested and the multicast program channels they did not requested. In other words, if a viewer is joining a specific multicast group, other viewers also receive the multicast program channel at the same time, even though they do not belong to the same multicast group.

This is, however, different from the traditional Internet Group Management Protocol (IGMP) as it does not allow immediate multicast program channel sharing as explained above. Moreover, the Connection ID (CID) must be used for transmitting the Multicast Broadcast Service (MBS) service flows in WiMAX. In this case, the viewers always need to perform two processes before they could view the program channels, i.e. performing the IGMP join/leave processes at network layer and obtaining the CID at Medium Access Control (MAC) layer. The mobile IPTV viewers over WiMAX access networks must endure, at least, several milliseconds of the channel change response time when changing program channels, i.e, time for IGMP processing and time for MBS configurations. Channel change response time is considered to be one of the most important parts of IPTV service quality. Particularly, IGMP join and leave delay is the main source of channel changing delay. Therefore, each viewer of the same multicast transmission zone in WiMAX would want to be able to watch immediately the shared IPTV program channels without the channel change response time.

This paper propose a new extended IGMP which can be apply in mobile WiMAX access network especially for mobile IPTV services to dramatically reduce the channel change response time caused by independent processes using the traditional IGMP at network layer and cause by delay during the issuance of CID at MAC layer.

The rest of the paper is organized as follows. In Section 2, we describe channel change response time on IPTV systems and mobile WiMAX protocol structure as the background of this study. In Section 3, details the cross-layer design of the Extended IGMP for mobile IPTV services in WiMAX. Section 4, we propose the extended IGMP protocol architecture for mobile WiMAX and its performance analysis is shown in Section 5. Finally, we conclude this paper in Section 6.

2 Background

2.1 Traditional IGMP Versions

IGMP, a multicasting protocol in the internet protocols family, is used by IP hosts to report their host group memberships to any immediately neighboring multicast routers. IGMP messages are encapsulated in IP datagram, with an IP protocol number of 2. IGMP has versions IGMP v1, v2 and v3 [2-6].

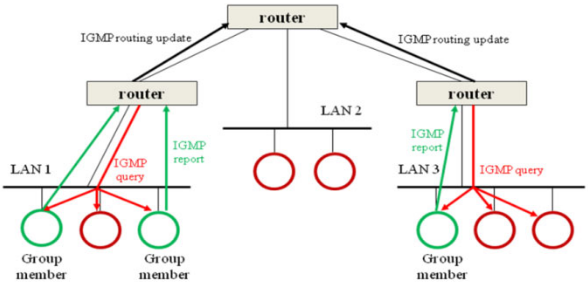


Fig. 1. IGMP basic mechanism

In IGMP version 1, as shown in Figure 1, hosts can join multicast groups. There were no leave messages. Routers were using a time-out based mechanism to discover the groups that are of no interest to its members. In IGMP version 2, leave messages were added to the protocol. It allows group membership termination to be quickly reported to the routing protocol, which is important for high-bandwidth multicast groups and/or subnets with highly volatile group membership. In IGMP version 3, the protocol has several major revisions. It allows hosts to specify the list of secured hosts from which incoming traffic is allowed. Traffic from other hosts is blocked from entering the network. It also allows hosts to block packets from sources that sent un-request traffic inside the network.

2.2 Channel Change Response Time in IPTV Services

The key quality of experience (QoE) element for IPTV is how quickly and correctly the subscribers can change TV channels. Acceptable channel change response time is generally considered to be around 1 second, end-to-end. A channel change response time of 100~200ms is considered, by viewers, to be instantaneous [7]. Sources of channel change response time response include network equipment and IPTV terminals. The IPTV terminals, an IPTV enabler at subscribers' side, add several

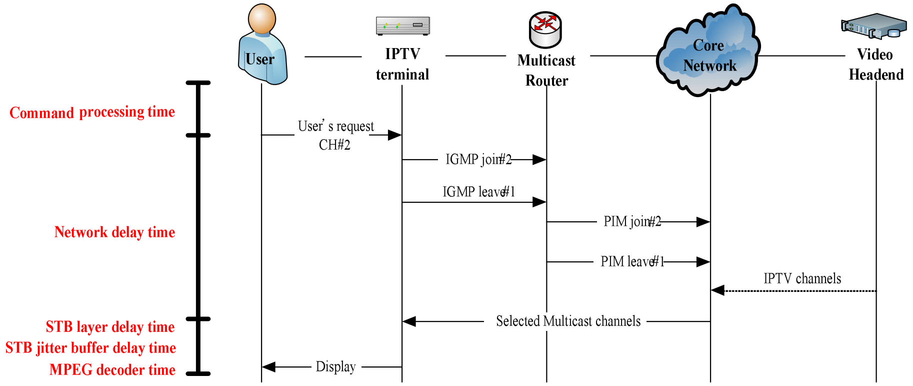


Fig. 2. IPTV channel change flows

hundred milliseconds of delay when changing channels due to command processing, buffer delay, MPEG decoder delay and video buffer delay. Fortunately, each IPTV terminal serves only one user and the main IPTV terminal functions are processed in hardware [8]. Therefore, IPTV terminal performance is relative stable and repeatable. Multicast protocols are used as the technique to enable channel change response time in network infrastructure. IGMP or MLD (Multicast Listener Discovery) leave/join delay is the main source of channel change response time. To keep overall channel change response time within one second, the target multicast leave/join delay of each network component needs to be about 10-200 ms.

3 Multicast Multimedia Delivery in WIMAX MAC Layer

3.1 Downlink MAP Message Monitoring to Receive MBS Data Bursts

The MBS service flows are managed through a DSx messaging procedure used to create, change, and delete a service flow for each MS [9-10]. The DSx message exchange between MS and BS carries important service flow information (SFID) such as quality of service (QoS), service flow identifier, and multicast CID (MCID). DSx messaging also provides an MS with the MBS_ZONE_ID for the subscribed service flows to indicate a service area through which an MCID and security association for a broadcast and multicast service flow are valid.

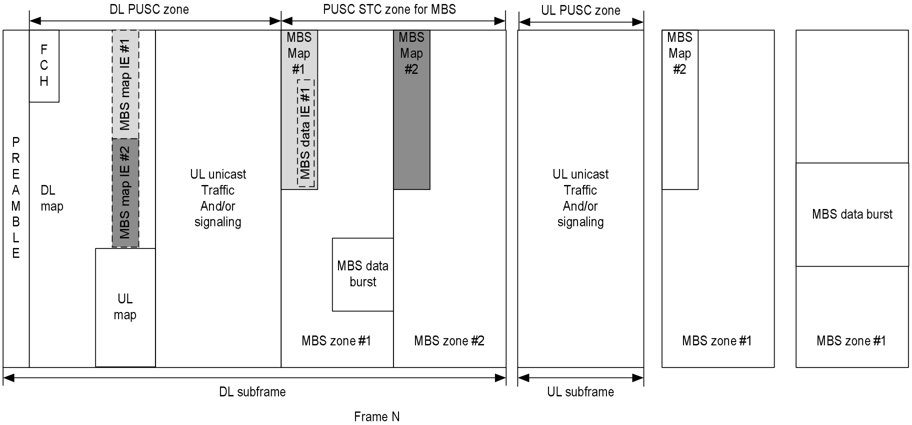


Fig. 3. MBS signaling in IEEE 802.16 MAC [9]

MBS typically involves multiple selectable content channels, the IEEE 802.16 standards [11] also define compound or group DSx, which can activate, delete, or change multiple connections and service flows using a single message exchange to reduce system overhead and latency. A BS supporting MBS includes the list of MBS zone identifier to which it belongs. One BS may belong to multiple MBS zones of the same or different sizes and coverage. From the MS perspective, the SFID assigned by the anchor authenticator is unique, and the MCID is also unique per MBS zone. This

MCID is common to all the MSs for that content in the MBS zone. As shown in Figure 3, the MS continues monitoring the broadcast channels (DL-MAPs) and looks up an information element called MBS-MAP-IE. Once the MS receives the MBS-MAP-IE, it verifies the associated MBS zone ID and, using the pointer within MBS-MAP-IE, locates the corresponding MBS permutation zone and the corresponding physical layer parameters. All MBS transmissions are sent in their designated subcarrier permutation zones. The MBS permutation zone starts with a management message called MBS-MAP, which includes one or more information elements called MBS-DATA-IEs, which list the MCIDs included in the upcoming MBS transmission and its also points to next occurrence of MBS-MAP as well as the location of MBS bursts. These pointers serve as daisy chain allocations allowing the MS to follow the MBS control and data transmissions without reading MAPs in every frame or interacting with the BS. MBS data bursts may contain different content channels, each mapping to different MCIDs. The standard allows parsing and selective discard/processing of content channels based on their corresponding MCIDs.

3.2 MBS Information Element Table Configure

In the future, the BS shall send an MBS_MAP message on the Broadcast CID to specify the location and size of multi-BS MBS data bursts which are located in downlink permutation zones designated for MBS in frames that ranges from two to five frames from a single frame containing the MBS MAP message. Figure 4 is an illustration of MAC service access point (SAP) which provides an MS with MBS information elements to network layer which an MCID and logical channel ID will be shared.

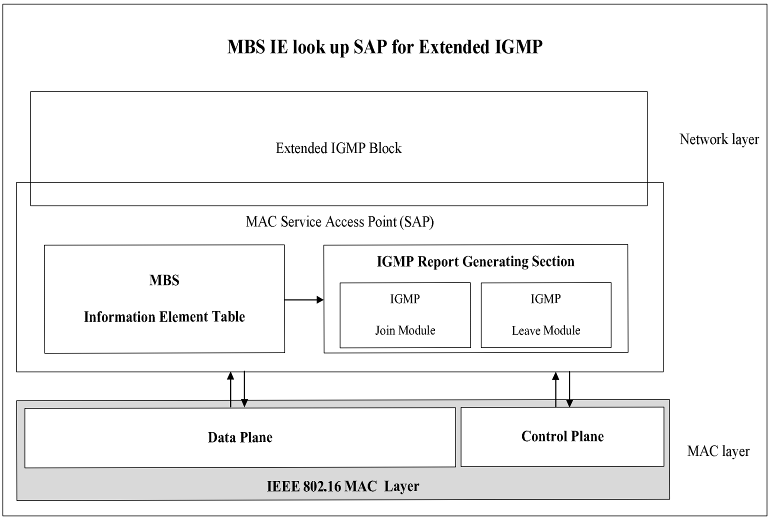

MBS IE look up SAP for Extended IGMP
Extended IGMP Block
Network layer
MAC Service Access Point (SAP)
MBS
Information Element Table
IGMP Report Generating Section
IGMP
Join Module
IGMP
Leave Module
Data Plane
Control Plane
MAC layer
IEEE 802.16 MAC Layer

Fig. 4. Basic function blocks with MCIDs mapping SAP for Extended IGMP

4 Extended IGMP Protocol Architecture in Network Layer

The proposed IGMP involves classifying multicast packets, updating the IPTV channel table, and managing channel control in network layer [12].

4.1 Multicast Packet Classifications in Extended IGMP Block

Figure 5 shows the extended IGMP protocol architecture in the network layer. Packet streams from ‘input module’, originated from service interface on mobile access network at the MS of viewer, are delivered to the ‘input packet classification section’. As shown in Figure 6, this section checks each octet and all specific field value in IP header for the packet’s destination IP address in order to classify packets into either general data packet/multicast packet for multimedia contents, or multicast packet for IGMP Group-Specific management. According to the address system of IPv4, the multicast IP is D-class (224.0.0.0~239.255.255.255) and leading 4 bits of uppermost octet is assigned as '1110'. The uppermost 8 bits start with hexadecimal '0xFF' in multicast address of IPv6 system. In this paper, the multicast address at IPv4 will be used for explanation. In other words, if the value of uppermost octet is greater than decimal '223' and less than '240' at the destination IP address of entered packet, the corresponding packet is a multicast packet. Therefore, the entered packet streams can be mainly classified into general data packets and multicast data packets using the method above. The separated general packets are sent to ‘general data packet handling section’. Next, all packets using contents for multimedia transmission (224.0.1.0~239.255.255.255), IGMP query (general query: 224.0.0.1, group-specific query: corresponding multicast address) and reserved address for specific protocol (random value between 224.0.0.2~224.0.0.255) are included in the separated multicast packets.

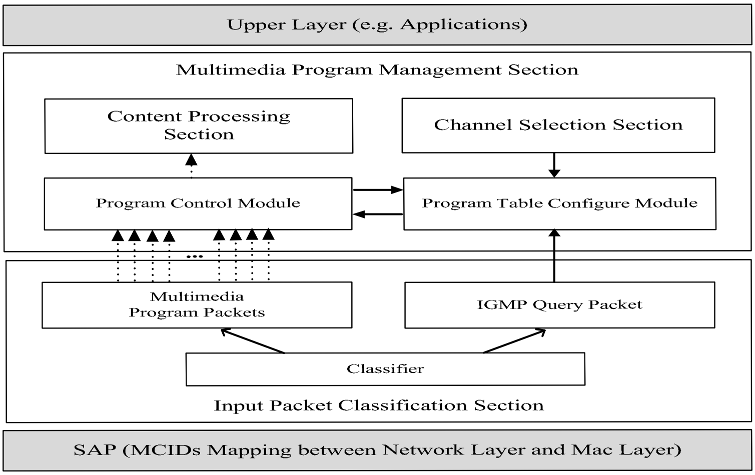

Upper Layer (e.g. Applications)
Multimedia Program Management Section
Content Processing Section
Channel Selection Section
Program Control Module
Program Table Configure Module
...
Multimedia Program Packets
IGMP Query Packet
Classifier
Input Packet Classification Section
SAP (MCIDs Mapping between Network Layer and Mac Layer)

Fig. 5. Extended IGMP protocol architecture in network layer

Therefore, in order to reclassify the multicast packets which is separated initially by usage, the second and third octet values are checked at the destination IP address using the same method as above. If both values are '0' and the value of the fourth octet '1', the corresponding packet is sent to IGMP general query 'program control module'. This is because all multicast packets, with the value of the fourth octet is not equivalent to '1', are often used by other specific protocol, therefore, they are delivered to the 'general data packet handling section'.

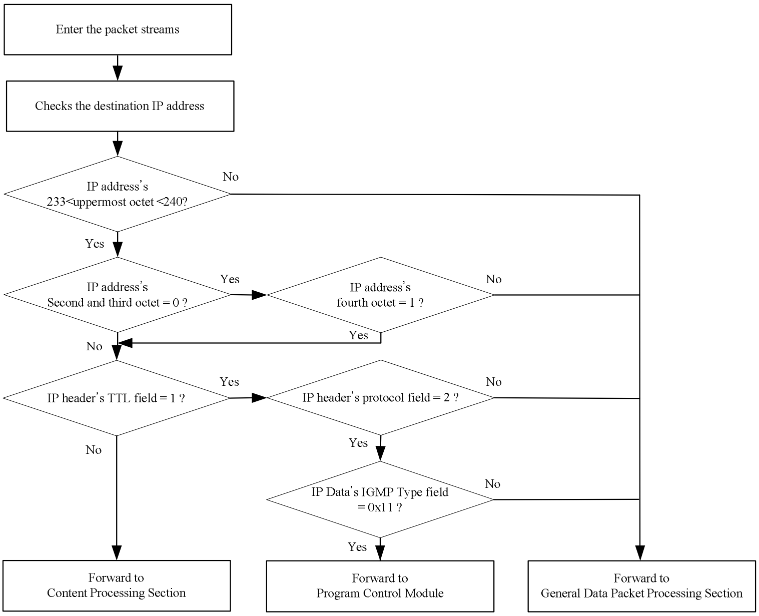

Enter the packet streams
Checks the destination IP address
IP address's
233<uppermost octet <240?
No
Yes
IP address's
Second and third octet = 0 ?
Yes
IP address's
fourth octet = 1 ?
No
Yes
No
IP header's TTL field = 1 ?
Yes
IP header's protocol field = 2 ?
No
No
Yes
IP Data's IGMP Type field
= 0x11 ?
No
Yes
Forward to
Content Processing Section
Forward to
Program Control Module
Forward to
General Data Packet Processing Section

Fig. 6. Flow chart of multicast packet classification

The remaining packets are content packets for multimedia transmission and packets for IGMP group-specific query. Group-specific query is a message inquiring existence of other participants on corresponding group in case multicast router has received IGMP leave report on specific group. Especially, the packets for IGMP group-specific query transmits using multicast address of corresponding group as destination IP address in contrast to the IGMP general query inquiring to all hosts (destination IP address: 224.0.0.1) on the network. Also, the TTL (Time To Live) field value of IP header is assigned as '1' and the value of protocol field is set as '2'. Therefore, in order to reclassify multicast packets other than two remaining types, the value of TTL field is checked at the IP header of corresponding packet. If the corresponding value is not '1', this is sent to 'program table configuration module' because the packet is for sending multimedia contents. Next, the remaining packets with TTL field value of '1' are send to 'program control module' after being classified as IGMP group-specific packet if the type field value of IGMP frame inserted to IP data field also corresponds to '0x11' specifying membership query after checking whether the protocol field value of IP header corresponds to '2' once again.

4.2 Multicast Program Channel Table Configure

In Figure 5, the ‘program table configure module’ of ‘program management section’ obtains multicast address which is destination address of corresponding packet and source address which is transmitting address once the multimedia content transmission packet is sent to organized this as multimedia program table. Also on the program table, the 'check viewing (CV)' field to check the program that the viewer is currently watching, 'reception time (RT)' field to check the time when program reception was started, 'max response time (MRT)' to check the maximum response time on programs that have IGMP group-specific query and 'waiting time' field to check continuity of program are organized together.

Table 1. IPTV Program table from extended IGMP

Destination IP	Source IP	CV	RT	MRT	WT
224.15.26.100	171.124.56.1	0	45000	45500	500
224.26.37.100	129.197.92.9	0	23000	23000	23500
224.37.48.100	166.214.55.7	1	1100	0	500
…	…	…	…	…	...

The 'check viewing' field value is expressed as 1 bit while being expressed as '1' if viewing is in progress and '0' if not. The 'reception time' field expresses the time from the point of starting program reception at the ‘program table organization module’ by counting as millisecond units. The 'Max Response Time' field is formed by adding the max response time field value of IGMP frame inserted to IP data domain of IGMP group-specific query packet handed over from ‘program control module’ to the 'reception time' field value on the program table. The initial value of 'max response time' is '0' and updated each time new value on group-specific query is handed over from ‘program control module’. While 'waiting time (WT)' field sets 'max response time' as one program receives IGMP group-specific query, a random 'waiting time' is assigned if the 'check viewing' value is maintained as '0' during the time when 'reception time' and 'max response time' becomes the same. This is to prevent immediate viewing until the continuity on multimedia program not participating directly in IGMP group-specific query is confirmed. Table 1 is a simple scenario of an IPTV program table from extended IGMP.

4.3 Multimedia Program Control

The ‘program control module’ receives IGMP query packet from the ‘input packet classification section’. If the received packet is IGMP general query, the ‘IGMP join module’ of ‘IGMP report generating section’ hands over the information of corresponding program to create IGMP join report to report join on the multimedia program with 'check viewing' value of '1' using multimedia program table. If the received packet is IGMP group-specific query, the status on whether multimedia program using multicast address of corresponding query is currently being viewed is checked using the multimedia program table. If the viewer is watching the corresponding program, the ‘IGMP join module’ of ‘IGMP report generating section’ hands over the information of corresponding program to create IGMP join report. If the corresponding program is not being watched, the received IGMP group-specific query is sent to ‘program organization module’ to form a 'max response time' field within the multimedia program table. Also, the response on corresponding query is not performed. Accordingly, the multicast router starts to leave corresponding group completely in case the response does not arrive within max response time of IGMP group-specific query.

4.4 Operations of the Extended IGMP

Figure 7 is an illustration of internet group management method proposed by this paper as sequence figure according to time. As element devices, it is assumed that the 'multimedia server' transmitting multimedia programs, the 'Edge Multicast Router' connecting transport network and access network, the 'Multicast Router' forwarding data according to multicast routing protocol of access network and the 'MS' that are subscriber terminals, exist. At first, the multimedia server transmits 'stream A' which is a multimedia program by multicast method.

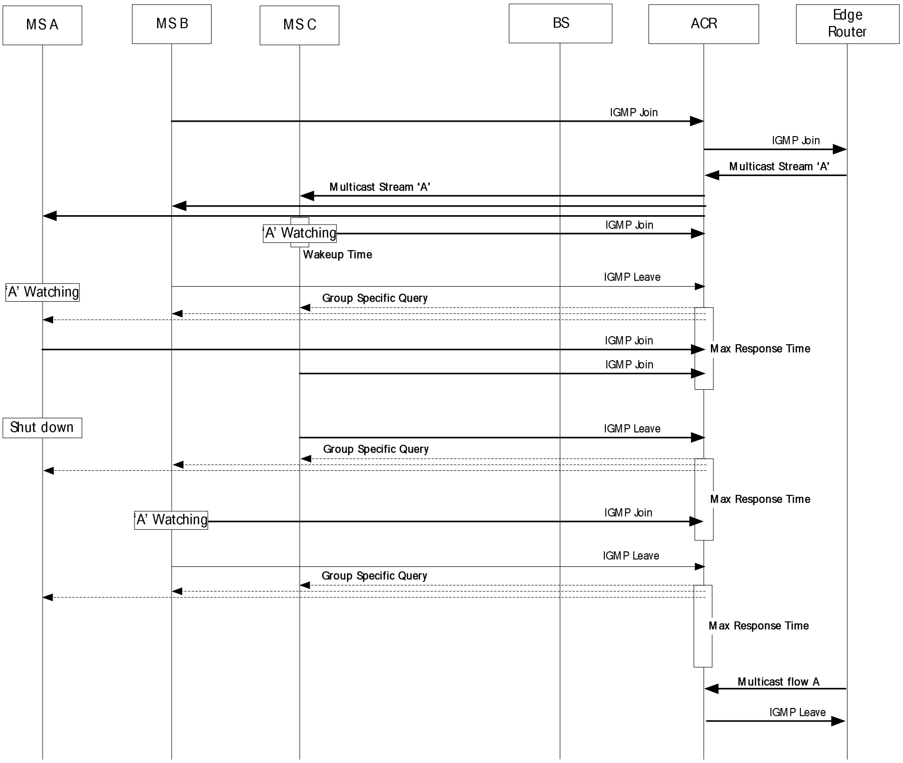


Fig. 7. Sequence diagram of the proposed extended IGMP in mobile WiMAX

The edge multicast router receive the corresponding program through transmission network. At this time, the subscriber terminal MS B sends IGMP join report to multicast router to view 'stream A'. Afterwards, mobile access networks are formed between multicast router and MS so that the 'stream A' requested by MS B is sent to both MS A and MS C existing on the same node. The 'stream A' transmitted this way gets to organize the program table at the MS. If A section is immediately perform after booting the MS or immediately perform after receiving new multimedia program, as shown in Figure 8, the multimedia program table is at an incomplete state. This is a section where the continuity of received program cannot be confirmed. Accordingly, a waiting time is set at the corresponding section and once the program is requested from the waiting time, an explicitly stated IGMP join report must be transmitted. Accordingly, the sequence for transmitting IGMP join report by MS C at the waiting time in order to view 'stream A' is illustrated. Next, once MS B sends IGMP leave report in order to change the multimedia program, the multicast router gets to send IGMP group-specific query. The MS A that had been viewing 'stream A' without explicitly stated IGMP join report at B section and MS C that had been viewing previously as illustrated in Figure 8 get to make a explicitly stated IGMP join report within max response time. Next, in case the MS of MS A that had been viewing 'stream A' becomes in a state of not viewing by being shut down and MS C that had been the last viewer also makes IGMP leave report on corresponding program, the multicast router gets to send a group-specific query.

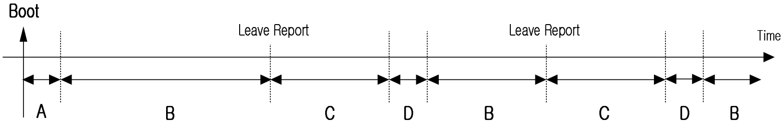


Fig. 8. Requesting sections of an IPTV program

For the MS B to watch the corresponding program once again at C section as illustrated in Figure 8, it gets to participate in viewing after making sure to send an explicitly stated IGMP join report. Next, if the MS B which is the last viewer of 'stream A' makes IGMP leave report, the multicast router gets to send a group-specific query and if the proper IGMP join report is failed to be received with max response time, the IGMP leave message is sent to multicast router that the data on 'stream A' is no longer received.

5 Performance Analysis

In order to analyses the performance of the new extended IGMP, we need analysis of the program popularity. Figure 9 shows the independent popularity change scenario and popularity-oriented change scenario when total numbers of program are 10, 15, and 20. For proper program popularity distribution, TV viewing behavior should be modeled. However, the program popularity is very difficult to be modeled. Therefore, we will model the channel popularity for a period of time.

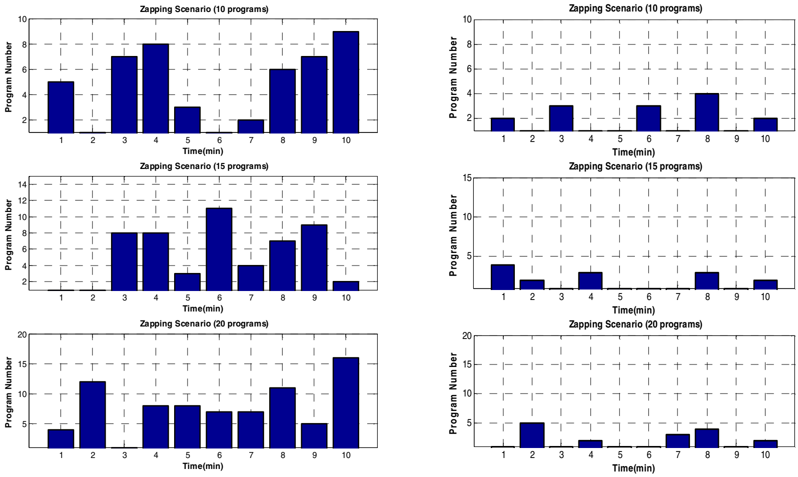


Fig. 9. Independent popularity change and popularity-oriented change scenario

As shown in Figure 10, if $\bar{C}_{select}$ is an element of $\bar{C}_{shared}$, a viewer immediately watches the selecting channel without an IGMP join report.

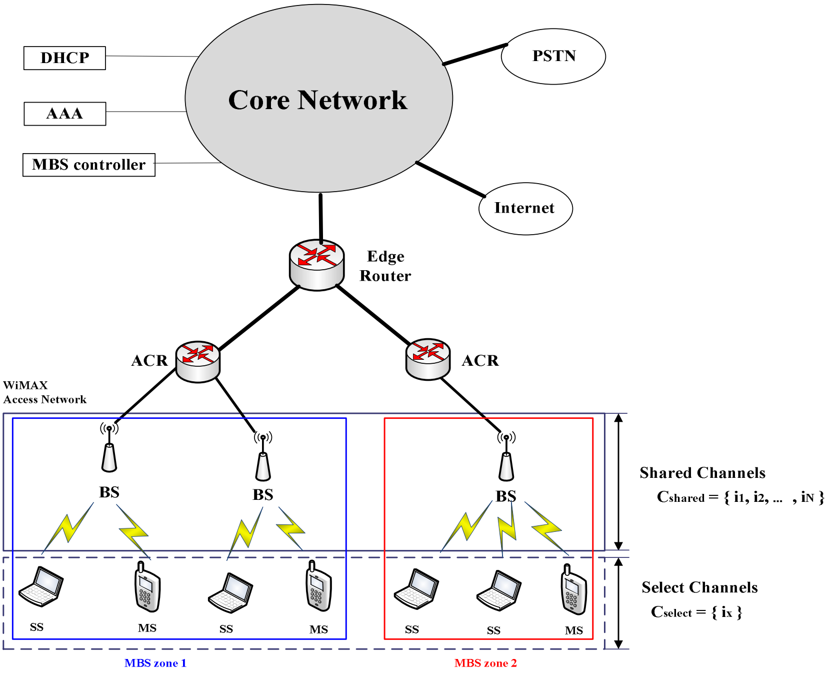


Fig. 10. Shared and select multimedia program channels

Accordingly, we attempt to investigate a channel popularity model in order to analyze the channel sharing. In Figure 11, there are 96 channels which are about Korea TV channel popularity density announced by TNS Media Research in Korea [13].

