Studies in Computational Intelligence 364

Roger Lee (Ed.)

Computer and Information Science 2011

Springer

Studies in Computational Intelligence, Volume 364

Further volumes of this series can be found on our homepage: springer.com

Vol. 345. Shi Yu, Léon-Charles Tranchevent, Bart De Moor, and Yves Moreau
Kernel-based Data Fusion for Machine Learning, 2011
ISBN 978-3-642-19405-4

Vol. 346. Weisi Lin, Dacheng Tao, Janusz Kacprzyk, Zhu Li, Ebroul Izquierdo, and Haohong Wang (Eds.)
Multimedia Analysis, Processing and Communications, 2011
ISBN 978-3-642-19550-1

Vol. 347. Sven Helmer, Alexandra Poulovassilis, and Fatos Xhafa
Reasoning in Event-Based Distributed Systems, 2011
ISBN 978-3-642-19723-9

Vol. 348. Beniamino Murgante, Giuseppe Borruso, and Alessandra Lapucci (Eds.)
Geocomputation, Sustainability and Environmental Planning, 2011
ISBN 978-3-642-19732-1

Vol. 349. Vitor R. Carvalho
Modeling Intention in Email, 2011
ISBN 978-3-642-19955-4

Vol. 350. Thanasis Daradoumis, Santi Caballé, Angel A. Juan, and Fatos Xhafa (Eds.)
Technology-Enhanced Systems and Tools for Collaborative Learning Scaffolding, 2011
ISBN 978-3-642-19813-7

Vol. 351. Ngoc Thanh Nguyen, Bogdan Trawiński, and Jason J. Jung (Eds.)
New Challenges for Intelligent Information and Database Systems, 2011
ISBN 978-3-642-19952-3

Vol. 352. Nik Bessis and Fatos Xhafa (Eds.)
Next Generation Data Technologies for Collective Computational Intelligence, 2011
ISBN 978-3-642-20343-5

Vol. 353. Igor Aizenberg
Complex-Valued Neural Networks with Multi-Valued Neurons, 2011
ISBN 978-3-642-20352-7

Vol. 354. Ljupco Kocarev and Shiguo Lian (Eds.)
Chaos-Based Cryptography, 2011
ISBN 978-3-642-20541-5

Vol. 355. Yan Meng and Yaochu Jin (Eds.)
Bio-Inspired Self-Organizing Robotic Systems, 2011
ISBN 978-3-642-20759-4

Vol. 356. Slawomir Koziel and Xin-She Yang (Eds.)
Computational Optimization, Methods and Algorithms, 2011
ISBN 978-3-642-20858-4

Vol. 357. Nadia Nedjah, Leandro Santos Coelho, Viviana Cocco Mariani, and Luiza de Macedo Mourelle (Eds.)
Innovative Computing Methods and their Applications to Engineering Problems, 2011
ISBN 978-3-642-20957-4

Vol. 358. Norbert Jankowski, Włodzisław Duch, and Krzysztof Grąbczewski (Eds.)
Meta-Learning in Computational Intelligence, 2011
ISBN 978-3-642-20979-6

Vol. 359. Xin-She Yang, and Slawomir Koziel (Eds.)
Computational Optimization and Applications in Engineering and Industry, 2011
ISBN 978-3-642-20985-7

Vol. 360. Mikhail Moshkov and Beata Zielosko
Combinatorial Machine Learning, 2011
ISBN 978-3-642-20994-9

Vol. 361. Vincenzo Pallotta, Alessandro Soro, and Eloisa Vargiu (Eds.)
Advances in Distributed Agent-Based Retrieval Tools, 2011
ISBN 978-3-642-21383-0

Vol. 362. Pascal Bouvry, Horacio González-Vélez, and Joanna Kolodziej (Eds.)
Intelligent Decision Systems in Large-Scale Distributed Environments, 2011
ISBN 978-3-642-21270-3

Vol. 363. Kishan G. Mehrotra, Chilukuri Mohan, Jae C. Oh, Pramod K. Varshney, and Moonis Ali (Eds.)
Developing Concepts in Applied Intelligence, 2011
ISBN 978-3-642-21331-1

Vol. 364. Roger Lee (Ed.)
Computer and Information Science, 2011
ISBN 978-3-642-21377-9

Roger Lee (Ed.)

Computer and Information Science 2011

Springer

Editor

Prof. Roger Lee
Central Michigan University
Computer Science Department
Software Engineering & Information
Technology Institute
Mt. Pleasant, MI 48859
U.S.A.
E-mail: lee1ry@cmich.edu

Guest Editors

Wencai Du
Simon Xu

ISBN 978-3-642-21377-9 e-ISBN 978-3-642-21378-6

DOI 10.1007/978-3-642-21378-6

Studies in Computational Intelligence ISSN 1860-949X

Typeset & *Cover Design:* Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

The purpose of the 10th International Conference on Computer and Information Science(ICIS 2011) held on May 16-18, 2011 Sanya, Hainan Island, China was to bring together researchers and scientists, businessmen and entrepreneurs, teachers and students to discuss the numerous fields of computer science, and to share ideas and information in a meaningful way. Our conference officers selected the best 20 papers from those papers accepted for presentation at the conference in order to publish them in this volume. The papers were chosen based on review scores submitted by members of the program committee, and underwent further rounds of rigorous review.

In Chapter 1, Honghao Gao et al. In this paper, to overcome these two challenges, we primarily present an On-the-fly approach to modeling the Web navigation behaviors, and apply bounded model checking (BMC) to verifying the On-the-fly navigation model. Finally, a prototype system is discussed.

In Chapter 2, Rahma Fourati et al. Within this context, we propose an approach that identifies anti-patterns in UML designs through the use of existing and newly defined quality metrics. Operating at the design level, our approach examines structural and behavioral information through the class and sequence diagrams. It is illustrated through five, well-known anti-patterns: Blob, Lava Flow, Functional Decomposition, Poltergeists, and Swiss Army Knife.

In Chapter 3, Juxiang Zhou et al. In this paper a novel feature extraction approach is proposed for facial expression recognition by using the curvelet and the LDP (Local Directional Pattern). First, the low frequency coefficients of Curvelet decomposition on expression region are selected as global facial features. Then, LDP descriptor is used to describe eyes region and mouth region respectively as local facial features.

In Chapter 4, Jinhui Yuan et al. In the scenario sensor nodes have the capability to adjust their transmission power with the transmission range, we approximately construct the maximum lifetime data gathering tree with the goal to balance the energy consumption among the sensor nodes to prolong the lifetime of the network. Our simulation shows that our approach is effective.

In Chapter 5, Dapeng Liu et al. The experiment results demonstrate that while novice programmers are diverse in terms of programming styles, good ones tend to control execution in finer granularity. Source code format can be a flag of programming performance. It seems that there is no direct correlation between the frequency of keystrokes and the quality of programs.

In Chapter 6, Kem Z.K. Zhang et al. In this study, we adopt a theory of planned behavior perspective and build up a switching model to explain blog service switching behavior. We employ a survey to explain how two quality beliefs (service quality and quality of alternatives) and two types of costs (sunk costs and relationship costs) exert influence in determining bloggers' switching behavior. Discussions and implications are provided to better understand the switching behavior of blog and other social technologies.

In Chapter 7, Kem Z.K. Zhang et al. In this study, we shed light on the user contribution of online review platforms. In particular, we attempt to understand how information technology (IT) enabled features can facilitate users' online review contribution. To achieve this objective, we conduct a preliminary study on a Chinese online review platform. The findings confirm that social networking technology and virtual community technology provide helpful IT-enabled features to attain a high level of user contribution on the platform. Implications for both researchers and practitioners are discussed.

In Chapter 8, Toukir Imam et al. This paper contains a detail description of the Fuzzy12 algorithm, its implementation and the set-up of the EDG simulation. Our simulation results demonstrate that the Fuzzy algorithm outperforms the LRU replacement algorithm with different perspective, for example, hit ratio, byte hit ratio, miss rate etc.

In Chapter 9, Natalia C. Silva et al. In this paper, we propose a methodology to tackle such a problem by naturally moving from informal business rules toward the implementation of abusiness process using complex event processing. The methodology allows for theactive participation of business people at all stages of the refinement process. This is important to guarantee the correct alignment between information systems and business needs. Throughout the paper, we present an example to illustrate the application of the methodology. The methodology was applied to implement a realprocess of a building company.

In Chapter 10, Hartinder Singh Johal et al. This manuscript tends to realize the above situation by proposing a three dimensional network address translation scheme with a tentative capability to support end-to-end connectivity based applications and at the same time retaining the benefits of conventional NAT model.

In Chapter 11 Majid Rafigh et al. This paper proposed a new routing algorithm schema based on event occurrence pattern to satisfying k-coverage of event paths and maintaining degree of coverage in maximum level as more as possible. This method improves the network lifetime by shifting the routing responsibility from covering nodes to communication nodes, while maximizing the degree of coverage in the main path of events.

In Chapter 12 Jin Zhang et al. This specification formally defines the regular behavior and fault tolerance behavior of priority queue. In particular, a priority-concatenation operator is defined to handle the ordering of data items to ensure the highest-priority item is removed first. A finite state machine as an implementation is built based on this specification. In addition, we also discuss a priority upgrading approach to handle possible starvation situation of low-priority data items in the priority queue.

In Chapter 13 Ming Ke et al. The global statistical properties of the network revealed the brain functional network for chewing of gum had small-world effect and scale-free property. Computing the degree and betweenness which belong to the centrality indices, we found that the neocortical hubs of the network were distributed in the sense and motor cortex, and the nodes in the thalamus and lentiform nucleus held the largest betweenness. The sense and motor cortices as well as thalamus and lentiform nucleus have the important roles in dispatch and transfer information of network.

In Chapter 14 Yosuke Motoki et al. In this paper, we propose a new mechanism based on GSP that is used in advertisement auctions. Each advertisement has some value, because users click the advertisement when it may be useful for them. We analyze the auctioneer's profit in comparison between normal GSP, normal VCG (Vickrey-Clarke-Groves Mechanism) and our proposed mechanism. The contribution of our research includes to clarify the features and advantages of advertisement auctions and effects to website owner's profit rate.

In Chapter 15 Li Yunpeng et al. As a theoretical creation, the research explores the coordinating pattern of value chain and has constructed the model of the value chain platform in tourism destinations and has proposed relevant theories. That is supporting the destination value chain operate effectively with dynamic integrated technology of tourism information system construction; the data searching and analyzing approaches to efficiently process the feedbacks from data users; studying the satisfaction level of tourism guests.

In Chapter 16 Ge Zhu et al. The result shows that China Mobile users' switching costs are significantly higher than those for customers of China Unicom, and the gap was increasing generally. The quantitative analysis demonstrates that reducing of consumer switching costs will relatively benefit small operators and intensify competition.

In Chapter 17 Hui Zhou et al. In our experiments, to identify a large ISP cloud, we spread vantage points inside the cloud and over the world, and collect topology information by probing a fixed list of IP addresses which consists of more than 25,000 routers and 36,000 links. Data analysis shows that sampling bias, if undetected, could significantly undermine the conclusions drawn from the inferred topologies.

In Chapter 18 Hui Zhou et al. The experiment result indicates that, after applying the object snapshot concept of software, RichMap can smoothly capture and present complete router-level snapshots and significantly decrease the network load that it generates.

In Chapter 19 Su Chen et al. The experimental results indicate that a sequential genetic algorithm with intensive interactions can be accelerated by being translated into CUDA code for GPU execution.

In Chapter 20 Haeng-Kon Kim. In this paper, we design and Implement a sensor framework systems related to medical and surveillance that are significantly considered for enhancing human life. These are employed under USN environment to construct multiple health care services in which medical sensors are inter-connected to provide efficient management of them.

It is our sincere hope that this volume provides stimulation and inspiration, and that it will be used as a foundation for works yet to come.