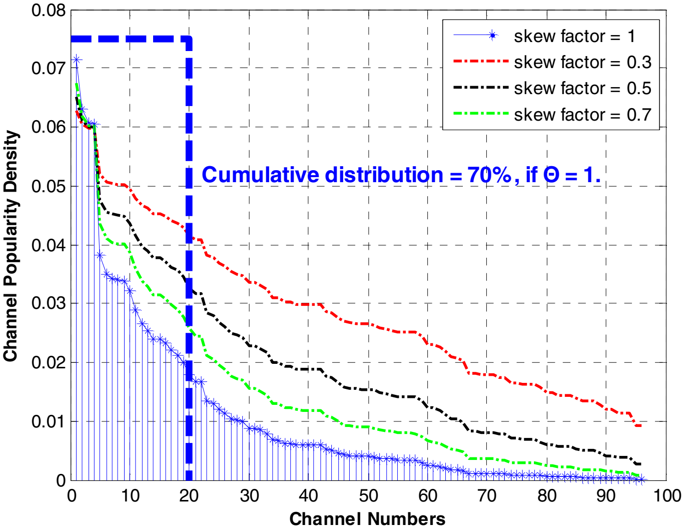


Fig. 11. Channel popularity density

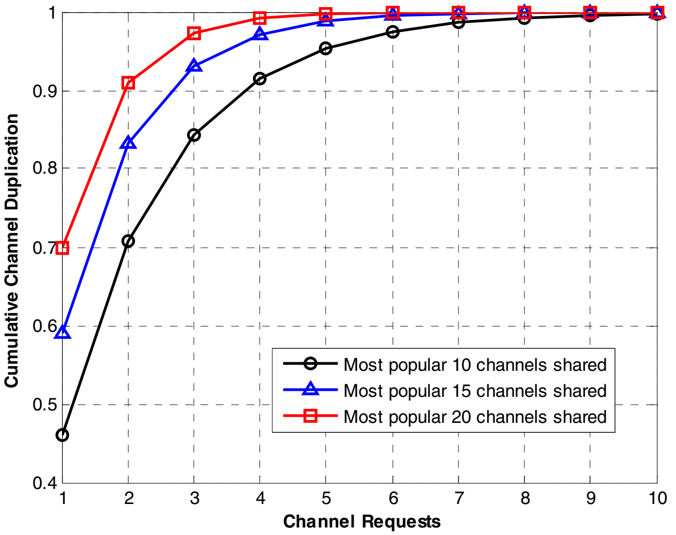


Fig. 12. Cumulative channel duplication probability on a MBS zone

$$\mathrm{p_X}(x) = \kappa x^{\Theta}, \quad \text{for } x=1, 2, \dots, \mathrm{N}\,, \tag{1}$$

where x is the channel's popularity-rank, Θ is the skew factor, κ is a normalization constant, and N is the number of channels [14]. We shall use this probability density to find the probability that X has values from 0 to most popular 10, 15, and 20 channels, respectivly. The probability is

$$\mathrm{P}\{0 \leq \mathrm{X} \leq \mathrm{N}\} = \int_0^{\mathrm{N}} \mathrm{p_X}(x)\mathrm{d}x = 1\,. \tag{2}$$

Thus, the event $\{0 \leq \mathrm{X} \leq 10\}$ is about 46%, the event $\{0 \leq \mathrm{X} \leq 15\}$ is about 59%, and the event $\{0 \leq \mathrm{X} \leq 20\}$ is about 70%. If the most popular channels are shared respectively in a service node, we can assume a channel duplication probability. The channel duplication probability implies that a viewer's selecting channel, $\mathrm{C_{select}}$, is already included in $\mathrm{C_{shared}}$ set. We shall repeat the experiment M times and determine the probability that $\mathrm{C_{select}} \in \mathrm{C_{shared}}$ is observed exactly r times out of the independent M trails.

$$\mathrm{P}\{\mathrm{C_{select}} \in \mathrm{C_{shared}}\} = \binom{\mathrm{M}}{\mathrm{r}}\mathrm{p^r}(1-\mathrm{p})^{\mathrm{M-r}}\,. \tag{3}$$

If most popular 10 channels are shared in a service node, the probability of a viewer choosing among the channels is 0.46 and M=1. If most popular 10 channels are shared in a service node, the probability of a viewer choosing among the channels is 0.46 and M=1. Figure 12 shows the cumulative channel duplication probability of 10 independent trials when total numbers of most popular shared channels are 10, 15, and 20. Notice that the newly proposed IGMP are able to let viewers immediately watch on the shared IPTV without channel change response time when more than 5 channel.

6 Conclusions

This paper proposes a new extended IGMP for IPTV services provided especially although mobile WiMAX access network. The proposed IGMP architecture includes MAC SAP, input packet classification section, multimedia program management section and IGMP report generation section. The MAC SAP provides downlink MAP information to an extended IGMP which can be allowed immediate channel switching. In the extended IGMP, general unicast packets and multicast packets are classified the packet streams entered to a MS of viewer from the service interface on the shared mobile access networks while classifying the separated multicast packets once again into multicast packets for sending IPTV contents and packets for IGMP query. Otherwise, the new IGMP with MAC SAP allows viewers to switch channels being watched by other viewers without going through complex traditional IGMP processes. In usual TV viewing situations, most popular channels are being watched by some views. Simulation results show that the newly proposed IGMP can immediately watch the shared IPTV channels without the channel change response time using the enhanced group join and leave process for mobile WiMAX, when more than 5 channel requests are made for most popular channels.

References

1. Teo, K.H., Tao, Z., Zhang, J.: The Mobile Broadband WiMAX Standard. IEEE Signal Processing Magazine 24, 144–148 (2007)
2. Deering, S.: Host Extensions for IP Multicasting. STD 5, RFC 1112 (August 1989)
3. Fenner, W.: Internet Group Management Protocol, Version 2. RFC 2236 (November 1997)
4. Cain, B., Deering, S., Thyagarajan, A.: Internet Group Management Protocol, Version 3. RFC 3376 (October 2002)
5. Deering, S., Fenner, W., Haberman, B.: Multicast Listener Discovery (MLD) for IPv6. RFC 2710 (October 1999)
6. Deering, S., Fenner, W., Haberman, B.: Multicast Listener Discovery (MLD) for IPv6. RFC 2710 (October 1999)
7. Cho, C., Han, I., Jun, Y., Lee, H.: Improvement of channel zapping time in IPTV services using the adjacent groups join-leave method. In: The 6th International Conference on Advanced Communication Technology, pp. 971–975 (2004)
8. Agilent Technologies Inc., Ensure IPTV Quality of Experience (2005)
9. Etemad, K., Wang, L.: Multicast and Broadcast Multimedia Services in Mobile WiMAX Networks. IEEE Communications Magazine 47, 84–91 (2009)
10. Jiang, T., Xiang, W.: Multicast broadcast services support in OFDMA-based WiMAX system. IEEE Communications Magazine 45(8), 78–86 (2007)
11. IEEE Std 802.16-2009, Part 16: Air Interface for Broadband Wireless Access Systems (May 2009)
12. Lee, E., Park, S.: Internet Group Mamagement Protocol for IPTV Services in Passive Optical Network. IEICE Trans. Commun. E93-B(2) (February 2010)
13. TNS Media Research in Korea, `http://www.tnsmk.co.kr`
14. Zipf, J.K.: Selective Studies and the Principle of Relative Frequency in Language (1932)

Optimal Expected Discounted Reward of a Wireless Network with Award and Cost[⋆]

Wenlong Ni[1], Wei Li[1], and Demetrios Kazakos[2]

[1] Department of Computer Science
[2] Department of Mathematics
Texas Southern University, Houston, TX 77004, USA
{niw,liw,kazakosd}@tsu.edu

Abstract. In this paper, we extend our previous optimization investigation on a single cell (IEEE Transactions on Wireless Communications, vol. 8, no. 2, pp. 1038-1044, 2009) to a whole network with multiple cells and further consider the call admission control (CAC) based on the total expected discounted reward. Here, the system will get the award for admitting a call, but will incur a cost for rejecting a call and incur a cost for holding a call. Call's routing between cells in the network is characterized probabilistically. Under several realistic assumptions on network in general, we consider the reserved channel scheme in the literature and applied the theory of continuous time Markov decision process to derive the optimal policy. We figure out the optimal CAC policy of when to admit or reject an arrival call (either new call or handoff call) in order to achieve the maximum total expected discounted reward for the scheme. Our numerical analysis confirms the correctness of our result for this whole network investigation. The result of this paper could be applied in designing wireless networks for optimal performance, and be extended to the multiple classes of calls in the deign of future wireless mobile multimedia networking.

Keywords: Reserved Channel Scheme, Optimal Call Admission Control, Control Limit Policy.

1 Introduction

In cellular wireless networks, the calls are normally divided into two groups: new calls and handoff calls. When a user moves from one cell to another, the base station in the new cell must take responsibility for this new arriving call and all calls previously already established the connections. Since premature termination of established connections is usually more objectionable than rejection of a new connection request, it is widely believed that a wireless network must give higher priority to the handoff connection requests as compared to new connection requests. Many different admission control strategies have been discussed in

[⋆] This work was supported in part by the National Science Foundation under Grants CCF-0829769 and CNS-0830011.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 427–438, 2010.

literature [1,2,3,4] to provide priorities to handoff requests without significantly jeopardizing the new connection requests. The basic idea of these admission control strategies is establish a scheme for admitting or rejecting an arriving call, with priority to handoff calls in each cell with limited resources and is widely used because of its simplicity.

In this paper, we consider the call admission control problem in a wireless network with multiple cells, and there is routing probability consideration among all cells. What makes the decision difficult is that, to achieve some sense of optimality, one needs to consider the future status of the network resources and the pattern of the future arrival requests to accept or reject the current incoming calls. Here, he optimization problem on maximum reward is modeled as a Continuous Time Markov decision process (CTMDP), [4,5,6,7]. To achieve the optimality on the expected rewards, the CTMDP model is described as an infinite-horizon problem with discounting. Due to the finite state space and action space, by using the Rate Uniformization technique [5], the CTMDP model can be transformed to a discrete MDP model, thus the theorems and algorithms for MDP models could be applied. The Value Iteration Method is used to solve this CTMDP problem, not in the way of Linear Programming as in [8,9]. We would like to pint out that in our recent paper [10], we considered the optimization problem in a single cell. However, the current paper is a major step to extend our previous result to the **whole network**.

The rest of this paper is organized as follows. Section 2 discusses the modeling and Section 3 describes the structure of optimal policy for the RCS schemes. Numerical analysis for the schemes are discussed in Section 4 and the final Section 5 is a conclusion for this paper.

2 Model Development

A. Assumptions: The assumptions and notations for this wireless cellular network using Reserved Channel Scheme [11,12] are as follows.

1. The network consists of a number of N cells.
2. New calls are generated in cell i according to a Poisson process with rate λ_i, $i = 1, 2, \ldots, N$. The requested call connection time (RCCT) of a new call at cell i is exponentially distributed with means $1/h_i$.
3. The call residence time in cell i, which is defined as the length of time a call stays in the cell i and which depends on the velocity and the direction of the mobile terminal, is exponentially distributed with means $1/r_i$.
4. The probability that a call moves from cell i to a neighboring cell j, given that it moves to a neighboring cell before the call is completed, is $p_{i,j}$, where $\sum_{j=1}^{N} p_{i,j} = 1$. Clearly, $p_{i,i} = 0$ and cell j is a neighboring cell of i if and only if $p_{i,j} > 0$.
5. There are $C_1, \ldots, C_N$ channels in each cell of the network. The reserved channels for handoff calls are $G_1, \ldots, G_N$ in each cell of the network. The new call or handoff request are rejected if there are not enough channels.

6. Accepting a new call in cell i would contribute R_i units of reward to the system, rejecting a handoff call into cell i would cost ϕ_i units of reward to the system. Let the number vector of calls in cells be $\mathbf{n} = (n_1, n_2, \cdots, n_N)$, n_i is the numbers for calls in cell i. The system incurs a holding cost rate $f(\mathbf{n})$ per unit time.

B. Objective Function: In CTMDP models, a decision rule prescribes a procedure for action selection in each state at a specified decision epoch. Decision rules range in generality from deterministic Markovian to randomized history dependent, depending on how they incorporate past information and how they select actions. Deterministic Markovian decision rules specify the action choice when the system occupies state s at decision epoch t. A policy π specifies the decision rule to be used at every decision epoch. It provides the decision maker with a prescription for action selection under any possible future system state or history. A policy is stationary if, for each decision epoch t, the decision $d_t = d$ is the same, which can be denoted by d^∞. For each policy π, let $v_\alpha^\pi(s)$ denote the total expected infinite-horizon discounted reward with α as the discount factor, given that the process occupies state s at the first decision epoch. Our objective is to find an optimal policy π that can bring the maximum total expected discounted reward $v_\alpha^\pi(s)$ for every initial state s, i.e the objective function is,

$$v_\alpha^\pi(s) = E_s^\pi \left\{ \sum_{k=0}^{\infty} e^{-\alpha t_k} r(s_k, a_k) \right\}, \tag{1}$$

where t_k is the time point of system at epoch k $(s_0 = s)$, s_k is the state of system at epoch k, a_k is the action to take at state s_k, and $r(s_k, a_k)$ represents the reward received during epoch k when taking action a_k in state s_k.

C. Construction of Models: Based on these assumptions, we can build the SMDP model for this wireless cellular network using Reserved Channel Scheme as follows:

- **State Space:** Let the state variable consists of number of calls in the system, the status of calls leaving or arriving to the system. So state space $S = \{<\mathbf{n}, b>\}$, where $b \in \{D_i, H_{ij}, A_i\}$, $n_i \leq C_i, i, j = (1, 2, \cdots, N)$, and $i \neq j$. Here b stands for the last call event, D_i means a departure from cell i, H_{ij} stands for a handoff request from cell i to cell j, A_i is an arrival of a new call in cell i.
- **Action Space:** In states $\langle \mathbf{n}, D_i \rangle$, set a_C as the action to continue, thus $A_{\langle \mathbf{n}, D_i \rangle} = \{a_C\}$. In states $\langle \mathbf{n}, A_i \rangle$ and $\langle \mathbf{n}, H_{ij} \rangle$, set a_R as the action to reject the call and a_A as the action to admit, so $A_{\langle \mathbf{n}, A_i \rangle} = \{a_R, a_A\}$ and $A_{\langle \mathbf{n}, H_{ij} \rangle} = \{a_R, a_A\}$, $i, j = (1, 2, \cdots, N)$, and $i \neq j$.
- **Decision epochs:** At each decision epoch, let $\tau(s, a)$ be the sojourn time starting from state s taking action a. Let $F(t|s, a)$ denotes the probability that the next decision epoch occurs within t time units, given that the decision maker chooses action a in state s,

$$P(\tau(s, a) \leq t) = F(t|s, a) = 1 - e^{-\beta(s,a)t}, t \geq 0.$$

For each state $s = \langle \mathbf{n}, b \rangle$ and action a, let $\beta_0(\mathbf{n}) = \sum_{i=1}^{N} \left[\lambda_i * 1_{(n_i < C_i - G_i)} + n_i(r_i + h_i)\right]$, so $\beta(s,a)$ can be written as

$$\beta(s,a) = \begin{cases} \beta_0(\mathbf{n}_i),\ b = D_i, a = a_C, \\ \beta_0(\mathbf{n}),\ b = A_i, a = a_R, \\ \beta_0(\mathbf{n}^i),\ b = A_i, a = a_A, \\ \beta_0(\mathbf{n}_i),\ b = H_{ij}, a = a_R, \\ \beta_0(\mathbf{n}_i^j),\ b = H_{ij}, a = a_A, \end{cases} \tag{2}$$

where $\mathbf{n}_i = (n_1, n_2, \ldots, \max(n_i - 1, 0), \ldots, n_N)$, $\mathbf{n}^i = (n_1, n_2, \ldots, n_i + 1, \ldots, n_N)$,

$$\mathbf{n}_i^j = \begin{cases} \mathbf{n}, i = j, \\ (n_1, n_2, \ldots, n_i - 1, \ldots, n_j + 1, \ldots, n_N), i < j, \\ (n_1, n_2, \ldots, n_j + 1, \ldots, n_i - 1, \ldots, n_N), i > j, \end{cases}$$

$i, j = (1, 2, \cdots, N)$. Here $1_{(\cdot)}$ is the indicator function.

- **Transition Probabilities:** Let $q(z|s,a)$ denote the probability that the system occupies state z in the next epoch, if at the current epoch the system is at state s and the decision maker takes action $a \in A_s$. For states $s = \langle \mathbf{n}, D_i \rangle$, $a = a_C$, the state transition probability, $q(z|\langle \mathbf{n}, D_i \rangle, a_C)$, is

$$\begin{cases} \lambda_k * 1_{((n_k - 1_{(k=i)}) < C_k - G_k)} / \beta_0(\mathbf{n}_i), & z = \langle \mathbf{n}_i, A_k \rangle, \\ (n_k - 1_{(k=i)}) h_k / \beta_0(\mathbf{n}_i), & z = \langle \mathbf{n}_i, D_k \rangle, \\ (n_k - 1_{(k=i)}) r_k p_{kl} / \beta_0(\mathbf{n}_i), & z = \langle \mathbf{n}_i, H_{kl} \rangle, \end{cases} \tag{3}$$

where $k, l = (1, 2, \cdots, N)$ and $k \neq l$. For states $s = \langle \mathbf{n}, A_i \rangle$, admitting a new call arrival in cell i would increase the calls in the system from $\mathbf{n}$ to $\mathbf{n}^i$, rejecting a new call arrival in cell i would keep the calls in the system unchanged, so we have that $q(z|\langle \mathbf{n}, A_i \rangle, a_A)$ is

$$\begin{cases} \lambda_k * 1_{((n_k + 1_{(k=i)}) < C_k - G_k)} / \beta_0(\mathbf{n}^i), & z = \langle \mathbf{n}^i, A_k \rangle, \\ (n_k + 1_{(k=i)}) h_k / \beta_0(\mathbf{n}^i), & z = \langle \mathbf{n}^i, D_k \rangle, \\ (n_k + 1_{(k=i)}) r_k p_{kl} / \beta_0(\mathbf{n}^i), & z = \langle \mathbf{n}^i, H_{kl} \rangle, \end{cases} \tag{4}$$

where $k, l = (1, 2, \cdots, N)$ and $k \neq l$. So, we can see that

$$q(z|\langle \mathbf{n}, A_i \rangle, a_R) = q(z|\langle \mathbf{n}^i, D_i \rangle, a_C), i = (1, 2, \cdots, N),$$
$$q(z|\langle \mathbf{n}, A_i \rangle, a_A) = q(z|\langle \mathbf{n}^i, A_i \rangle, a_R), i = (1, 2, \cdots, N).$$

For states $s = \langle \mathbf{n}, H_{ij} \rangle$, allowing a handoff request H_{ij} would change the number of calls in system from $\mathbf{n}$ to $\mathbf{n}_i^j$, rejecting such request would leave $\mathbf{n}_i$ calls in the system for all $i, j = (1, 2, \cdots, N)$, we have

$$q(z|\langle \mathbf{n}, H_{ij} \rangle, a_A) = q(z|\langle \mathbf{n}^j, D_i \rangle, a_C), i \neq j,$$
$$q(z|\langle \mathbf{n}, H_{ij} \rangle, a_R) = q(z|\langle \mathbf{n}, D_i \rangle, a_C), i \neq j.$$

- **Reward Functions:** Because the system state does not change between decision epochs, the expected discounted reward starting from state s taking action a satisfies

$$\begin{aligned} r(s,a) &= k(s,a) + c(s,a)E_s^a\left\{\int_0^{\tau(s,a)} e^{-\alpha t}dt\right\}, \\ &= k(s,a) + c(s,a)E_s^a\left\{[1-e^{-\alpha\tau(s,a)}]/\alpha\right\}, \\ &= k(s,a) + \frac{c(s,a)}{\alpha+\beta(s,a)}, \end{aligned} \tag{5}$$

where

$$k(s,a) = \begin{cases} 0, & b = D_i, a = a_C, \\ R_i, & b = A_i, a = a_A, \\ 0, & b = A_i, a = a_R, \\ 0, & b = H_{ij}, a = a_A, \\ -\phi_j, & b = H_{ij}, a = a_R, \end{cases} \tag{6}$$

and $c(s,a)$ is the holding cost rate function if the system is at state s and takes action a. It can be defined as

$$c(s,a) = \begin{cases} -f(\mathbf{n}_i), & b = D_i, a = a_C, \\ -f(\mathbf{n}^i), & b = A_i, a = a_A, \\ -f(\mathbf{n}), & b = A_i, a = a_R, \\ -f(\mathbf{n}_i^j), & b = H_{ij}, a = a_A, \\ -f(\mathbf{n}_i), & b = H_{ij}, a = a_R. \end{cases} \tag{7}$$

Thus, from equation (1) and (5) the objective function $v_\alpha^\pi(s)$ can be written as

$$E_s^\pi\left\{\sum_{k=0}^{\infty} e^{-\alpha t_k}\left[k(s_k,a_k) + \frac{c(s_k,a_k)}{\alpha+\beta(s_k,a_k)}\right]\right\}. \tag{8}$$

3 Optimal Policy

Based on the assumptions, for the admission control problem, both the state space S and the action space A_s are finite, the reward function $r(s,a)$ is also finite. From *Theorem 11.3.2* of [5], the optimal policy is a stationary deterministic policy d^∞, so the problem can be reduced to finding a deterministic decision rule d. For each deterministic decision rule d, let $q_d(z|s) = q(z|s,d(s))$, $r_d(s) = r(s,d(s))$ and $\beta_d(s) = \beta(s,d(s))$, from equation (8) we have,

$$\begin{aligned} v_\alpha^{d^\infty}(s) &= r_d(s) + E_s^\pi\{e^{-\alpha\tau(s,d(s))}v_\alpha^{d^\infty}(s_1)\}, \\ &= r_d(s) + \sum_{z\in S}\left[\int_0^\infty \beta_d(s)e^{-[\alpha+\beta_d(s)]t}dt\right] q_d(z|s)v_\alpha^{d^\infty}(z), \\ &= r_d(s) + \frac{\beta_d(s)}{\alpha+\beta_d(s)}\sum_{z\in S} q_d(z|s)v_\alpha^{d^\infty}(z). \end{aligned} \tag{9}$$

We use rate uniformization technique to calculate $v_\alpha^{d^\infty}(s)$. Based on the assumptions, our process fits the condition of *Assumption 11.5.1* of [5], which is $[1 - q(s|s,a)]\beta(s,a) \leq c, \forall s \in S, a \in A_s$, here $c = \sum_{i=1}^{N}[\lambda_i + C_i * (r_i + h_i)]$ is a constant. So, we can define a uniformization of our process with components denoted by $\sim$. Let $\tilde{S} = S$ and $\tilde{A}_s = A_s$, we have

$$\tilde{q}(z|s,a) = \begin{cases} 1 - \frac{[1-q(s|s,a)]\beta(s,a)}{c}, & z = s, \\ \frac{q(z|s,a)\beta(s,a)}{c}, & z \neq s. \end{cases} \tag{10}$$

For the reward functions, we have

$$\tilde{r}(s,a) \equiv r(s,a)\frac{\alpha + \beta(s,a)}{\alpha + c}. \tag{11}$$

From *Proposition 11.5.1* [5], for each d^∞ policy and $s \in S$, we have

$$\tilde{v}_\alpha^{d^\infty}(s) = v_\alpha^{d^\infty}(s). \tag{12}$$

From equations (9) and (12), the optimality equation of $v(s)$ for maximum $v_\alpha^\pi(s)$ would have the form of

$$v(s) = \tilde{v}(s) = \max_{a \in A_s}\{\tilde{v}(s,a)\} = \max_{a \in A_s}\left\{\tilde{r}(s,a) + \frac{c}{c+\alpha}\sum_{z \in S}\tilde{q}(z|s,a)v(z)\right\} \tag{13}$$

After uniformization, the transition process from one state to another can be described by a discrete-time Markov chain which allows fictitious transitions from a state to itself.

From equations (5), (6), (7), (13) and (14), we have for states with $b = D_i$, there is only one action a_C, the reward is

$$\tilde{r}(\langle \mathbf{n}, D_i \rangle, a_C) = \frac{-f(\mathbf{n}_i)}{\alpha + c}. \tag{14}$$

From equations (3), (10), the transition probability $\tilde{q}(z|\langle \mathbf{n}, D_i \rangle, a_C)$ is

$$\begin{cases} \lambda_k * 1_{((n_k - 1_{(k=i)}) < C_k - G_k)}/c, & z = \langle \mathbf{n}_i, A_k \rangle, \\ (n_k - 1_{(k=i)})h_k/c, & z = \langle \mathbf{n}_i, D_k \rangle, \\ (n_k - 1_{(k=i)})r_k p_{kj}/c, & z = \langle \mathbf{n}_i, H_{kj} \rangle, \\ (c - \beta_0(\mathbf{n}_i))/c, & z = \langle \mathbf{n}, D_i \rangle, \end{cases} \tag{15}$$

for $k, j = (1, 2, \cdots, N)$ and $i \neq j$. From equations (13), (14) and (15), we have

$$\begin{aligned} v(\langle \mathbf{n}, D_i \rangle) = \frac{1}{\alpha + c}[& -f(\mathbf{n}_i) + \sum_{k=1}^{N}\lambda_k v(\langle \mathbf{n}_i, A_k \rangle) * 1_{((n_k - 1_{(k=i)}) < C_k - G_k)} \\ & + \sum_{k=1}^{N}(n_k - 1_{(k=i)})h_k v(\langle \mathbf{n}_i, D_k \rangle) \\ & + \sum_{j=1}^{N}\sum_{k=1}^{N}(n_k - 1_{(k=i)})r_k p_{kj} v(\langle \mathbf{n}_i, H_{kj} \rangle) \\ & + (c - \beta_0(\mathbf{n}_i))v(\langle \mathbf{n}, D_i \rangle)]. \end{aligned} \tag{16}$$

In the same way, for states $s = \langle \mathbf{n}, A_i \rangle$, from equations (3), (4) and (13), we have

$$\begin{aligned}\tilde{v}(\langle \mathbf{n}, A_i \rangle, a_R) = &\frac{1}{\alpha + c}[-f(\mathbf{n}) + \sum_{k=1}^{N} \lambda_k v(\langle \mathbf{n}, A_k \rangle) * 1_{(n_k < C_k - G_k)} \\ &+ \sum_{k=1}^{N} n_k h_k v(\langle \mathbf{n}, D_k \rangle) + \sum_{j=1}^{N} \sum_{k=1}^{N} n_k r_k p_{kj} v(\langle \mathbf{n}, H_{kj} \rangle) \\ &+(c - \beta_0(\mathbf{n})) v(\langle \mathbf{n}, A_i \rangle)], \\ = &\frac{1}{\alpha + c}[(\alpha + \beta_0(\mathbf{n})) v(\langle \mathbf{n}^i, D_i \rangle) + (c - \beta_0(\mathbf{n})) v(\langle \mathbf{n}, A_i \rangle)], \qquad (17)\end{aligned}$$

and,

$$\begin{aligned}\tilde{v}(\langle \mathbf{n}, A_i \rangle, a_A) = &\frac{1}{\alpha + c}[(\alpha + \beta_0(\mathbf{n}^i)) R_i - f(\mathbf{n}^i) \\ &+ \sum_{k=1}^{N} \lambda_k v(\langle \mathbf{n}^i, A_k \rangle) * 1_{((n_k + 1_{(k=i)}) < C_k - G_k)} \\ &+ \sum_{k=1}^{N} (n_k + 1_{(k=i)}) h_k v(\langle \mathbf{n}^i, D_k \rangle) \\ &+ \sum_{j=1}^{N} \sum_{k=1}^{N} (n_k + 1_{(k=i)}) r_k p_{kj} v(\langle \mathbf{n}^i, H_{kj} \rangle) \\ &+(c - \beta_0(\mathbf{n}^i)) v(\langle \mathbf{n}, A_i \rangle)], \\ = &\frac{1}{\alpha + c}[(\alpha + \beta_0(\mathbf{n}^i))(R_i + v(\langle \mathbf{n}^{ii}, D_i \rangle)) \\ &+(c - \beta_0(\mathbf{n}^i)) v(\langle \mathbf{n}, A_i \rangle)]. \qquad (18)\end{aligned}$$

Also, we have

$$v(\langle \mathbf{n}, A_i \rangle) = \begin{cases} \max\left[\tilde{v}(\langle \mathbf{n}, A_i \rangle, a_R), \tilde{v}(\langle \mathbf{n}, A_i \rangle, a_A)\right], & n_i < C_i - G_i, \\ \tilde{v}(\langle \mathbf{n}, A_i \rangle, a_R), & n_i \geq C_i - G_i. \end{cases}$$

From equation (16), it is seen that $v(\langle \mathbf{n}^i, D_i \rangle) = v(\langle \mathbf{n}^j, D_j \rangle)$, for $i, j \in (1, 2, \cdots, N)$. So the value of $v(\langle \mathbf{n}^i, D_i \rangle)$ is not dependent on i, but is dependent on $\mathbf{n}$. Let $g(\mathbf{n}) = v(\langle \mathbf{n}^i, D_i \rangle)$, from equations (17) and (18), the calculation of $v(\langle \mathbf{n}, A_i \rangle)$ can be simplified to

$$v(\langle \mathbf{n}, A_i \rangle) = \begin{cases} \max\left[g(\mathbf{n}), R_i + g(\mathbf{n}^i)\right], & n_i < C_i - G_i, \\ g(\mathbf{n}), & n_i \geq C_i - G_i. \end{cases} \qquad (19)$$

Similarly for events $b = H_{ij}$, we have

$$v(\langle \mathbf{n}, H_{ij} \rangle) = \max\left[v(\langle \mathbf{n}, D_i \rangle) - \phi_j, v(\langle \mathbf{n}^j, D_i \rangle)\right]. \qquad (20)$$

Let $\beta_1(\mathbf{n}) = \sum_{i=1}^{N} (\lambda_i + n_i(r_i + h_i))$. If $(n_k - 1_{(k=i)}) \geq C_k - G_k$, which means the arrival of new calls in cell k can only be rejected, we have $v(\langle \mathbf{n}_i, A_k \rangle) = v(\langle \mathbf{n}, D_i \rangle)$. Equation (16) can be transformed to

$$\begin{aligned}v(\langle \mathbf{n}, D_i\rangle) = \frac{1}{\alpha + c}[&-f(\mathbf{n}_i) + \sum_{k=1}^{N} \lambda_k v(\langle \mathbf{n}_i, A_k\rangle)\\ &+\sum_{k=1}^{N}(n_k - 1_{(k=i)})h_k v(\langle \mathbf{n}_i, D_k\rangle)\\ &+\sum_{j=1}^{N}\sum_{k=1}^{N}(n_k - 1_{(k=i)})r_k p_{kj} v(\langle \mathbf{n}_i, H_{kj}\rangle)\\ &+(c - \beta_1(\mathbf{n}_i))v(\langle \mathbf{n}, D_i\rangle)].\end{aligned} \tag{21}$$

Definition: A function $f : R^k \to R$ is supermodular if

$$f(x \vee y) + f(x \wedge y) \geq f(x) + f(y),$$

for all x, y $\in R^k$, where $x \vee y$ denotes the componentwise maximum and $x \wedge y$ the componentwise minimum of x and y. If $-f$ is supermodular, then f is called submodular.

Theorem: If $v(\langle \mathbf{n}, D_i\rangle)$ is a submodular function on $\mathbf{n}$, then the optimal admission control policy with RCS scheme for new call arrivals and handoff calls are control limit policies.

Proof: We already know that the optimal policy is a stationary deterministic policy. Let $\triangle v_i(\langle \mathbf{n}, D_i\rangle) = v(\langle \mathbf{n}^{ii}, D_i\rangle) - v(\langle \mathbf{n}^{i}, D_i\rangle)$ and $\triangle v_i^j(\langle \mathbf{n}, D_i\rangle) = v(\langle \mathbf{n}^{j}, D_i\rangle) - v(\langle \mathbf{n}, D_i\rangle)$. So $\triangle v_i(\langle \mathbf{n}, D_i\rangle)$ and $\triangle v_i^j(\langle \mathbf{n}, D_i\rangle)$ are nonincreasing, we have the decision rule for the new call arrivals as

$$d(\langle \mathbf{n}, A_i\rangle) = \begin{cases} a_A, \triangle v_i(\langle \mathbf{n}, D_i\rangle) > -R_i, \\ a_R, \triangle v_i(\langle \mathbf{n}, D_i\rangle) \leq -R_i. \end{cases}$$

For the handoff requests,

$$d(\langle \mathbf{n}, H_{ij}\rangle) = \begin{cases} a_A, \triangle v_i^j(\langle \mathbf{n}, D_i\rangle) > -\phi_j, \\ a_R, \triangle v_i^j(\langle \mathbf{n}, D_i\rangle) \leq -\phi_j. \end{cases}$$

So, if $d(\langle \mathbf{n}, A_i\rangle) = a_R$, since

$$v(\langle \mathbf{n}^{i}, D_i\rangle) + v(\langle \mathbf{n}^{iix}, D_i\rangle) \leq v(\langle \mathbf{n}^{ix}, D_i\rangle) + v(\langle \mathbf{n}^{ii}, D_i\rangle),$$

which means $\triangle v_i(\langle \mathbf{n}^x, D_i\rangle) \leq \triangle v_i(\langle \mathbf{n}, D_i\rangle))$, so we have $d(\langle \mathbf{n}^x, A_i\rangle) = a_R$. Similarly if $d(\langle \mathbf{n}, H_{ij}\rangle) = a_R$, we have $d(\langle \mathbf{n}^x, H_{ij}\rangle) = a_R$, and so on as x, i, j goes through $1, 2, \ldots, N$. Consequently the optimal policy for both new calls and handoff requests is a control limit policy (or threshold policy).

Remark: We observe that the function $v(\langle \mathbf{n}, D_i\rangle)$ will not be a submodular function on $\mathbf{n}$ if the cost function is not a supermodular function on $\mathbf{n}$. Please see function $f_4(\mathbf{n})$ in the section of numerical analysis. Therefore, in order to make the optimal admission control policy with RCS scheme for either new call arrivals or handoff calls to be a control limit policy, we must carefully select the cost function from the class of supermodular functions.

4 Numerical Analysis

Without loss of generality, we suppose there are $N = 3$ cells in the wireless network. The routing probabilities among cells are set in Table 1. For simplification, let the discount factor $\alpha = 0.1$, and set the other parameters for analysis as in Table 2 to study the performance of the RCS scheme. We get the $v(s)$ values for states in the following Tables.

Table 1. Routing Probablities

Cell	1	2	3
1	0	0.4	0.6
2	0.5	0	0.5
3	0.8	0.2	0

Table 2. Parameters Setting

Cell	C_i	λ_i	h_i	r_i	R_i	ϕ_i	G_i
1	5	4	6	4	2	1	2
2	5	2	4	5	1.6	1.1	1
3	5	1	3	2	1.5	1.2	0

In Table 2, λ is the arrival rate of new calls, h is the call connection rate, r is the cell residence rate, R is the reward for new calls, ϕ is the cost for rejecting a handoff call, and G is the number of reserved channels.

Next, we show the performance of the RCS scheme with different cost functions. First, let us set the cost function be $f_1(\mathbf{n}) = 4n_1^2 + 3n_2^2 + 2n_3^2$, which is a supermodular function. The corresponding $v(\langle \mathbf{n}, D_3\rangle)$ are shown in Table 3. It can be seen that $\triangle v_1(\langle(3, n_2, 1), D_3\rangle)$, $n_2 = 0, \ldots, 5$ are less than $-R_1$ (R_1=2), which means that the system would reject cell-1 new call arrivals if there are already 3 calls in cell-1.

Table 3. RCS scheme: Function 1

$v(\langle \mathbf{n}, D_3\rangle)$	$n_2 = 0$	1	2	3	4	5
$n_1 = 0$	34.7039	33.8302	32.6263	31.1079	29.2705	27.1087
1	33.5482	32.6423	31.4261	29.9008	28.0585	25.8929
2	31.9847	31.0542	29.8283	28.2975	26.4513	24.2828
3	30.0533	29.1045	27.8714	26.3365	24.4875	22.3168
4	27.7431	26.7785	25.5391	24.0007	22.1492	19.9765
5	25.0446	24.0665	22.8216	21.28	19.4263	17.252

Set the cost function to $f_2(\mathbf{n}) = 2(n_1 + n_2 + n_3)^2$, which is also a supermodular function. The corresponding $v(\langle \mathbf{n}, D_3\rangle)$ are shown in Table 4. It can be seen that $\triangle v_1(\langle(4, 2, 1), D_3\rangle)$ and $\triangle v_1(\langle(3, 3, 1), D_3\rangle)$ are less than $-R_1$, so if there are 2

Table 4. RCS scheme: Function 2

$v(\langle \mathbf{n}, D_3\rangle)$	$n_2 = 0$	1	2	3	4	5
$n_1 = 0$	39.1698	38.3064	37.23	35.9378	34.4342	32.7217
1	38.1851	37.099	35.8102	34.3136	32.6092	30.7056
2	36.9665	35.6818	34.1923	32.4959	30.6012	28.5016
3	35.5502	34.0692	32.382	30.496	28.4058	26.107
4	33.9247	32.262	30.3884	28.3084	26.0198	23.5199
5	32.1286	30.2752	28.2083	25.9307	23.4414	20.7389

calls at cell-2, the system stops accepting cell-1 arrivals when there are 4 calls in cell-1, but if there are 3 calls at cell-2, the system stops accepting cell-1 arrivals when there are 3 calls in cell-1.

Set the cost function to $f_3(\mathbf{n}) = 2(n_1 + 1) * (n_2 + 1) * (n_3 + 1)$, which is a supermodular function. The corresponding $v(\langle \mathbf{n}, D_3\rangle)$ are shown in Table 5. It can be seen that all the $\triangle v_1(\langle \mathbf{n}, D_3\rangle)$ are greater than $-R_1$, so the system would always accept new call arrivals in cell-1 if there are free space.

Table 5. RCS scheme: Function 3

$v(\langle \mathbf{n}, D_3\rangle)$	$n_2 = 0$	1	2	3	4	5
$n_1 = 0$	29.6495	28.9132	28.1433	27.3726	26.564	25.7666
1	28.812	27.9492	27.0791	26.2057	25.2985	24.3877
2	27.9251	26.964	25.9916	25.0148	24.0073	22.9906
3	27.0115	25.9447	24.8719	23.7918	22.6844	21.5697
4	25.9856	24.8219	23.6537	22.4783	21.2794	20.0791
5	24.9774	23.7152	22.4498	21.1788	19.8864	18.5936

Finally, set the cost function to $f_4(\mathbf{n}) = 4n_1^2 + 4n_2^2 + 4n_3^2 - 8n_1n_2$. Let $x = (1, 0, 0)$ and $y = (0, 1, 0)$, so we have $f(x \vee y) = f(x \wedge y) = 0$, and $f(x) = f(y) = 4$. Based on the definition of supermodular function, $f_4(\mathbf{n})$ is not a supermodular function. The values of $v(\langle \mathbf{n}, D_3\rangle)$ are shown in Table 6. If $\triangle v_2(\langle \mathbf{n}, D_3\rangle)$ are less than $-R_2$ ($R_2 = 1.6$), the corresponding action for new call arrivals in cell-2 is to reject.

Table 6. RCS scheme: Function 4

$v(\langle \mathbf{n}, D_3\rangle)$	$n_2 = 0$	1	2	3	4	5
$n_1 = 0$	36.0884	35.2532	34.0446	32.4941	30.5935	28.3406
1	35.02	34.5633	33.715	32.5176	30.977	29.0896
2	33.6422	33.5472	33.0549	32.1976	31.0001	29.4636
3	31.9552	32.2213	32.079	31.5552	30.6671	29.4635
4	29.9821	30.5827	30.7694	30.5569	29.9566	29.0432
5	27.7088	28.6533	29.1793	29.2955	28.9977	28.3771

The actions for cell-2 new call arrivals of cost function 4 are shown in Table 7. In Table 7 '1' stands for the action to accept and '0' is for action reject. It is

Table 7. RCS scheme: for states $n_3 = 0$

$a(\langle \mathbf{n}, A_2 \rangle)$	$n_2 = 0$	1	2	3	4	5
$n_1 = 0$	1	1	1	0	0	0
1	1	1	1	1	0	0
2	1	1	1	1	0	0
3	1	1	1	1	0	0
4	1	1	1	1	0	0
5	1	1	1	1	0	0

seen that when there no calls in cell-1, the systems starts to reject the new call arrivals of cell-2 when there are 3 calls in cell-2, but it starts to reject the new call arrivals of cell-2 when there are 4 calls in cell-2 if there are more than 1 calls in cell-1, so the optimal policy is not a control limit policy.

Based on the four cost functions studied, for those states with no calls in cell-3, if the cost functions are supermodular functions, from Table 3 to Table 5 we can see that the values of $\triangle v_1(\langle \mathbf{n}, D_3 \rangle)$ and $\triangle v_2(\langle \mathbf{n}, D_3 \rangle)$ are nonincreasing in both $n_1 \downarrow$ and $n_2 \rightarrow$ directions, which fits our conclusion of the conjecture. Similar results can also be achieved for new calls and handoff calls to all other states. But if the cost function is not a supermodular function, as shown in Table 6, the resulting $v_3(\langle \mathbf{n}, D_3 \rangle)$ may not be a submodular function, consequently the optimail policy is not a control limit policy.

5 Conclusion

In this paper, we consider a wireless network with multiple cells, with routing probabilities between each cells. Assume that the arrival process of new calls in each cell is a Poisson process, the call connection time and cell residence time follow exponential distributions, accepting each new call would contribute some units of reward to the system, rejecting a handoff call would bring some cost to the system, and the system incurs a holding cost per unit time for the calls in the system. The result of this paper could easily be used in designing wireless networks for optimal performance, and be extended to the multiple classes of calls in the deign of future wireless mobile multimedia networking.

References

1. Bartolini, N., Chlamtac, I.: Call Admission Contro. In: Wireless Multimedia Networks. IEEE PIMRC (2002)
2. Li, W., Chao, X.: Call Admission Control for an Adaptive Heterogeneous Multimedia Mobile Network. IEEE Transactions On Wireless Communications 6(2) (February 2007)
3. Moretta, B., Zideins, I.: Admission controls for Erlang's loss system with service times distributed as a finite sum of exponential random variables. Applied Mathematics & Decision Sciences 2(2), 119–132 (1998)

4. Chao, X., Chen, H., Li, W.: Optimal Control for a Tandem Network of Queues with Blocking. ACTA Mathematicae Applicatae Sinica 13(4) (October 1997)
5. Puterman, M.L.: Markov Decision Process: Discrete Stochastic Dynamic Programming. Wiley-Interscience, Hoboken (2005) ISBN 0-471-72782-2
6. Lippman, S.: Applying a New Device in the Optimization of Exponential Queuing Systems. Operations Research 23, 687–710 (1975)
7. Kyung, Y., Shaler, S.: Optimal Service-Rate control of M/G/1 Queuing Systems Using Phase Methods. Applied Probability 15, 616–637 (1983)
8. Chen, H., Huang, L., Kumar, S., Kuo, C.J.: Radio Resource Management For Multimedia QoS Support In Wireless Networks, 2nd edn. John Wiley & Sons, Chichester (1983) ISBN 0-471-09942-2
9. Ho, C.J., Lea, C.T.: Improving call admission policies in wireless networks. Wireless Networks 5, 257–265 (1999)
10. Ni, W., Li, W., Alam, M.: Determination of Optimal Call Admission Control Policy in Wireless Networks. IEEE Transactions on Wireless Communications 8(2), 1038–1044 (2009)
11. Li, W., Fang, Y.: Performance Evaluation of Wireless Cellular Networks with Mixed Channel Holding Times. IEEE Transactions On Wireless Communications 7(6), 2154–2160 (2008)
12. Li, W., Fang, Y., Henry, R.: Actual Call Connection Time Characterization for Wireless Mobile Networks Under a General Channel Allocation Scheme. IEEE Transactions On Wireless Communications 1(4) (2002)

A Spectrum Sharing Scheme in Two Cellular Wireless Networks

Yuhong Zhang[1] and Wei Li[2]

[1] Department of Engineering Technology
[2] Department of Computer Science
Texas Southern University
3100 Cleburne Street, Houston, TX 77004, USA
{zhangya,liw}@tsu.edu

Abstract. In this paper, we propose and investigate a spectrum sharing scheme in two cellular wireless networks. This scheme specifies various considerations of how to share the licensed radio spectrum of one network with another, including how to rent out the spectrum to another network and how to withdraw the original spectrum from the renting network. For each network, we figure out explicit expressions for important system performance measures, which include blocking probability of new calls and handoff calls and system throughput. We also show how to adjust our spectrum sharing scheme to achieve a better result for each of above performance measures.

Keywords: Spectrum sharing scheme, radio spectrum rent, call admission control, blocking probability.

1 Introduction

FCC in its reports “Spectrum policy task force” [1] and “Notice of proposed rule making and order” [2] indicated respectively that most licensed spectra are underutilized. Since then, there are many papers studied various problems of spectra sharing between various wireless networks to efficiently increase the utilization of the radio spectra. For example, in [3] and [4] the authors studied threshold call admission control scheme in a cellular wireless network with the spectrum renting feature and gave optimal values of the admission thresholds. The paper [5] outlined the issues related to how to make the spectrum sharing approach more close to the reality. The paper [6] addressed spectrum regulators with ways to increase the utility of future unlicensed allocations by improving the sharing of such bands between diverse systems. A survey of dynamic spectrum access techniques is provided in [7]. There are some more papers studied problems related the spectrum sharing schemes such as [8] and [9].

The key of the radio spectrum sharing is that the idle radio spectrum can be rented by other wireless networks. As radio spectrum can be normally divided into radio channels by using multiple access methods such as TDMA and FDMA etc., the spectrum sharing concept can be further explained as that one network can borrow

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 439–450, 2010.

idle radio channels from another network, and the system renting out channels may also withdraw its radio channels when these channels are needed. That is, when mobile users suffer insufficient channels in one radio system, they may attempt to use idle channels in other radio systems.

If we restrict our consideration on an environment in which there are only two wireless networks with possible spectrum sharing features, there are two situations we should pay attention to. In the first situation, only one network may borrow a channel from another one. In order to decrease the blocking probability for handoff calls, we assume only the handoff calls have this priority. Without loss of generality, we assume that the handoff calls of the network-1 may borrow channels from network-2 but not vise verse. In the second situation, in which handoff calls in any networks may rent a channel from another. There is another situation when network-1 and network-2 are two independent systems, which means no one will rent channels from another. In this case, the performance of the network is determined completely by its own capacity and the call arriving rate and call resident time. There are many research papers in the literature for this independent situation, such as [10-15]. In our paper, this situation will only act as a comparison sample. We will derive analytic results for four cases and then make comparison among these total five cases based on performance measures.

The rest of this paper is organized as follows. The parameter description and theoretic analysis of two network system is given in section 2. The explicit expression for each of the performance measures is provided in section 3. Section 4 devoted to the numerical results. Finally, our conclusion is given in section 5.

2 Description and Analysis of the Two-Network System

We consider two network systems with spectrum renting feature and withdrawn procedure. When an arrived handoff call finds all home channels are being used, the network may set up a renting procedure to rent a channel from another network if there is a free channel available. In another side, when an owner network needs its rented out channels, it can active its withdrawn protocol too. To give a detailed description of the input parameters of this general model, we introduce the following assumptions:

- The new call and handoff call arrival process to network-*k* are Poisson process with a rate λ_k^N and λ_k^H (k =1, 2), respectively. Hence, the total arrived rate to network-*k* is $\lambda_k = \lambda_k^N + \lambda_k^H$ (k =1, 2).
- The lifetime of a new call or a handoff call with network-*k* is exponentially distributed with a rate h_k (k =1, 2).
- The cell residence time of a new call or a handoff call with network-*k* is exponentially distributed with a rate r_k (k = 1, 2).
- When all channels in the network-*k* are being occupied and there is at least one channel in another network available, an arrived network-*k* call may rent a channel from another network. This will active the renting procedure and

the call arrival rate to network-*k* may change. To include the no-rental or independent networks as a special case of our model, we will assume the total arrival process of network-*k* call, under the condition that there is at least one free channel from another network, is a Poisson process with a new rate λ_k^T $(k = 1, 2)$.

- If all channels from two networks are occupied and one of the networks is using some channels borrowed from another network, then the withdrawn procedure will be active if the owner network needs the rented channels. In this situation. The arrival process of the owner network calls will be adjusted depending on the withdrawn protocol. We use $\overline{\lambda_k}$ $(k = 1, 2)$ to represent the arrival rate of the adjusted Poisson process of owner arrived calls.

We assume that there are totally *M* channels for network-1 and *N* channels for network-2. The purpose of this section is to find the steady-state probability of the system when there are *m* calls from network-1 and *n* calls from network-2 in the status of connection, where $m = 0, 1, \cdots, M + N$ and $n = 1, 2, \cdots, N + M - m$. In order to reach this goal, we will introduce two stochastic processes. One is the number of calls from network-1 in the status of connection at time *t*, $I(t)$, and the other one is the number of calls from network-2 in the status of connection at time *t*, $J(t)$. Based on the description of the scheme proposed in the previous section, it is not hard to show that $\{(I(t), J(t)\})$ forms a two-dimensional Markov process with the state space

$$\Omega = \{(m, n) : m + n \le M + N\}$$

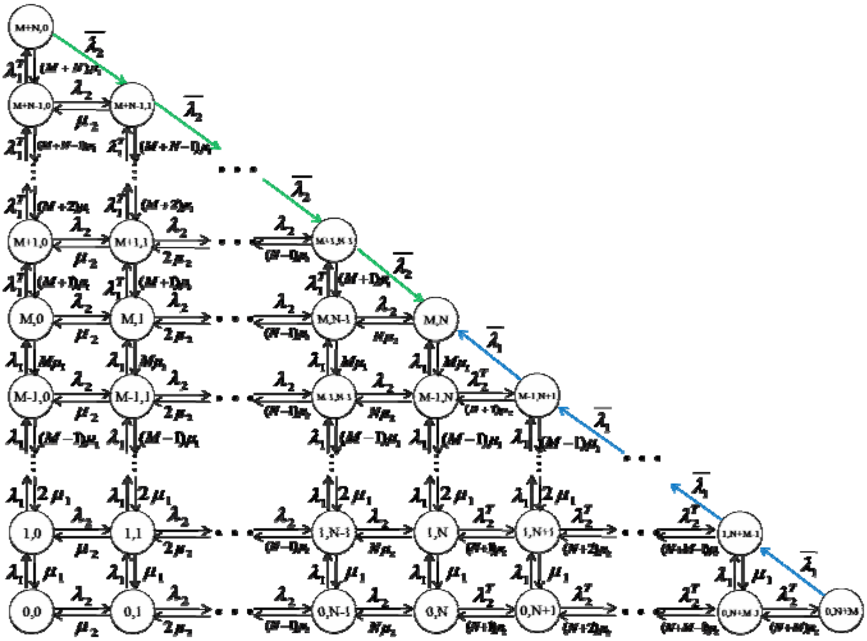


Fig. 1. Transition rate diagram

Based on the description of the system in Section 2, if we assume $\mu_k = h_k + r_k$ and $\lambda_k = \lambda_k^H + \lambda_k^N$ for $k = 1, 2,$ the transition rate diagram of the corresponding Markov process can be depicted as in Fig. 1.

We now take the step to find the steady state probability for this system. By using the transition rate diagram, the corresponding transition rate matrix Q of the general two dimensional Markov Process $\{(I(t), J(t))\}$ can be derived as follows

$$\boldsymbol{Q} = \begin{bmatrix} E_0 & A_0 & 0 & \cdots & 0 & \cdots & 0 & 0 \\ B_1 & E_1 & A_1 & \cdots & 0 & \cdots & 0 & 0 \\ 0 & B_2 & E_2 & \cdots & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & E_M & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & E_{M+N-1} & A_{M+N-1} \\ 0 & 0 & 0 & \cdots & 0 & \cdots & B_{M+N} & E_{M+N} \end{bmatrix}$$

where the matrices A_i, B_i and E_i are briefly explained in the following:

- Matrix A_i $(i = 0, 1, 2, \cdots, M + N - 1)$ refers to the event that an arrived network-1 call successfully receives the connection with the system when there are already *i* connections of network-1 calls in the system. The expression for A_i is provided as follows.
  - if $i = 0, 1, 2, \cdots, M - 1$, A_i is a matrix with size of $(M + N - i + 1) \times (M + N - i)$ given by

$$A_i = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 & 0 \\ 0 & \lambda_1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \lambda_1 & 0 \\ 0 & 0 & \cdots & 0 & \lambda_1 \\ 0 & 0 & \cdots & 0 & \overline{\lambda_1} \end{bmatrix}.$$

  - if $i = M, M + 1, M + 2, \cdots, M + N - 1,$ A_i is also a matrix with size of $(M + N - i + 1) \times (M + N - i)$ and has the expression

$$A_i = \begin{bmatrix} \lambda_1^T & 0 & \cdots & 0 & 0 \\ 0 & \lambda_1^T & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \lambda_1^T & 0 \\ 0 & 0 & \cdots & 0 & \lambda_1^T \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

- Matrix B_i ($i = 1, 2, \cdots, M + N$) refers to the event that a connected network-1 call departs from the system when there are i network-1 connected calls receiving the service. The expression for B_i is given as follows:
  - if $i = 1, 2, \cdots, M$, B_i is a matrix with size of $(M + N - i + 1) \times (M + N - i + 2)$ and is given by

$$B_i = \begin{bmatrix} i\mu_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & i\mu_1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & i\mu_1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 0 & i\mu_1 & 0 \end{bmatrix}.$$

  - if $i = M + 1, M + 2, \cdots, M + N$,

    B_i is a matrix with size of $(M + N - i + 1) \times (M + N - i + 2)$ and is given by

$$B_i = \begin{bmatrix} i\mu_1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & i\mu_1 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & i\mu_1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & i\mu_1 & \overline{\lambda_2} \end{bmatrix}$$

- Matrix E_i ($i = 0, 1, 2, \cdots, M + N$) refers to the event that there are no changes in the total numbers of network-1 connected calls in the system when there are i network-1 calls receiving the service. The expression for E_i is given by,
  - if $i = 1, 2, \cdots, M - 1$, E_i is a square matrix with size of $(M + N - i + 1)$ given by

$$E_i = -i\mu_1 I + \begin{bmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{bmatrix}.$$

    where,

$$E_{11} = \begin{bmatrix} -(\lambda_1+\lambda_2) & \lambda_2 & 0 & \cdots & 0 & 0 \\ \mu_2 & -(\lambda_1+\lambda_2+\mu_1) & \lambda_2 & \cdots & 0 & 0 \\ 0 & 2\mu_2 & -(\lambda_1+\lambda_2+2\mu_2) & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -[\lambda_1+\lambda_2+(N-2)\mu_2] & \lambda_2 \\ 0 & 0 & 0 & \cdots & (N-1)\mu_2 & -[\lambda_1+\lambda_2+(N-1)\mu_2] \end{bmatrix}.$$

$$E_{12} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \lambda_2 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}. \quad E_{21} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & N\mu_2 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

$$E_{22} = \begin{bmatrix} -[\lambda_1 + \lambda_2^T + N\mu_2] & \lambda_2^T & 0 & \cdots & 0 & 0 \\ (N+1)\mu_2 & -[\lambda_1 + \lambda_2^T + (N+1)\mu_2] & \lambda_2^T & \cdots & 0 & 0 \\ 0 & (N+2)\mu_2 & -[\lambda_1 + \lambda_2^T + (N+2)\mu_2] & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -[\lambda_1 + \lambda_2^T + (N+M-i-1)\mu_2] & \lambda_2^T \\ 0 & 0 & 0 & \cdots & (N+M-i)\mu_2 & -[\overline{\lambda}_1 + (N+M-i)\mu_2] \end{bmatrix}.$$

- if $i = M$, E_i is a square matrix with size of $(N+1)$ and is given by

$$E_M = -M\mu_1 I + \begin{bmatrix} -(\lambda_1^T + \lambda_2) & \lambda_2 & 0 & \cdots & 0 & 0 \\ \mu_2 & -(\lambda_1^T + \lambda_2 + \mu_2) & \lambda_2 & \cdots & 0 & 0 \\ 0 & 2\mu_2 & -(\lambda_1^T + \lambda_2 + 2\mu_2) & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -[\lambda_1^T + \lambda_2 + (N-1)\mu_2] & \lambda_2 \\ 0 & 0 & 0 & \cdots & N\mu_2 & -N\mu_2 \end{bmatrix}.$$

- if $i = M+1, M+2, \cdots, M+N-1$,
  E_i is a square matrix with size of $(M+N-i+1)$ given by

$$E_i = -i\mu_1 I + \begin{bmatrix} -(\lambda_1^T + \lambda_2) & \lambda_2 & 0 & \cdots & 0 & 0 \\ \mu_2 & -(\lambda_1^T + \lambda_2 + \mu_2) & \lambda_2 & \cdots & 0 & 0 \\ 0 & 2\mu_2 & -(\lambda_1^T + \lambda_2 + 2\mu_2) & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -[\lambda_1^T + \lambda_2 + (M+N-i+1)\mu_2] & \lambda_2 \\ 0 & 0 & 0 & \cdots & (M+N-i)\mu_2 & -[(M+N-i)\mu_2] - \overline{\lambda_2} \end{bmatrix}.$$

- if $i = M+N$, we have that $E_{N+M} = -[\overline{\lambda_2} + (M+N)\mu_1]$.

Based on the result of Lemma 3 in [16], we will have the following conclusion.

Theorem 1: The steady-state probability of the investigated system can be calculated by

$$\pi_i = \pi_0 \prod_{j=1}^{i} \left[A_{j-1} \left(-D_j \right)^{-1} \right], \text{ for } i = 1, 2, \cdots, M+N, \tag{1}$$

where $\prod_{j=1}^{K} a_j \equiv a_1 a_2 \cdots a_{K-1} a_K$ for any matrix $a_j,$ and $D_j \; (j = 0, 1, 2, \cdots, M + N)$ is recursively derived by $D_{N+M} = E_{N+M}$ and

$$D_j = E_j - A_j D_{j+1}^{-1} B_{j+1}, \quad \text{for } j = 0, 1, 2, \cdots, M + N - 1 \;, \tag{2}$$

and π_0 can be calculated by $\pi_0 D_0 = 0$ and $\pi_0 \left[\boldsymbol{e} + \sum_{i=1}^{M+N} \left(\prod_{j=1}^{i} A_{j-1} \left(- D_j^{-1} \right) \right) \boldsymbol{e} \right] = 1$,

where $\boldsymbol{e}$ is a column vector of suitable size with all its components equaling to one.

3 Performance Evaluations

Given the stationary probability distribution of the network as expressed in (1), many interesting performance measures of the system can be derived. In this section, the explicit expressions for the blocking probabilities and the system throughput are provided.

3.1 Blocking Probabilities

Call blocking probability is always important and has been considered to be a key measure of the quality of a network system. Let us first consider the new call blocking probability for network-1. When a new call arrives to the network-1, it will be blocked when all channels of network-1 are busy if we assume that the new call cannot borrow channels from other networks Therefore, by using the PASTA (Poisson Arrivals See Time Averages) rule [3], the new call blocking probability of network-1, defined as $P_1(B)$, can be expressed as

$$\begin{aligned} P_{1,N}(B) &= \left(1 - \frac{\overline{\lambda}_1}{\lambda_1} \right) \sum_{m=0}^{M-1} \pi_m \cdot e_{M+N-m} + \sum_{m=M}^{M+N} \pi_m \cdot e \\ &= \left(1 - \frac{\overline{\lambda}_1}{\lambda_1} \right) \sum_{m=0}^{M-1} \pi_{m,M+N-m} + \sum_{m=M}^{M+N} \sum_{n=0}^{M+N-m} \pi_{m,n} , \end{aligned}$$

where e_{M+N-m} is a $(M+N-m+1)$-dimensional column vector with 1 in the last element and zeroes for other elements. Similarly, the new call blocking probability for network-2 can be given by

$$P_{2,N}(B) = \left(1 - \frac{\overline{\lambda}_2}{\lambda_2} \right) \sum_{n=0}^{N-1} \pi_{M+N-n,n} + \sum_{n=N}^{M+N} \sum_{m=0}^{M+N-n} \pi_{m,n} \cdot$$

Next we will proceed to the formulae for the handoff blocking probability of network-1. When a handoff call arrives to the network-1, it will be blocked when the channels for both networks-1 and network-2 are occupied. Therefore, by using the PASTA rule again, the handoff call blocking probability of network-1, defined as $P_{1,H}(B)$, can be expressed as

- $\lambda_1^T = \lambda_1^H$

$$P_{1,H}(B) = \left(1 - \frac{\overline{\lambda_1}}{\lambda_1}\right) \sum_{m=0}^{M-1} \pi_m \cdot e_{M+N-m} + \sum_{m=M}^{M+N} \pi_m \cdot e_{M+N+1-m} .$$

$$= \left(1 - \frac{\overline{\lambda_1}}{\lambda_1}\right) \sum_{m=0}^{M-1} \pi_m \cdot e_{M+N-m} + \sum_{m=M}^{M+N} \pi_{m,M+N-m}$$

where $e_{M+N+1-n}$ is a (*M*+*N*+*1*-*m*)-dimensional column vector with 1 in the last element and zeroes for other elements.

- $\lambda_1^T = 0$

$$P_{1,H}(B) = \left(1 - \frac{\overline{\lambda_1}}{\lambda_1}\right) \sum_{m=0}^{M-1} \pi_m \cdot e_{M+N-m} + \sum_{n=0}^{N} \pi_{M,n} \cdot$$

Similarly, the handoff call blocking probability for network-2 can be obtained by

- $\lambda_2^T = \lambda_2^H$

$$P_{2,H}(B) = \left(1 - \frac{\overline{\lambda_2}}{\lambda_2}\right) \sum_{n=0}^{N-1} \pi_{M+N-n,n} + \sum_{n=N}^{M+N} \pi_{M+N-n,n} \cdot$$

- $\lambda_2^T = 0$

$$P_{2,H}(B) = \left(1 - \frac{\overline{\lambda_2}}{\lambda_2}\right) \sum_{n=0}^{N-1} \pi_{M+N-n,n} + \sum_{m=0}^{M} \pi_{m,N} \cdot$$

3.2 Throughput

We will consider the throughput of the network-*k* (*k* =1, 2) and the throughput of the whole system respectively. The throughput of network-*k*, denoted by TH_k, is defined as the long-run rate at which handoff call are processed through network-*k*. Since the handoff calls arrive at network-*k* according to a Poisson process with rate λ_k^H, we have

$$TH_1 = \lambda_1^H [1 - P_{1,H}(B)] = \lambda_1^H \left[1 - \sum_{m=0}^{M+N} \pi_{m,M+N-m}\right],$$

and

$$TH_2 = \lambda_2^H [1 - P_{2,H}(B)] = \lambda_2^H \left[1 - \sum_{n=0}^{M+N} \pi_{M+N-n,n}\right].$$

Because both network-1 and network-2 have the same handoff call blocking probability, the overall throughput from the all system, *TH*, can be obtained by,

$$TH = TH_1 + TH_2 = \lambda_1^H \left[1 - \sum_{m=0}^{M+N} \pi_{m,M+N-m}\right] + \lambda_2^H \left[1 - \sum_{n=0}^{M+N} \pi_{M+N-n,n}\right].$$

4 Numerical Analysis

To verify the validity of the analytical expressions obtained in the previous section and make a comparison for different situations, we have implemented the proposed model for five different cases:

Case 1. network-1 and network-2 are two independent networks, which means no one will borrow from others, i.e, in this case, $\lambda_k^T = 0$, and $\overline{\lambda_k} = 0$, $k = 1, 2$.