May 2011 Guest Editors

George Du
Simon Xu

Contents

List of Contributors

Maghsoud Abbaspour
Shahid Beheshti University, Iran

Hanêne Ben Abdallah
University of Sfax, Tunisia
Email: hanene.benabdallah@fsegs.rnu.tn

Nadia Bouassida
University of Sfax, Tunisia
Email: nadia.bouassida@isimsf.rnu.tn

Shan Ao
Beijing Info. Science & Tech
University, China

Hong Chen
Renmin University of China
Email: chong@ruc.edu.cn

Huaping Chen
University of Science and Technology
of China, China
Email: hpchen@ustc.edu.cn

Shengbo Chen
Shanghai University, China
Email: schen@shu.edu.cn

Su Chen
Arkansas State University, U.S.A.
Email: su.chen@smail.astate.edu

Xuhui Chen
Lanzhou University of Technology,
China
Email: xhchen@lut.cn

Jianhua Dai
Beijing Info. Science & Tech
University, China

Spencer Davis
Arkansas State University, U.S.A.
Email: spencer@smail.astate.edu

Wencai Du
Hainan University, China
Email: wencai@hainu.edu.cn

Rahma Fourati
University of Sfax, Tunisia
Email: rahma.fourati10@gmail.com

Honghao Gao
Shanghai University, China
Email: gaohonghao@shu.edu.cn

Lijin Gao
Yunnan Normal University, China

Dewen Hu
National University of Defense
Technology, China

Gongzhu Hu
Central Michigan University, USA
E-mail: hu1g@cmich.edu

Toukir Imam
North South University, Bangladesh

Hai Jiang
Arkansas State University, U.S.A.
Email: hjiang@smail.astate.edu

Hartinder Singh Johal
Lovely Professional University, India
Email: hs.johal@lpu.co.in

Ming Ke
Lanzhou University of Technology,
China

Haeng-Kon Kim
Catholic University of Daegu, Korea
Email: hangkon@cu.ac.kr

Kewal Krishan
Lovely Professional University, India
Email: kewal.krishan@lpu.co.in

Matthew K.O. Lee
City University of Hong Kong, China
Email: ismatlee@cityu.edu.hk

Roger Lee
Central Michigan University, USA
Email: lee1ry@cmich.edu

Ricardo M.F. Lima
Federal University of Pernambuco, Brazil
Email: rmfl@cin.ufpe.br

Qi Lina
Beijing University, China

Dapeng Liu
The Brain Tech., U.S.A.
Email: dliu@the brain.com

Tokuro Matuso
Yamagata University, Japan

Jia Mei
School of Computer Engineering and Science
Shanghai University, China
Email: me269@shu.edu.cn

Huaikou Miao
Shanghai University, China
Email: hkmiao@shu.edu.cn

Ni Min
Beijing Yong You Software Company, China

Yosuke Motoki
Yamagata University, Japan

Amandeep Nagpal
Lovely Professional University, India
Email: amandeep.nagpal@lpu.co.in

Andy Novobilski
Arkansas State University, U.S.A.
Email: anovobilski@smail.astate.edu

C´esar A. L. Oliveira
Federal University of Pernambuco, Brazil
Email: calo@cin.ufpe.br

Majid Rafigh
Shahid Beheshti University, Iran
Email: m.rafigh@mail.sbu.ac.ir

Rashedur M. Rahman
North South University, Bangladesh

Cecília L. Sabat
Federal University of Pernambuco, Brazil
Email: cls@cin.ufpe.br

Yoshihito Saito
Yamagata University, Japan

Hui Shen
National University of Defense Technology Changsha, China

Natália C. Silva
Federal University of Pernambuco, Brazil
Email: ncs@cin.ufpe.br

Balraj Singh
Lovely Professional University, India
Email: balraj.13075@lpu.co.in

Satoshi Takahashi
Tsukuba University, Japan

Yunqiong Wang
Yunnan Normal University, China

Qinling Xin
Central China University of Technology, China

Shaochun Xu
Algoma University, Canada
Email: simon.xu@algomau.ca

Tianwei Xu
Yunnan Normal University, China
Email: xutianwei@ynnu.edu.cn

Rongfang Yang
Yunnan Normal University, China

Xie Yongqiu
Capital University of Economics and Business,
China

Hao Yu
Capital University of Economics and Business,
China

Jinhui Yuan
Renmin University of China
Email: jcyjh@126.com

Li Yunpeng
Beijing University, China

Kem Z.K. Zhang
City University of Hong Kong, China
Email: zikzhang@cityu.edu.hk

Jin Zhang
Hainan University, China
Email: zj001_cn@163.com

Sesia J. Zhao
USTC-CityU Joint Advanced Research Center, China
Email: sesiazj@mail.ustc.edu.cn

Hui Zhou
Hainan University, China

Hongwei Zhou
Renmin University of China, China
Email: hong wei zhou@hotmail.com

Juxiang Zhou
Yunnan Normal University, China
Email: zjuxiang@126.com

Xiaolin Zhou
National University of Defense Technology, China

Zongtan Zhou
Peking University, Beijing

Ge Zhu
Beijing Yong You Software Company, China

Applying Bounded Model Checking to Verifying Web Navigation Model

Honghao Gao, Huaikou Miao, Shengbo Chen, and Jia Mei

Abstract. With the development of Web applications, formal verification of Web navigational behaviors has been a significant issue in Web engineering. Due to the features of Web technologies, such as caching, session and cookies, Web users can press the Back or Forward buttons to revisit Web pages. But these complex interactions between users and Web browsers may negatively influence the overall functionalities and navigations of Web applications. There are two challenges: One is that it is hard to model all possible navigation paths because the number of dynamic interactions and the personalized pages generated by different Web users may be huge or even infinite. Another is that how to improve the efficiency of verification because counterexamples usually manifest in a small number of navigation model. In this paper, to overcome these two challenges, we primarily present an On-the-fly approach to modeling the Web navigation behaviors, and apply bounded model checking (BMC) to verifying the On-the-fly navigation model. Finally, a prototype system is discussed.

1 Introduction

Along with the appearance and rapid improvement of Internet, more and more companies are rushing to employ Web applications to support their business in order to accelerate the cooperation with their customers, i.e., B2B, C2C, G2C Web sites. Due to the distribution of Web environment, hyperlinks linked by *page-to-page* paradigm have been used to binding the connection between Web pages. Moreover, the basic Web browser features provide an adequate set of navigational facilities for Web users to revisit Web pages, including the Back and Forward buttons, Refresh, Favorites, Link menu, URL-rewriting etc. Therefore, Web users can interact with not only the Web pages but also the Web browsers. However,

Honghao Gao · Huaikou Miao · Shengbo Chen · Jia Mei
School of Computer Engineering and Science, Shanghai University, 200072, Shanghai, P.R. China
and
Shanghai Key Laboratory of Computer Software Evaluating&Testing, 201112, Shanghai, P.R. China
e-mail:{gaohonghao,hkmiao,schen,me269}@shu.edu.cn

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 1–15.
springerlink.com

negatively pressing the Back or Forward buttons may influence the overall functionalities and navigations of Web applications, especially in *Safety Critical Region* (SCR)[3].

At present, how to modeling and verifying Web applications is still a complex task. To the best of our knowledge, the navigation model is one of the important research areas, also named navigation graph, which can help for clarifying requirements and specifying implementation behaviors. e.g., it is valuable for checking the conformance between a designed navigation behavior and an implemented navigation behavior. Actually, the navigation of a Web application is the possible sequence of Web pages a user has visited, where nodes represent Web pages and edges represent direct transitions between pages. The next page is determined by the current page and the action, i.e, back, forward, reload and hyperlink. But it also brings up two challenges: (1) the number of dynamic interactions and the personalized pages generated by different Web users may be huge or even infinite. It is nearly always impractical to model all possible navigation paths; (2) counterexamples only manifest in a small number of navigation model. It is inefficient to check all states of each path, many of which do not contribute to counterexample detection.

In our paper, there are two novelties: (1) capturing dynamic navigation model on the fly and considering not only Web pages' hyperlink but also Web browser buttons' state; (2) using BMC to checking the properties in the small scope of navigation model for improving the efficiency of verification. Concretely, our approach involves three major steps: First, a Web Browser Loading Model (BLM) is used to construct the integrated model of the Web application incorporating the abstract behavior of the internal session control, caching mechanism and enabled buttons of web browser. Second, based on BLM, Kripke structure is employed to describe the On-the-fly navigation models by gradually combining each page's navigation associations. Third, BMC is used to verify the safety and liveness property against the On-the-fly navigation model for enhancing the capacity of detecting faults.

The remainder of this paper is organized as follows: Section 2 analyzes the Web applications behaviors, and introduces Kripke structure to construct the Web navigation model on the fly based on our previous study [3]. Section 3 gives a brief introduction to BMC, and then applies BMC to verifying the liveness and safety properties of Web applications. Section 4 reviews the related works. Section 5 draws a conclusion and future works.

2 Modeling Web Application Behaviors

The purpose of this section is to discuss the features of Web browsers and Web pages for formally modeling the Web Application behaviors and the definitions of On-the-fly Web navigation model.

The pages in *SCR* require higher security requirements. They prohibit the user from entering to ***SCR*** through pressing the Web button from *no-SCR*, even though the page has been visited in the past. As Fig.1 shown, we first introduce an

example of *Audit System*, which will be used throughout the paper. After receiving a request of login from a user, *Audit System* will authenticate his identity. Once he login system successfully, two things may happen: user can check out a bank receipt and audit the authenticity and reliability. Otherwise, user can declare this receipt invalid. During this process, the *login page* P2 can be linked to the set of ***SCR*** {P3, P4, P5, P7}. At page P4 or P7, the user can press the Back and Forward button to its predecessor and successor page, respectively. But at the boundary {P2, P3} or {P5, P6}, user can press the Web buttons to revisit pages{P3, P5} form pages{P2,P6}.Otherwise, the user will be confused to be redirected to an unreachable or unsecure Web page. To model these navigational behaviors of Web Applications, we should take the Web browser button click and the Web page hyperlink action into consideration.

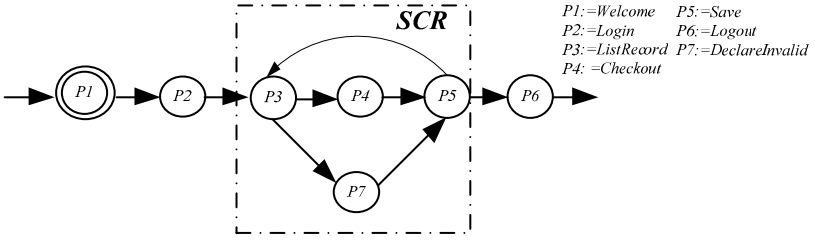


Fig. 1 A motivation example

Actually, many Web browsers cache page contents. When a user triggered a series of operations, the Web browser will maintain a *history stack* for the revisited Web pages. The history stack consists of stack pointer, top pointer and bottom pointer specifying the button states. We mainly investigate the *Replacement* hyperlinks paradigm where the destination page replaces the source one[1]. Some of sophisticated situations [1], such as *OpenInNewWindow* and *ShowInTooltipWindow*, are not considered because our work only requires to obtain the available pages states.

Definition 1 (Button enabled). If the history stack has more than one item and the stack pointer does not point to the bottom item stored, the Back button is enabled. Similarly, the Forward button is enabled when the history stack has more than one item and the stack pointer does not point to the top item [3].

Let n be the length of history stack. *pno* indicates the page position in the stack. For Back and Forward buttons, we consider four states [2]: (1) Back Disabled and Forward Disabled(BDFD), $pno=1$, $n=1$; (2) Back Disabled and Forward Disabled(BDFD), $pno=1$, $n>1$; (3)Back Enabled and Forward Disabled (BEFD), $pno = n$, $n>1$; (4)Back Enabled and Forward Enabled (BEFE), $pno>1$, $pno<n$.

According to the button states, a Web Browser loading model (BLM) [3] is used to modeling each page and the behaviors of the Web browsers which loaded the page.

Definition 2 (Browser Loading Model, *BLM*). BLM is a quadruple BLM = (*Pre*, *Br, Cur*, *Post*), where

- *Pre* is a state of precursor page, {-}indicates that its precursor page is absent;
- *Br* is the state of the browser behaviors, *Br*∈{00, 10, 01, 11}, where, 00 stands for *BDFD*, 10 stands for *BEFD*, 01 stands for *BDFE* and 11 stands for *BEFE*;
- *Cur* is a state of current page;
- *Post* is a state of successor page, {-} indicates that the successor page is absent.

The BLM prescribes all potential behaviors of Web applications. Tab.1 shows the BLMs of the example depicted in Fig.1. Considering the security of *SCR*, these *BLMs* are divided as follows:

Definition 3 (Normal States BLM, *n-BLM*). At the boundary, the button state of login page and logout page, which specifies the ability of entering *SCR*, is disabled, such as (P1, 10, P2, P3). At the other places, the button state described in *Br* is enabled (or disabled) and its corresponding page state *Pre* or *Post* is present (or absent), e.g., (-, 01, P1, P2).

Definition 4 (**Error States BLM, *e-BLM*).** At the boundary, the button state of login page and logout page, which specifies the ability of entering *SCR*, is enabled, such as (P1, 11, P2, P3). At the other places, the button state described in *Br* is enabled (or disabled) and its corresponding page state *Pre* or *Post* is absent (or present), e.g., (-, 01, P1, -).

Table 1 An example of BLMs

page	*n-BLM*	*e-BLM*
P1	(-,00,P1,-), (-,01,P1,P2)	(-, 01, P1, -), (-, 11, P1, P2)
P2	(P1,10,P2, -), (P1,10,P2,P3)	(P1, 11, P2, -), (P1, 11, P2, P3)
P3	(P2,00,P3,-),(P2,01,P3,P4),(P5,11,P3,P4)(P2, 01, P3, P7), (P5,11,P3,P7)	(P2,11,P3,-),(P2,10,P3,P4),(P5,00,P3,P4)(P2, 10, P3, P7), (P5,00,P3,P7)
P4	(P3, 10, P4, -), (P3, 11, P4, P5)	(P3,01, P4, -), (P3, 00, P4, P5)
P5	(P4,10,P5,-),(P4,10,P5,P6),(P7,10,P5,P6),(P7, 10, P5,-),(P7,11,P5,P3),(P4,11, P5, P3)	(P4,01,P5,-),(P4,01,P5,P6),(P7,01,P5,P6),(P7, 01, P5,-),(P7,00,P5,P3),(P4,00, P5, P3)
P6	(P5,00,P6,-)	(P5,11,P6,-),(P5,10,P6,-)
P7	(P3,10,P7,-), (P3,11,P7,P5)	(P3,11,P7,-),(P3,01,P7,-),(-,11,P7,P5), (P3,00,P7,-)

When links are followed, the Web browser builds a complex sequence of visited pages. Based on the BLM, each page state denoted as Pi(XY) labels each page and the Web browser where X and Y represent the enable state of Back and Forward button, respectively. Each link trigged by *actions*∈(*login*, *link*, *back*, *forward*, *logout*) is a transition which means that the Web application undergoes the different actions. By combining these associations, we can model these

navigational behaviors. For instance, from the relations P3(00)-«*link*»-P4(10), P4(10)-«*link*»-P5(10) and P5(10)-«*back*»-P4(11), we can get a new navigation P3(00)→P4(11).

Compared to the static analysis of Web applications, the navigation model has more complexity because the complex interactions will be stochastically triggered by Web users at runtime. Intuitively, it is impractical to construct the complete navigation model without On-the-fly strategy. Encapsulating all human interactions into a completed system model is hard to implement. Instead, in our study, we consider to apply the technology of incremental modeling for dynamically analyzing Web applications, which is called On-the-fly navigation models [3].