Case 2. The handoff calls of system-1 and system-2 can borrow from each other and occupy the borrowed channel until the call is finished, i.e, in this case, $\lambda_k^T = \lambda_k^H$, and $\overline{\lambda_k} = 0, k = 1, 2$.

Case 3. The handoff calls of system-1 and system-2 can borrow from each other, but need to return the borrowed channels if the owner needs the channels, i.e, in this case, $\lambda_k^T = \lambda_k^H$, and $\overline{\lambda_k} = \lambda_k, k = 1, 2$.

Case 4. Only the handoff call of one of the networks, for example, network-1, can borrow channels from another one and occupy the borrowed channels until the call is finished, i.e, in this case, $\lambda_1^T = \lambda_1^H, \lambda_2^T = 0$ and $\overline{\lambda_k} = 0$, $k = 1, 2$.

Case 5. Only the handoff call of one of the networks, for example, network-1, can borrow channels from another one, but need to return the borrowed channels if the owner needs them, i.e, in this case, $\lambda_1^T = \lambda_1^H, \lambda_2^T = 0$ and $\overline{\lambda_1} = \lambda_1, \overline{\lambda_2} = 0$, $k = 1, 2$.

The performance measures considered here are the new call and handoff call blocking probability, the whole system throughputs. The parameters for the network are assumed as follows:

1) The capability of both network-1 and network-2 is 60 channels, i.e., *M=N=60;*
2) The new call arrival rate λ_k^N of network-*k* is $\frac{4}{60}$, i.e., $\lambda_k^N = \frac{4}{60}, k = 1, 2$;
3) The handoff call arrival rate λ_k^H of network-*k* $(k = 1, 2)$ changes from 0 to $\frac{12}{60}$, i.e., $\lambda_k^H \in \left[0, \ \frac{12}{60}\right], k = 1, 2$
4) The average channel holding time of any call is 400 second, i.e., $\frac{1}{\mu_k} = 400, k = 1, 2$;

Fig. 2 illustrates the handoff call blocking probability of netwok-1 under varied traffic loads for five cases. The traffic load is measured in terms of traffic intensity unit, Erlang, which is equal to the number of calls originating in the mean holding time. It is obvious that the probabilities increase when the traffic load increases and any borrowing strategies have smaller handoff call broking probability than the case1: network-1 and network-2 are independent which means no one will borrow channels from another. Among those four kinds of borrowing strategies, the handoff calls in case 5 and case 3 have smaller blocking probability than that in case 2 and case 4, which means that returning the borrowed channel whenever the owner needs is a better strategy.

The comparison of new call blocking probability among five cases is presented in Fig. 3. It is obvious that the new call blocking probability for any cases with borrowing feature is bigger. This is reasonable. Based on our algorithm, any handoff call which is using the borrowed channel will switch to its own network channel

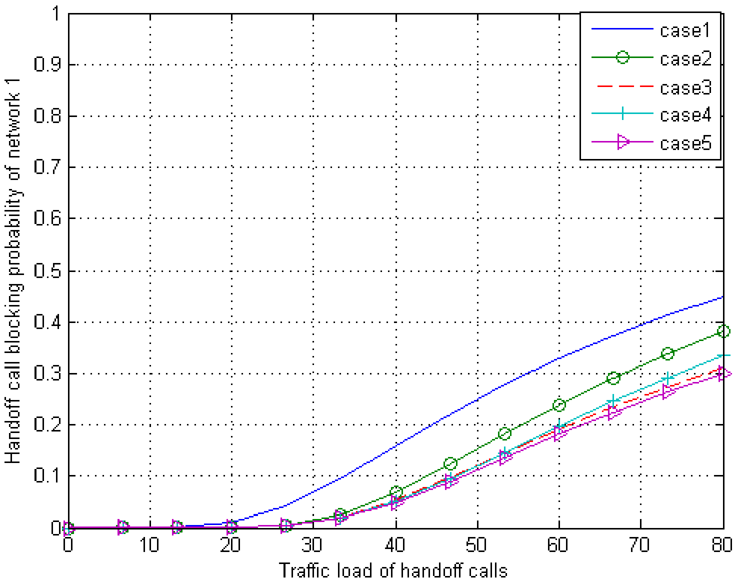


Fig. 2. Hand off call blocking probability of network-1 for five cases

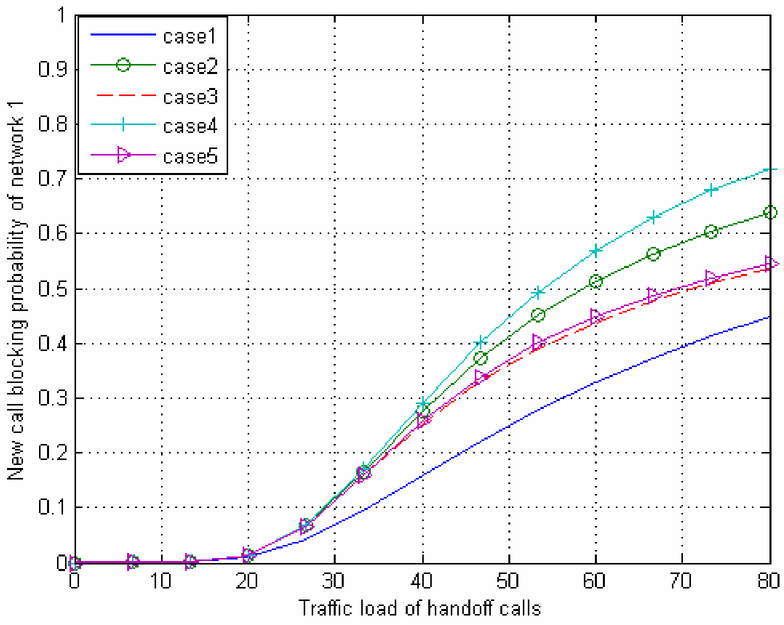


Fig. 3. New call blocking probability of network-1 for five cases

immediately as long as such a channel available. This approach actually gives a priority to handoff call and reduces the chance of new call getting connected in the same time. Among the "case 2" to the "case 5", the new call blocking probability in the "case 5" and the "case 3" show the lower value than that in the "case 2" and "case 4", which indicates the scheme that returning the borrowed channel in case the owner needs is also benefit to the new calls.

The throughputs of the whole system for five cases are shown in Fig. 4. The throughput is an important metric to judge a system. Fig. 4 shows that the "case 3" is the best choice in the view of whole system. The "case 3" means that the handoff call of both network-1 and network-2 can borrow the channel from the other and return the borrowed channel to the owner as long as the owner needs it.

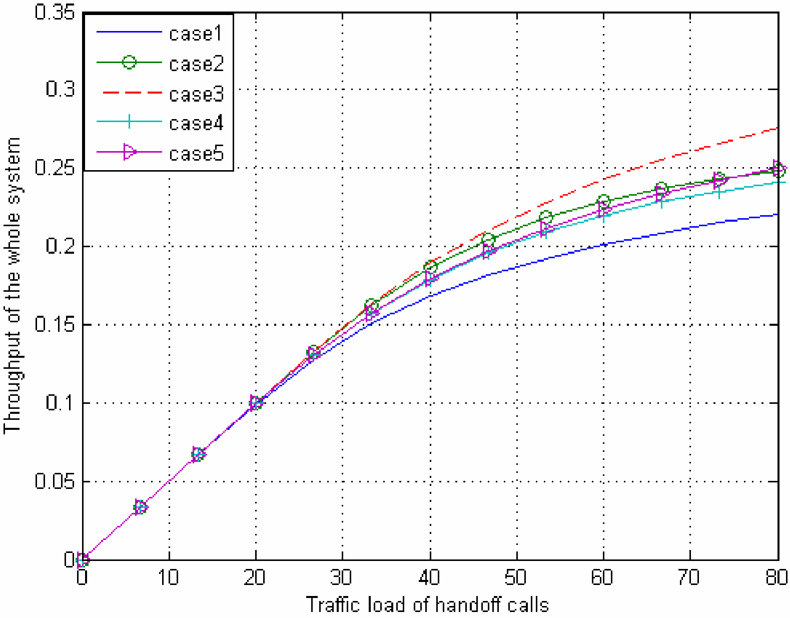


Fig. 4. Throughputs of the whole system for five cases

5 Conclusions

A novel two-network system with spectrum renting feature in the same area has been investigated in this paper. In order to better understand our scheme for the renting strategy, we developed five specific models: 1) network-1 and network-2 are two independent networks, which means no one will rent from another; 2) The handoff calls of network-1 and network-2 can rent from each other and occupy the borrowed channel until the call is finished; 3) The handoff calls of network-1 and network-2 can borrow from another, but need to return the borrowed channel if the owner needs the channel; 4) Only the handoff call of one of the network, for example, network-1, can borrow channels from another one and occupy the borrowed channel until the call is finished; 5) Only the handoff call of one of the networks, for example, network-1, can borrow channels from another one, but needs to return the borrowed channel if the owner needs it. For these five cases, analytical formulae of some important performance measures, such as new call and handoff call blocking probability, the throughput and utilization of the network are derived. The numerical implementation indicated that the proposed mathematical model is very accurate and the corresponding theoretic results are consistent with the simulation results. In addition, we made a comparison study among different situations. This comparison study shows that with the renting feature, the whole system can achieve higher throughput than that one with two independent networks; the handoff calls will always obtain high priority to new calls; in "case 3", the system achieves the best performance in a view of the whole system.

References

1. F.C. Commission: Spectrum policy task force. Rep. ET Docket, pp.02–135 (2002)
2. F.C. Commission: Notice of proposed rule making and order. Rep. ET Docket, pp. 03–222 (2003)
3. Tzeng, S.S.: Call admission control policies in cellular wireless networks with spectrum renting. Computer Communications (2009)
4. Tzeng, S.S., Huang, C.W.: Threshold based call admission control for qos provisioning in cellular wireless networks with spectrum renting. In: CISSE (2008)
5. Tonmukayakul, A., Weis, M.B.H.: Secondary use of radio spectrum: A feasibility analysis (2004)
6. De Vries, P., Hassan, A.: Spectrum sharing rules for new unlicensed bands. Technical report, Microsoft Corporation (2003)
7. Zhao, Q., Swami, A.: A survey of dynamic spectrum access: signal processing and networking perspectives. In: ICASSP 2007, Honolulu, Hawaii, vol. 4 (2007)
8. Mangold, S., Challapali, K.: Coexistence of wireless networks in unlicensed frequency bands. vol. 0, Zurich, Switzerland (2003)
9. Raychaudhuri, D., Jeng, X.: A spectrum etiquette protocol for efficient coordination of radio devices in unlicensed bands. Beijing, China (2003)
10. Alfa, A.S., Li, W.: PCS networks with correlated arrival process and retrial phenomenon. IEEE Trans. Wireless Commun. 1, 630–637 (2002)
11. Fang, Y.: Thinning scheme for call admission control in wireless networks. IEEE Trans. on Comput. 52, 685–687 (2003); changing environments. Advanced in Applied Probability 16, 715–731(1984)
12. Li, W., Fang, Y.: Performance evaluation of wireless cellular networks with mixed channel holding times. IEEE Transactions on Wireless Communications 7, 2154–2160 (2008)
13. Ni, W., Li, W., Alam, M.: Determination of optimal call admission control policy in wireless networks. IEEE Transactions on Wireless Communications 8, 1038–1044 (2009)
14. Zhang, Y., Salari, E.: Utilisation analysis and comparison for multimedia wireless networks. International Journal Ad Hoc and Ubiquitous Computing 3, 185–190 (2008)
15. Zhang, Y., Salari, E.: A hybrid channel allocation algorithm with priority to handoff calls in mobile cellular networks. Computer Communications 32, 880–887 (2009)
16. Gaver, D.P., Jacobs, P.A., Labouche, G.: Finite birth and death models in randomly

A Perspective on Estimation of Available Capacity in Wireless Networks

H.S. Ramesh Babu[1], Gowrishankar[2], and P.S. Satyanarayana[3]

[1] Department of Information Science and Engineering,
Acharya Institute of Technology,
Bangalore – 560090, Karnataka, India
[2] Department of Computer Science and Engineering,
B.M.S. College of Engineering,
Bangalore – 560019, Karnataka, India
[3] Department of Electronics and Communication Engineering,
B.M.S. College of Engineering,
Bangalore – 560019, Karnataka, India
rameshbabu@acharya.ac.in,
{gowrishankar.cse,pssvittala.ece}@bmsce.ac.in

Abstract. To understand the characteristics of the wireless networks, the network usage data from wireless measurement tools are essential. The data collection is a process of collecting the network time-varying information in standardized formats and from standard interfaces. The characteristics of the wireless networks include, signal propagation, received signal quality, network traffic, active applications and mobility of the MT. The purpose of the measurement is to collect vital data of the wireless network. There are several tools available for this purpose. The most widely used network measurement tools are client side measurement tool, Syslog, Simple Network Management protocol(SNMP), network sniffing, wireless sniffing. This paper discusses the different wireless measurement tools like Syslog, Simple Network Management protocol, network sniffing, wireless sniffing and their benefits and limitations.

Keywords: Wireless networks, Syslog, Simple Network Management protocol, network sniffing, wireless sniffing**.**

1 Introduction

The data collection is a process of collecting the network time-varying information in standardized formats and from standard interfaces. This needs a Portable tool for data collection. The collected data need to be processed effectively without losing the "tail" of the data and identifying holes and cleaning data. In the pre-processing mechanism, the time-varying network parameters are arranged in an order. These time series may have few missing entries, due to the minor flaws in the measurement tools, which are estimated and filled using time series techniques.

There are many implicit differences in wired and wireless medium. Wired medium will have clear points of connection but wireless medium is physically dispersed. The

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 451–461, 2010.

mobility in wireless networks and novel devices used inspires new usage patterns. In this prevailing scenario, the measurement of wireless network information is essential. This strengthens our understanding of user and network behaviours. The better understanding leads to better network models. The improved network models are momentous to improvement in terms of network protocols, distributed algorithms, applications and improved deployment strategy.

The NGWN provides users with a wide range of services across HWNs coexisting with diverse throughput and coverage with a single MT. The existing cellular networks will provide communication services over a wide geographical area but has limited bandwidth to support emerging data services. But the future 3G cellular and 4G systems, such as UMTS, Wi-Max (802.16), have lesser coverage and higher bandwidth when compared to cellular networks. The WLAN (IEEE 802.11a/b/g/n) is able to provide higher data rate but with lesser coverage compared to cellular and 4G systems. Therefore an integration of cellular networks, Wireless Local Area networks (WLAN) and Wi-MAX would result in higher bandwidth, more network coverage and will also help in enhanced user mobility and with choice of new services and enhanced QoS [1]. The Speed v/s Mobility comparison for wireless networks is represented in Figure1. The characteristics of the different wireless networks are depicted in table 1.

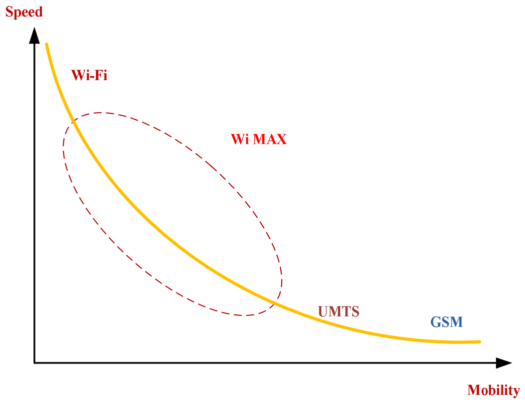


Fig. 1. Speed v/s Mobility comparisons of different wireless networks

The process of network switching will involve the following three phases – network discovery, switching decision and execution [2]. The decision phase will play an important role in balancing network utilization, fulfilling the user requirements and QoS requirements of network applications. Thus, the need of effective decision mechanism is crucial. The decision mechanism is driven by a set of QoS parameters [3-6]. The QoS parameters are bandwidth, BER and cost. The criteria that affect these QoS parameters are wireless link quality and the current network load. The factors that influence link quality are noise and signal fading [7]. The Signal to Noise Ratio (SNR) value of the wireless channel can be considered as the measure of the channel quality in a wireless network. The network load is measured based on the number of active users and their network sessions and is also called as network traffic [8].

The signal fading in a wireless system is common phenomena of the radio channel. They are classified into two types, *Flat fading* and *Frequency selective fading*. In a narrowband wireless channel, the consistency bandwidth of the channel is larger than the bandwidth of the signal. In such channels all frequency components of the signal will experience the same amount of fading. Such a fading is called as *'Flat fading'*. On the other hand, in a wideband wireless channel the coherence bandwidth of the channel is smaller than the bandwidth of the signal. This result in Different frequency components of the signal, experiencing the different amount of fading called as *'frequency selective fading'*. Apart from these two types of fading, when the MT is moving at a high speed, the signal strength varies severely and undergoes deep fading within the small time frame. This type of fading is named as 'Fast fading' [9].

The next generation wireless systems typically have higher bandwidth and support optimal mobility, need to challenge with the frequency selective fading and fast fading. The next generation wireless systems make use of low complexity techniques such as Orthogonal Frequency Division Multiplexing (OFDM) in the physical layer and Orthogonal Frequency Division Multiple Access (OFDMA) mechanisms in the link layer to prevail over the effect of frequency selective fading [10].

2 Wireless Network Measurements

To understand the characteristics of wireless networks, the network usage data from wireless measurement tools are essential. The characteristics include signal propagation, received signal quality, network traffic, active applications and mobility of the MT. The purpose of the measurement is to collect vital data of the wireless network. There are several tools available for this purpose. The most widely used network measurement tools are client side measurement tool, Syslog, Simple Network Management protocol (SNMP), network sniffing, wireless sniffing.

2.1 Client Side Network Management Tools

The wireless measurement tools mentioned above i.e. Syslog, SNMP, and Network sniffing and wireless sniffing tools are intended to monitor the network from the viewpoint of the network. In client side methods the measurement tools are installed in client to measure the activities at the client side. This client side measurement has many advantages.

A client side tool can accurately determine what exactly a client is doing. While Syslog will provide information about set of clients which are associated to the particular AP/BS, a client side tool can list all the APs/BSs that a client can handle, which are useful for mobility tracing. A client side tool can list all the applications that are running on it, rather than just those applications that generate network traffic. Client side tools are extensively used in WMAN and WWAN measurements [11] [12].

Writing a generic client side program, such as *tcpdump, Wireshark* formerly called *Ethereal* and *kismet*, will be a challenging task because it has to run on varieties of operating systems and different device drivers.

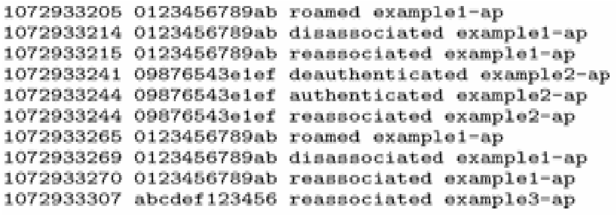
Table 1. Attribute comparisons of Different Wireless Networks

Wireless Network	*Bandwidth (Mbps)*	*Modulation Technique*	*Freq (GHz)*	*Coverage*	
				Indoor Coverage	*Outdoor Coverage*
IEEE802.11a	20	OFDM	5	35 Meters	120 Meters
IEEE802.11b	11	DSSS	2.4	38 Meters	140 Meters
IEEE802.11g	54	OFDM/ DSSS	2.4	38 Meters	140 Meters
IEEE802.11n	600	OFDM	5	70 Meters	250 Meters
HiperLAN2	54	OFDM	5	50 Meters	50 Meters
802.16e	Up to 125	OFDMA	2-6	Up to 35000 Meters (35Kms)	
802.16m	Up to 300	OFDM	Upto6	Up to 50000 Meters (50 Kms)	
EDGE Evolution	9.6- 384	TDMA/ FDD	900/ 1800/1 900 MHz	Up to 40000 Meters (40kms)	
UMTS W-CDMA	2	FDD, TDD	2	Up to 20000 Meters (20kms)	

2.2 Syslog

Syslog records detail steps of association, and have been used effectively for studying user activity patterns [13] [14]. To all intents and purposes Syslog is a standard for sending and receiving of log messages [15]. The wireless APs and BSs can be configured to log appropriate events in the network. The Syslog messages are used to understand the state of an MT in the wireless network. The AP or BS can generate a time stamped message whenever an MT *authenticates, de-authenticates, associates, disassociates or roams* to that AP or BS. By collecting these messages it is possible to determine the state of the MTs on the network. The Syslog messages are stored and analyzed locally in the BS or transmitted across the network for storage and analysis by a dedicated computer.

There is no standard format for Syslog messages. The messages that APs or BSs send can vary in format and amount of information contained. In most of the cases APs and BSs manufactured from same manufacturer will have different Syslog message formats. In certain cases the message formats differ for each version of the same product. In a heterogeneous wireless environment, multiple type of APs and BSs with varieties of Syslog message formats. It is necessary to translate these messages in to an intermediate format prior to the data analysis. In some of the measurement studies [16] [17], the multiple Syslog message formats are translated to general, intermediate parsed format for the purpose of analysis. Figure 2 indicates the parsed Syslog trace data format.



```
1072933205 0123456789ab roamed example1-ap
1072933214 0123456789ab disassociated example1-ap
1072933215 0123456789ab reassociated example1-ap
1072933241 09876543e1ef deauthenticated example2-ap
1072933244 09876543e1ef authenticated example2-ap
1072933244 09876543e1ef reassociated example2-ap
1072933265 0123456789ab roamed example1-ap
1072933269 0123456789ab disassociated example1-ap
1072933270 0123456789ab reassociated example1-ap
1072933307 abcdef123456 reassociated example3-ap
```

Fig. 2. Parsed Syslog Format

2.3 SNMP

The SNMP is a generic tool in measuring and managing a network device, called *'network object'* in the network management terminology [18]. The SNMP provides information on both traffic volume and the number of active users. This makes the SNMP the most suitable technique used for both traffic studies [14] [19] [20] and user mobility studies [21].

A network administrator runs a tool known as *'manager'*, which communicates with SNMP *'agents'*. Agents run on network objects and provide interface between the object and manager. A network object can contain several objects, such as statistics or configuration items, arranged in a database known as *Management Information Base (MIB).* The network statistics are stored in the MIB variables and these variables are represented in a standard format known as Abstract Syntax Notation (ASN) .The manager queries the agent for the purpose of measurement and agent replies by extracting information from the MIB variables. Both request and reply will be in the standard SNMP message format [22]. In the recent version of SNMP few MIB variables, like MAC address, IP address, Signal strength, Power saving mode, Network session length and Traffic of the MT associated with AP or BS, are specific to the wireless network [23].

Some of the advantages of the SNMP are

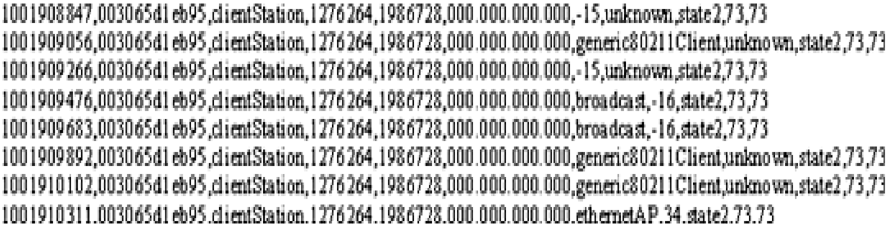
- SNMP messages provide more detailed information about the status of the network than Syslog messages.
- SNMP provides information on both traffic volume and the number of active users. Hence it is suitable to be used for both traffic studies and user mobility studies.
- SNMP messages are generally device independent and are usually available in a standard format.

The drawbacks of SNMP are

- SNMP-based approaches is that they require an interval between SNMP polls (typically every 1–5 minutes), and it has been shown that long poll intervals may miss wireless clients that associate with APs for less than this poll interval [24].
- The SNMP-based approaches may be able to retrieve such detailed wireless MAC/PHY information through the use of a properly defined MIB, the most existing SNMP MIBs for APs (MIB-I (RFC 1066), MIB-II (RFC 1213), and 802.11 MIB (IEEE Std 802.11-1999)) provide very limited visibility into MAC-level behaviour.

2.4 Network Sniffing

The network or packet *sniffing* refers to the process of capturing of the network traffic at the network interface. For the purpose of sniffing, the network interface should be in a promiscuous mode. In this mode the interface will ignore its assigned address and captures all the frames/packets present in the network. There are programs, such as *tcpdump*, *Ethereal* and *kismet,* which will capture and analyze the frame/packet [25] [26] [27].



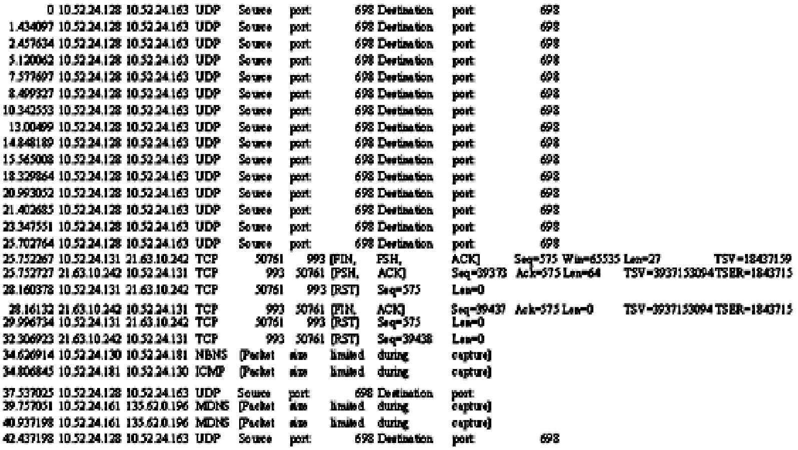
```
1001908847,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,-15,unknown,state2,73,73
1001909056,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,generic80211Client,unknown,state2,73,73
1001909266,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,-15,unknown,state2,73,73
1001909476,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,broadcast,-16,state2,73,73
1001909683,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,broadcast,-16,state2,73,73
1001909892,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,generic80211Client,unknown,state2,73,73
1001910102,003065d1eb95,clientStation,1276264,1986728,000.000.000.000,generic80211Client,unknown,state2,73,73
1001910311.003065d1eb95.clientStation.1276264.1986728.000.000.000.000.ethernetAP.34.state2.73.73
```

Fig. 3. Set of SNMP Messages

Kismet is an 802.11 layer2 wireless network detector, sniffer, and intrusion detection system. Kismet will work with any wireless card which supports raw monitoring (rfmon) mode, and (with appropriate hardware) can sniff 802.11b, 802.11a, 802.11g, and 802.11n traffic. *Kismet* is good for WLAN surveillance. It is capable to sense the details of all wireless access points (WAPs) and WLAN nodes, showing channels, use of encryption and signal strength.

Ethereal is a network packet analyzer. A network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible. You could think of a network packet analyzer as a measuring device used to examine what's going on inside a network cable. The Ethereal is not an intrusion detection system. It will not warn when someone does strange things on the network that he/she isn't allowed to do. However, if strange things happen, Ethereal might help you figure out what is really going on. Ethereal will not manipulate things on the network, it will only "measure" things from it. Ethereal doesn't send packets on the network or do other active things (except for name resolutions, but even that can be disabled). The trace of an ethereal is shown in figure 4.

The important concern with network sniffing is that the volume of data generated from the sniffing process is much larger than Syslog and SNMP. A typical sniffing of 802.11b wireless network operating at 11 Mbps speed can generate several gigabits of data within few minutes. It is vital to ensure that sufficient disk space is available to store the captured frames/packets in the hard disk. Another major concern in the network sniffing is the privacy of captured information.



```
        0 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 1.434097 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 2.457634 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 5.120062 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 7.577697 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 8.499327 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
10.342553 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
 13.00499 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
14.848189 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
15.565008 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
18.329864 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
20.993052 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
21.402685 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
23.347551 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
25.702764 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
25.752267 10.52.24.131 21.63.10.242 TCP      50761   993 [FIN,   PSH,       ACK]      Seq=575  Win=65535 Len=27          TSV=18437159
25.752727 21.63.10.242 10.52.24.131 TCP        993 50761 [PSH,   ACK]       Seq=39373 Ack=575 Len=64    TSV=3937153094 TSER=1843715
28.160378 10.52.24.131 21.63.10.242 TCP      50761   993 [RST]   Seq=575    Len=0

 28.16132 21.63.10.242 10.52.24.131 TCP        993 50761 [FIN,   ACK]       Seq=39437 Ack=575 Len=0     TSV=3937153094 TSER=1843715
29.996734 10.52.24.131 21.63.10.242 TCP      50761   993 [RST]   Seq=575    Len=0
32.306923 21.63.10.242 10.52.24.131 TCP        993 50761 [RST]   Seq=39438  Len=0
34.626914 10.52.24.130 10.52.24.181 NBNS [Packet size    limited during     capture]
34.806845 10.52.24.181 10.52.24.130 ICMP [Packet size    limited during     capture]

37.537025 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port:
39.757051 10.52.24.161 135.62.0.196 MDNS [Packet size    limited during     capture]
40.937198 10.52.24.161 135.62.0.196 MDNS [Packet size    limited during     capture]
42.437198 10.52.24.128 10.52.24.163 UDP  Source  port:     698 Destination  port      698
```

Fig. 4. Network Sniffing Trace

The frame/packet that is captured through sniffing may contain sensitive data especially when the data within the frame/packet is not encrypted. The issue of privacy may be alleviated by only capturing the header data, which may be sufficient for a network measurement. Even with this, the privacy problem is not completely overcome as some vital information, such as packet size, MAC/IP address, higher layer protocol and inter-arrival time, stand exposed. The result of such a sniffing is referred to as a trace.