Definition 5 (On-the-fly navigations model). Formally, On-the-fly navigation model in Web application is described as an extended Kripke structure $WA_N = (S, S_0, AP, R, L)$.

In WA_N, S is a finite set of states. Each state Pi(XY) consists of one Web page and the behaviors of Web browser which loaded the page; $S_0 \subseteq S$ is the set of initial states; AP is a set of atomic propositions specifying the Web pages and actions. The proposition of each page Pi(XY) is expressed as (Pi∧X∧Y); $R \subseteq S \times S$ is a transition relation that must be total, that is, for every state $s \in S$ there is a state $s' \in S$ such that $R(s, s')$; L: $S \rightarrow 2^{AP}$ is a function that labels each state with the set of atomic propositions true in that state.

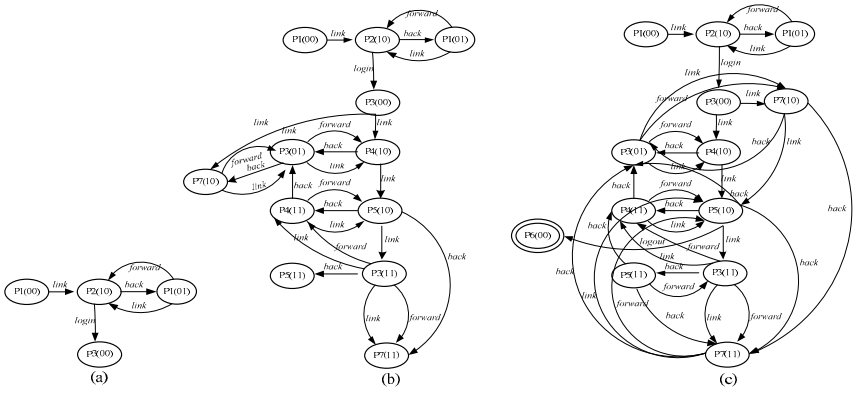


Fig. 2 The On-the-fly navigation model

To illustrate the related definitions, the following simple examples are used to help users comprehend how the navigation model of Fig.1 is constructed on the fly. In Fig.2.(a), pages {P1, P2, P3} are involved. The state P3(00) is special state, which characterizes the *SCR* representing the private page. Thus, only the login action has access to the state P3(00) so that the Forward button of Web browser at

page P3 need to be disenabled. When the pages {P1, P2, P3, P4, P5} participate, the more complex model is shown in Fig.2.(b). Finally, the completed model is depicted in Fig.2.(c).

3 Bounded Model Checking Web Application

After constructing a Web navigation model, how to effectively search and find counterexamples plays an important role in verifying the On-the-fly navigation model. In the following section, we will focus on using BMC to verify the On-the-fly navigation models to improve the efficiency of verification. The motivation of BMC is to reduce the verification of a property to search for a counterexample in small scope of executions whose length is bounded by some integer k [12,13]. If no bugs are found then the search is continued for larger k until either a bug is found or the bound k is touched. Unlike model checking studies of the Web application [3,10,11], our verification method has shorter time-consuming and lower memory-consuming for checkingproperty.

First, we introduce the definition of BMC to reader. Given a navigation model M, a temporal logic formula φ and a bound k, the BMC formula is defined as $[[M,\varphi]]_k = [[M]]_k \wedge [[\varphi]]_k$,which will be satisfiable if and only if the formula φ is valid along some navigation paths of M [4,13], that is,

- The leftmost expression $[[M]]_k = I(s_0) \wedge \bigwedge_{i=0}^{k-1} R(s_i,s_{i+1})$ represents a valid navigation path starting from an initial state by unrolling the transition relation $(s_0, \ldots, s_k)$ within bound k, where $I(s_0)$ is the characteristic function of the set of initial states, and $R(s_i,s_{i+1})$ is the characteristic function of the transition relation.
- The rightmost expression $[[\varphi]]_k$ will be true if and only if the property formula φ is valid along the navigation path of length k.

3.1 Bounded Semantics

To find the bounded witnesses, the bounded semantics is needed to be introduced, which allows us to interpret the formulas over a fragment of the considered model only. Here, the k-loop transition is defined as (k, l)-loop $R(s_k,s_l)$ that means the successor of s_k is s_l.

Definition 6 (Successor function). Given l is the start position of the loop, k is bound, and i is the current position, the successor function is defined as:

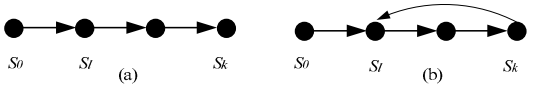


Fig. 3 Two transition ways

As Fig.3.(a) shown, if the path is loop free, then $succ(i) := i + 1 \quad for\ i < k$; otherwise $succ(i) := \varnothing \quad for\ i = k$;

As Fig.3.(b) shown, if the path has a loop, then $succ(i) := i + 1 \quad for\ i < k$; otherwise $succ(i) := l \quad for\ i = k$;

Definition 7 (LTL Formula with loop). Let φ and ϕ be LTL formulae, k, l, $i \geq 0$, with l, $\leq$ k. The ${}_l[\![\varphi]\!]^i_k$ is defined as loop-based LTL formula:

$${}_l[\![p]\!]^i_k := p(s_i)$$
$${}_l[\![\neg\varphi]\!]^i_k := \neg\varphi(s_i)$$
$${}_l[\![\varphi\wedge\phi]\!]^i_k := {}_l[\![\varphi]\!]^i_k \wedge {}_l[\![\phi]\!]^i_k$$
$${}_l[\![\varphi\vee\phi]\!]^i_k := {}_l[\![\varphi]\!]^i_k \vee {}_l[\![\phi]\!]^i_k$$
$${}_l[\![\varphi\rightarrow\phi]\!]^i_k := {}_l[\![\varphi]\!]^i_k \vee {}_l[\![\phi]\!]^i_k$$
$${}_l[\![\mathbf{X}\phi]\!]^i_k := {}_l[\![\phi]\!]^{succ(i)}_k$$
$${}_l[\![\mathbf{G}\phi]\!]^i_k := {}_l[\![\phi]\!]^i_k \wedge {}_l[\![\phi]\!]^{succ(i)}_k$$
$${}_l[\![\mathbf{F}\phi]\!]^i_k := {}_l[\![\phi]\!]^i_k \vee {}_l[\![\phi]\!]^{succ(i)}_k$$
$${}_l[\![\phi\,\mathbf{U}\varphi]\!]^i_k := {}_l[\![\varphi]\!]^i_k \vee ({}_l[\![\phi]\!]^i_k \wedge {}_l[\![\phi\,\mathbf{U}\varphi]\!]^{succ(i)}_k)$$

Definition 8 (General Translation). Let φ be an LTL formula, M a navigation model and $k \geq 0$. The general translation is

$$[\![M_k\varphi]\!]_k = I(s_0) \wedge \bigwedge_{i=0}^{k} R(s_i, s_{i+1}) \wedge [\![\varphi]\!]_k$$

where $[\![\varphi]\!]_k = (\neg L_k \wedge [\![\varphi]\!]^0_k) \vee \bigvee_{l=0}^{k} ({}_lL_k \wedge {}_l[\![\varphi]\!]^0_k)$ is the disjunction of the translation without a loop and the translation with a loop. And ${}_lL_k = \mathrm{R}(s_k, s_l)$, $L_k = \bigvee_{l=0}^{k} {}_lL_k$.

In Fig.2.(a), suppose k=2 and the initial state is I(P1(00)). To check the property φ, the BMC formula $[\![M, \varphi]\!]_2$ is created as follows:

The transition with a loop R(P1(01),P2(10)) is

${}_1[\![\varphi]\!]^0_2 = \varphi(\mathrm{P1(00)}) \wedge {}_1[\![\varphi]\!]^1_2$, ${}_1[\![\varphi]\!]^1_2 = \varphi(\mathrm{P2(10)}) \wedge {}_1[\![\varphi]\!]^2_2$, ${}_1[\![\varphi]\!]^2_2 = \varphi(\mathrm{P1(01)}) \wedge {}_1[\![\varphi]\!]^3_2 = \varphi(\mathrm{P1(01)}) \wedge [\![\varphi]\!]^1_2$,

The transition without loops is

$[\![\varphi]\!]^0_2 = \varphi(\mathrm{P1(00)} \wedge [\![\varphi]\!]^1_2$, $[\![\varphi]\!]^1_2 = \varphi(\mathrm{P2(10)}) \wedge [\![\varphi]\!]^2_2$, $[\![\varphi]\!]^2_2 = \varphi(\mathrm{P3(00)})$, $[\![\varphi]\!]^3_2 = \varnothing$.

The fully formula is constructed as follow:

$[\![M,\varphi]\!]_3 = [\![M]\!]_3 \wedge [\![\varphi]\!]_2 = I(\mathrm{P1(00)}) \wedge (R(\mathrm{P1(00),P2(10)}) \wedge R(\mathrm{P2(10),P1(01)}) \wedge R(\mathrm{P2(10),P3(00)})) \wedge ((R(\mathrm{P1(01),P2(10)}) \wedge \varphi(\mathrm{P1(00)}) \wedge \varphi(\mathrm{P2(10)}) \wedge \varphi(\mathrm{P1(01)})) \vee ((\neg R(\mathrm{P1(01),P2(10)}) \wedge \varphi(\mathrm{P1(00)}) \wedge \varphi(\mathrm{P2(10)}) \wedge \varphi(\mathrm{P3(00)}))$).

Normally, a finite unfolding path of $[\![M,\varphi]\!]_k$ is CNF (conjunctive normal form)[4]. In other words, it can be translated to check the propositional satisfiability problem (SAT), where SAT solver can be used to get an answer though conflict clauses discover technology [18].

3.2 Verifying On-the-Fly Navigations Model

As mentioned above, users are not allowed to revisit private pages of ***SCR*** from public pages of ***non-SCR***. In this paper, we mainly consider two properties (1) liveness ($\mathbf{F}\varphi$) that the good thing will be occurred eventually, and (2) safety ($\mathbf{G}\neg\varphi$) that every state is in contradiction to the bad thing.

Definition 9 (BMC Liveness Property). A notion of liveness property states that, under certain conditions, something will ultimately occur. By examining all finite sequences of length k starting from initial states, the satisfiability of the formula (1) is adopted to check whether φ can be satisfied in k steps against our On-the-fly navigation model.

$$[[M, \mathbf{F}\varphi]]_k = I(s_0) \wedge \bigwedge_{i=0}^{k-1} R(s_i, s_{i+1}) \rightarrow \bigvee_{i=0}^{k} \varphi(s_i) \quad (1)$$

- $I(s_0)$ is the characteristic function for the set of initial relations. The page state s_0 corresponds to P1(00) where Forward disabled and Back disabled. Initially, $I(s_0)$ = (P1∧¬back∧¬forward).
- $\bigwedge_{i=0}^{k-1} R(s_i, s_{i+1})$ enumerates the unfolding paths of the navigation models. The semantic of transition Pi(XY)-«*action*»-Pj(X'Y') in On-the-fly navigation model means that there exists an association between Pi(XY) and Pj(X'Y'). The R(Pi(XY), Pj(X'Y')) is expressed as (Pi∧X∧Y) → (Pj∧X'∧Y'). For example, R(P1(00), P2(10)) corresponds to (P1∧¬back∧¬forward) → (P2∧back∧¬forward).
- $\varphi(s_i)$ is the conjoint of atomic propositions at state s_i. For instance, φ(P1(10))=(P1∧back∧¬forward).
- $\bigvee_{i=0}^{k} \varphi(s_i)$ is the disjoint of $\varphi(s_i)$ which means that at least one page state s_i satisfies the property φ that $\exists s_i \in S \bullet s_i \models \varphi$

If liveness property $[[M, \mathbf{F}\varphi]]_k$ is unsatisfiable, then the bounded k needs to be increased. The verification continues to verify whether the following formula (2) is satisfied,

$$[[M,\mathbf{F}\varphi]]_{k+1} = I(s_0) \wedge \bigwedge_{i=0}^{k} (R(s_i, s_{i+1}) \wedge \neg\varphi(s_i)) \rightarrow \varphi(s_{k+1}) \quad (2)$$

The procedure will terminate if the liveness property holds at length k. Note that the bound k-1 represents the length of the longest sequence from an initial state without hitting a state where φ holds.

Theorem 1. $M \models \mathbf{F}\varphi$ iff $\exists k \bullet [[M, \mathbf{F}\varphi]]_k$ is valid.

For the formal proof of theorem 1, interested readers are referred to [12]. According to theorem 1, we need to search for a k that makes the negation of $[[M, \mathbf{F}\varphi]]_k$ unsatisfiable. In our experiments, we adopt *Normal States BLM* to check the liveness property of navigation models for verifying the functional correctness. Let us consider the example as shown in Fig.2.(a). Suppose the maximal bound k is 5. Liveness property requires that the desired property should be satisfied in future. The semantic of verification process is shown as follows:

If k=1 with $[[M, \mathbf{F}\varphi]]_1$, the initial formula is
= P1(00) → φ(P1(00))
=(P1∧¬back∧¬forward)→φ(P1(00))

If k=3 with $[[M, \mathbf{F}\varphi]]_3$, one path may be selected:

=P1(00)∧(P1(00)-«*link*»-P2(10))∧(P2(10)-«*back*»-P1(01))→φ(P1(00))∨φ(P2(10))∨φ(P1(01)).

=(P1∧¬back∧¬forward)∧(P1∧¬back∧forward→P2∧back∧¬forward)∧((P2∧back∧¬forward)→(P1∧¬back∧forward))→φ(P1(00))∨φ(P2(10))∨φ(P1(01))

If k=5 with $[[M, \mathbf{F}\varphi]]_5$, one path may be selected:

=P1(00)∧(P1(00)-«*link*»-P2(10))∧(P2(10)-«*back*»-P1(01))∧(P1(01)-«*forward*»-P2(10))∧(P2(10)-«*login*»-P3(00))→φ(P1(00))∨φ(P2(10))∨φ(P1(01))∨φ(P3(00))

=(P1∧¬back∧¬forward)∧(P2∧¬back∧forward)∧(P1∧¬back∧forward)∧(P2∧back∧forward)∧(P3∧¬back∧¬forward)→φ(P1(00))∨φ(P2(10))∨φ(P1(01)∨φ(P2(01))∨φ(P3(00)).

These Boolean calculations can be atomically verified by using model checker NuSMV with SAT solver. Tab.2 shows the results of liveness verification for Fig.2.(a), where φ is the property formula; ***k*** is the bound; ***R*** is the verification result where 'T' is *true* and 'F' is *false*; ***i*** is the cursor; ***T*** is the time consumption; ***M*** is the memory consumption; - is the maximum size of states.

Table 2 The results for liveness verification

Pages	**Fφ**	***k*=2**				***k*=5**			
		R	***i***	**T**	**M**	**R**	***i***	**T**	**M**
P1(00)	**F**(P1(00))	T	1	<0.1	25	T	1	<0.1	25
P2(10)	**F**(P2(10))	T	2	<0.1	32	T	2	<0.1	32
P1(01)	**F**(P1(01))	F	-	11	46	T	3	31	56
P2(11)	**F**(P2(11))	F	-	11	46	F	-	45	61
P3(00)	**F**(P3(00))	F	-	11	46	T	3	34	60

As expected, some properties can be immediately checked in the small model without constructing the fully model. In addition, the performance of verification has been significantly improved in time consuming and memory consuming. Note that the lines recoded as " R='F', S='-', T='11', M='46' " indicates that the property φ doesn't satisfy in Fig.2.(a) after the model has been checked completely. In this case, it is hard to decide whether the liveness is guaranteed in the complete model. The more outgoing actions imply that some counterexamples will be discovered in a larger model in the future. But if the verification returns "T" we ensure that the complete model also satisfies liveness.

Definition 10 (BMC Safety Property). A notion of safety property states that, under certain conditions, an undesirable event will never happen. By examining all finite sequences of length k starting from initial states, the satisfiability of the formula (3) is adopted to check whether $\neg\varphi$ is violated in k steps against our On-the-fly navigation model.

$$[[M, \mathbf{G}\neg\varphi]]_k = \mathrm{I}(s_0) \wedge \bigwedge_{i=0}^{k-1} \mathrm{R}(s_i, s_{i+1}) \wedge \bigwedge_{i=0}^{k} \neg\varphi(s_i) \tag{3}$$

The formula $\mathbf{G}\neg\varphi$ indicates that every state of a path avoids the bad thing φ. The major difference from formula (1) is that safety property requires that every state is not a counterexample within length k. In other words, unsafe property, such as the threat events and attack behaviors, will not happen in the safety model. The navigation model is called unsafe when at least one page state violated the safety properties. In other words, it makes formula (3) unsatisfiable.

If safety property $[\![M, \mathbf{G}\neg\varphi]\!]_k$ is unsatisfiable, then the bounded k needs to be increased. The verification continues to verify whether the following formula (4) is satisfied,

$$[\![M,\mathbf{G}\neg\varphi]\!]_{k+1}=\mathbf{I}(s_0)\wedge\bigwedge_{i=0}^{k}(\mathrm{R}(s_i,s_{i+1})\wedge\neg\varphi(s_i))\wedge\varphi(s_k) \tag{4}$$

A counterexample to property $[\![M, \mathbf{G}\neg\varphi]\!]_k$ is a trace of states, where the last state contradicts the property. Different from unbounded transition, although $\mathbf{G}\neg\varphi$ holds along all the states from s_0 to s_k, a counterexample of $\mathbf{G}\neg\varphi$ may appear in state s_{k+1}. Once the $[\![M, \mathbf{G}\neg\varphi]\!]_k$ outputs a counterexample, we confirm that the safety property is unsatisfied in M.

Theorem 2. $M\not\models\mathbf{G}\neg\varphi$ iff $\exists k\bullet[\![M, \mathbf{G}\neg\varphi]\!]_k$ is invalid.

According to the Theorem 2 [12], the basic idea of our safety verification is to prevent the *Error States BLM* from occurring in navigation models. Suppose P2 has safety $\mathbf{G}(\neg(\mathrm{P2}(10)\rightarrow\mathbf{X}(\mathit{login}\wedge\mathrm{P3}(00))))$. A counterexample will be detected at k=3 because a counterexample P2(10)-«*login*»-P3(00) satisfies to the formula $\mathbf{F}(\mathrm{P2}(10)\rightarrow\mathbf{X}(\mathit{login}\wedge\ \mathrm{P3}(00)))$.

As Tab.3 shown, if the result prints "T", we assert that there is no counterexample. But we can't arbitrarily make an conclusion that the safety property is satisfied. If the result prints "F", we confirm that the safety property is unsatisfied. Let us discuss the results in Tab.3, some interesting things will confuse us when k=5. The different results between P3(11)-«*forward*»-P5(11) and P7(11)-«*back*»-P3(11) show that even though the bound k is reached to the limited length we can't draw that whether the model is safe or not because not all navigations have been constructed in the current model. Maybe the bad state of counterexample will be found in a larger state space. As expected, when k=10, the counterexample of P7(11)-«back »-P3(11) is found.

Table 3 The results for safety verification

Interactions	$\mathbf{G}\neg\varphi$	***k*=2**				***k*=5**			
		R	***i***	**T**	**M**	**R**	***i***	**T**	**M**
P1(00)-«*forward*»-P2(10)	$\mathbf{G}(\mathrm{P1}(00)\rightarrow\mathbf{X}(\mathit{forward}\wedge\mathrm{P2}(10))$	F	2	<0.1	25	F	2	<0.1	25
P1(00)-«*link*»-P2(10)	$\mathbf{G}\ (\mathrm{P1}(00)\rightarrow\mathbf{X}(\mathit{link}\wedge\mathrm{P2}(10))$	T	-	156	216	T	-	156	216
P3(00)-«*forward*»-P4(10)	$\mathbf{G}(\mathrm{P3}(00)\rightarrow\mathbf{X}(\mathit{forward}\wedge\mathrm{P4}(10))$	F	4	48	113	F	4	48	113
P3(00)-«*link*»-P4(10)	$\mathbf{G}\ (\mathrm{P3}(00)\rightarrow\mathbf{X}(\mathit{link}\wedge\mathrm{P4}(10))$	T	-	156	216	T	-	156	216
P3(00)-«*forward*»-P7(10)	$\mathbf{G}(\mathrm{P3}(00)\rightarrow\mathbf{X}(\mathit{forward}\wedge\mathrm{P7}(10))$	F	3	25	48	F	3	35	48
P3(00)-«*link*»-P7(10)	$\mathbf{G}\ (\mathrm{P3}(00)\rightarrow\mathbf{X}\ (\mathit{link}\wedge\mathrm{P7}(10))$	T	-	156	216	T	-	156	216
P4(10)-«*forward*»-P5(10)	$\mathbf{G}(\mathrm{P4}(10)\rightarrow\mathbf{X}(\mathit{forward}\wedge\mathrm{P5}(10))$	F	4	35	95	F	4	35	95
P4(10)-«*link*»-P5(10)	$\mathbf{G}\ (\mathrm{P4}(10)\rightarrow\mathbf{X}\ (\mathit{link}\wedge\mathrm{P5}(10))$	T	-	156	216	T	-	156	216

Table 3 (*continued*)

P5(10)-«*forward*»-P3(11)	**G**(P5(10)→**X**(*forward*∧P3(11))	F	5	46	154	F	5	46	154
P5(10)-«*link*»-P3(11)	**G** (P5(10)→**X**(*link* ∧P3(11))	T	-	156	216	T	-	156	216
P5(11)-«*forward*»-P3(11)	**G**(P5(11)→**X**(*forward*∧P3(11))	T	-	156	216	T	-	75	160
P3(11)-«*forward*»-P5(11)	**G**(P3(11)→**X**(*forward*∧P5(11))	T	-	156	216	F	7	77	134
P7(11)-«*forward*»-P5(10)	**G**(P7(11)→**X**(*forward*∧P5(10))	F	-	156	216	F	7	68	164
P5(10)-«*back*»-P7(11)	**G**(P5(10)→**X**(*back*∧P7(11))	F	5	65	183	F	5	65	183
P2(10)-«*back*»-P1(00)	**G**(P2(10)→**X**(*back*∧P1(00))	T	-	156	216	T	-	156	216
P1(00)-«*link*»-P2(10)	**G**(P1(00)→**X**(*link*∧P2(10))	T	-	156	216	T	-	156	216
P4(10)-«*back*»-P3(00)	**G**(P4(10) →**X** (*back*∧P3(00))	F	4	38	101	F	4	38	101
P5(10)-«*back*»-P4(10)	**G** (P5(10)→**X**(*back*∧P4(10))	F	5	70	200	T	5	70	200
P7(11)-«*back*»-P3(11)	**G**(P7(11)→**X**(*back*∧P3(11))	T	-	156	216	F	7	68	164
P3(11)-«*link*»-P7(11)	**G**(P3(11)→**X**(*link*∧P7(11))	T	-	156	216	T	-	156	216
P4(11)-«*back*»-P3(11)	**G**(P4(11)→**X**(*back*∧P3(11))	F	-	156	216	F	6	60	148
P3(11)-«*link*»-P4(11)	**G**(P3(11)→**X**(*link*∧P4(11))	T	-	156	216	T	-	156	216
P2(10)-«*forward*»-P3(00)	**G**(P2(10)→**X**(*forward*∧P3(00))	F	2	25	48	T	2	25	48
P2(10)-«*login*»-P3(00)	**G**(P2(10)→**X**(*login*∧P3(00))	T	-	156	216	T	-	156	216
P3(00)-«*back*»-P2(10)	**G**(P3(00)→**X**(*back*∧P2(10))	T	3	30	51	T	3	30	51
P5(10)-«*logout*»-P6(00)	**G**(P3(00)→**X**(*logout*∧P2(10))	F	-	156	216	F	-	156	216

3.3 Prototype System

Different from the general model checking, BMC can reduce the complexity of verification without fully numerating the reachable states in the navigation model. Using BMC for navigation model of Web application is a truly On-the-fly verification. A reachable violation state may be reached during verification, where the whole model needn't to be built completely so that BMC does not suffer from the space explosion problem.

Fig.4 presents the work process of our approach. We step up different bound k for each model M_i when BMC is used to check property φ. If the model M_i outputs a counterexample, then the verification process stops immediately. If the model M_i has no counterexamples after examining all finite sequences of length k, then we move to depth k=k+1. If k reaches the size of currently model state space, then the larger model M_{i+1} is to be constructed on the fly for further processing. Finally, if the completed model M_n doesn't generate any counterexample, then we can summary that the model satisfies the property φ.

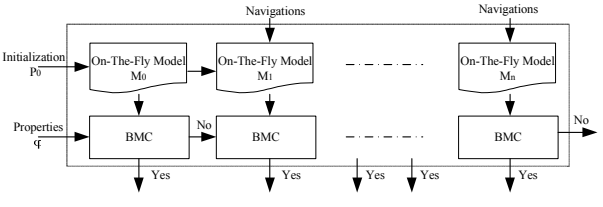


Fig. 4 The work process

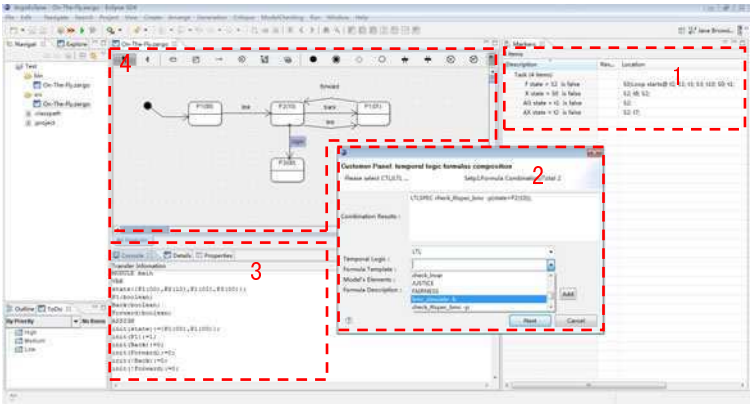


Fig. 5 Prototype system

As Fig.5 shown, a prototype of the system based on Eclipse IDE platform is constructed to facilitate logs analysis, visual monitoring (referring to dashed area 4), automatic model transformation (referring to dashed area 3), model checking, and counterexamples generation (referring to dashed area 1). In our prototype, ArgoUML is integrated with model checker NuSMV, from which developer can verify navigation model easily. First, the logs file of Web applications is dynamically extracted from IIS Server for constructing navigation model in form of ArgoUML file format ".zargo". Second, the model can be automatically translated to NuSMV code. Third, on the TL(Temporal Logic) specification panel (referring to dashed area 2), the developer can specify bound *k* and compose custom formulae. Four, the prototype system verifies the properties automatically, and output the counterexamples to developer if the properties are unsatisfied. Moreover, commands such as *bmc_simulate*, *ltlspec_bmc, ltlspec_bmc*_onepb are also supported for generating the simulation trace, checking properties and analyzing counterexamples.

4 Related Works

As a popular application of Internet software, the Web application has promoted the development in e-commerce, e-education and e-government area. A number of modeling and verifying techniques for Web applications have already been proposed.

Navigation models such as [5,6,7] use statecharts notations. They model Web navigation, Web elements and the interactions among them when the user traverses Web applications. The models presented in [8,9] model the behavior of Web applications using FSMs. However, the Web browser interactions are not concerned.