2.5 Wireless Sniffing

The wireless sniffing is a WLAN measurement tool [28]. Syslog, SNMP and network sniffing are the generic measurement tools which will be used in measuring all types of wireless as well as wired networks. The wireless sniffing is a measurement tool useful only for a wireless network. It will operate at AP/BS or at a switch that connects wireless network to the wired backbone. The disadvantage of wire side measurement is that not all wireless data observable from the wired network, such as management frames, beacons, retransmissions and collisions, send traffic via wired network. The wireless sniffer is widely used to collect the MAC level frame information in a wireless network. Even though wireless sniffer can be installed on a host under measurement, but in majority of cases, it is installed on an autonomous device. This independent device could be a laptop or any MT or a PDA system. This makes the wireless sniffer to monitor the wireless network in promiscuous mode without interfering with the stations under study/monitoring. Wireless sniffers capture both the data frames as well as management frames. The management frames captured by wireless sniffer includes beacon frames, request to send (RTS) frames, clear to send (CTS) frames and Acknowledgement (ACK) frames. Nevertheless, there is need of special hardware and software in form of drivers is essential for effective working of a wireless sniffer. *Ethereal* and *Kismet* are the most admired wireless sniffer and analyzer software. There are good amount of research works reported on wireless performance using Wireless sniffers .The measurement of streaming media over wireless link using independent sniffers [29][30], measurement of congestion in wireless LAN [31],the network monitor research in [32],a complete wireless sniffer system is implemented and used to characterize a typical computer science department WLAN traffic.

Wireless measurement can be applied to the mobile host. This is accomplished by placing wireless network interface card in a *monitor* mode. In this mode, the wireless card captures all types of frames/packets. These frames/packets may be analyzed similar to those of network sniffing. Since this mode is not a promiscuous mode it limits the wireless sniffer in the mobile host as a simple network monitoring tool. Figure 6 shows an example of wireless sniffing trace.

The advantages and disadvantages of wireless sniffing are as listed below

Advantages of wireless sniffing are:

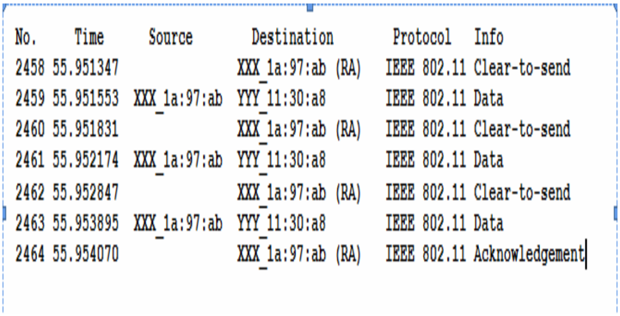
- Wireless Sniffing done be an independent sniffer in a promiscuous mode will not cause any interference with the hosts under test in wireless experiment. Therefore, sniffing can be used to measure these devices, such as the wireless game consoles, which do not provide general accesses for measurement purpose.
- Wireless sniffing can provide frame level information and wireless network conditions, such as the RSSI and sending capacity.
- Wireless sniffers can be used as wireless network diagnostic tools as they are capable to capture wireless management frames, such as RTS, CTS, Authentication/De-authentication frames, and Association/Disassociation frames.

Disadvantages of Wireless sniffers are:

- Wireless sniffers cannot record all the frames that are transmitted over the network [31] [33] since the sniffer is only capturing the frames at its own location this results in non-capturing of the packets lost due to a hidden terminal and packets lost due bit errors.
- The Received Signal Strength Indicator (RSSI) is measured relative to the wireless sniffer installation location. This measurement of received signal strength may not be same as the AP or the clients that are remote from the wireless sniffer installation location.
- The location of the sniffer plays an important role in the wireless sniffing. For example, a location very close to an AP is helpful when studying the AP behaviour, but may miss some traffic sent from a distant client due to signal attenuation and on the other hand the similar effect is experienced when the sniffer is near to the client and away from the AP. This results in ‘Generic losses.

- The wireless sniffing suffers from ‘AP losses due to the firmware incompatibility between AP and monitoring device. These losses can be minimized by using redundant sniffers or sniffers with interface cards having different chipset and using antennas of different gains and positioning the sniffers at strategic places [34].

The sample of wireless sniff trace is shown in figure 5.



```
No.     Time       Source        Destination        Protocol  Info
2458 55.951347                XXX_1a:97:ab (RA)  IEEE 802.11 Clear-to-send
2459 55.951553 XXX_1a:97:ab  YYY_11:30:a8        IEEE 802.11 Data
2460 55.951831                XXX_1a:97:ab (RA)  IEEE 802.11 Clear-to-send
2461 55.952174 XXX_1a:97:ab  YYY_11:30:a8        IEEE 802.11 Data
2462 55.952847                XXX_1a:97:ab (RA)  IEEE 802.11 Clear-to-send
2463 55.953895 XXX_1a:97:ab  YYY_11:30:a8        IEEE 802.11 Data
2464 55.954070                XXX_1a:97:ab (RA)  IEEE 802.11 Acknowledgement
```

Fig. 5. Wireless Sniffing Trace in WLAN

3 Conclusion

The wireless Measurement is an important stage of any study on wireless networks. The data collection phase acts as the building stone of the study of wireless measurements. The various wireless measurements tools used have their own strength and weaknesses. The wireless sniffing is one of the measurement techniques that could be used for effective measurement of wireless network time varying characteristics. The data collection of wireless networks can be supported by standardization of interfaces and format, information from network vendors and archival of the network data. Our future works includes the building up the effective measurement framework and step ahead for predicting the missing values in measurements by applying intelligent techniques.

References

[1] Kuran, M.S., Tugcu, T.: A Survey on Emerging Broadband Wireless Access Technologies. Computer Networks 51(11), 3013–3046 (2007)

[2] Siddiqui, F., Zeadally, S.: Mobility Management across Hybrid Wireless Networks: Trends and Challenges. Computer Communications 29(9), 1363–1385 (2006)

[3] Chen, W., Shu, Y.: Active Application Oriented Vertical Handoff in Next-generation Wireless Networks. IEEE Wireless Communication and Networking Conference 3, 1383–1388 (2005)

[4] Al-Gizawi, T., Peppas, K., Axiotis, D., et al.: Interoperability Criteria, Mechanisms and Evaluation of System Performance for Transparently Interoperating WLAN and UCLIENTS-HSDPA Networks. IEEE Networks 19(1), 66–72 (2005)

[5] Song, Q., Jamalipour, A.: Network Selection in an Integrated Wireless LAN and UCLIENTS Environment using Mathematical Modeling and Computing Techniques. IEEE Wireless Communication Magazine 12(3), 42–48 (2005)

[6] Zhu, F., McNair, J.: Optimization for Vertical Handoff Decision Algorithms. In: IEEE Wireless Communication and Networking Conference, vol. 2, pp. 867–872 (2004)

[7] Zhang, J., Cheng, L., Marsik, I.: Models for Non-intrusive Estimation of Wireless Channel Bandwidth. In: 9th IFIP International Conference on Personal Wireless Communication Conference, pp. 334–348 (2003)

[8] Papadopouli, M., Shen, H., Raftopoulos, E., et al.: Short-term Traffic Forecasting in Campus-wide Wireless Networks. In: 16th IEEE International Symposium on Personal, Indoor and Mobile Wireless Communications, pp. 1446–1452 (2005)

[9] Pahlvan, K., Krishanamurthy, P.: Principles of Wireless Networks - A Unified Approach. Prentice-Hall, Inc., Englewood Cliffs (2002)

[10] Prasad, R.: OFDM for Wireless Communication Systems. Artech House Inc., Boston (2004)

[11] Tang, D., Barker, M.: Analysis of a Metropolitan-Area Wireless Network. Wireless Networks 8, 107–120 (2002)

[12] Claypool, M., Kinicki, R., Lee, W., Li, M., Ratner, G.: Characterization by Measurement of a CDMA 1xEVDO Network. In: 2nd International Workshop on Wireless Internet, p. 2-es (2006)

[13] Chinchilla, F., Lindsey, M., Papadopouli, M.: Analysis of Wireless Information Locality and Association Patterns in a Campus. In: Proceedings of INFOCOM 2004, Hong Kong, China (March 2004)

[14] Kotz, D., Essien, K.: Analysis of a Campus-wide Wireless Network. In: Proceedings of MOBICOM 2002, Atlanta, GA (September 2002)

[15] Lonvik, C.: The BSD Syslog Protocol., IETF RFC 3164 (August 2001)

[16] Henderson, T., Kotz, D., Abyzov, I.: The Changing Usage of Mature Campus-wide Wireless Network. In: 10th ACM International Conference on Mobile Computing and Networking, pp. 187–201 (2004)

[17] Kotz, D., Essien, K.: Analysis of a Campus-wide Wireless Network. Wireless Networks 11(1-2), 115–133 (2005)

[18] Mc Cloghire, K., Perkins, D., Schoenwaelder, J.: Structure of Management Information Version 2 (SMIv2). IETF RFC 2578 (April 1999)

[19] Balachandran, G.M., Voelker, P.B., Rangan, V.: Characterizing User Behavior and Network Performance in a Public Wireless LAN. In: Proceedings of ACM SIGMETRICS 2002, Marina Del Rey, CA (June 2002)

[20] Tang, D., Baker, M.: Analysis of a Local-Area Wireless Network. In: Proceedings of MOBICOM 2000, Boston, MA (August 2000)

[21] Balazinska, M., Castro, P.: Characterizing Mobility and Network Usage in a Corporate Wireless Local-Area Network. In: Proceedings of MOBISYS 2003, San Francisco, CA (May 2003)

[22] Subramanian, M.: Network Management: Principles and Practice. Addison-Wesley, Reading (2000)

[23] Flick, J., Jhonson, J.: Definitions of Managed Objects for Ethernet-like Interface Types. IETF RFC 2665 (August. 1999)

[24] Subramanian, M.: Network management. PearsonEducation, London

[25] Ethereal Protocol Analyzer, http://www.ethrereal.com

[26] Kismet Wireless Sniffing Software, http://www.Kismetwireless.net

[27] Tcpdump packets capture software, http://www.tcpdump.org

[28] Shenoy, R., Ananda, A.L., Chan, M.C., Ooi, W.T. (eds.): Mobile, Wireless and Sensor Networks: Technology, Application and Future Directions. John Wiley & Sons, Chichester (2006)

[29] Kuang, T., Williamson, C.: RealMedia Streaming Performance on an IEEE 802.11b Wireless LAN. In: Proceedings of IASTED Wireless and Optical Communications (WOC), pp. 306–311 (July 2002)

[30] Bai, G., Williamson, C.: The Effects of Mobility on Wireless Media Streaming Performance. In: Proceedings of Wireless Networks and Emerging Technologies (WNET), pp. 596–601 (July 2004)

[31] Jardosh, A.P., Ramachandran, K.N., Almeroth, K.C., Belding-Royer, E.M.: Understanding Congestion in IEEE 802.11b Wireless Networks. In: Proceedings of the Internet Measurement Conference (IMC), Berkeley, CA, USA (October 2005)

[32] Yeo, J., Youssef, M., Agrawala, A.: A framework for wireless lan monitoring and its applications. In: ACM Workshop on Wireless Security (WiSe 2004) in conjunction with ACM MobiCom 2004, Philadelphia, PA, USA (October 2004)

[33] Claypool, M.: On the 802.11 turbulence of nintendo ds and sonypsp hand-held network games. In: Proceedings of the 4th ACM Network and System Support for Games (NetGames), Hawthorne, NY, USA (October 2005)

[34] Yeo, J., Banarjee, S., Agarwaala, A.: Measuring Traffic on the Wireless Medium: Experience and pitfalls., Technical Reports, CS-TR-4421, Department of Computer Science, University of Maryland (December 2002)

Mobile Healthcare Infrastructure with Qos and Security

Afsheen Mughal, Mohammed Kanjee, and Hong Liu

University of Massashusetts Dartmouth
Department of Electrical And Computer Enginering
285 Old Westport Road, North Dartmouth, MA 02740, USA
{amughal,mkanjee,hliu}@UmassD.edu

Abstract. This paper proposes a security framework with quality of service (QoS) mechanisms embedded in a Next Generation Internet architecture to support healthcare infrastructure with mobile wireless sensors. The framework shields the complexity of internetworking with a policy management system to subscribe quality of service and level of security. The framework also hides the intricacy of mobile wireless-networked sensors with a middle ware component to deliver sensing data and retrieve patient monitoring information. Complying with the Internet Design Philosophy, the complexity is pushed to the end processing nodes for healthcare information comprehension and manipulation. Both security and service requirements for healthcare infrastructure are achieved with the established architecture of Next Generation Internet.

Keywords: Next Generation Internet (NGI) architectures; security framework; Quality of Service (QoS) mechanisms; Mobile Wireless Sensor Network (WSN); Healthcare Sensor Network (HSN).

1 Introduction

Healthcare infrastructure deploys both the Internet and wireless sensor networks (WSN) to achieve mobility in monitoring patient conditions. Sensors attached to patients are used to monitor and measure vital signs such as patient's heart rate or body temperature [1]. Having these body area wireless sensors allows the patients to be mobile and not be confined to one area. Critically ill patients have to be put under constant bedside monitoring for the healthcare practitioners to effectively react in a timely manner in case of emergency. If these patients were allowed to be mobile they would benefit from physical exercise as well as better patient recovery in a favorable environment away from the hospital, without sacrificing effective reaction time during emergency. These measurements are sent periodically to the medical staff. Because of the sensitive nature of the data, it is critical to provide mechanisms to protect patient's medical files. Security becomes an important design goal in such applications. Sensor nodes have constraints in computation, memory and power resources, therefore, it becomes challenging when designing a secure WSN application [2]. A trade-off must be made between the level of security provided and the resources consumed.

Recently, there have been several WSN applications proposed and designed specifically for the medical and healthcare industry. These include CodeBlue

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 462–473, 2010.

developed for emergency medical care [3], AlarmNet to monitor continuously assisted-living and independent-living residents [4], and SNAP (Sensor Network Assessment of Patients) with some security mechanism in its architecture [5]. Unfortunately, security issues have not been systematically addressed in Healthcare Sensor Network (HSN) applications [6]. Besides serious consequences led by patient data adversaries, the complying requirement with HIPAA (Health Insurance Portability and Accountability Act) makes security the top priority in all healthcare settings. However, before a potential security mechanism can be integrated into a HSN, we must understand and develop the measurement to assess and evaluate the requirements and security goals for the application.

The remaining paper is organized as the follows. In Section 2, we discuss possible threats and attacks a healthcare sensor network may face. Next, we examine the requirements and characteristics of the healthcare environment, which distinguishes it from other mobile wireless sensor network applications. Section 4 describes our Next Generation Internet (NGI) architecture of security framework enabled with Quality of Service (QoS) mechanisms for HSN applications. We give our conclusion and future work in the last section.

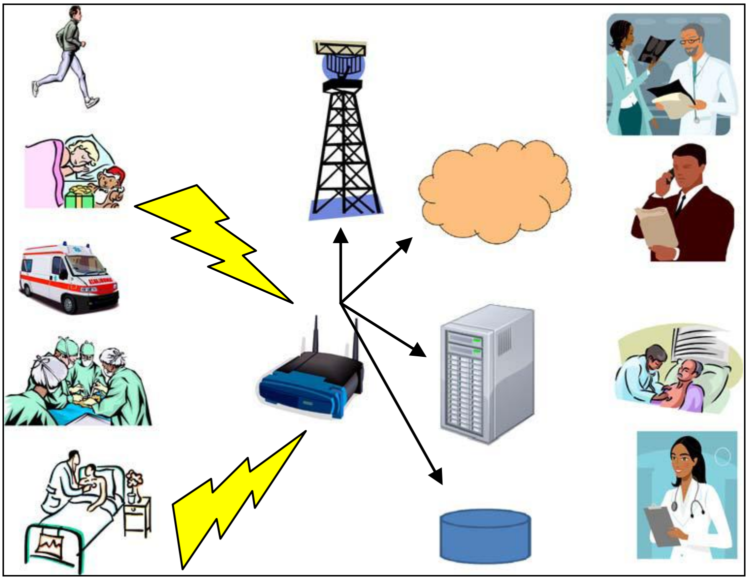


Fig. 1. Typical architecture of wireless sensor networks in healthcare applications [16]

2 Threats and Attacks on HSN

There are numerous different types of attacks or threats that a healthcare sensor network may face. These attacks can be classified according to the intended target: patient nodes or a healthcare system. A patient node attack is one in which the patient end is specifically targeted by malicious acts. In a healthcare system attack, the adversary aims to disrupt/destroy the link between the medical personnel and the patient nodes as well as the central system.

Attacks at the patient level include eavesdropping (snooping or traffic analysis), unauthorized modification, masquerading, and node compromising. Attacks at the system level include Denial-of-Service (DoS), system intrusion, and impostor.

A malicious sensor node intercepts and/or overhears packets which are being transmitted between a patient and a member of the medical staff. By overhearing packets not intended for it, an intruder can perform an analysis on the traffic patterns and potentially steal private and sensitive patient information. This leads to a violation of the patient's privacy. A malicious node can also gain information on the encryption scheme being used between the patient node and medical staff. This node can decrypt future communications between the patient node and medical staff. There can be a possible misuse of information by the malicious node.

Attacks can be classified into two abstract categories passive and active [16]. Passive attacks change the path of the information within the network in order to obtain specific information or to make routing inconsistent. Active attacks are more harmful in nature as they can be life threatening. They may try to corrupt vital patient information within the network, thereby preventing the monitoring entity to take the right action in a timely fashion. Some of the types of attacks that can be injected on a HSN are [16]

- Data Modification – The attack could modify or delete vital patient data and send the modified version back to the original receiver causing the patient to be misdiagnosed.
- Impersonation attack – The attack could eavesdrop and obtain the node identification information, which can be used to deceive other nodes.
- Eavesdropping – The attack could eavesdrop on information that is being sent on the open channel and use sensitive patient data for criminal acts.
- Replaying – The attack could reply stale information to the receiver and prevent real-time patient data from reaching the original receiver.

Security in HSN can be divided into two tiers [16]. The first tier consists of the System Security, which includes access to the physical systems – sensor nodes, gateways and centralized server. Accesses to these systems have to be controlled via measures of authentication and authorization and use of firewalls to prevent non-authorized users form assessing the system. System level security has to be applied at three levels – Administrative, Physical and Technical.

- Administrative Level Security is applied to check security breaches by the people who are responsible for system operation. Authentication measures along with access mechanism to prevent unauthorized users from accessing sensitive patient data.
- Physical Level Security is applied to prevent physical access to the devices attached to the patient and other equipment through which information is channeled or stored. These devices are open to attacks who would want to tamper with the devices in order to gain access to sensitive patient information. Physical Level Security is the hardest to implement in a distributed and scaled environment of HSN.

- Technical Level Security is implemented on hardware such as servers. In a network oriented design data is sent to central servers, server based security measures have to be implemented. Secure routing will have to be implemented to prevent attackers from causing routing inconsistencies resulting in erroneous destination. Due to sensitive nature of the data in the healthcare domain it is necessary to implement encryption schemes.

The second tier of security consists of Information Security, which prevents the tremendous amount of sensitive patient data traversing through the HSN to fall into the hands of attackers [16]. The data is at risk of sabotage, theft, exploitation and manipulation. The information security apparatus should be able to provide the following security services

- Data Encryption – Information traversing the HSN is encrypted so that it is not easy for eavesdroppers to gain access to data while it is in transit.
- Data Integrality – Sensitive patient information has to be authenticated against the sender, while also making sure that it stands the test of integrity. It provides against data modification attacks.
- Authentication – Various devices in the network have to be authenticated against some central system to make sure imposter devices are not induced into the HSN. False nodes masquerading as authentic devices can reroute sensitive information or induce false information into the HSN producing devastating effects for patients.
- Freshness Protection – Freshness provides protection against replay attacks.

3 Requirements and Security Goals in HSN

The main issue of the existing security solutions for HSN is that not all security goals or application requirements are satisfied. Although many security solutions have been proposed for WSN [7-9], they are not designed for healthcare in mind. Furthermore, among the HSN applications, not all have addressed security in their original design [3, 4]. As a result, a gap exists between the security requirements of HSN and the state-of-the-art WSN security solutions for medical applications.

Because HSN has different characteristics compared to other WSN applications, existing WSN security technologies are not suitable to or overkill HSN. The table below lists the key differences between a healthcare application and a general wireless sensor network.

Unlike a general WSN, a HSN application deploys two levels of different security goals: the node (patient) level and the system level. At the patient level, confidentiality ensures that patient's medical files are protected from eavesdropping or traffic analysis. Integrity prohibits altering medical reports, at the nodes as well as during transmission from patients to medical staff, by any external or unauthorized source. Patient data freshness keeps the information recent. Patient data availability ensures patient data obtainable to doctors and other medical personnel at all times. Authentication verifies the legitimacy of an entity while authorization grants the access to that confirmed entity to access the patient data.

A healthcare sensor network is a network of sensors deployed on human bodies to monitor patients' health. These sensors collect personal medical data, therefore, security and privacy are important requirements in healthcare sensor networks. At the same time the network should be able to transmit the data in a robust manner in order for it to be readily available in case of a medical emergency. Any delay or latency can prove to be fatal for the patient if the medical staff is not able to respond in time.

Despite the increased range of potential health care applications – ranging from pre-hospital, in-hospital, ambulatory and home monitoring, to long term database collection for analysis – the security gap that exists between wireless sensor networks and the requirements of the medical applications and community has yet to be resolved. Wireless sensor networks are limited in terms of power and computation, and are deployed in areas where they can be easily accessed causing security vulnerabilities. Dynamic ad hoc topology, multicast transmission, location awareness, critical data acquisition, and co-ordination of diverse sensors of health care applications further exacerbate the security challenges [1].

Table 1. Contrast HSN vs. WSN

Characteristics	**Healthcare**	**Wireless Sensor Network**
Energy Efficient	Batteries replaced by medical staff	Energy source not usually replenish
Privacy	Protect from unauthorized users	
Real-Time Response	Time-critical	Delay tolerable
Accurate Patient Results	For better treatment/diagnose	Application-dependent
ID-Centric Addressing	Patient identity as important as data	Data-Centric addressing scheme
In-network Processing	Limited redundancy	Communication cost reduction
Robustness	Limited redundancy	Node failure tolerable
Scalability	Vary patient density	Nodes enter/leave network
Mobility	Both patients and doctors are mobile	Application-dependent

Table 2. Compare Node vs. System

Security Goals	**Patient Level**	**System Level**
Confidentiality	YES	
Integrity	YES	
Freshness	YES	
Availability	YES	YES
Authentication	YES	YES
Authorization	YES	YES

Sensor nodes in the healthcare environment are semi-permanently deployed, since the topology of such networks changes over time, due to new sensors being introduced into the environment and in the case of mobile patients, the patients themselves moving in and out of the network. Since each sensor node is acquiring critical medical data the nodes should be able to coordinate with each other with to acquire the data, perform computation and selection to transmit the required information. The purpose of deployment of sensor nodes in the healthcare environment is to allow patients to be mobile and not be confined to one location; some type of location awareness implementation is required. This implementation becomes even more critical if security is involved, as the security key shared between a group of sensors and the cluster node would change if the group of sensors transgresses into an area covered by another cluster node.

At the network level, availability guarantees that the system remains operational 24/7. Authentication is used to establish legitimate communication between sensor nodes and the system. Authorization is used to make sure that authorized medical personal are accessing patient data. Once authentication and authorization are in place, confidentiality and integrity would be implied at the system level. Therefore, confidentiality, integrity, and freshness (guaranteed by patient nodes) have no need to be addressed specifically at the system level. Table 2 summarizes our findings.

4 Our Approach: NGI Architecture

A sensor network is composed of a large number of sensor nodes that are densely deployed either inside the phenomenon or closely around it. The positions of sensor nodes need not be engineered or predetermined, which allows random deployment in an inaccessible terrain or disaster relief operations. On the other hand, such a feature requires self-organizing capabilities in sensor network protocols and algorithms. Another unique feature of sensor networks is the cooperative effort of sensor nodes. Sensors fit with an onboard processor, instead of sending raw data to a cluster head responsible for data fusion, sensor nodes use their processing abilities to locally carry out simple computations and transmit processed data.

In healthcare applications, sensor nodes are deployed to monitor patients and assist disabled. Our research is focused on designing a wireless sensor network that collects, transmits, and processes sensitive patient information for medical personals to monitor patients in real time. Since security is of significant challenge in transmitting data wirelessly and timely, we propose a security architecture enabled Quality of Service (QoS) to support mobile healthcare infrastructure. Our approach is unique in that we place security in the center of the architectural design. The goals of our research are to

- Develop an architectural design that positions security as a core component,
- Implement a security structure that provides low latency encryption and decryption, and
- Design security algorithms that are not resource intensive, permitting its deployment on sensor nodes.

We present our work in five subsections. Subsection 4.1 describes our two-tier architecture that places security in its core design and makes security implementation feasible under resource scarce computing environment. Our assumptions of sensor nodes are given in the second subsection. The next two subsections discuss the two tiers: the low-tier structure is a middle ware to shield the diversity of sensing nodes in security implementation, and the high-tier structure deploys policy management for the Next Generation Internet (NGI) to adapt changes in security requirements. The assessment of our architecture towards its design goals is shown in Subsection 4.5.

4.1 Two Tier Architecture

Our two-tier networking architecture untangles long haul medical communications on the Internet from short-range transmissions within individual wireless clusters of patient sensor nodes. The low-tier structure deals with diverse patients' data: being assistant living, clinic heart monitoring, or a sudden epidemic like swine flu. The high-tier structure provides medical staff communication, real-time observation, or health data processing. Naturally, the patient-oriented low tier contains wireless sensor networks while the doctor-oriented high tier deploys the Internet. Figure 2 depicts our two-tier architecture. The two tiers interact as if each wireless cluster of patient sensor nodes is a periphery of a medical system plugged through an end node of the Internet.

The two tiers possess drastically divergent features that lead to different approaches. For security, the low tier fulfils the security goals at the patient level, listed in Table 2,

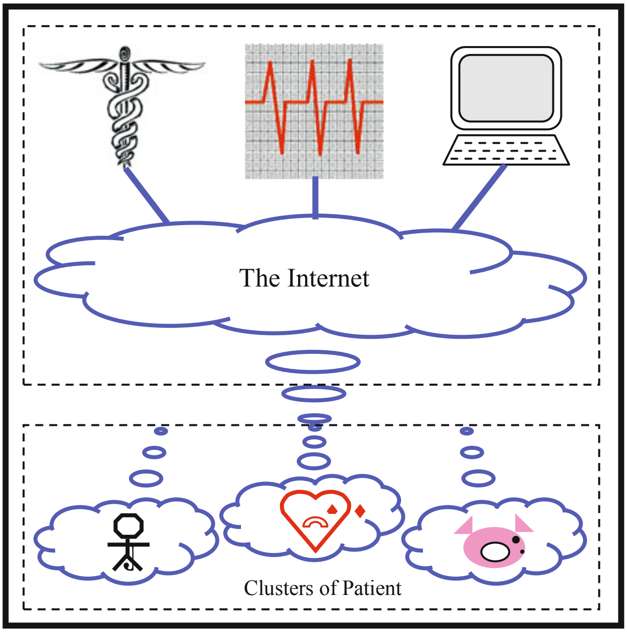


Fig. 2. Two Tier Architecture

while the high tier covers those at the system level. However, a secured infrastructure would diminish its purpose if it provides poor Quality of Service (QoS) such as data loss and long delay. For QoS requirements, the high tier, using the Internet, has the traditional QoS issues that can be dealt with mechanisms proposed for the Next Generation Internet (NGI), to be discussed in Subsection 4.4. The low tier, composed of wireless sensor networks (WSN), faces new QoS challenges due to unreliable communication service of wireless links and limited computation resource of sensor nodes. Pay attention to the unique characteristics of healthcare sensor network (HSN) summarized in Table 1; Subsection 4.3 presents our solution.

The two tiers possess drastically divergent features that lead to different approaches. For security, the low tier fulfils the security goals at the patient level, listed in Table 2, while the high tier covers those at the system level. However, a secured infrastructure would diminish its purpose if it provides poor Quality of Service (QoS) such as data loss and long delay. For QoS requirements, the high tier, using the Internet, has the traditional QoS issues that can be dealt with mechanisms proposed for the Next Generation Internet (NGI), to be discussed in Subsection 4.4. The low tier, composed of wireless sensor networks (WSN), faces new QoS challenges due to unreliable communication service of wireless links and limited computation resource of sensor nodes. Pay attention to the unique characteristics of healthcare sensor network (HSN) summarized in Table 1; Subsection 4.3 presents our solution.

4.2 Sensor Node

As shown in Figure 3 below [2], a sensor node consists of four units: a sensing unit, a processing unit, a transceiver unit, and a power unit. A sensor node might also equip some application-driven components such as a location finding system, a mobilizer, and a power generator. Its sensing unit contains sensors and analogue-to-digital converters (ADC). It performs sensing data and delivers the data to the processing unit for analyses. The processing unit has a processor and a small storage. The transceiver unit transmits the processed data and receives control signals for nodal

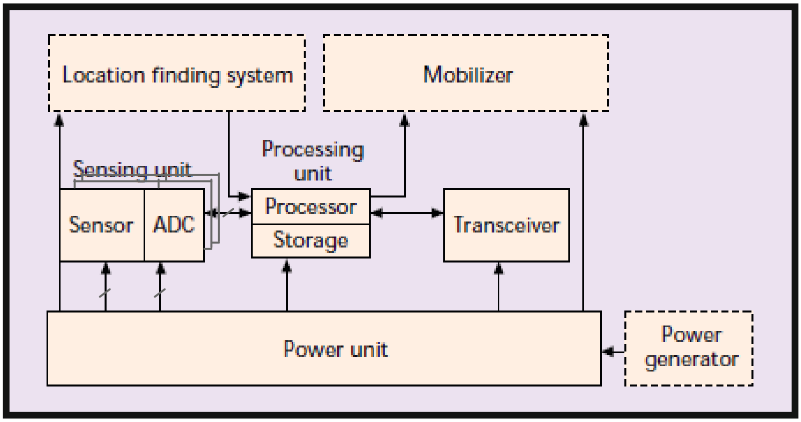


Fig. 3. Components of a Sensor Node

reconfiguration or for data relay. The power unit supplies electricity. For mobile patients, a location finding system keeps track of him for emergency response. Most medical sensor nodes, wearable, do not need a mobilizer to propel the node. A power generator is included if the node does not use batteries or a wall plug.