Donini et al.[10] used Web Application Graph (WAG), a proposed mathematical model that partitions the usual Kripke structure into windows, links, pages and actions, to support for automated verification of the UML design of a Web application. Minmin Han et al. [11] used Statecharts to formally model adaptive navigation, and show how important properties of a navigation model are verified using existing model-checking tools. The verifying models above are all static models and not taken the browser interactions into account.

To check a given specification, model checking with BDDs (Binary Decision Diagrams) or ROBDD (Reduced Ordered Binary Decision Diagrams) technique has been pushed the barrier to systems with 10^{20} states and more [12] because it can't fully enumerate the reachable states of the system. This obstacle (or bottleneck) is the state grows exponentially. Practically, Bounded model checking (BMC) [12,13], it uses a SAT procedure instead of BDDs such as GRASP [14] and SATO [15], has attracted attention as an alternative to model checking, which seldom requires exponential space.

A related development has become the extension of BMC to timed systems [16,17]. It uses SAT solver to verify the location invariant constraints of the related time. A framework of abstraction/refinement [18] has been proposed. BDDs is used to prove the abstract model, and SAT solvers are used to check whether the counterexamples constructed in the abstract space are real or spurious, and also to derive a refinement to the abstraction being applied. The authors of [19] showed that BMC has a promise to verify a large class of hardware designs. However, to the best of our knowledge, the issues in bounded model checking Web application have seldom been addressed.

5 Conclusions

The Web application must satisfy very high requirements for reliability, availability and usability. In this paper, we propose an approach to modeling On-the-fly navigation models for Web application considering both the Web browser button click and the Web page hyperlink action, during which BMC is introduced to implement On-the-fly verification of Web navigation model for checking safety and liveness properties. Our future work includes checking data consistency of Web application and extending the navigation model for database interactions.

Acknowledgments. This work is supported by National Natural Science Foundation of China (NSFC) under grant No. 60970007 and 61073050, the National Grand Basic Research Program (973 Program) of China under grant No. 2007CB310800, the Natural Science Foundation of Shanghai Municipality of China under Grant No.09ZR1412100, Project of Science and Technology Commission of Shanghai Municipality under Grant No. 10510704900 and Shanghai Leading Academic Discipline Project (Project Number: J50103).

References

[1] Nielsen, J.: Hypertext and Hypermedia. Academic Press, Inc., San Diego (1990)

[2] Di Lucca, G.A., Di Penta, M.: Considering Browser Interaction in Web Application Testing. In: Proceedings of the 5th IEEE International Workshop on Web Site Evolution, pp. 74–78. IEEE Press, New York (2003)

[3] Chen, S., Miao, H., Qian, Z.: Modeling and Verifying Web Browser Interactions. In: Proceedings of the 15th Asia-Pacific Software Engineering Conference (APSEC 2008), pp. 351–358. IEEE Press, New York (2008)

[4] Clarke, E., Biere, A., Raimi, R., Zhu, Y.: Bounded Model Checking Using Satisfiability Solving. Journal Formal Methods in System Design 19(1), 7–34 (2001)

[5] Turine, M.A.S., de Oliveira, M.C.F., Masiero, P.C.: A Navigation-oriented Hypertext Model Based on Statecharts. In: Proceedings of the 8th ACM Conference on Hypertext (Hyper-text 1997), pp. 102–111. ACM Press, New York (1997)

[6] Leung, K.R.P.H., Hu, L.C.K., Yiu, S.M., Tang, R.W.M.: Modeling Web Navigation by Statechart. In: Proceedings of the 24th Annual International Computer Software and Applications Conference (COMPSAC 2000), pp. 41–47. IEEE Computer Society Press, Washington, DC, USA (2000)

[7] de Oliveira, M.C.F., Turine, M.A.S., Masiero, P.C.: A Statechart-based Model for Hypermedia Applications. Journal ACM Transactions on Information Systems(TOIS) 19(1), 28–52 (2001)

[8] Andrews, A.A., Offutt, J., Alexander, R.T.: Testing Web Applications by Modeling with FSMs. Software and Systems Modeling 2005(4), 326–345 (2005)

[9] Dargham, J., Al-Nasrawi, S.: FSM Behavioral Modeling Approach for Hypermedia Web Applications: FBM-HWA Approach. In: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), pp. 199–204. IEEE Computer Society Washington, DC, USA (2006)

[10] Doninia, F.M., Mongiellob, M., Rutac, M., Totarod, R.: A Model Checking-based Method for Verifying Web Application Design. Electronic Notes in Theoretical Computer Science 151(2), 19–32 (2006)

[11] Han, M., Hofmeister, C.: Modeling and verification of adaptive navigation in web applications. In: Proceedings of the 6th international Conference on Web Engineering (ICWE 2006), vol. 263, pp. 329–336. ACM, New York (2006)

[12] Biere, A., Cimatti, A., Clarke, E.M., Strichman, O., Zhu, Y.: Bounded model checking. Advances in Computers 58, 118–149 (2003)

[13] Biere, A., Cimatti, A., Clarke, E.M., Zhu, Y.: Symbolic Model Checking without BDDs. In: Cleaveland, W.R. (ed.) TACAS 1999. LNCS, vol. 1579, pp. 193–207. Springer, Heidelberg (1999)

[14] Marques-Silva, J.: Search Algorithms for Satisfiability Problems in Combinational Switching Circuits. Ph.D. Dissertation, EECS Department, University of Michigan (1995)

[15] Zhang, H.: SATO: An Efficient Propositional Prover. In: McCune, W. (ed.) CADE 1997. LNCS, vol. 1249, pp. 272–275. Springer, Heidelberg (1997)

[16] Wozna, B., Zbrzezny, A.: Checking ACTL Properties of Discrete Timed Automata via Bounded Model Checking. In: Larsen, K.G., Niebert, P. (eds.) FORMATS 2003. LNCS, vol. 2791, pp. 18–33. Springer, Heidelberg (2004)

[17] Audemard, G., Cimatti, A., Kornilowicz, A., Sebastiani, R.: Bounded model checking for timed systems. In: Peled, D.A., Vardi, M.Y. (eds.) FORTE 2002. LNCS, vol. 2529, pp. 243–259. Springer, Heidelberg (2002)

[18] Clarke, E., Gupta, A., Kukula, J., Strichman, O.: SAT Based Abstraction-Refinement Using ILP and Machine Learning Techniques. In: Brinksma, E., Larsen, K.G. (eds.) CAV 2002. LNCS, vol. 2404, pp. 265–279. Springer, Heidelberg (2002)

[19] Baumgartner, J., Kuehlmann, A., Abraham, J.: Property Checking via Structural Analysis. In: Brinksma, E., Larsen, K.G. (eds.) CAV 2002. LNCS, vol. 2404, pp. 151–165. Springer, Heidelberg (2002)

A Metric-Based Approach for Anti-pattern Detection in UML Designs

Rahma Fourati, Nadia Bouassida, and Hanêne Ben Abdallah

Abstract. Anti-patterns are poor solutions of recurring design problems, which decrease software quality. Numerous anti-patterns have been outlined in the literature as violations of various quality rules. Most of these anti-patterns have been defined in terms of code quality metrics. However, identifying anti-patterns at the design level would improve considerably the code quality and substantially reduce the cost of correcting their effects during the coding and maintenance phases. Within this context, we propose an approach that identifies anti-patterns in UML designs through the use of existing and newly defined quality metrics. Operating at the design level, our approach examines structural and behavioral information through the class and sequence diagrams. It is illustrated through five, well-known anti-patterns: Blob, Lava Flow, Functional Decomposition, Poltergeists, and Swiss Army Knife.

1 Introduction

Design patterns [11] propose "good" solutions to recurring design problems. To benefit from design patterns, a developer must have a thorough understanding of and a good practice with design patterns in order to identify the appropriate patterns to instantiate for his/her application. The absence of such high-level of expertise often results in *anti-patterns*. In fact, anti-patterns [19] describe commonly occurring solutions to problems but generate negative consequences on the quality of object-oriented software in terms of quality factor (so called in the ISO 9126 norm) as complexity, reusability. Evidently, anti-patterns detection and correction improves substantially the software quality. This benefit motivated several researchers to propose assistance for inexperienced designers through the detection of anti-patterns, *cf.*, [13] [14], [7], [4] [15] and [2]. Existing propositions differ mainly in: *i)* the level they consider (code *vs.* design level); and *ii)* their objectives

Rahma Fourati · Nadia Bouassida · Hanêne Ben Abdallah
Miracl Laboratory, University of Sfax, Tunisia
e-mail: rahma.fourati10@gmail.com, nadia.bouassida@isimsf.rnu.tn, hanene.benabdallah@fsegs.rnu.tn

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 17–33.
springerlink.com

(code improvement through re-engineering for maintenance purposes, *cf.*, [2], [15], [4], [13] and [14] *vs.* design improvement, *cf.*, [17], and [3]).

In reality, the code is rich with information useful for the detection of anti-patterns, *e.g.,* variable instances used by a method, number of lines in the code, comments and global variable used. However, the detection of anti-patterns at the code level is considered too late and may not reduce the correction cost. Hence, in our work, we are interested in anti-pattern detection at the design level. During the design phase, a designer (inexperienced with design patterns) may inadvertently specify a design fragment that "resembles" a design pattern. The resemblance can be manifested either as an anti pattern, or as a "poor/bad" design solution. Adopting our anti-pattern detection approach at this phase helps the designer in improving the quality of his/her design, henceforth the code.

In this paper, we propose a metric-based approach for anti-patterns detection in UML designs, *i.e.,* the class and sequence diagrams. In our detection approach, we have selected a set of most pertinent metrics from the works of [16] and [9]. To these metrics, we added a set of specific metrics useful for anti-patterns detection.

The remainder of this paper is organized as follows: Section 2 overviews currently proposed approaches for anti-pattern identification. Section 3 first presents the OO software metrics used and, secondly, introduces our approach. Section 4 illustrates our approach through an example of a Functional Decomposition anti-pattern [19]. Section 5 summarizes the paper and outlines our future work.

2 Related Works

Several works have tackled software (code) quality through various techniques. Amongst these works, design patterns and heuristics have been considered as the most promising approaches. In fact, Gamma [11] introduced *design patterns* as "good" solutions that can be instantiated and composed to produce software faster; being agreed up-on solutions, design patterns guarantee the good-quality of the produced software. In addition, Riel [1] presented more than sixty guidelines as object oriented *heuristics* to evaluate manually existing software and to improve its quality. On one hand, design patterns and heuristics are considered as good design practices. On the other hand, *anti-patterns* and *bad smells* are considered as results of bad design practices. Bad smells are code-level symptoms indicating the possible presence of an anti-pattern (also called '*Design Flaw'* [12]). Anti-patterns and bad smells are sometimes merged into one term: *design defects* [13] [14]. However, bad smells are fine-grained and strongly linked to the code level; on the other hand, anti-patterns are coarse-grained and can be represented at the design level.

Since we are treating quality at the design level, in this paper, we are interesting in detecting only anti-patterns. We next overview the definitions of anti-patterns and then discuss current identification approaches.

2.1 Anti-pattern Definition

Anti-patterns are *bad solutions* to recurring design problems. For example, the Blob, also called God class [1], corresponds to one single complex controller class that monopolizes the processing and that is surrounded by simple data classes [19]. Brown [19] defines the relation between patterns and anti-patterns. He indicates that, patterns are applicable in a well-known context, while anti-patterns are structures that appear to be beneficial, but produce more bad consequences than good ones. Furthermore, Dodani [11] added that developing patterns is a bottom-up process, whereas developing anti-patterns is a top-down process. He insisted that we should learn from our 'mistakes' which are anti-patterns.

On the other hand, while design patterns are well-described/documented thanks to class diagrams [8], anti-patterns presented in the literature are only informally described in natural language. This hinders the development of CASE tools to assist in their detection. Nevertheless, several detection approaches have been proposed. We next discuss the most complete approaches.

2.2 Anti-patterns Detection Approaches

Current approaches for the identification of anti-patterns operate either at the code level (for software re-engineering purposes), or at the design level (for design quality improvement purposes).

2.2.1 Code Level

Marinescu [15] [4] proposed a semi-automatic approach that detects design flaws [12] through a set of metric-based rules which called them 'detection strategy'. The rules first analyze the code to separate correlated symptoms (e.g., excessive method complexity, high coupling) that can be measured by a single metric. Secondly, for each symptom, the rules apply filters (*e.g.,* "HigherThan") proposed to detect various flaws. This approach was shown experimentally capable to detect ten design flaws with the tool named iPlasma; the evaluation is presented through precision, while the recall rate was ignored. One advantage of this work is that it uses correlated metrics instead of separate metrics that produce individual measurements whose interpretation does not reflect the cause of a flaw. In addition, Marinescu specified a process to organize the detection strategy. This process facilitates the passage from the description of a design flaw to a detection strategy.

Trifu et al. [2] also proposed an approach to detect design flaws. They used a tool called "jGoose" that creates a design database with relevant information about the system's source code. The design DB contains information about all structural design entities of the system such as classes, methods and attributes along with their relations. Moreover, the design DB contains the values of some basic metrics such as control flow, complexity of methods ([18]) and lines of code. This design DB is stored as an XMI model accessed through the standard query language XQuery to find design flaws. An important aspect in the approach of Trifu is that he related the quality factor to the design flaw. However, this work showed how to detect only design flaws that have a well known form expressible through the

handled metrics. Furthermore, expressing the rule into a query is complex and the authors do not provide for an assistance means.

Alikacem et al. [7] proposed an approach to detect violations of quality rules in object-oriented programs. For this, they classified quality rules into three categories: metric-based rules, structural information-based rules and rules expressing abstract notions. They defined a meta-model for representing the source code and a language to express a quality rule and its metrics independent of the programming language (java, C...). Moreover, they used fuzzy thresholds during the application of the quality rule to decide up-on the quality of the source code. The advantages of this proposition are pending a validation.

Moha [13] [14] combined the idea of fuzzy thresholds proposed by Alikacem et al. [7] and the idea behind the detection strategy presented by Marinescu [15]. She proposed to specify each design defect by a rule card. The rule card is a specification of a defect in terms of their measurable, structural and lexical properties. Given the rule card of a given design defect, the approach of Moha generates automatically the algorithm for detecting it. One advantage of this Moha's work is that the passage from the specification of a rule card to the algorithm is transparent. A second advantage is that Moha's work is the first complete approach containing both detection and correction of the detected design defects. Moreover, the efficiency of this approach depends on the designer capacity in defining manually the rule card necessary to detect a particular design defect. While correcting in the code source, the traceability between design and implementation will be lost (i.e. the code source does not reflect the design since it is modified).

We remark that all these previous works do not operate directly on the code source but they define a language or a model to represent the source code in order to make its manipulation easier. Note that, this supplementary step is not necessary when working at the design level.

2.2.2 Design Level

Besides the source code level, other works detect anti-patterns at the design level, *cf.,* [17], [5] and [6]. For example, Grotehen *et al.* [17] proposed an approach named METHOOD that checks the conformity of a set of heuristics ([1]) using measures like: size, hiding, coupling and cohesion. In addition to their detection, heuristics deviations can be corrected in METHOOD by a transformation rule which proposes an alternative fragments that respect the set of metrics used by the violated heuristic. However, in general an anti-pattern violates more than one heuristic, thus its detection is more complex. Finally, note that while METHOOD has been implemented in a tool named MEX (MEthood eXpert), this approach has not been evaluated experimentally.

Ballis et al. [5] [6] detect both patterns and anti-patterns using a rule-based matching method to identify all instances of a pattern/anti-pattern in the graph which underlies the designer's diagram [5]. A pattern/anti-pattern is defined either textually or in a graphical language that extends UML by adding few graphical primitives [6]. The strong point of this approach is that it allows the designer either to redefine easily the descriptions of canonical patterns (anti-patterns) to specify his/her customizations, or to define new ones from scratch. Evidently, the

graphical notation can specify only patterns/anti-patterns with a well-defined structural form.

Our contribution consists in: i) detecting anti-patterns at the design level, and ii) modeling both the structure and the behavior of five anti-patterns. Detecting anti-patterns at the design level allows the designer to anticipate the problems that could be generated by an implementation. It is true that, when moving from the code to the design, we lose information. But we compensate this loss by using the sequence diagrams (which highly reflects the dynamic information in the code) and by defining new metrics. Another distinction of our approach is that it exploits the semantics carried by the names; this information can easily characterize several aspects that are not captured through design metrics. Finally, similar to Moha's approach, our detection approach can be extended to assist in the correction step since it extracts a clear and significant report of the detected anti-pattern.

3 A New Anti-pattern Identification Approach

Our anti-pattern detection approach is based on OO software metrics used to measure quantifiable properties. In this section, we begin by listing a set of the most pertinent, existing metrics used in the detection of anti-patterns. Afterwards, we describe the metrics we added. Finally, we explain our detection rules which use the metrics for the class and sequence diagrams and rely on structural, behavioral and semantic information.

3.1 Useful Existing and New Metrics

Several metrics have been defined in the OO software engineering field. We selected the most important metrics and we classified them according to four categories: Coupling, cohesion, complexity and inheritance. Next, we explain each category and we present its associated metrics.

3.1.1 Coupling

Coupling measures the degree of interdependency between classes/objects. Two objects X and Y are coupled if at least one of them acts upon the other, *i.e.,* there is at least one method in X that calls methods or uses instance variables in Y and vice versa [16]. Coupling could, essentially, be measured with the following two metrics:

- **CBO** (Coupling Between Objects): *"The CBO for a class is a count of the number of other classes to which it is coupled"* [16]. The CBO value should be minimized. In fact, when it increases, the sensibility to changes is higher and therefore maintenance is more difficult [16].
- **RFC** (Response For Call) is the cardinality of the response set containing the methods called by each method in the class and the set of methods defined in the class. The larger the number of methods that can be invoked by a class, the greater the complexity of the class is [16].

3.1.2 Cohesion

Cohesion is a measure of how strongly-related and focused the various responsibilities of a class. A cohesive class is a class all of whose methods are tightly related to the attributes defined locally. The cohesion should be maximized to get a design with a good quality. The essential metrics measuring cohesion are:

- **LCOM** (Lack Of Cohesion in Methods)**:** It counts the number of method pairs whose similarity is zero, minus the count of method pairs whose similarity is not zero; the similarity of a pair of methods is the number of joint instance variables used by both methods. The larger the number of similar methods, the more cohesive the class is [16].
- **TCC** (Tight Class Cohesion)**:** It measures the cohesion of a class as the relative number of directly connected methods, where methods are considered to be connected when they use at least one common instance variable [9].
- **LCC** (Loose class Cohesion)**:** It counts, for each class, the percentage of method pairs related either directly or indirectly [9].
- **Coh** for a class with N methods M(1)…M(N) with N sets of parameters I(1)…I(N) and M is the number of the disjoint sets of parameters formed by the intersection of these N sets [10], then

$$Coh=M/N * 100\%$$

Note that Coh is insufficient to quantify the cohesion (useless), when most of the methods do not have parameters. In addition, LCOM, TCC and LCC are based on "instance variables used by methods" while this information is not available during the design phase. For this reason, we use the 'Coh' metric and we rely on semantic information.

3.1.3 Complexity

Complexity measures the simplicity and understandability of a design. Many complexity measures have been proposed in the literature, amongst which we find:

- **WMC** (Weighted Methods per Class)**:** This metric determines the complexity of a class by summing the complexities of its methods. Note that, WMC cannot be applied at the design level since the code of the method is not available.
- **NAtt:** the Number of the Attributes of a class [13].
- **NPrAtt:** the Number of Private Attributes [13].
- **NOM:** the Number Of Methods of a class including the constructor [13].
- **NII:** the Number of Imported Interfaces [13].

3.1.4 Inheritance

Inheritance measures the tree of inheritance and the number of children. In this category, we find:

- **DIT** (Depth of Inheritance of a class) is the depth in the inheritance tree. If multiple inheritances are involved, then the DIT will be the maximum length from the node to the root of the tree [16].
- **NOC** (Number Of Children) is the number of immediate subclasses subordinated to a class in the class hierarchy [16].

In addition to the above existing metrics, our detection approach uses the following new metrics for a class:

- **NAcc:** the Number of the Accessors in a class.
- **NAss:** the Number of Associations (association link, agregation, composition, dependency link) of a class.
- **NInvoc:** the Number of the Invoked methods (Call Action in the sequence diagram) of a class.
- **NReceive:** the Number of the Received messages that invoke methods of this class.

3.2 The Metric Threshold Issue

Choosing useful metrics is not enough to ensure an efficient detection; it is necessary to fix the threshold values which highly influence the efficiency of the detection process.We should caution that, even in the software engineering field, in general, there is not yet a precise guideline for how to fix thresholds and good interpretation of the proposed quality metrics.

Marinescu [15] proposed three means to fix metric thresholds. The first mean consists in using metrics from the literature with already predefined thresholds; this might require the adaptation of the thresholds to the system size [15]. In our case, there are several defined threshold metrics, but a few of them are validated; for instance, DIT is fixed to six according to [1]. In addition, to adapt such thresholds to the system size, the detection process would take more time since it needs to calculate the maximum size of each used metric (the maximum NAtt, the maximum NOM, etc). The second means to choose thresholds is to define a tuning machine which tries to find automatically the proper threshold values. In order to determine correct threshold values, this means requires a large repository of design fragments containing anti-patterns. Finally, the third means is to enhance the detection process by combining detection strategies applied on a single version with additional information about the history of the system. This means would be appropriate for evolving systems. However, when the design is being constructed for the first time, this means is not applicable. Marinescu [15] used in his detection strategy the boxplot technique which is a statistical method by which the abnormal values in a data set can be detected (http://en.wikipedia.org/wiki/Box_plot). On the other hand, Alikacem [7] resolved this problem by introducing fuzzy thresholds. In this case, the main goal is to encompass the exact value, for example a class with 19 or 21 methods will not be considered as a large class when the NOM is fixed to 20. As for Moha whose work combines the above two approaches, she used the boxplot and fuzzy thresholds at the same time.

In our approach, we try, on the one hand, to use already predefined thresholds and to let the designer parameterize the threshold value when necessary. Determining appropriate thresholds empirically is our ongoing work. In the next section, we assume that we have four thresholds delimiting each metric: very low, low, high and very high.

3.3 Detection

In this section, we present the detection of five anti-patterns: Blob, Lava Flow, Functional Decomposition, Poltergeists and Swiss Army knife. Table 1 lists the relationship between the different metric categories and these five anti-patterns. Note that, not all the anti-patterns defined by Brown [19] can be detected at the design level. For example, Spaghetti Code is a complex class containing methods without parameters and *using global variables*. The last symptom cannot be detected at the design level.

Table 1 Classification of anti-patterns

Anti-Patterns	Metrics Category			
	Coupling	Cohesion	Complexity	Inheritance
Blob	high	low	High	low
Lava Flow	low		High	
Functional Decomposition		high	Low	Very low
Poltergeists	high		low	
Swiss Army Knife	high		high	

The structural detection of an anti-pattern is insufficient. Behavioral information extracted from the sequence diagram and semantic information, in terms of anti-pattern class names and method names within the classes, are also required to confirm the presence of the anti-pattern and, hence, the necessity of a correction. Thus, we divide our anti-pattern detection process into three steps:

Structural anti-pattern detection: This step relies on the transformation of the class diagram into an XML document and a straightforward calculation of metrics such as NOM, NAtt. In fact, the transformation of UML diagrams into XML documents is rather trivial and can be handled by all existing UML editors.

Behavioral anti-pattern detection: Similar to the structural detection, this step also relies on the transformation of the sequence diagram into an XML document and metric calculation such as method calling, the sender, the receiver…

Semantic anti-patterns detection: After examining anti-pattern symptoms described in the literature, we found out that, in some cases the anti-pattern has not only quantifiable symptoms (structural and behavioral) but also semantic symptoms. It relies on linguistic information to identify the classes of the anti-pattern

problem. In fact, the determination of the semantic characteristics of anti-patterns can be handled through lexical dictionaries such as WordNet (http://fr.wikipedia.org/wiki/WordNet).

For the semantic identification, we use the following semantic information:

- **IsController(ClassName):** It determines if a class has a name as 'Controller', 'Main', or a synonym.
- **IsAccessor(MethodName):** It tests if a method is named as 'Get', 'Set' or their synonym.
- **Coh1(Att, Arg):** It determines if the argument 'Arg' is a synonym of the attribute 'Att'.
- **Coh2(Att, Meth):** It indicates if the name of the method 'Meth' contains the name of the attribute 'Att'.
- **IsFunctional(ClassName):** It tests if a class has a name as a function i.e. a verb or a noun action like 'Creation', 'Making',…
- **IsStart(MethodName):** It indicates if a method is named as 'Start','Initiate' or their synonym.

Table 2 shows the correlation between the symptoms and the category of metrics. It helps us to choose the convenient metrics for each anti-pattern.

Table 2 Correlation symtoms/metrics

Metrics	Symptoms Nature		
Category	Structural	Behavioral	Semantic
Coupling		CBO, RFC, NInvoc, NReceive	
Cohesion	Coh		Coh1, Coh2
Complexity	NPrAtt, NAtt, NOM, NAcc, NII, NAss		
Inheritance	DIT, NOC		

In the following sub-sections, we present our detection approach for the five anti-patterns in two categories: structurally by applying metrics on the class diagram, and both structurally and behaviorally through metrics applied on the class and sequence diagrams. We also present 'Correction proposition' which are useful later at the correction phase.

3.3.1 Structural Detection

Swiss Army Knife

Swiss army knife implements a *large number of interfaces* to expose the maximum possible functionalities. In addition, the classes associated to Swiss Army knife offer public interfaces. The Swiss Army Knife structure is illustrated in Fig. 1. The detection rule of this anti-pattern is: NII high.

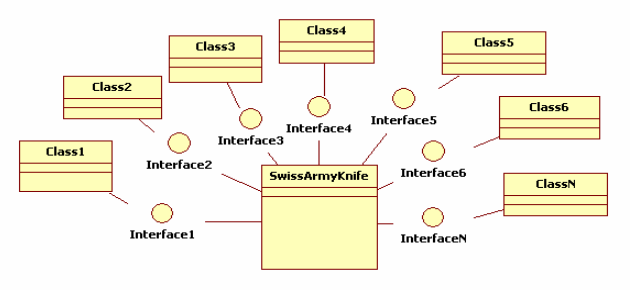


Fig. 1 Swiss Army Knife structure

3.3.2 Structural and Behavioral Detection

BLOB

The Blob anti-pattern consists in a class containing a large number of attributes and methods which makes it *complex*. This *large class* depends on the classes that surround it, and which are called *data classes*. A data class is a simple class that has only attributes and their method accessors (Fig. 2). The methods in the large class use attributes of the data classes (Fig. 3). Thus, the blob class is *highly coupled* with its data classes. In addition, the Blob anti-pattern contains methods that do not operate on local data, this makes it *non cohesive.*

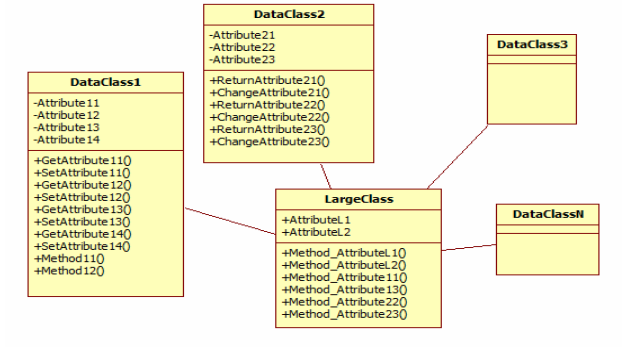


Fig. 2 Blob structure

We noticed that cohesion is necessary for detecting especially the blob; we overcome the insufficiency of the Coh metric by relying on the sequence diagram to find methods using attributes (by the accessors Fig. 3). In addition, we use the semantics for cohesion: we search sets of methods and attributes that have a common word or a term such as Title_Book and Borrow_Book().