Process/storage limitation and real-time requirement are the driving forces behind the design of a security protocol tailored for HSN.

4.3 Low-Tier Structure: Middle Ware

The low tier is a sky topology of wireless sensor networks (WSN), each of which is a star topology. A star stems at a Base Station and rays in several Cluster Heads, as shown in Figure 4. Although the figure provides one cluster head for each patient, a cluster head can accommodate several patients geographically nearby without increasing computational complexity because only simple addressing not routing is involved to locate a patient under a cluster. A *Patient Node* deploys sensors to gather various medical data such as temperature, blood pressure, and EKG. It then processes/encrypts data and sends them to its cluster head located in the near vicinity. A *Cluster Head* in our HSN does not deploy any sensors, and its function is to fuse and relay data. A *Base Station* acts as the interface between the low tier and the high tier.

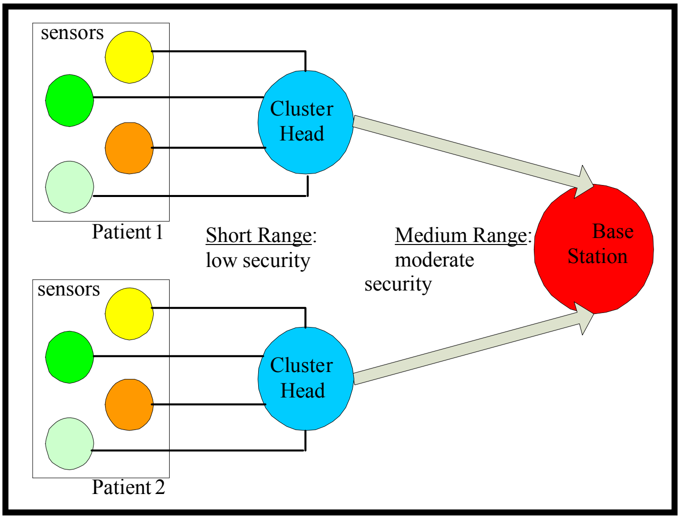


Fig. 4. A Star in Low Tier

The distances from the sensors of a patient node to its cluster head are short, body length in our prototype. The short range of communication minimizes interception of data by foreign entities; therefore, a low degree of security is sufficient enough for ad hoc settings as individual patients come and go. Even with the resource limitation on these nodes, it is feasible to develop low-latency or real-time security protocols for patient's sensor nodes and its cluster head with small overhead on encryption.

Communications between cluster heads and their base station is medium range, vulnerable to security breaches. Fortunately, cluster heads and a base station have abundant resources, capable of running a moderate degree security protocol without causing much latency [10, 11].

A star topology easily houses a *Middle Ware*, a warehouse of software/ firmware/hardware to process data for security and QoS while hide the intricacy of various wireless networking and sensing technologies. To counterattack *Traffic Analysis*, study of traffic patterns without knowing their contents, the transmissions both of short range (from patient's sensors to its cluster head) and of medium range (from cluster heads to the base station) are kept in regular intervals. All data in transmission at the low tier are encrypted, and no decryption is involved until the base station. This simplicity works as a double-edged knife that ensures the desired level of security in Table 2 and promises QoS with ignoble loss, low delay, and no jitter in Table 1. Depending on its configuration dictated by the policy (to be discusses in the next subsection), a base station filters traffic before forwards it to the high tier. It uses partial decryption (full decryption at medical systems of the high tier) and prunes noise/insignificant traffic to reduce aggregated traffic towards the high tier.

4.4 Low-Tier Structure: Middle Ware

The high tier incorporates policy management into the differentiated service (DiffServ) model for the Internet. *DiffServ* is the most prominent QoS model for NGI, evolving from the original Internet with a best-effort service model, by handling classes of traffic in different ways. Instead of trying the best to deliver all packages equally poor, a DiffServ-capable router offers subscribed QoS to aggregated traffic by their service class [12]. DiffServ routers are classified into edge routers and core routers. An *Edge Router*, at the "edge" of the Internet, connects to end systems, base stations of the low tier or medical systems of the high tier in HSN. A *Core Router*, within the Internet, finds a path to forward a packet with the QoS by that packet's class. Figure 5 depicts policy management in DiffServ. A Policy Enforcement Point (PEP), added to each edge router, executes configured policies. A Policy Decision Point (PDP) performs complex policy interpretations for PEPs [13]. The Policy Information Base (PIB) stores policies, which is created and maintained by a Policy Management Tool (PMT) whose performance is feedback by a QoS monitoring at each edge router.

Policy management in DiffServ has been successfully applied to offer QoS by major Internet Service Providers (ISP) [14] and to combat Denial-of-Service attacks [15]. Applied to support mobile healthcare infrastructure, we need to address the QoS requirements in Table 1 and the three security goals at system level in Table 2. Real-time response [14] and system availability [15] at the high tier are readily done. New policies need to be developed for authentication and authorization.

4.5 Assessment

We have conducted qualitative evaluations of our architecture's suitability to healthcare. As discussed in Subsections 4.3 and 4.4, both the two tiers satisfy the QoS requirements in Table 1; the low tier achieves the security goals in Table 2 at the patient level while the high tier at the system level. The details of our assessment with a prototype will be presented in a sequel paper.

We need to design a quantitative matrix for assessing performance vs. efficiency with respect to its real-time response, data accuracy, privacy protection, system vulnerability, scalability, and mobility. We also need to devise a comparative study of our architecture with other architectures applicable to healthcare.

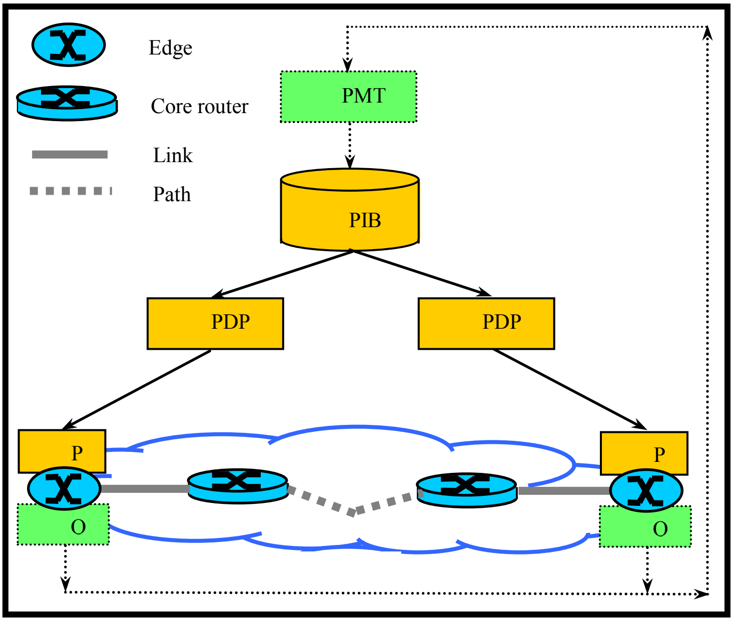


Fig. 5. Policy Management in High-Tier

5 Conclusion

This work is the first attempt to provide security in mobile healthcare infrastructure at architectural level. The unique two-tier architecture offers end-to-end total security from patients to doctors across long haul internetworking and covering wireless communication islands. The secure framework with QoS mechanisms oversees complicated processes of security and quality assurance with ease, where existing piecewise protocols/algorithms fit seamlessly and new ones would be justified to fill in security holes. The low tier also acts as a middle ware to hide the diversity of wireless medical sensing techniques, and the high tier utilizes NGI's DiffServ model enhanced with policy management to outlive the unpredictable security challenges.

Another contribution is the unique approach we use to solve security problems. Instead of focusing on surfaced problems, we analyze the characteristics of the application and identify its security goals. We let the application lead to a natural architecture for security and design testing procedures before a purposeful implementation. The method is applicable to other field of WSN applications.

The architecture lays a grant future work. Besides its self-correction, we need to choose and design specific security techniques for its components. We also need to assess the work systematically.

References

[1] Tan, C.C., Wang, H., Zhong, S., Li, Q.: Body sensor network security: an identity-based cryptography approach. In: Proceedings of the 1st ACM Conference on Wireless Network Security, Alexandria, VA, USA, March 31-April 02, 2008, pp. 148–153 (2008)

[2] Karl, H., Willig, A.: Protocols and architecture for wireless sensor networks. Wiley, Boston (2007)

[3] Malan, D., Fulford-Jones, T., Welsh, M., Moulton, S.: Codeblue: An ad hoc sensor network infrastructure for emergency medical care. In: International Workshop on Wearable and Implantable Body Sensor Networks (2004)

[4] Wood, A., Virone, G., Doan, T., Cao, Q., Selavo, L., Wu, Y., Fang, L., He, Z., Lin, S., Stankovic, J.: ALARM-NET: Wireless sensor networks for assisted-living and health monitoring, Technical Report CS-2006–01, University of Virginia (2006)

[5] Malasri, K., Wang, L.: Addressing security in medical sensor networks. In: Proceedings of the 1st ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments, San Juan, Puerto Rico, June 11-13 (2007)

[6] Wang, Y., Attebury, G., Ramamurthy, B.: A survey of security issues in wireless sensor networks. IEEE Communications Surveys & Tutorials 8(2), 2–23 (2006)

[7] Malan, D.J., Welsh, M., Smith, M.D.: Implementing public-key infrastructure for sensor networks. ACM Transactions on Sensor Networks 4(4), 22–45 (2008)

[8] Karlof, C., Sastry, N., Wagner, D.: TinySec: A link layer security architecture for wireless sensor networks. In: Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems, Baltimore, Maryland, USA (2004)

[9] Liu, A., Ning, P.: TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks. In: Proceedings of the 7th International Conference on Information Processing in Sensor Networks, April 22-24, pp. 245–256 (2008)

[10] Kurian, J., Sarac, K.: A security framework for service overlay networks: access control. In: BroadNets 2008, Internet Track 3: Overlays and Traffic Estimation London, UK, September 8-11 (2008)

[11] Wang, Y., Ramamurthy, B., Xue, Y., Zou, X.: A key management framework for wireless sensor networks utilizing a unique session key. In: BroadNets 2008, Wireless Track 6: MAC and Key Management London, UK, September 8-11 (2008)

[12] Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An architecture for differentiated services. RFC2475 (December 1998)

[13] Rajan, R., Verma, D., kamat, S., Felstaine, E., Herzog, S.: A policy framework for integrated and differentiated services in the Internet. IEEE Network, 36–41 (September 1999)

[14] Liu, H., Dempsey, H.H.: Multi-facet Internet resource management system. In: Proceedings of IFIP/IEEE International Symposium on Integrated Network Management (IM), Boston, MA, USA, May 24-28 (1999)

[15] Yu, Q., (Liu, H., advisor): Denial-of-Service Countermeasure with Immunization and Regulation: Ph.D. Dissertation University of Massachusetts Dartmouth, Dartmouth (2005)

[16] Ameen, M., Jingwei, L., Kyungsup, K.: Security and Privacy Issues in Wireless Sensor Networks for Healthcare Applications. Journal of Medical Systems (March 2010)

A Survey on the Cross-Layer Design for Wireless Multimedia Sensor Networks

Honggang Wang[1], Wei Wang[2], Shaoen Wu, and Kun Hua

[1] University of Massachusetts, Dartmouth, MA, USA
hwang1@umassd.edu
[2] South Dakota State University, Brookings, SD, USA
wei.wang@sdstate.edu

Abstract. Small low-cost multimedia sensors ubiquitously capture and transmit multimedia information from fields to central stations in support of applications. However, Multimedia sensors in Wireless Multimedia Sensor Networks (WMSNs), have limited resources in terms of their computational capability, memory capability, bandwidth, and battery power, which hinder a wide application of WMSNs. The challenges, issues and solutions in WMSNs regarding QoS, energy efficiency and security have been recently studied. It is highly desirable to incorporate the characteristics of multimedia and information security at the application layer into adaptation and optimization at lower layers in a cross-layer approach. In this paper, we conducted a survey on the recent development of the cross-layer design in WMSNs. Based on our studies, we concluded that that cross-layer approaches are promising solutions and can efficiently support future multimedia application in WMSNs.

Keywords: Cross-layer design, Wireless Multimedia Sensor Network.

1 Introduction

Small low-cost multimedia sensors ubiquitously capture and transmit multimedia information from fields to central stations in support of applications such as instance multimedia surveillance networks, target tracking, environmental monitoring, and multimedia-aided navigation systems. These multimedia sensors are wirelessly connected to retrieve still image, video, audio and scalar data in Wireless Multimedia Sensor Networks (WMSNs). Multimedia sensors in WMSNs, have limited resources in terms of their computational capability, memory capability, bandwidth, and battery power, which hinder a wide application of WMSNs. They generate a large volume of data and consume a great deal of energy in data processing and transmission. With limited bandwidth and resource constraints in sensors and relay nodes, network bottlenecks in WMSNs likely occur and thus degrade network performance.

Multimedia Quality of Service (QoS) is another major concern in WMSNs. The error prone wireless channel causes packet loss and then degrades the multimedia quality. It is challenging to efficiently utilize limited network resources in order to protect the quality of multimedia transmissions. Multimedia

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 474–486, 2010.

sensor nodes are vulnerable to a variety of malicious attacks and compromises when they are deployed in a hostile environment. Therefore, security methods must be designed judiciously in sensor network environments to enhance error resilience, and provide security and perceptual integrity under limited power, memory, and computational constraints. A dynamically varying network topology and resource configuration becomes one of the major challenges to security protection efforts.

In response to these challenges and issues regarding QoS, energy efficiency and security, it is highly desirable to incorporate the characteristics of multimedia and information security at the application layer into adaptation and optimization at lower layers. Multimedia applications require secure, effective and efficient communication, as well as in-network processing platforms, where the entire multimedia system performance can be optimized as a whole. The error-prone and shared nature of dynamic wireless channel in WSNs allows us to break the traditional layer architecture for pursuing optimized network performance of multimedia delivery in a cross-layer manner. Given fundamental resource constraints, secure quality-driven energy-efficient cross-layer architecture for multimedia delivery must be developed carefully in WSNs to enhance error resilience, gain energy efficiency, and provide security protection. The joint design and control of multimedia source coding, resource allocation and information security creates a new perspective from which to explore the interaction among multiple equivalent layers, and fill the gaps among networking, cryptography and signal processing.

However, traditional cross-layer optimization and architecture [1-4] limit themselves in real practices due to their associated complexity. These challenges are even more critical for multimedia data delivery in WSNs due to the nature of multimedia data. The major focus of this survey is to investigate the current cross-layer design research development in addressing the QoS, security and energy efficiency issues in WMSNs.

1.1 Wireless Multimedia Sensor Hardware Structure

A basic multimedia sensor hardware structure can process multimedia data and be capable of networking. In the multimedia processing part, different types of physical sensors such as cameras, audio and scalar sensors acquire environment information in the form of multimedia. The Central Processing Unit (CPU) and memory implement algorithms to manipulate multimedia data such as data compression. The communication system handles the multimedia transmission over wireless environments, which includes the system software module and network interface as well as the wireless transceiver. The motor part is controlled to offer a platform for mobile multimedia sensor applications. The localization system provides information on the location of sensors in mobile application environments. Typical multimedia sensors include Cyclops image sensor (CMOS camera + MICA) with low resolution, medium-resolution imaging motes based on the Stargate platform , and the Imote 2 developed by Crossbow and Intel. The wirelessly-connected multimedia sensor nodes enable the interplay between multimedia processing and networking. To achieve overall quality network performance, the communication system protocol design and network resource management must provide an efficient support for multimedia processing techniques (e.g., compression); inversely, the processing techniques must be adaptive to the networking and communication capabilities.

1.2 Wireless Multimedia Sensor Networks Communication Structure

In WMSNs, heterogeneous multimedia sensors (audio sensors, scalar sensors, high- and low-end image sensors) have different capabilities in data processing and transmission. In WMSNs, a set of multimedia processing gateways formulates the distributed multimedia processing architecture. The multimedia contents are delivered and relayed through multimedia processing gateways. These multimedia processing gateways fuse and process multimedia data locally. A multi-tiered structure allows sensors to handle different levels of processing according to their memory, CPU computation and bandwidth capability. Wireless gateway functions as a sink node to collect multimedia data, which are ultimately delivered to users through the Internet or through satellite networks. In practice, this multimedia network of heterogeneous sensors has advantages over networks with homogeneous sensors, especially for multimedia processing and transmission. For example, more complex hardware and extra batteries can be selectively embedded in a few gateways such as cluster heads or high-end nodes rather than in all sensors for comprehensive multimedia processing, thereby reducing the hardware cost of the network.

1.3 WMSNs Applications

WMSNs support multimedia information retrieval and delivery over wireless environments, which enable new potential applications and enhance many existing network applications. These major WMSNs applications include:

- Target tracking applications
- Home automations
- Multimedia surveillance
- Environmental monitoring
- Multimedia-aided navigation systems and traffic avoidance
- Healthcare monitoring and delivery applications
- Environmental monitoring in the form of acoustic and video
- Manufacturing process controls for semiconductor chip, food or pharmaceutical products

A typical application, for example, is a secure multimedia surveillance over image/video sensor networks. the video sensor monitors the secure military area. Suddenly, an enemy robot is coming and trying to attack the secure area. The video sensor captures this scene and transmits the emergency multimedia information to the monitoring office through wireless links. However, the intruders can access and modify the image content so that the enemy robot is hidden or scratched in transmitted images. In this scenario, the monitoring office receives this faked image and cannot detect this incoming enemy robot. An authentication solution for recognizing this malicious activity is studied by utilizing watermarking techniques in this work, which allows the monitoring office to detect the modified image.

Many of the above applications require new mechanisms to deliver multimedia content with a certain level of quality of service (QoS), energy efficiency and security. These mechanisms should not only include energy efficient communications, but also the interplay between multimedia processing techniques and the communication process. In this paper, intelligent cross-layer architecture is studied to offer such new mechanisms to efficiently support WMSNs applications.

2 Research Challenges for WMSNs

Many WMSNs' applications as mentioned above require sensor networks to deliver multimedia content with a certain level of quality of service (QoS) and security protections as well as resource efficiency. To ensure multimedia deliveries are secure, energy-efficient, and high-quality, the following four issues are major challenges:

2.1 Resource Constraints and QoS Requirements

Sensor devices are constrained in terms of CPU computation and memory capability, bandwidth and battery support. These resource constraints make it difficult for Wireless Sensor Networks (WSNs) to provide a required QoS (e.g., multimedia quality, real-time performance) in many applications.

2.2 Layer Interactions and Complexity

The variable and shared nature of a wireless channel and uniqueness of multimedia in WSNs provide an opportunity to break the traditional layer structure and allow the interactions among different equivalent layers to optimize WMSNs system performance as a whole. This requires a secure energy-efficient cross-layer architecture that can couple several layer functionalities. However, only a few studies on cross-layer design have been conducted for multimedia delivery in WSNs, while much research focuses on image/video delivery over general wireless networks. As cross-layer design violates the layer structure, an optimization framework is needed to concurrently model multiple parameters from equivalent layers. The design complexity is thus intensified and needs to be addressed along with overheads.

2.3 Interplay between Multimedia Processing and Networking

Networked multimedia sensors can conduct in-network multimedia processing. In traditional designs, multimedia processing is independent of delivery of multimedia contents, while in WMSNs their interplay has a significant impact on the levels of QoS. The multimedia contents and source coding techniques cannot be designed without wireless network conditions and resource support. Inversely, the network protocol and resource management must consider multimedia contents and source coding techniques when multimedia sensors acquire and transmit multimedia data.

2.4 Resource Constrained Multimedia Security

Multimedia sensor nodes and data transmission among this nodes are vulnerable to a variety of malicious attacks and compromises when they are deployed in a hostile wireless environment. Security protection methods must be provided to guarantee multimedia content security and integrity in such environments. Given the fundamental issues (i.e., resource constraints and source coding) related to information security, they must be judiciously designed and implemented for secure multimedia delivery over WSNs.

Typical challenges include reliable secure resource-efficient multimedia processing and communication in WMSNs, such as multimedia quality definition with heterogeneous sensors, security protection, and tight QoS expectations. They boost developing feasible solutions to design and deploy wireless networked multimedia sensor systems. Comprehensive cross-layer architecture should be proposed to achieve the goal of resource efficiency, certain level multimedia QoS and high security. In this architecture, it is essential to jointly control efficient source coding, intelligent resource allocation and information security in a cross-layer manner.

The optimization problem for cross-layer multimedia delivery over WSNs is formulated as follows:

$$\begin{aligned}&\{(APP_1, APP_2, ...APP_q), (NET_1, NET_2...NET_n),\\&(MAC_1, MAC_2, ...MAC_m), (PHY_1, PHY_2, ...PHY_p\\&)\} = \arg\max(Target)\ or\\&\ \arg\min(Target),\end{aligned}$$

$$s.t. \quad \{Con_1, Con_2...Con_i\},$$

where we either maximize or minimize the target. This target function is defined as any metric related to QoS requirements, energy efficiency or security performance of multimedia transmissions. The four sets, $(APP_1, APP_2, ...APP_q)$, $(NET_1, NET_2...NET_n)$, $(MAC_1, MAC_2, ...MAC_m), (PHY_1, PHY_2, ...PHY_p)$ denote the cross-layer parameters at equivalent application layer, network layer, MAC layer and physical layer, respectively. $\{Con_1, Con_2...Con_i\}$ represents the constraints at equivalent layers with low or high bound.

3 Current Research Development of the Cross-Layer Design for WMSN

In WSNs, it is necessary to break Open Systems Interconnection (OSI) architecture, as there is a tradeoff between QoS gain and resource cost in optimizing multimedia delivery performance. The cross-layered design for multimedia delivery in WSNs has more advantages than traditional layered approaches in transmissions. First, the traditional layered architecture is hierarchical and layer-independent, which forbids direct communication between nonadjacent layers. The layers in the cross-layer design are dependent, and can communicate directly or share variables between nonadjacent layers. Second, although traditional architecture performs well in wired networks, it does not function well in wireless networks. Supporting multimedia applications and services over wireless networks is challenging due to constraints and heterogeneities such as limited battery power, limited bandwidth, random time-varying fading effect, and stringent quality of service (QoS) requirements. These challenges cannot be solved via traditional layered architecture. The cross-layer design, instead, provides a new venue to enable optimal communication over wireless links and multimedia data processing at the application layer. Third, in WMSNs, new patterns of communication (e.g., channel broadcast nature, variance channel) through wireless medium allow new interfaces, merging of adjacent layers, and vertical calibration across layers. The adaptive strategy at each individual layer in traditional layer architecture is always suboptimal as the dependence of these layers is ignored and the optimization is localized at each layer. In a cross-layer approach, the layers share systematic information and can achieve global or systematic optimization of the multimedia network performance. Finally, the application-specific and energy-resource limitations of WMSNs pose challenges for cross-layer architecture design. To address these issues and challenges, optimal solutions need to be proposed to explore the benefits of this cross-layer approach.

There are three primary approaches to cross-layer architecture design [17]. The first approach allows direct communication between layers, where the information is shared in real time through visible variables (e.g., protocol headers). The second approach enables several layers to share a common database that is used for service storage and information retrieval. This approach is suitable for vertical calibration across layers. The third approach is to provide a complete new abstraction to organize protocols with flexibility. These cross-layer approaches in WSNs have been studied in two main contexts. One is focused on cross-layer interactions, where each layer has the information about other layers while the traditional layered structure has information at each layer. The second context reconsiders the mechanism of network layers in a unified way to provide a single communication module for efficient communication.

In WSNs, the cross-layer optimization considers the fundamental tradeoff between application-specific QoS gain and resource cost. Sensornet Protocol [1] uses the link layer abstraction and allows cooperation between the link layer and network layer, where limited resources can be utilized efficiently. EYES MAC [2] models the interaction between the MAC and routing protocol. It can improve traffic routing performance with consideration of network topologies, power duty cycling and node failure. In [3], the authors studied an energy consumption minimization problem by developing a joint design of MAC, Link and routing schemes. The link adaptation, optimal routing and scheduling are modeled for calculating the energy consumption. [4] proposed a unified cross-layer protocol to achieve energy-efficient and reliable event communication, which integrates the transport, network and MAC functionalities into one single module called XLM. In [5], the end-to-end congestion control at the transport layer and the power control at the physical layer are optimized through the JOCP algorithm. In [6], the authors form a network lifetime optimization problem under the constraints of transmission rates, energy budget and communication range. The optimal solution suggests an optimal rate control and link scheduling. In [7], the authors proposed a Low Energy Self-Organizing Protocol (LESOP) specifically for target tracking applications in dense WSNs. The LESOP can achieve high protocol efficiency through direct interactions between the application layer and MAC layer. In [8][9], we studied the cross-layer design for distributed source coding (DSC) in sensor networks by a joint design of the routing, link assignment and coding rate allocation at the application layer. However, these approaches are not easy to apply to in the multimedia transmission over WSNs, as their resource management, adaptation, and protection strategies at lower layers (PHY, MAC, Network/Transport) are suboptimal without considering characteristics of multimedia applications. In [10], we studied the joint design of the resource allocation at the Link-PHY layer with the rate distributions at the application layer for collaborative multimedia transmission. In [11], the single hop scenario for wavelet-based image transmissions in WSNs was also studied. The key idea is to differentiate the importance of digital images and allocate extra network resources to protect position values (P values).

More general cross-layer studies for multimedia delivery in wireless networks (not limited to sensor networks) have been conducted extensively in recent years. In [12], the authors proposed an adaptive cross-layer strategy to enhance the robustness and efficiency of scalable video transmission by optimizing MAC retransmission strategy, application-layer forward error correction, bandwidth-adaptive compression and adaptive packetization strategies. The research in [13] describes a cross-layer framework that selects and adapts different strategies available at the various OSI layers in terms of multimedia quality, consumed power, and spectrum utilization. In [14], the authors investigated how several APP (application layer) and MAC strategies can be jointly optimized to improve multimedia quality. Specifically, the experimental results show that the decoded video quality can be maximized by optimizing the MAC retry limit along with the application layer rate adaptation and prioritized scheduling strategies. In [15], an application-centric cross-layer approach is studied, where the APP layer selects the optimal MAC and PHY parameters. Incorporating the APP layer information into the cross-layer optimization, this approach offsets the disadvantages of the suboptimal multimedia delivery performance of the single MAC-PHY cross-layer approach. In [16], the source coding, allowable retransmissions, adaptive modulation and channel coding have been jointly optimized within a rate-distortion theoretical framework. Through the joint selection of parameters at physical, data link and application layers, the transmitted video quality over wireless networks can be improved.

Table 1. Existing Research and Cross-Layer Architecture

Cross-Layer Design Application	**Cross-Layer Architecture** (PHY—physical layer; APP- transportation layer; MAC/LINK- MAC and Link layer; Equivalent)	**Detail Methodologies**
For general data transmission in Wireless Sensor Network	MAC+PHY	[1]-energy consumption analysis for Physical and MAC layers is performed for three different MAC protocols [6]- Network lifetime maximization
	MAC+Transport+Routing	[4]- single XLM module for energy efficiency and reliable event communication
	MAC+Routing	[2]-traffic routing with the consideration of network topologies, power duty cycling and node failure
	Routing+MAC/LINK	[3]- form an energy consumption minimization LP problem
	Transport+PHY	[5]-an cross-layer optimization solution for power control and congestion control
	APP+MAC layer	[7]-Low energy self-organizing protocol(LESOP) for target tracking applications
	APP+Routing+MAC/LINK	[8]-joint the rate allocation, link assignment and rate-oriented routing for distributed source coding based applications
	APP+PHY+MAC/LINK	[9]-Join the resource allocation at LINK-PHY with the source coding adaptation
For multimedia transmission over Wireless Networks	APP+Routing+ MAC/LINK+PHY (In WSNs)	[10]-the rate-oriented routing, unequal resource allocation (PHY+LINK) and rate distribution for collaborative image transmissions.
	APP+MAC/LINK+PHY	[11]-joint optimizations of the resource allocation at LINK-PHY, and Position-Value based wavelet coding at APP layer(In WSNs) [15]-application-centric approach incorporating APP parameters into MAC-PHY optimization (General Wireless Networks) [16]-within rate-distortion frameworks, joint source coding, retransmission, adaptive modulation and channel coding
	OSI equivalent Layers	[13]-the adaptive strategy from various OSI layers for multimedia quality, power and spectrum utilization
	MAC+APP	[12]-joint the MAC retransmission, APP forward error correction and adaptive compression [14]-joint retransmission limit at MAC and rate adaptation at APP

These cross-layer approaches have difficulty in WMSN applications. First, generic wireless networks are more concerned with throughput, rather than energy efficiency, while in WSNs, the throughput is not critical as other metrics such as energy consumption and network life time. Energy efficiency is one of major goals for the cross-layer optimization of multimedia delivery in WSNs. Second, the existing cross-layer design for the general form of data transmission in sensor networks does not consider multimedia characteristics or content. Without incorporating multimedia information from the application layer, the cross-layer optimization at other lower layers is always sub-optimal. Therefore, the multimedia compression and streaming algorithms at the application layer should consider the mechanisms provided by the lower layers for error protection, scheduling, resource management, and so on. In addition, security is another major concern in WSNs. The security protection of transmitted multimedia requires both communication and computational resources. There is always a tradeoff between resource cost and security protection. The multimedia data protected by either encryption or authentication methods usually contain major information, and extra network resources should be provided to guarantee its transmission quality. Therefore, a joint design of the security and other cross-layer parameters such as resource allocation is necessary.

The current research is lack of an efficient cross-layer platform that can interplay among the multimedia processing, crytopography, and sensor networking to optimize overall WSNs performance. With the new development of multimedia applications over WSNs, the large volume of multimedia data transmission, strict QoS and security requirements need new efficient cross-layer architecture to support them under limited resource constraints in WSNs. Table 1 summarizes the current related research works in this field.

4 Case Study: Cross-Layer Based Collaborative Transmissions

An image sensor array deployed in the field with the same field views can provide distributed imaging of many applications. However, transmitting and processing a large volume of image data often causes bottlenecks in WSNs due to their limited resources (e.g. battery, and bandwidth). In many sensor network applications, they can drastically decrease the network performance and lifetime. One solution is to exploit the inter-image correlation among multiple sensors and remove inter-image redundancies. Many previous studies include cooperative methods [19]-[23] or predictive methods [24]-[28] to take advantage of sensor correlations. However, the former cannot be easily applied in image sensor applications due to heavy inter-sensor image communication overheads. Utilizing the sensor correlation model for efficient image transmissions should not only consider source image sensors themselves, but also take into account the network parameters such as the routing patterns and MAC/Link design. In our previous work [10], a cross-layer approach is proposed to distribute and protect the transmission of the overlap image regions shared by multiple sensors in a collaborative manner. We studies cross-layer architecture to support collaborative transmission by utilizing inter-sensor correlations. The maximum energy consumption saving bounds for collaborative transmissions are analyzed in this section. Let L denote the size of overlap regions, S be the size of the captured image, and N be the number of correlated sensors in a cluster group. In a two-sensor scenario for target monitoring, let sensor S_1 be at (x_1, y_1, z_1) position, and S_2 be at (x_2, y_2, z_2). d_1 is the distance between the target and S_1, and d_2 represents the distance between the target and S_2. To determine the overlap regions for these two sensors, several reference points in the target can be selected. For example, if the reference point i is at location $(x_i,\ y_i,\ z_i)$ and f is the focal length of the sensor, the coordinates of the projection point on the image plane (p, q) can be derived as:

$$p = \frac{x_i - x_1}{z_i - z_1} \times f,\ q = \frac{y_i - y_1}{z_i - z_1} \times f.$$

The errors in each dimension of the reference points are assumed to be normally distributed,N $\sim (0,\ \sigma^2)$, with zero mean and the variance of σ^2. The error term (e) is added to $\{p, q\}$,

$$p' = \frac{x_i - x_1}{z_i - z_1} \times f + e_p;\ q' = \frac{y_i - y_1}{z_i - z_1} \times f + e_q,$$

To determine if the two image sensors overlap, the intersection of these captured image shapes constructed by the reference points need to be determined. Thus, the size of overlap regions (L) can be represented as a Δ function of $\{p', q'\}$, which is denoted as

$$\Delta(\frac{x_i - x_s}{z_i - z_s} \times f + e_p,\ \frac{y_i - y_s}{z_i - z_s} \times f + e_q) \quad ,$$

where $\{x_s, y_s, z_s\}$ represents a sensor camera location. In non-collaborative multimedia transmission approaches, the total amount of transmitted image data Ψ_{no} is expressed as $(S + H_c) \times N$, where H_c denotes the communication protocol overheads. In collaborative image transmission, the overlap regions are transmitted to the base station collaboratively, and the total amount of communication load Ψ_{co} is expressed as:

$$N \times (S + H) - (N - 1) \cdot \Delta(\frac{x_i - x_s}{z_i - z_s} \times f + e_p, \frac{y_i - y_s}{z_i - z_s} \times f + e_q)$$

When the proposed secret sharing scheme (see Chapter 4) is applied, the amount of transmitted data is increased due to the redundancies of secret shares. In this proposed $(r,\ n)$ threshold image secret sharing scheme for overlap regions, the size of each image secret is L/r . Then the total amount of communication load is

$$\Psi_{se} = N \times (S + H) - \Delta(\frac{x_i - x_s}{z_i - z_s} \times f + e_p, \frac{y_i - y_s}{z_i - z_s} \times f + e_q) \times (N - \frac{n}{r})$$

The relationship $\Psi_{co} <= \Psi_{se} <= \Psi_{no}$ is always true when there are overlap regions. Due to this ordered relationship among Ψ_{co}, Ψ_{se} and Ψ_{no},this collaborative approach can achieve the minimal communication loads for any overlap regions.

Table 2. Analytical PMES Results for Collaborative Transmissions

Number of sensors	PMES (Collaborative; no energy minimization)	PMES (Collaborative; energy minimization)
5	30%	86.72%
6	32%	87.10%
7	33.33%	87.35%

Let $\rho_{\min}$ denote the minimum energy consumption per transmitted bit, and $\rho_{\max}$ represent the energy maximum energy consumption per transmitted bit. Compared with non-collaborative approaches, the Percentage of Maximum Energy Saving (PMES) without the low layer energy minimization is $[(\Psi_{no} - \Psi_{co})/\Psi_{no}]\%$ for the collaborative transmissions. The PMES with the energy minimization mechanism for the collaborative approach is

$$[\frac{(\Psi_{no} \times e_{\max} - \Psi_{co} \times e_{\min})}{(\Psi_{no} \times e_{\max})}]\%$$

In this case study, the image size is 10k bytes, and the total overlap regions are 4K bytes. $\rho_{\max}$ is 6.324 e-4 joules/bit , and $\rho_{\min}$ is 1.2e-4 joules/bit. The (r=6, n=10) secret sharing scheme is applied in this case. When the number of participated image sensors varies, the percentage of maximum ES is shown in Table II.

Table 2 shows a theoretical bound for energy savings with determined multimedia sensor network configuration. It is observed that the energy savings are increased with the number of correlated sensors. The values of PMES with energy minimization (see Table 2) indicate that the collaborative approach in the proposed architecture can achieve up to 87% maximum energy savings compared with the non-collaborative multimedia transmission approaches.

5 Conclusions and Future Directions

The main goal of this paper is to investigate the current development of cross-layer design research in WMSNs and provide research directions in these fields. Current research mainly focuses on three objectives: The first objective is to improve resource efficiency. Multimedia sensor has limited resources in terms of CPU computational capability, memory and bandwidth as well as battery support capability. These resources must be utilized efficiently or be optimized for applications. Among them, energy efficiency is the major objective. The second objective is to improve multimedia transmission quality over WSNs. The error-prone wireless channel causes packet loss and degrades multimedia transmission quality. The cross-layer adaptive strategy of multimedia delivery must be designed to prevent packet loss through efficient resource allocation. The third objective is to enhance security. There is a critical need to provide privacy and security assurances for distributed multimedia sensor networking in applications such as military surveillance and healthcare monitoring. In order to achieve these three objectives, we believe that the cross-layer design provides a promising solution by incorporating the characteristics of multimedia and information security at the application layer into adaptation and optimization at lower layers.

References

1. Polastre, J., Hui, J., Levi, P., Zhao, J., Culler, D., Shenker, S., Stoica, I.: A unifying link abstraction for wireless sensor networks. In: Proc. third International Conference on Embedded Networked Sensor Systems (Sensys), San Diego, CA (2005)
2. Hoesel, L.V., Nieberg, T., Wu, J., Havinga, P.J.M.: Prolonging the lifetime of wireless sensor networks by cross-layer interaction. IEEE Wireless Communications Magazine, 78–86 (2004)
3. Cui, S., Madan, R., Goldsmith, A.J., Lall, S.: Joint routing MAC and link layer optimization in sensor networks with energy constraints. In: Proc. IEEE ICC, pp. 725–729 (2005)
4. Akyildiz, I.F., Vuran, M.C., Akan, O.B.: A cross-layer protocol for wireless sensor networks. In: Proc. the Conference on Information Science and Systems, CISS (2006)
5. Chiang, M.: Balancing transport and physical layers in wireless multihop networks: jointly optimal congestion control and power control. IEEE Journal on Selected Area in Communications (JSAC) 23, 104–116 (2005)
6. Madan, R., Cui, S., Lall, S., Goldsmith, A.: Cross-layer design for lifetime maximization interference-limited wireless sensor networks. IEEE Communications on Wireless Communications 5, 3142–3152 (2005)
7. Song, L., Hatzinakos, D.: A Cross-layer Architecture of Wireless Sensor Networks for Target Tracking. IEEE/ACM Trans. on Networking 15(1), 145–158 (2007)
8. Wang, H., Peng, D., Wang, W., Sharif, H., Chen, H.H.: Cross-layer Routing Optimization in Multirate Wireless Sensor Networks for Distributed Source Coding based Applications. IEEE Transactions on Wireless Communications (TWC) 7(10) (2008)
9. Wang, W., Peng, D., Wang, H., Sharif, H., Chen, H.H.: Cross-layer Multirate Interaction with Distributed Source Coding in Wireless Sensor Networks. IEEE Transaction on Wireless Communication 8(2), 787–795 (2009)
10. Wang, H., Peng, D., Wang, W., Sharif, H., Chen, H.H.: Image Transmission with Security Enhancement Based on Region and Path Diversity in Wireless Sensor Networks. IEEE Transactions on Wireless Communications (TWC) 8(2), 757–765 (2009)
11. Wang, W., Peng, D., Wang, H., Sharif, H., Chen, H.H.: Energy-Constrained Distortion Reduction Optimization for Image Transmission in Wireless Sensor Networks. IEEE Transaction on Multimedia 10(6) (October 2008)
12. van der Schaar, M., Krishnamachari, S., Choi, S., Xu, X.: Adaptive cross-layer protection strategies for robust scalable video transmission over 802.11 WLANs. IEEE J. Sel. Areas Commun. 21(10), 1752–1763 (2003)
13. van der Schaar, M., Shankar, S.: Cross-layer wireless multimedia transmission: challenges, principles and new paradigms. IEEE Wireless Commun. Mag. 12(4), 50–58 (2005)

14. Li, Q., van der Schaar, M.: Providing Adaptive QoS to Layered Video over Wireless Local Area Networks through Real-Time Retry Limit Adaptation. IEEE Trans. Multimedia (March 2004)
15. van der Schaar, M., Turaga, D.: Content-Gased Cross-Layer Packetization and Retransmission strategies for Wireless Multimedia Transmission. IEEE Trans. Multimedia 9(1) (2007)
16. Wu, D., Ci, S., Wang, H.: Cross-layer optimization for video summary transmission over wireless networks. IEEE Journal of Selected Areas of Communications (JSAC) 25(4), 841–850 (2007)
17. Srivastava, V., Motani, M.: Cross-layer design: a survey and the road ahead. IEEE Communications Magazine 43(12), 112–119 (2005)
18. Wang, H., Peng, D., Wang, W., Sharif, H., Chen, H.H.: Resource-Efficient Watermarking for Image Authentication in Wireless Multimedia Sensor Networks. IEEE Transactions on Wireless Communications, TWC (2009) (submitted to)
19. Chong, C., Kumar, S.: Sensor networks: Evolution, opportunities, and challenges. Proc. IEEE 91(8), 1247–1256 (2003)
20. Back, S., Veciana, G., Xun-Su.: Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation. IEEE J. Sel. Areas Commun. 22(6), 1130–1140 (2004)
21. Pattern, S., Krishnamachari, B., Govindan, R.: The impact of spatial correlation on routing with compression in wireless sensor networks. In: Proc.3rd Int. Symp. Information Processing in Sensor Networks, pp. 28–35 (2004)
22. Petrovic, D., Shah, R., Ramchandran, K., Rabaey, J.: Data funneling: Routing with aggregation and compression for wireless sensor networks. In: Proc. IEEE Int. Workshop Sensor Network Protocols and Applications, pp. 156–162 (2003)
23. Krishnamachari, L., Estrin, D., Wicker, S.: The impact of data aggregation in wireless sensor networks. In: Proc. 22nd Int. Conf. Distributed Computing Systems, pp. 575–578 (2002)
24. Ramchandran, K.: Distributed signal processing: New opportunities and challenges. In: IEEE Workshop on Statistical Signal Processing, vol. 4 (2003)
25. Pradhan, S., Kusuma, J., Ramchandran, K.: Distributed compression in a dense microsensor network. IEEE Signal Process. Mag. 19(2), 51–60 (2002)
26. Wagner, R., Nowak, R., Baraniuk, R.: Distributed image compressionfor sensor networks using correspondence analysis and super-resolution. In: Proc. Int. Conf. Image Processing, vol. 1 (2003)
27. Xiong, Z., Liveris, A., Cheng, S.: Distributed source coding for sensor networks. IEEE Signal Process. Mag. 21(5), 80–94 (2004)
28. Tang, C., Raghavendra, C., Prasanna, V.: An energy efficient adaptive distributed source coding scheme in wireless sensor networks. In: Proc. IEEE Int. Conf. Communications, vol. 1, pp. 732–737 (2003)

A Survey on Cognitive Radio Networks

Jingfang Huang[1], Honggang Wang[2], and Hong Liu[3]

[1] University of Massachusetts, Dartmouth, MA, USA
jhuang@umassd.edu
[2] University of Massachusetts, Dartmouth, MA, USA
hwang1@umassd.edu
[3] University of Massachusetts, Dartmouth, MA, USA
hliu@umassd.edu

Abstract. The limitation of the spectrum bands is a major bottleneck for the development of next generation wireless networks. Cognitive Radio (CR) aims at improving the spectrum utilization by taking advantage of licensed but currently unused spectrum. CR has broad applications including dynamic spectrum access and interference management, which will largely impact the next generation of wireless devices and networks. In this paper, we conducted a survey on CR networks from various aspects such as waveform, spectrum management and sensing, testbeds, performance evaluations and etc.

Keywords: Cognitive Radio, Primary User, Secondary User, Waveform, Testbed.

1 Introduction

When spectrum has become a scarce resource, using of existing spectrum efficiently is critical. Cognitive Radio (CR), a software system, enables unlicensed users to utilize allocated spectrum for licensed users when the spectrum is temporarily unused. It should be noted that, CR has two most important characteristics [1]: 1) Cognitive ability: through constant interaction with the environment, CRs are able to figure out the portion of spectrums which are currently unused. Consequently, CRs can decide the best spectrum (spectrum selection) to utilize, hence share it with other CRs, and exploit this spectrum without interference on primary users; 2) Reconfigurability: CRs should transmit and receive on different frequency bands in order to choose the best spectrum band and most appropriate parameters. Further, CR networks can access unlicensed as well as licensed but currently abandoned spectrum, which can be concluded into two main processes of CR networks [1]:

1. Licensed band operation: Since licensed band are primarily used by Primary Users (PUs), the main job of CRs focus on the detection of reappearance of PUs. As soon as a reappearance of PU is detected, CR must evacuate this spectrum band and leave immediately This process is called channel mobility.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 487–498, 2010.

2. Unlicensed band operation: For a certain free spectrum abandoned by PU, all CRs have the same rights to access it. Consequently, effective spectrum sharing algorithms are of primary importance for CR networks to develop.

The rest of this paper is organized as follows: new waveforms aimed at improving the transmitting efficiency and throughput are surveyed in section 2. Section 3 introduces new algorithms for spectrum sensing/ sharing, channel allocation/ selection. Cross-layer and Media Access Control (MAC) protocols are studied with the cooperative communication in section 4. Moreover, to test new algorithms or new CR systems, testbeds suitable for different environments are introduced in section 5. Further, security aspects of CR are discussed in section 6. Finally, we studied performance and reliability of CR in section 7.

2 Waveforms for Cognitive Radio

The most important job of the Cognitive Radio (CR) is to efficiently use spectrum hole which is assigned to a primary user (PU). In order to achieve this goal, CRs have to detect the reappearance of PU frequently. They should quit the spectrum immediately as soon as a PU is detected in order to minimize their reciprocal interference. This suggests that CR has to change its transmitting waveform and adapt to the spectrum environment. Therefore, the adaptive waveform [2] techniques have been investigated.

The term adaptive waveform stands for "a time domain pulse in the radio frequency (RF) range that has the desired frequency response" [2]. In this technique, CRs will periodically monitor the RF spectrum (spectrum sensing) and choose the best available spectrum (spectrum decision). On basis of the spectrum information obtained, CRs generate an adaptive carrier waveform which fits only the free band. As soon as the waveform is generated, digital data will be modulated using this waveform and transmitted. Figure 1 shows the process of the adaptive waveform generation.

Obviously, how to decide and select a waveform for transmitting based on environmental measures is one of the most important problems for CR. A new on adaptive carrier waveform scheme is proposed in [3] to adapt to any band without bringing about harmful interference. It is useful in accessing TV spectrum with high spectrum utilization efficiency. In addition, a pulse generation

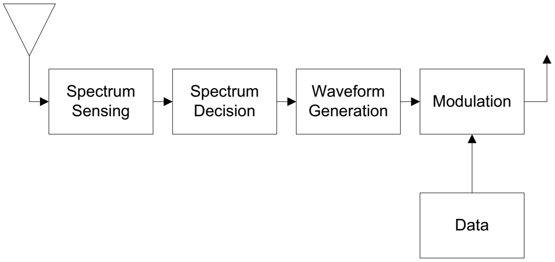


Fig. 1. Adaptive Waveform Generation [2]

method is introduced [4]. This time domain pulse is generated according to required frequency response using band pass filter. In [5], a hybrid overlay/underlay waveform generation is proposed to minimize the interference of secondary user on primary user. This method is aimed at exploiting both unused (white) and underused (gray) spectrum resources. In Ultra-wideband (UWB) systems there are two main kinds of waveforms: pulsed and chirp waveforms. A modified chirp waveform [6] is proposed to avoid the spectrum authorized to other existing systems.

Additionally, statistical knowledge of Primary User (PU) can be exploited in Spectrally Modulated, Spectrally Encoded (SMSE) waveform designs to maximize system throughput. Based on this knowledge, the authors in [7] proposed a SMSE algorithm using parametric variation in both waveform update latency and update rate. The benefit of this algorithm is apparent when using moderate value of latency and update rate. However, its performance will degrade when large latency value or high update rate are adopted. Further research of cognitive radio waveform will mainly focus on obtaining various carrier waveforms based on different pulses and developing the optimal design for adaptive transmissions.

3 Algorithms for Spectrum Management

The great challenges in CR networks are the interference issues due to CRs' coexistence with primary users. In addition, CR networks must provide seamless communication regardless of the reappearance of primary users [1]. These challenges can be addressed by spectrum management techniques, which mainly include four components: spectrum sensing, spectrum decision (spectrum allocation), spectrum sharing, and spectrum mobility. In this paper, we mainly investigate the process of spectrum sensing, spectrum decision and spectrum sharing. The applications of CRs in Multi-Input Multi-Output (MIMO) systems are introduced.

3.1 Spectrum Sensing

Particularly, as cognitive radio is designed to be sensitive to the changing environment, spectrum sensing becomes an important requirement for CRs. Generally speaking, spectrum sensing process includes obtaining the spectrum usage characteristics across multiple dimensions such as time, space, frequency, code, as well as determining what types of signals are occupying the spectrum [8]. Spectrum sensing techniques consist of primary transmitter detection, primary receiver detection and interference temperature management, which are illustrated in figure 2.

In [9], bandwidth problem of reporting channel during spectrum sensing is addressed. Contrasting to traditional sensing algorithms, this new method only allows cognitive users who have the highest performance to report in the absence of reliable cognitive user, which is proved to have better sensing performance.

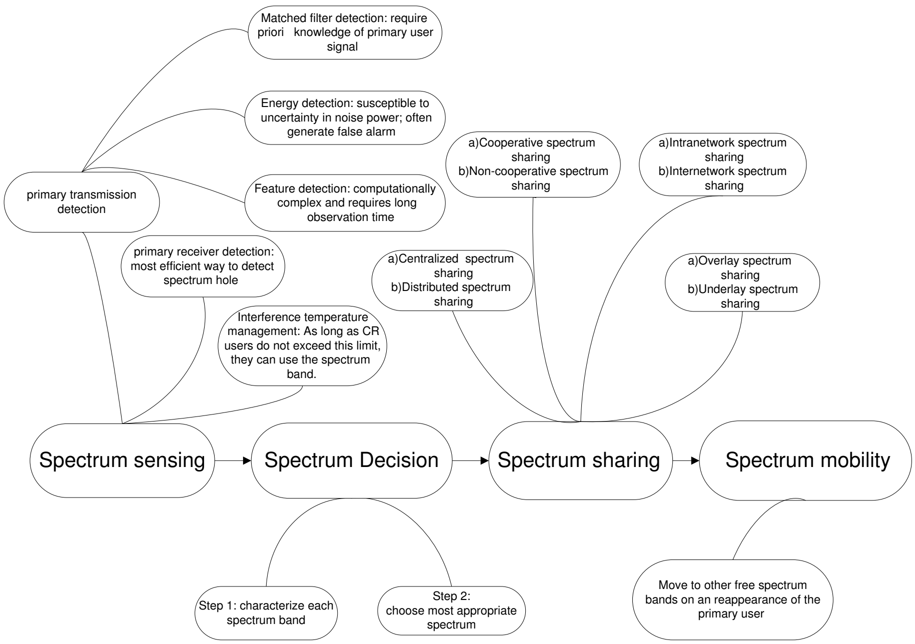

Matched filter detection: require priori knowledge of primary user signal
Energy detection: susceptible to uncertainty in noise power; often generate false alarm
Feature detection: computationally complex and requires long observation time
primary transmission detection
primary receiver detection: most efficient way to detect spectrum hole
Interference temperature management: As long as CR users do not exceed this limit, they can use the spectrum band.
a)Cooperative spectrum sharing b)Non-cooperative spectrum sharing
a)Intranetwork spectrum sharing b)Internetwork spectrum sharing
a)Centralized spectrum sharing b)Distributed spectrum sharing
a)Overlay spectrum sharing b)Underlay spectrum sharing
Spectrum sensing
Spectrum Decision
Spectrum sharing
Spectrum mobility
Step 1: characterize each spectrum band
Step 2: choose most appropriate spectrum
Move to other free spectrum bands on an reappearance of the primary user

Fig. 2. Spectrum Management Process

3.2 Spectrum Decision and Spectrum Sharing

As shown in Figure 2, after sensing and choosing the best spectrum for CRs, the next step is to decide which CR should get the access right of this spectrum. The coexistence of primary users and secondary users makes the process of spectrum sharing more difficult. Recent research aiming at meeting this challenge can be classified by four aspects: the architecture, spectrum allocation behavior, spectrum access technique, and scope [1].

In cognitive radio networks, priority mechanism is adopted for CRs to share spectrum resource called channel allocation [11]. Many algorithms such as game theory and reinforcement learning are developed and proved to be effective for the channel allocation.

Game theory [12] is a mathematical tool that can help to solve the competitive situations among all the players and enable rational decision makers to choose their actions based on their interests. In [1], non-cooperative game theory is applied to address the waveform adaption . In [13] a game-theoretic based adaptive channel allocation scheme is proposed for cognitive radio networks. This scheme provides a natural framework for the design and analysis of noncooperative behavior.

Further, as allocated spectrum is not always occupied by licensed users, by predicting the leaving time and duration of licensed users from this spectrum [14], CRs can utilize this free spectrum and improve spectrum efficiency. In

[15], the authors introduces a novel predictive channel allocation scheme called Intelligent Channel Allocation (ICA), which is based on long term call statistics, instantaneous call statistics and event driven decisions for a fair utilization of spectrum. A reinforcement learning algorithm is proposed to learn the spectrum usage by interacting with the environment [16]. In [17], a Q-learning module is studied , in which CR users can choose their task through interacting with the environment and self-learning and consequently improve sensing efficiency. A improved algorithm is further studied in [18], in which a learning algorithm is combined with a reasoning engine to enable CRs to remember information learnt in the past and react quickly in the future.

In particular, secondary users need to get the position information of primary users for guidance. A robust distributed localization algorithm that enables secondary users to obtain the location information of primary users is proposed in [19]. However, traditional dynamic channel assignment algorithm cannot be directly used on CRs without modification. Thus in [11], the authors proposed methods of channel assignment and channel reallocation that are suitable for cognitive radio systems. In [10], both algorithms and spectrum access protocols are developed to control the dynamic allocation of spectrum resources between cooperating networks.

3.3 Application of CR in MIMO System

Recent research has emphasized on the utilization of cognitive radio on MIMO systems, in which each CR has Multiple-Input and Multiple-Output antennas. MIMO system has the advantages in sensing environment information because multiple antennas energy detector is more efficient when sensing primary users' signal. The selection mechanism of receiving antenna is proposed [2] to further reduce the computational and feedback complexity of the CR networks in a MIMO system. In addition, multiple antennas can be used to a) improve SNR, b) provide diversity, c) introduce an extra signaling dimension (i.e., spacial multiplexing), and d) mitigate interference [21].

4 Protocols for Cooperative Communication

In order to solve the problem of spectrum scarcity, new protocols are required to realize better spectrum access and high spectrum efficiency. The authors in [22] have done a survey on MAC protocols for cognitive radio, which mainly focuses on channel selection and channel sensing policies. Recently, new MAC protocols and cross-layer designed protocols have been studied by researchers to achieve better CR performance.

In [23], the authors propose a leasing oriented MAC protocol, in which secondary users are divided into several groups and each group bid for the right to use the spectrum occupied by a primary user who is going to leave it. This protocol can promise the fairness and dynamic allocation of spectrum resources. Moreover, in [24] the authors develope an adaptive MAC protocol which enhances the throughput of CR. The protocol allows cognitive channels to change

Table 1. Advantages of new CR protocols

New Protocols	Models Adopted	Merits of the proposed protocol
A Leasing Oriented MAC Protocol	Property-Rights Model	1. Maximizing the utilization of spectrum resources; 2. Achieving revenue maximization for primary users; 3. Allocating channels among groups fairly and evenly; 4. If a group is allocated channels, the group's minimum bandwidth requirement must be satisfied
A Decentralized Adaptive Medium Access Control (AMAC) Protocol	Cognitive Radio Networks (CRNs)	1. Has no dedicated global common control channel (CCC), Solve potential bottleneck problem(CCC); 2. Aggregated throughput is higher even in poor channel condition; 3. Solves the multichannel hidden terminal problem
Cognitive Radio-Enabled Multi-channel MAC (CREAM-MAC) Protocol	Wireless Ad Hoc Networks	1. Integrates the spectrum sensing at physical layer and packet scheduling at MAC layer; 2. Solve both the traditional and multi-channel hidden terminal problems by introducing the four-way handshakes of control packets over the control channel
Protocol That Combats The Hidden Incumbent Problem	Satellite Assisted Cognitive Radio Networks	1. Use of satellites in a cognitive radio setting is quite beneficial in addressing the yet unresolved problems in cognitive radio networks; 2. Avoids the hidden node problem while taking the mobility pattern into consideration

from transmitting mode to frame recovery mode when there is frame errors. It ultimately increases the throughput of the CR channels. Another method to improve the system throughput is proposed in [25]. This protocol decentralizes the system from dedicated common control channel (CCC) and every CR will have a table and index itself. In the method, the most stable channel will become CCC and thus available resource utilization is improved. Further, the challenges such as multichannel hidden terminal problem and timevarying channel availability are addressed by [26]. A cognitive radio-enabled multi-channel MAC (CREAM-MAC) Protocol is proposed to integrate spectrum sensing at physical layer and packet scheduling at MAC layer. The spectrum efficiency is improved by enabling each secondary user with a transceiver and a multiple channel sensor because it helps avoiding the collisions between primary users and secondary users as well as collisions among secondary users. In [27], the authors propose a handover protocol to address a hidden node problem and discover the mobility pattern of both primary and secondary users in satellite assisted cognitive radio networks. In [28], the author exploits a new direction for spectrum allocation called cooperative relay, in which the secondary user can performs as common channel to obtain information of free spectrum and assists transmission.

The reconfiguration of the radio is another challenge in cognitive radio networks. In [29], the author addressed this problem by proposing a radio-independent authentication protocol for CRs which used user-specific information, such as location information, as a key seed. This protocol is dependent of underlying radio protocols and could support EAP (Extensible Authentication Protocol) transport. In addition, a cognitive tree-based routing protocol for (CTBR) cognitive wireless access networks [30] is proposed to adapt the protocol to support multiple systems. This CTBR protocol is then proved to be more effective than the known TBR protocol. The comparison of these new protocols is shown in table 1.

5 Testbed Suitable for Different Environment

A testbed is a platform for evaluation of software, hardware and networking components in Cognitive Radio Networks. Most of existing wireless research uses simulations as its major validation technique. However, the simulations for cognitive radio techniques may not be convincing in some situations [31]. Accordingly, many different research testbeds for cognitive radio are proposed and developed.

In [32], the authors develop a cognitive UWB testbed, which generates an adaptive pulse applicable for different spectrum environment. Moreover, a new cooperative relay for resource allocation in cognitive radio networks [28] is studied based on this testbed.

A specific testbed is developed for ad-hoc cognitive radio network [33]. This testbed is designed for the crosslayer configuration and performance optimization which includes adaptive MAC layer and network layer as well as cross layer management interface. The testbed is developed for verification of new algorithms in Cognitive radio networks. In [34], the author provides a systematic testbed model to verify effectiveness of new algorithms such as Genetic Algorithm (GA) for channel selection. It demonstrated that primary and secondary users can coexist in a spectrum sharing manner.

It is worth noticing that cross-layer testbed design is addressed in [35] for spectrum sensing and interference analysis. This testbed is able to achieve the following three points: (a) cognitive radio system concept demonstration; (b) multi-resolution spectrum sensing (MRSS) receiver IC evaluation; (c) interference analysis for UWB coexistence with WiMax.

6 Security Consideration

The security concerns become critical in cognitive radio networks because a selfsh secondary user can modify its air interface to mimic a primary user for occupation of the spectrum and can mislead the spectrum sensing performed by primary users. Security threats and attacks against cognitive radio networks includes Denial of Service, selfish misbehaviours, licensed user emulation, attacks on spectrum managers and eavesdropping [36].

There are currently two major methods which address the security issues of cognitive radio. One is to identify an attacker by using position of the transmitter. The other one is to prevent secondary users from mimic of primary users [8] by using public key encryption based primary user identification. In the latter method, primary users are required to transmit encrypted values (signatures) along with their information. This legitimate primary user is then recognized by this signature. However, a disadvantage is that this approach requires all the CRs to have the mechanism of encrypting and decoding system.

Further, a new concept concerning the security of cognitive radio is proposed in [36] to allow two cognitive radio nodes to authenticate each other before conducting any confidential channel communication.

7 Study on Performance and Reliability of the Cognitive Radio

As CR will change its objective spectrum according to the change of primary users' status, it is difficult to obtain a firm understanding of the relationship between the primary and secondary users. In [38], the author suggests that the performance metrics of CRs should include spectrum utilization, impact to other SU nodes or incumbent ratios, power efficiency, communication cost for end users, as well as link reliability. Recently, many researches address the performance evaluation of cognitive radio networks. A study on the spectral efficiency of adaptive modulation is conducted in [39]. This study mainly focuses on a cross-layer combination of adaptive modulation and proves that cross-layer design could improve the performance of cognitive radio systems. A new algorithm is proposed for maximizing throughput of cognitive radio networks in [4], and it requires minimal interaction between primary and secondary users. Further, a queuing analytic framework is developed [41] in order to allocate available spectrum in a spectrum overlay scenario. The authors present a step-by-step procedure to derive key parameters for facilitating cross-layer design and improving QoS in CR networks. What's more, reliability of cognitive radio networks is addressed in [42], and effect of parameters (such as number of channels, radios, and simultaneous flows) on reliability of a CR is analyzed. This method can find optimal routes on basis of reliability and tune network parameters in order to improve performance.