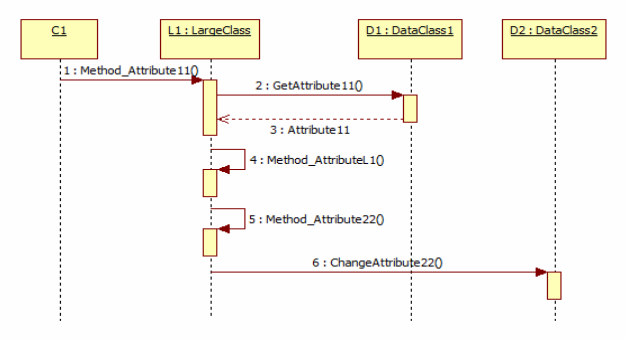


Fig. 3 A sample of a sequence diagram showing the low cohesion of a Blob

- **Large class:** NAtt high and NOM high and Coh low and Coh1 is true and Coh2 is true and DIT low and NOC low and RFC high and CBO high and IsController is true.
- **Data class:** NAcc high and NOM low and DIT low and NOC low and IsAccessor is true.

Correction proposition

- Methods of the large class should be moved to the data class.
- Methods and attributes of the large class should be moved to a new class.

Lava Flow

Lava Flow is known as a dead code. In terms of design, it consists in an isolated class which makes it uncoupled with others classes. Furthermore, on one hand, it is characterized by a large number of attributes and it is complex, so as a result its code is not clear and often developers cannot understand the main functionality of this class. On the other hand, it has no interactions, thus in general this class is isolated (i.e., it does not figure as an interacting object in any sequence diagram).

This anti-pattern can be detected through the following rule:

NAtt high and NOM high and NAss low and DIT zero and NOC zero and absence of interaction in the sequence diagram.

Functional Decomposition

This anti-pattern is materialized by classes having functional names (e.g., *Calculate_Interest*) and associations with cardinality "1". Moreover, all class attributes are private and used only inside the class; hence, there is a *high cohesion*. In general, it has classes with a *single action* such as a function, which makes them *simple* classes. Fig. 4 and Fig. 5 sketch the typical structure and the behavior of this anti-pattern, respectively. Note that, in the sequence diagram, when following the sequence of method calls, we observe that there is an invocation of one method on each class ensuring a chronological order.

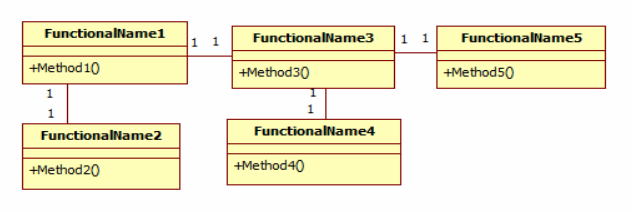


Fig. 4 Functional Decomposition structure

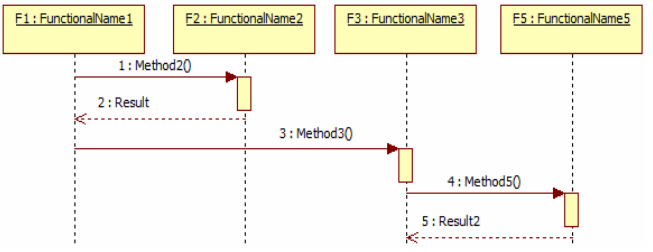


Fig. 5 A sample of a sequence diagram showing typical behavior of the Functional Decomposition anti-pattern

The Functional Decomposition can be identified through the following rule:

IsFunctional is true and NPrAtt low and NOM low and NAcc zero and DIT zero and NOC zero, some classes having NInvoc zero and other classes have successive invocations. Also the cardinality between classes has a value of "1".

Correction Proposition

- Classes having NInvoc zero should be merged with the class which invokes her method.
- Classes having successive invocations form the correct modelisation.

Poltergeists

This anti-pattern is characterized by a class containing a *single method* having *redundant navigation paths* to all other classes to 'seed' or '*invoke'* other classes through temporary associations (see Fig. 7). As a result, classes surrounding Poltergeists are *coupled* with it. Often, a Poltergeists is a class with 'control–like' method names such as *start_process* since it just initiates the system (Fig. 6).

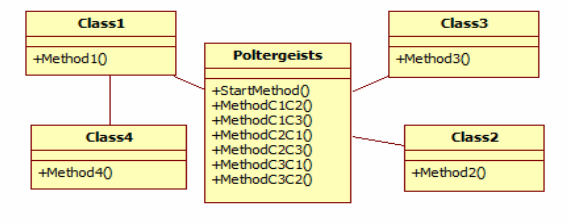


Fig. 6 Poltergeists Structure

When observing the sequence diagram, we find that all classes communicate with one another always through a central class which is the poltergeists (Fig. 7). In other word, many classes invoke method of the poltergeists which, in turn, invokes method of the destination class.

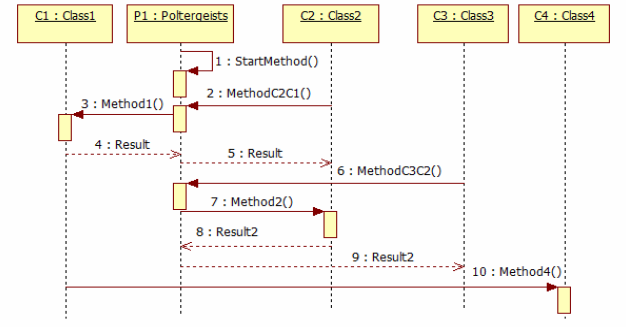


Fig. 7 A sequence diagram showing typical behavior and Centralized invocations of Poltergeists

We can identify this anti-pattern according to the following rule:

NAss high and NOM low and NInvoc=NReceive+1(the start method + methods for transient communication.) and IsStart is true.

Correction Proposition

- Class Poltergeist should be redesigned.
- Classes communicating through Poltergeist should be associated.

4 Illustration of Our Approach

In this section, we illustrate our anti-pattern detection approach through an example of a Functional Decomposition anti-pattern, adapted from the literature [19]. In fact, we added attributes and methods, essentially for clarity and understandability purposes. Fig. 8 presents the diagram class of the **Calculate Loan** example.

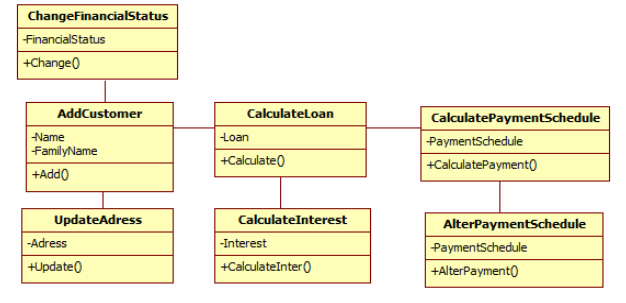


Fig. 8 Class diagram of the Calculate loan example

In addition, we specified the sequence diagram of the Calculate loan example by examining the documentation and scenario presented by Brown et al., [19]. Fig. 9 presents the sequence diagram illustrating the object interactions of the Calculate loan example. Next, we illustrate this scenario:

1. Adding a new Customer
2. Updating a Customer Address
3. Calculating a Loan to Customer
4. Calculating the Interest on a Loan
5. Calculating a Payment Schedule for a Customer Loan
6. Altering a Payment Schedule.

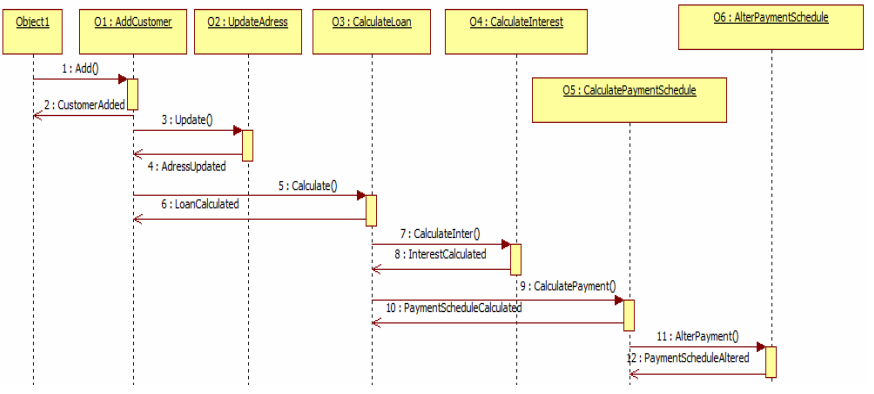


Fig. 9 Sequence diagram of the **Calculate loan** example

In order to detect the functional decomposition anti-pattern, the following 6 steps are applied:

Anti-pattern symptoms detection on the Class diagram

Step 1: Find the first class, having 'functional name' (IsFunctional is true).

Step 2: Follow the associated classes to this class having cardinality '1'.

Step 3: Repeat Step1 and Step2 to determine sets of related classes having 'functional name' (IsFunctional is true) and cardinality '1'.

Step 4: Calculate metrics NAtt, NOM, DIT, NOC and NAcc.

Anti-pattern symptoms detection on the sequence diagram

Step 5: Extract Classes that exist only to serve just one class. These classes are highly coupled and have to be merged together (NInvoc=0).

Step 6: Extract Classes that invoke methods of another class, just after receiving a message invoking one of its methods. Note that, these classes are related by association link.

The Results of the detection steps

Step 1, 2 and 3 : These steps detect the following sets of classes which have associations with cardinality "1" and which have functional names:

{ChangeFinancialStatus, AddCustomer}, {AddCustomer, UpdateAdress, CalculateLoan}, {CalculateLoan, CalculateInterest, AddCustomer, CalculatePaymentSchedule}, {CalculatePaymentSchedule, AlterPaymentSchedule, CalculateLoan}

Step 4: Table 3 shows the metric values relative to the set of classes already determined in the previous step.

Table 3 Calculated metrics on each class of the suspious Functional Decompossition

Class Name	Metrics				
	NPrAtt	NOM	DIT	NOC	NAcc
ChangeFinancialStatus	1	1	0	0	0
AddCustomer	2	1	0	0	0
UpdateAdress	1	1	0	0	0
CalculateLoan	1	1	0	0	0
CalculateInterest	1	1	0	0	0
CalculatePaymentSchedule	1	1	0	0	0
AlterPaymentSchedule	1	1	0	0	0

Step 5: Table 4 lists the sets of classes that interact only together.

Table 4 Result of Step 5

Class Name	Properties	
	NInvoc	ClassInvocator
UpdateAdress	0	AddCustomer
CalculateInterest	0	CalculateLoan
AlterPaymentSchedule	0	CalculatePaymentSchedule

Step 6: Now, we extract the sequence of ordered invocation methods, as illustrated in Fig.9. Add()→ Calculate()→ CalculatePayment()→ AlterPayment()

{AddCustomer, CalculateLoan, CalculatePaymentSchedule}

Finally, Fig. 10 shows the correction of the functional decomposition anti-pattern detected in the example of Fig. 8. Note that, we added the attributes and methods defined in each class.

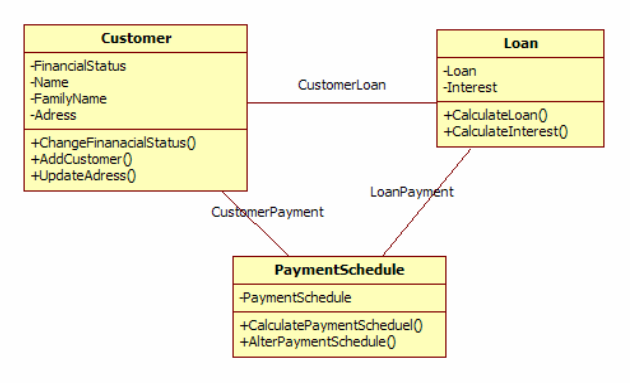


Fig. 10 The Correction of the **Calculate loan** example

5 Conclusion and Future Works

In this paper, we proposed an approach for detecting five anti-patterns described by Brown [19]. Working at the design level, our approach has the merit of anticipating anti-patterns at the code level and thus reduces their correction cost. Similar to existing approaches, ours relies on a set of existing and new design metrics defined for the class and sequence diagrams. It uses these metrics in a set of rules that, in addition, exploit the form of the anti-pattern such as the successive invocation in the Functional Decomposition anti-pattern. On the other hand, one main distinction of our approach from others is that it considers three types of information: structural and *semantic* from the class diagram, and behavioral from the sequence diagram.

Evaluated on several designs collected from the literature, our design approach produces very satisfactory levels of recall and precision. However, a thorough evaluation is underway to better place our approach amongst the existing approaches.

Furthermore, we are currently working on how to complement our detection approach with a correction phase. Here, we will exploit the report produced during the detection phase in terms of metrics, problematic classes, relationships and naming choices. Such detailed information will be used in a set of correction rules to produce alternative fragments. We will adapt the algorithm generation approach proposed by Moha [13].