To improve the performance of cognitive radio networks, two directions are promising. The first one is to increase the cooperative behavior of CR. With cooperative cognitive radio, information obtained from observation stage and knowledge benefited from a learning stage can be communicated and shared among CRs. The second is to conduct spectrum sensing, sharing and decision making processes from different layers. With the changed information and decision made among layers, better performance improvement can be obtained.

8 Conclusion

With the development of wireless network, spectrum resource is becoming more and more precious. Cognitive radio (CR) can largely improve the utilization of licensed spectrum.

In this paper, waveform designs which help obtain better spectrum utilization are introduced; New algorithms concerning spectrum sensing and channel allocation are investigated; Protocol designs which address cross-layer cooperation are given out; New designs of testbed for simulation and verification of new algorithms and CR designs are explained; Also, security of CR wireless network is given consideration and performance of CR systems is evaluated. CR is also facing cooperative and cross-layer communication challenges [10]. Since wireless communication is conducted in different layers, problems will arise from exclusion of information helpful for communication process. Consequently, both cross-layer design and cross-layer protocols are in urgent requirement.

The future research of CRs will include improving cooperation of different stages among different layers with a MIMO system to improve the performance and spectrum utilization of Cognitive Radio system. We anticipate that this article will provide better understanding of CRs and foster the research in the Cognitive research field.

References

1. Akyildiz, I.F., Lee, W.-Y., Vuran, M.C., Mohanty, S.: A Survey on Spectrum Management in Cognitive Radio Networks. IEEE Communications Magazine, 40–48 (2008)
2. Buzzi, S., Poor, H., Saturnino, D.: Noncooperative Waveform Adaptation Games in Multiuser Wireless Communications. IEEE Signal Processing Magazine 26(5), 64–76 (2009)
3. Mathew, M., Premkumar, A.B., Lau, C.T.: Pulse Based Adaptive Carrier Waveform Generation for Cognitive Radio Applications. In: Mathew, M. (ed.) 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CrownCom 2008, pp. 1–6 (2008)
4. Manju, M., Premkumar, A.B., Lau, C.T.: An Adaptive Waveform Generation Technique for Cognitive Radio. In: Manju, M. (ed.) Vehicular Technology Conference VTC Spring 2008, pp. 1291–1295. IEEE, Los Alamitos (2008)

5. Chakravarthy, V., Li, X., Zhou, R., Wu, Z., Temple, M.: A novel hybrid overlay_underlay Cognitive Radio waveform in frequency selective fading channels. In: 4th International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CROWNCOM 2009, pp. 1–6 (2009)
6. Shen, H., Zhang, W., Kwak, K.S.: Modified Chirp Waveforms in Cognitive UWB System. In: IEEE International Conference on Communications Workshops, ICC 2008, pp. 504–507 (2008)
7. Like, E.C., Temple, M.A.: Coexistent Intra-Symbol SMSE Waveform Design: Variation in Waveform Update Latency and Update Rate. In: 4th International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CROWNCOM 2009, pp. 1–7 (2009)
8. Yucek, T., Arslan, H.: A survey of spectrum sensing algorithms for cognitive radio applications. IEEE Communications Surveys & Tutorials 11(1), 116–130 (2009)
9. Zhang, L., Xia, S.: A new cooperative spectrum sensing algorithm for cognitive radio networks. In: ISECS International Colloquium on Computing, Communication, Control, and Management, CCCM 2009, vol. 1, pp. 107–110 (2009)
10. Harrold, T.J., Wang, L.F., Beach, M.A., Salami, G.: Spectrum Sharing and Cognitive Radio. In: Harrold, T.J. (ed.) International Conference on Ultra Modern Telecommunications & Workshops, ICUMT 2009, pp. 1–8 (2009)
11. Hiraga, K., Akabane, K., Shiba, H., Uehara, K.: Channel Assignment and Reallocation Algorithms for Cognitive Radio Systems. In: 14th Asia-Pacific Conference on Communications, APCC 2008, pp. 1–4 (2008)
12. Niyato, D., Hossain, E.: Cognitive Radio for Next-Generation Wireless Networks: An Approach to Opportunistic Channel Selection. IEEE Wireless Communications 16(1), 46–54 (2009)
13. Nie, N., Comaniciu, C.: Adaptive Channel Allocation Spectrum Etiquette for Cognitive Radio Networks. In: Proc. IEEE DySPAN 2005, pp. 269–78 (November 2005)
14. Akbar, I.A., Tranter, W.H.: Dynamic Spectrum Allocation in Cognitive Radio Using Hidden Markov Models: Poisson Distributed Case 2007. In: Proceedings of SoutheastCon 2007, pp. 196–201. IEEE, Los Alamitos (2007)
15. Choudhary, S., Mishra, S., Desai, N., Priya, N.S., Chudasama, D.: A fair cognitive Channel Allocation method for cellular networks. In: Second International Workshop on Cognitive Radio and Advanced Spectrum Management, CogART 2009, pp. 138–142 (2009)
16. Ge, F., Chen, Q., Wang, Y., Bostian, C.W., Rondeau, T.W.: Cognitive Radio: From Spectrum Sharing to Adaptive Learning and Reconfiguration. In: IEEE Aerospace Conference 2008, pp. 1–10 (2008)
17. Li, M., Xu, Y., Hu, J.: A Q-Learning based sensing task selection scheme for cognitive radio networks. In: International Conference on Wireless Communications & Signal Processing, WCSP 2009, pp. 1–5 (2009)
18. Clancy, C., Hecker, J., Stuntebeck, E., O'Shea, T.: Applications of Machine Learning to Cognitive Radio Networks. IEEE Wireless Communications, 47–52 (2007)
19. Zheng, Y., Wan, L., Men, S.: A robust distributed localization algorithm for cognitive radio. In: 14th Asia-Pacific Conference on Communications, APCC 2008, pp. 1–4 (2008)
20. Hamdi, K., Zhang, W., Letaief, K.: Opportunistic Spectrum Sharing in Cognitive MIMO Wireless Networks. IEEE Transactions on Wireless Communications 8(8), 4098–4109 (2009)
21. MacKenzie, A.B., Reed, J.H., Athanas, P., Bostian, C.W., Buehrer, R.M.: Cognitive Radio and Networking Research at Virginia Tech. Proceedings of the IEEE 97(4), 660–688 (2009)

22. Wang, H., Qin, H., Zhu, L.: A Survey on MAC Protocols for Opportunistic Spectrum Access in Cognitive Radio Networks. In: International Conference on Computer Science and Software Engineering, vol. 1, pp. 214–218 (2008)
23. Song, H., Lin, X.: A Leasing Oriented MAC Protocol for High Spectrum Usage in Cognitive Radio Networks. In: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WIMOB 2009, pp. 173–178 (2009)
24. Lee, B., Lhee, S.H.: Adaptive MAC Protocol for Throughput Enhancement in Cognitive Radio Networks. In: International Conference on Information Networking, ICOIN 2008, pp. 1–5 (2008)
25. Joshi, G.P., Kim, S.W., Kim, B.-S.: An Efficient MAC Protocol for Improving the Network Throughput for Cognitive Radio Networks. In: Third International Conference on Next Generation Mobile Applications, Services and Technologies, NGMAST 2009, pp. 271–275 (2009)
26. Su, H., Zhang, X.: CREAM-MAC: An Effcient Cognitive Radio-EnAbled Multi-Channel MAC Protocol for Wireless Networks. In: International Symposium on World of Wireless, Mobile and Multimedia Networks, WoWMoM 2008, pp. 1–8 (2008)
27. Gozupek, D., Bayhan, S., Alagoz, F.: A novel handover protocol to prevent hidden node problem in satellite assisted cognitive radio networks. In: 3rd International Symposium on Wireless Pervasive Computing, ISWPC 2008, pp. 693–696 (2008)
28. Jia, J., Zhang, J., Zhang, Q.: Cooperative Relay for Cognitive Radio Networks. In: INFOCOM 2009, pp. 2012–2034. IEEE, Los Alamitos (2009)
29. Kuroda, M., Nomura, R., Trappe, W.: A Radio-independent Authentication Protocol (EAP-CRP) for Networks of Cognitive Radios. In: 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, SECON 2007, pp. 70–79 (2007)
30. Zhang, B., Takizawa, Y., Hasagawa, A., Yamaguchi, A., Obana, S.: Tree-based Routing Protocol for Cognitive Wireless Access Networks. In: Wireless Communications and Networking Conference, WCNC 2007, pp. 4204–4208. IEEE, Los Alamitos (2007)
31. Jia, J., Zhang, Q.: A Testbed Development Framework for Cognitive Radio Networks. In: IEEE International Conference on Communications, ICC 2009, pp. 1–5 (2009)
32. Choi, N.H., Hwang, J.H., Zheng, G., Han, N., Kim, J.M.: A Cognitive UWB Testbed Employing Adaptive Pulse Generation. In: 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CrownCom 2008, pp. 1–6 (2008)
33. Wang, S., Zheng, H.: A resource management design for cognitive radio ad hoc networks. In: Military Communications Conference, MILCOM 2009, pp. 1–7. IEEE, Los Alamitos (2009)
34. Kim, J.M., Sohn, S.H., Han, N., Zheng, G., Kim, Y.M.: Cognitive Radio Software Testbed using Dual Optimization in Genetic Algorithm. In: 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CrownCom 2008, pp. 1–6 (2008)
35. Park, J., Kim, K.-W., Song, T., Lee, S.M., Hur, J.: A Cross-layer Cognitive Radio Testbed for the Evaluation of Spectrum Sensing Receiver and Interference Analysis. In: 3rd International Conference on Cognitive Radio Oriented Wireless Networks and Communications, CrownCom 2008, pp. 1–6 (2008)

36. Safdar, G.A., O'Neill, M.: Common Control Channel Security Framework for Cognitive Radio Networks. In: IEEE 69th Vehicular Technology Conference, pp. 1–5. VTC Spring (2009)
37. Jesuale, J., Eydt, B.C.: A Policy Proposal to Enable Cognitive Radio for Public Safety and Industry in the Land Mobile Radio Bands. In: 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, DySPAN 2007, pp. 66–77 (2007)
38. Zhao, Y., Mao, S., Neel, J.O., Reed, J.H.: Performance Evaluation of Cognitive Radios: Metrics, Utility Functions, and Methodology. Proceedings of the IEEE 97(4), 642–659 (2009)
39. Foukalas, F.T., Karetsos, G.T.: A Study on the Performance of Adaptive Modulation and Cross-Layer Design in Cognitive Radio for Fading Channels. In: 13th Panhellenic Conference on Informatics, PCI 2009, pp. 158–162 (2009)
40. Hoang, A.T., Liang, Y., Islam, M.H.: Maximizing Throughput of Cognitive Radio Networks with Limited Primary Users' Cooperation. In: IEEE International Conference on Communications, ICC 2007, pp. 5177–5182 (2007)
41. Rashid, M., Hossain, M., Hossain, E., Bhargava, V.: Opportunistic Spectrum Scheduling for Multiuser Cognitive Radio: A Queueing Analysis. IEEE Transactions on Wireless Communications 8(10), 5259–5269 (2009)
42. Pal, R., Idris, D., Pasari, K., Prasad, N.: Characterizing Reliability in Cognitive Radio Networks. In: First International Symposium on Applied Sciences on Biomedical and Communication Technologies, ISABEL 2008, pp. 1–6 (2008)

A Comprehensive P-Persistent Algorithm in Multi-channel and Multi-interface Cognitive Network*

Junwei Lv, Weili Lei, and Tigang Jiang

School of Communication and Information Engineering,
University of Electronic Science and Technology of China, Chengdu, China
ljwsunny@sina.com

Abstract. In this paper, we design a multi-interface model on single mobile node and every node in the cognitive network is implemented multiple transceivers. *P*-persistent slotted CSMA is used as the access protocol. Interfaces with different channels may have different values of *p*s. To avoid crosstalk, multiple transceivers of each node must work in the same state, only sending or receiving, at one time. This requirement limits the performance of *p*-persistent algorithm. To solve this problem, we propose the Comprehensive *P*-Persistent Algorithm (CPPA), which is to find an optical value of *p* shared by one node's multiple interfaces. Simulation experiments are conducted to evaluate the performance of our algorithm in terms of the throughput and the jitter. Results show that our algorithm performs better.

Keywords: cognitive radio, OFDM, p-persistent slotted CSMA, multi-interface.

1 Introduction

The concept of cognitive radio (CR) was introduced to improve the frequency spectrum utilization in wireless networks. CR is able to do self-cognition, user-cognition and radio-cognition [1]. It is adapted for use of frequency without interference to primary user. Therefore, the secondary user in cognitive network should continuously perceive channels and master the channel occupancy of the primary user. Then it makes decision on which channel can be used. To meet this demand, in this paper, the mobile node in the cognitive network is designed with multiple interfaces, and every interface can work for different purposes independently.

OFDM (Orthogonal Frequency Division Multiplexing) is chosen as the modulation. OFDM may be viewed as using many slowly-modulated narrowband signals rather than one rapidly-modulated wideband signal. The low symbol rate makes the use of a guard interval between symbols affordable, making it possible to handle time-spreading and eliminate inter-symbol interference (ISI) [2]. It can support flexible selection of frequency, implement adaptive bandwidth allocation, and divide the

* This work is supported by the National Natural Science Foundation of China under Grant No.60802024, the 863 Project of China under Grant No.2009AA011801, the National S&T Major Project of China under Grant No. 2008ZX03006-001 and 2008ZX03003-005).

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 499–506, 2010.

channels into several narrow sub-channels. So that OFDM provides good implementation foundation of spectrum sensing for cognitive radio. Due to the Frame synchronization of OFDM, [3] shows that OFDM with slotted aloha does a better job than unslotted aloha, and [4] indicates that CSMA provides better performance than aloha. So we adopt *p*-persistent slotted CSMA as the access protocol for single interface. Moreover, all of the multiple transceivers of one node only can be either transmitters or receivers (ETOR) at one time. It is not admitted that some transceivers of one node are transmitting packets while the others are receiving packets. To meet this requirement, the performance of *p*-persistent slotted CSMA is limited. To solve this problem, in this paper, the Comprehensive *P*-Persistent Algorithm (CPPA) is put forward, which is to find the optimal value of *p* shared by the multiple interfaces of a mobile node.

The rest part of the paper is organized as follows: First, we describe the system model in Section II. Then we present the Comprehensive *P*-Persistent Algorithm (CPPA) in Section III. In Section IV, we analyze the performance of the CPPA according to simulation results.

2 System Model

In this paper, the mobile nodes work in the hybrid network including centralized and distributed network. In overlapping coverage area of the two kinds of networks, one or more nodes become the gateways, interconnecting the centralized and distributed network.

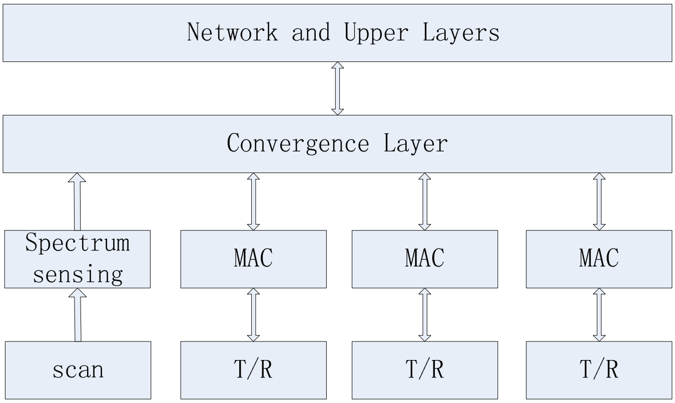


Fig. 1. The protocol stack of a mobile node

As described in Fig.1, each node has one PHY/MAC pair for spectrum sensing and three pairs for data communication. In the PHY, multiple transceivers work concurrently and independently. The channel device for spectrum sensing has to scan all channels continuously, sense and master the state of each channel, and provide the information for data channel devices to choose spectrum holes to work. The three data channel devices are independent with each other and have the same software and hardware structures. Moreover, the data channel devices work in the same range of frequency, so that one's working frequency is close to other ones. In order to avoid the crosstalk between different transceivers, the transceivers working roles of a single mobile node are confined to be either transmitters or receivers (ETOR) uniformly. For example, at *t*th time slot, if a transceiver is sending packets, the other two also have to be in sending state or shut down even if there is no packet to be sent.

The three MACs use *p*-persistent slotted CSMA as the access protocol separately. Each MAC entity is implemented with two working modes, one working in the centralized network and the other working in the distributed network. With the restriction of ETOR mode, given that the three MACs may determine different transmission probability *p*, the channel device that has lower probability can neither transmit nor receive data packets when the one that has higher probability is sending data packets. This may lower the channel utilization. The CPPA calculates the optimal value of transmission probability that can be used by three data channel devices commonly so that they can work under the same policy.

The convergence layer, where the CPPA locates, decides working modes of MAC entities and completes the spectrum resource allocation and synchronization management of the three data channels.

3 The Comprehensive P-Persistent Algorithm (CPPA)

In this paper, we assume that the channel is error free and there is no capture phenomenon. So the collision is the only reason of packet error or packet loss. The collision is divided into two kinds: a) Primary users arrive when secondary users are using the same channel. B) Two or more secondary users using the same channel to send packets simultaneously.

The CPPA is aimed to lessen the performance loss from the ETOR mode. So the optimal value of *p* is a number which can make the throughput difference between CPPA with ETOR and non-CPPA without ETOR as small as possible.

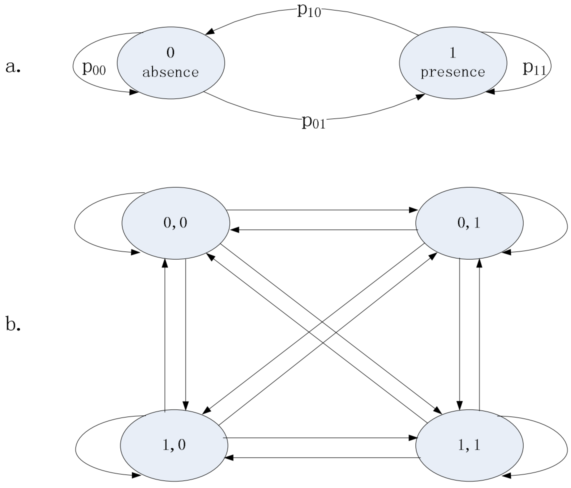


Fig. 2. State transition diagrams

In the environment of cognitive radio network, secondary user scans channels and selects a spectrum hole to send packets. According to [5], we use a multi-state Markov process to model this environment.

As illustrated in Fig.2.a, the primary user transits from state 0 (absence) to state 1 (presence) with probability p_{01} and stays in state 1 with probability p_{11}. Then the state is defined as (*Xt*, *Yt*). *Xt* denotes the real status of the primary users at time *t*, and *Yt* denotes the sensed status of the primary users by the secondary users. In addition, we define the probabilities of misdetection and false alarm as p_{md} and p_{fa}. Then we get the transition probability matrix of multi-state Markov process described in Fig.2.b as follows:

$$\begin{pmatrix} p_{00}(1-p_{fa}) & p_{00}p_{fa} & p_{01}p_{md} & p_{01}(1-p_{md}) \\ p_{00}(1-p_{fa}) & p_{00}p_{fa} & p_{01}p_{md} & p_{01}(1-p_{md}) \\ p_{10}(1-p_{fa}) & p_{10}p_{fa} & p_{11}p_{md} & p_{11}(1-p_{md}) \\ p_{10}(1-p_{fa}) & p_{10}p_{fa} & p_{11}p_{md} & p_{11}(1-p_{md}) \end{pmatrix}$$

The secondary user can only send packets in the states (0, 0) and (1, 0). When the state is (1, 0), the collision probability of the secondary users is 1, because this is a misdetection state. Based on the above transition probability matrix, we can determine the steady-state probability π_{00} and π_{10} which indicate the stationary probabilities of state (0, 0) and (1, 0) respectively.

We consider the case when the number of stations is much larger than that of sub-channels. Let *N* represents traffic load (the number of stations waiting for data transmission at *t*th slot) while there are *M* sub-channels. If *m* sub-channels are idle, with a probability *p* the station selects one idle sub-channel and transmits a packet; or with a probability $1-p$, the station defers the decision for transmission by one time slot. In sub-channel selection, the station randomly selects one among idle sub-channels.

We assume that the activity of each sub-channel is independent from each other. Let $M_{idle}(t)$ be the stochastic process representing the number of idle sub-channels among the $M-1$ sub-channels for the time slot index *t*. We can get the probability equation (1) that *k* stations transmit their packets in the sub-channel 1, conditioned on that *m* + 1 sub-channels including sub-channel 1 are idle [4].

$$\Pr(A_1 = k \mid M_{idle} = m) \equiv \sum\nolimits_{a=k}^{\infty} \Pr(A(m+1) = a) \bullet \begin{pmatrix} a \\ k \end{pmatrix} \frac{m^{a-k}}{(m+1)^a} \qquad (1)$$

$$\Pr(A(m) = k) = \frac{(Np)^k}{k!} e^{-Np}, k = 0, 1, \cdots \qquad (2)$$

When calculating the comprehensive *p*, we consider the channel occupancy of all nodes in the network is similar. Let *m* be a constant. We denote the steady-state probabilities by P_i, P_s, and P_c that the Markov process is in the idle, success, and collision states, respectively. According to [4], we can get the equations as follows:

$$P_s = P_i \cdot \pi_{00} \cdot \Pr(M_{idle} = m)\Pr(A_1 = 1 \mid M_{idle} = m), \tag{3}$$

$$P_c = P_i \cdot \pi_{00} \cdot \{\Pr(M_{idle} = m) \cdot \sum\nolimits_{k=2}^{\infty} \Pr(A_1 = k \mid M_{idle} = m)\} + \pi_{10}, \tag{4}$$

$$P_i = 1 - P_s - P_c \tag{5}$$

The probability that the transmission on a sub-channel finishes at a certain slot is given by $\frac{\sigma}{\mathrm{LM}}$ for the packet transmission time *LM* and the slot duration *σ* [6]. The saturated throughput of a sub-channel is

$$S = \frac{P_s LM}{P_i \sigma + (P_s + P_c)LM} \tag{6}$$

Substituting the expressions (6) obtained for P_s (3), P_c (4) and P_i (5), we have the finally established (7).

$$S = \frac{L \cdot M \cdot \Pr(M_{idle} = m) \cdot (1 - \pi_{10}) \cdot \pi_{00} \cdot e^{-\frac{Np}{m+1}} \cdot \frac{Np}{m+1}}{(1 - \pi_{10}) \cdot \sigma + L \cdot M \cdot \pi_{10}}, \tag{7}$$

Using equation (7), we can get the value of the comprehensive *p* in theoretically. *P*-persistent algorithm is researched in many articles [7]-[9], and in this paper, it does not be discussed.

Supposed at *t*th time slot, the transmission probabilities of the three channels are p_1, p_2, p_3, and the throughputs are S_1, S_2, S_3. The throughput of one interface with transmission probability *p* is *S*. *F(p)* is defined as below:

$$F(p) = S_1 + S_2 + S_3 - 3S$$

When *F(p)*=0, *p* is the optimal value, thus we can get the formula as below:

$$e^p \cdot p = \frac{1}{3}(e^{p_1} \cdot p_1 + e^{p_2} \cdot p_2 + e^{p_3} \cdot p_3) \tag{8}$$

4 Simulation and Analysis

First, we analyze the influence of the primary user on the throughput of the secondary user using the equation (7) we proposed.

To simplify the simulation, we assume that the number of channels is the same as that of the interfaces of the mobile node. For the parameters, we set the slot duration to 250μs; the packet transmission time, *L*, to 2.5 ms; misdetection, p_{md}, to 0.2; false alarm, p_{fa}, to 0.2; and the transmission probability, *p*, to 0.1.

The saturated throughput of a sub-channel for p_{00} = 0.1, p_{00}=0.5 and p_{00} = 0.9 are plotted in Fig.3. p_{01} is set as the same as p_{00}. From the Fig.3, we notice that the higher the probability of absence is, the higher the throughput of the secondary user is, which is in line with the realities.

In order to characterize the feature of the algorithm accurately, we compare the performance of CPPA to that of non-CPPA by simulation. We simulated a system of 50 nodes on a 500×500 grid. The nodes could move in all possible directions with displacement varying uniformly between 0 to 5, per unit time. Each node had 3 interfaces, and transceivers worked in ETOR mode as mentioned above. The simulation started at $t = 0s$, and data flows increased as time went on.

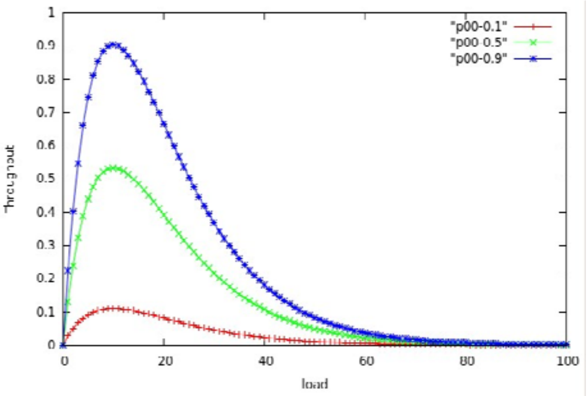


Fig. 3. The throughputs under different p00s

Fig. 4 and Fig. 5 show the performances of a data flow at $t = 19s$. p.out is the throughput with CPPA and 3p.out is that without CPPA. As shown in the figure, the throughput of CPPA is greater than that of non-CPPA. With the same transmission probability, the chance that three interfaces of one node are in the same state is larger, so that the channel utilization is improved. This will make throughput higher.

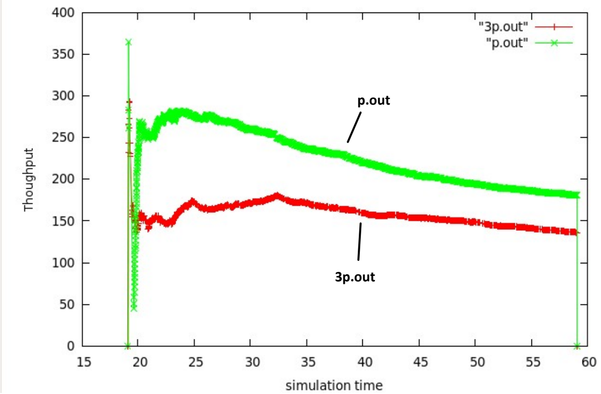


Fig. 4. The comparison of throughputs

But in the long haul, 3p.out curve is more stable, and p.out curve declines while time goes on. The reason is that each interface decides its own p according to its using channel condition, so as to keep its performance at a good level. With the data flows increasing, the state difference between channels may be greater. At this time, the way of using the common policy may not be suitable. The probability p calculated by CPPA is a compromise value, which may make a channel in good condition has a low probability p, and one in bad condition has a high probability p. It needs more researches on that in what range of the difference between the minimum and maximum of p_i (i is the number label of a interface), the CPPA is more efficient. Fig. 5 is the comparison of jitters. Obviously, the difference is not very significant.

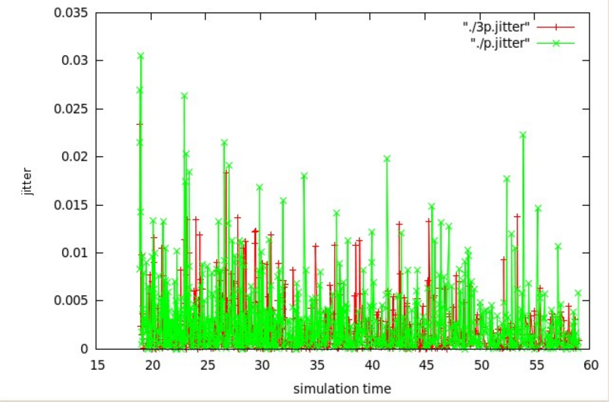


Fig. 5. The comparison of jitter

Future work is needed to determine that under what conditions the CPPA perform well. As mentioned above, the throughput degrades when the channel states among interfaces are very different. A threshold value should be calculated to guarantee the efficiency of the CPPA. An adjustment algorithm should be applied when the maximum difference goes up to threshold. For example, we choose the minimal p_i among the three interfaces as the comprehensive p to ease the network collision, and choose the p calculated by (8) when the maximum difference is less than threshold.

5 Conclusion

We propose the comprehensive p-persistent algorithm (CPPA) which is an optimization of p-persistent algorithm when the multiple transceivers of a mobile node in the multi-channel and multi-interface cognitive network work in the ETOR mode. CPPA can minimize the performance loss and improve the channel utilization, so that the throughput of the network can be enhanced by CPPA. Furthermore, the simulation result shows the algorithm need further researches to perform better.

References

1. Haykin, S.: Cognitive Radio: Brain-Empowered Wireless Communications. IEEE JSAC 23(2), 201–220 (2005)
2. Keller, T., Hanzo, L.: Adaptive multicarrier modulation: A convenient framework for time-frequency processing in wireless communications. Proc. IEEE 88, 611–640 (2000)

3. Chung, J.-M.: OFDM frame synchronization in slotted aloha mobile communication systems. In: Proc. IEEE Vehicular Technology Conference 2001, vol. 3, pp. 1373–1377 (2001)
4. Kwon, H.: Generalized CSMA/CA for OFDMA systems: protocol design, throughput analysis, and implementation issues. IEEE Trans. Wireless commun. 8, 4176–4187 (2009)
5. Jeon, S.-Y.: An ARQ mechanism considering resource and traffic priorities in cognitive radio systems. IEEE Communication Letters 13, 504–506 (2009)
6. Park, S.Y., Lee, B.G.: An analysis on the state-dependent nature of DS/SSMA unslotted ALOHA. J. Commun. Networks 8, 220–227 (2006)
7. Bruno, R., Conti, M., Gregori, E.: Optimal capacity of p-persistent CSMA protocols. IEEE Communications Letters 7(3), 139–141 (2003)
8. Long, K.P., Li, Y., Zhao, W.L., Wang, C.G., Sohraby, K.: p-RWBO: a novel low-collision and QoS-supported MAC for wireless ad hoc networks. Science in China Series F: Information Sciences 51(9), 1193–1203 (2008)
9. Zha, W., Hu, R.Q., Qian, Y., Cheng, Y.: An adaptive MAC scheme to achieve high channel throughput and QoS differentiation in a heterogeneous WLAN. In: Cheng, X.Z. (ed.) Proc. of the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, pp. 26–35. ACM, New York (2006)

Author Index