References

1. Riel, J.: Object-Oriented Design Heuristics. Addison Wesley, Reading (1996)
2. Trifu, A., Seng, A., Genssler, T.: Automated design flaw correction in object-oriented systems. In: Proceedings of the 8th Conference on Software Maintenance and Reingineering (CSMR 2004), pp. 174–183. IEEE Computer Society Press, Los Alamitos (2004)

3. Bouhours, C., Leblanc, H.: Christian Percebois. Bad smells in design and design patterns. Journal of Object Technology 8, 43–63 (2009)
4. Marinescu, C., Marinescu, R., Mihancea, P.F., Wettel, R.: iPlasma: An integrated platform for quality assesment of object-oriented design. Loose Research group. Politechiea University of Tinoisoora (ICSM 2005), pp. 77–80. Society Press (2005)
5. Ballis, D., Baruzzo, A., Comini, M.: A rule-based method to match Software Patterns against UML Models. In: Proceedings of International Workshop on Rule Based Programming (RULE 2007), vol. 219, pp. 51–66. Theoretical Computer Science Press (2008)
6. Ballis, D., Baruzzo, A., Comini, M.: A Minimalist Visual Notation for Design Patterns and Antipatterns. In: Proceedings of the 5th International Conference on Information Technology: New Generations (ITNG 2008), April 2009, pp. 51–66 (2009) (in press)
7. Alikacem, E., Sahraoui, H.A.: Détection d'anomalies utilisant un langage de description de règle de qualité. In: Rousseau, R., Urtado, C., Vauttier, S.(eds.) 12 Conference on Languages and Models with Objects (LMO 2006), March 2006, pp. 185–200 (2006)
8. Gamma, E., Helm, R., Johnson, R., Vlissides, J.M.: Design patterns: Elements of reusable Object Oriented Software. Addisson-Wesley, Reading (1995)
9. Bieman, J.M., Kang, B.K.: Cohesion and reuse in an object oriented system. In: Proceedings ACM Symp., On Software Reusability, pp. 259–262 (1995)
10. Chen, J.Y., Lu, J.F.: A new metric for object-oriented design. Information and Software Technology 35, 232–240 (1993)
11. Dodani, M.: Patterns of antipatterns. Journal Of Object Technology 5, 29–33 (2006)
12. Fowler, M., Beck, K., Brant, J., Opdyke, W., Roberts, D.: Refactoring-Improving the Design of Existing Code, 1st edn. Addison-Wesley, Reading (1999)
13. Moha, N., Guéheneuc, Y.G., Leduc, P.: Automatic generation of detection algorithms for design defects. In: Proceedings of the 21st Conference on Automated Software Engi-neering (ASE 2006), September 2006, pp. 297–300. IEEE Computer Society Press, Los Alamitos (2006)
14. Moha, N., Guéhéneuc, Y.-G., Le Meur, A.-F., Duchien, L.: A domain analysis to specify design defects and generate detection algorithms. In: Fiadeiro, J.L., Inverardi, P. (eds.) FASE 2008. LNCS, vol. 4961, pp. 276–291. Springer, Heidelberg (2008)
15. Marinescu, R.: Detection strategies: Metrics-based rules for detecting design flaws. In: Proc. of the 20th International Conference on Software Maintenance (ICSM 2004), pp. 350–359. IEEE Computer Society Press, Los Alamitos (2004)
16. Chidamber, S.R., Kemerer, C.F.: A metric suite for object oriented design. IEEE Transactions on Software Engineering 20, 476–493 (1994)
17. Grotehen, T., Dittrich, K.R.: The MeTHOOD Approach: Measures, Transformation Rules, and Heuristics for Object-Oriented Design. Technical Report: ifi-97.09 (1997)
18. McCabe, T.J.: A Complexity Measure. IEEE Transactions on Software Engineering 4, 308–320 (1976)
19. Brown, W.J., Malveau, R.C., McCormick, H.W., Mowbray, T.J.: Antipatterns: Refactorin Software, Architectures, and Projects in Crisis, 1st edn. John Wily and Sons, West Sussex (1998)

A Novel Feature Extraction for Facial Expression Recognition via Combining the Curvelet and LDP

Juxiang Zhou, Tianwei Xu*, Yunqiong Wang, Lijin Gao, and Rongfang Yang

Abstract. In this paper a novel feature extraction approach is proposed for facial expression recognition by using the curvelet and the LDP (Local Directional Pattern). First, the low frequency coefficients of Curvelet decomposition on expression region are selected as global facial features. Then, LDP descriptor is used to describe eyes region and mouth region respectively as local facial features. Finally we obtain the fusion of these two different miscellaneous features for facial expression and this combination makes use of Curvelet and LDP their respective characteristics in global and local features simultaneously. A simple NN (Nearest Neighbor) classifier is used on the JAFFE and Cohn-Kanade these two benchmark databases to show the effectiveness.

1 Introduction

With broad applications of the image analysis and pattern recognition technology, facial expression recognition has attracted more and more attention. It is not only important to human-computer interaction, but also an significant component of the artificial psychology and the research in emotion computing [1].The full process of facial expression recognition includes three main stages: images acquisition, feature extraction and expression classification, and where feature extraction is a vital step because the performance of a method depends on whether an effective and discriminative facial representation is derived.

From the view of facial features, there are two common approaches to extract facial features: geometric feature-based methods and appearance-based method

Juxiang Zhou · Tianwei Xu · Yunqiong Wang · Lijin Gao · Rongfang Yang
College of information
Yunnan Normal University
Kunming, China
e-mail: zjuxiang@126.com, xutianwei@ynnu.edu.cn

* Corresponding author.

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 35–46.
springerlink.com

[3][4][5]. The Geometric feature-based methods extract the shape and locations of facial components (including mouth, eyes, and nose) to represent the face geometry [6][7]. Appearance-based methods deal with the whole face or specific face-regions to extract appearance changes of face using image filters such as Gabor-wavelet and LBP [8][9]. Some research suggests that, geometric feature-based approaches provide similar or better performance than appearance- based methods [10]. However, the geometric feature-based methods usually require accurate and reliable facial feature detection and tracking, which is difficult to accommodate in practical applications [9]. Hence, the appearance-based methods are the most widely used at present.

Among the appearance-based feature extraction methods, Gabor-wavelet methods are used broadly. But some findings are made in the process of applications that wavelet transform is unable to express directions of image edges accurately and also cannot achieve sparse representation of images. So, it is difficult for wavelet to express the important information about facial contours and the curves of facial feature. In order to overcome these limitations of wavelet transform, a new multi-scale analysis tool-Curvelet transform, is proposed recently [13]. Curvelet transform treats curves as basic representation elements and has strong directivity, which is beneficial to high-efficiency expression of image. In recent years, some researchers attempted to use Curvelet features in face recognition and has achieved satisfactory results [2][25].

As one of another potential appearance-based feature extraction method, LBP has many applications in the field of face detection, face recognition and facial expression recognition [8][14]. Although LBP has advantages of efficient computation and robustness to monotonic illumination changes, it is sensitive to non-monotonic illumination variation and also shows poor performance in the presence of random noise [15]. In order to overcome this weakness of LBP, Taskeed Jabid and others proposed LDP (Local directional pattern) method [15] which was applied successfully in face recognition, object description, gender recognition and facial expression recognition. [16][17][18].

Motivated by the fact that combining appearance and geometric features is a promising way for better facial representation [3], we propose a novel method for feature extraction in this paper by considering of the good characteristics of curvelet transform and LDP. That is to combine the global curvelet features and local LDP features effectively. On one hand, we extract the low curvelet features after decomposition on the whole face as global facial features, which can express the important information about facial contours and the curves of facial feature in a whole; On the other hand, we use LDP to describe the special regions of eyes and mouth, which have important contribution to expression variance. In this way, it not only grasps the principal directional information of local regions but also avoids to using geometric feature-based methods to express the local features. Our proposed method was verified on the JAFFE and Cohn-Kanade these two benchmark databases with NN classifier. The results show that the proposed method can achieve higher classification accuracy and have better performance for low-resolution images.

The structure of the paper is organized as follows. In section 2, we will introduce the curvelet transform and Local Directional Pattern first, and then describe the process of features fusion and the facial expression recognition algorithm. In section 3, we will do extensive experiments on two benchmark datasets and analyze the results, and the conclusions are given in section 4.

2 Facial Featrues Extraction

2.1 Curvelet Transform

Candes and Donoho first proposed the curvelet transform based on the ridgelet transform in 1999, i.e., the first generation of curvelet transform [19]. Within a few years, the theoretical research of curvelet transform has good development and it has wide application mainly in the field of image-denoising and image fusion. In order to make the implementations simpler, faster and less redundant, the second generation of curvelet transform is proposed in 2006 [13]. Curvelet transform is an effective analytical method for multi-resolution, band pass and directions which are considered as three important characteristics the "optimal" image representation should have from the perspective of biological point of view. Therefore, the curvelet transform has better representation capability than wavelet transform for image edges. Figure 1 shows the scale and angle segmentations in curvelet transform and the coefficients in the wedge (represented in the shaded area in Fig.1.) represent both angle and scale information as explained in [13][20].

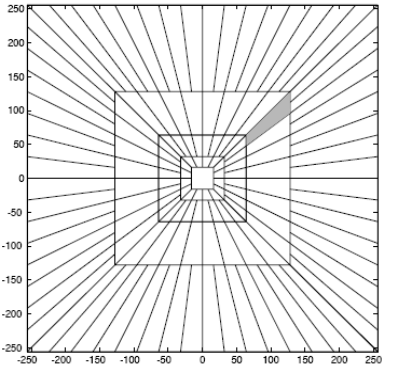


Fig. 1 The scale and angle segmentations of Curvelet transform

In fact Candes and Donoho proposed two different digital implementations for curvelet: one is based on the USFFT (Unequally Spaced Fast Fourier Transform) and the other is based on wrapping idea [13]. We select the one with wrapping because its high speed, and the algorithm is follows:

a) Apply the *2D* discrete Fast Fourier Transform and obtain the Fourier samples $\hat{f}[n_1,n_2],\ -n/2 \le n_1, n_2 \le n/2$

b) In frequency domain, for each pair *(j,l)*(scale, angle), resample $\hat{f}[n_1,n_2]$, and obtain the sampled values $\hat{f}[n_1,n_2-n_1\tan\theta_l]$ *for* $(n_1,n_2)\in p_j$

c) Multiply the interpolated object $\hat{f}$ with the parabolic window $\tilde{U}_j$, and obtain $\hat{f}[n_1,n_2]=\hat{f}[n_1,n_2-n_1\tan\theta_l]\tilde{U}_j[n_1,n_2]$

d) $\hat{f}[n_1,n_2]$ is wrapped around the origin.

e) Apply the inverse *2D* FFT to each $\hat{f}_{j,l}$, hence collecting the discrete coefficients $C^D(j,l,k)$

2.2 Local Directional Pattern

1) LBP (Local binary pattern)

The LBP as a gray-scale invariant texture primitive has gained significant popularity for describing the texture of an image [18]. The LDP value of every pixel in image is defined by

$$LBP_{P,R}(x_c,y_c)=\sum_{p=0}^{p-1}s(g_p-g_c)\cdot 2^p,\ s(x)=\begin{cases}1, & x\ge 0,\\ 0, & x<0,\end{cases} \tag{1}$$

where g_c denotes the value of the center pixel (x_c,y_c) and g_p corresponds to the gray value of equally spaced pixels *P* on the circumference of a circle with radius R [9][18]. As shown in Fig.2., it labels each pixel of an image by thresholding its *P*-neighbour values with center value and converts the result into a binary number by using (1).

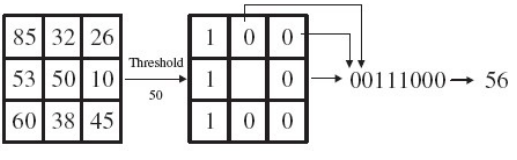


Fig. 2 The basic LBP operator

2) LDP(Local Directional Pattern)

The LBP operator encodes the micro-level information of edges, spots and other local characteristic in an image using information of intensity changes around pixels [18]. Motivated by the fact that finding that the gradient magnitude value at a pixel position has less sensitive to non-monotonic illumination variation than its

intensity value, Taskeed Jabid et al. proposed LDP Code which has more stability when describing different type of image features and preserves more image information [16][17].

The LDP descriptor is an eight bit binary code assigned to each pixel of an input image that can be calculated by comparing the relative edge response value of a pixel in different directions. So that eight directional edge response values $\{m_i\}$, $i = 0,1,\hbar\ 7$ of a particular pixel are computed using Kirsch masks in eight different orientations M_i centered on its own position. These Kirsch masks are shown in the Fig.3, and Fig. 4 shows eight directional edge response positions and LDP binary bit positions. Because different importance of the response values, the k most prominent directions are considered to generate the LDP. So the top k values $|m_j|$ are set to 1, and the other positions are set to 0. Finally, the LDP code is derived by formula (2), where m_k is the k-th most significant directional response value. Fig.5 shows an exemplary LDP code with k=3.

$$LDP_k = \sum_{i=0}^{7} b_i(m_i - m_k)\cdot 2^i,\ b_i(a) = \begin{cases} 1, & a \geq 0, \\ 0, & a < 0, \end{cases} \tag{2}$$

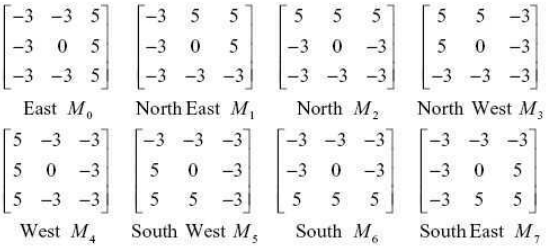


Fig. 3 Kirsch edge response masks in eight directions

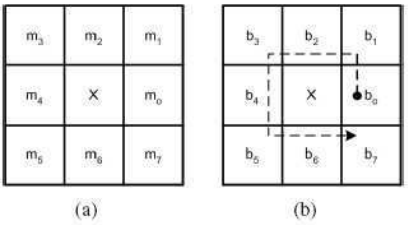


Fig. 4 (a) Eight directional edge response positions; (b) LDP binary bit positions

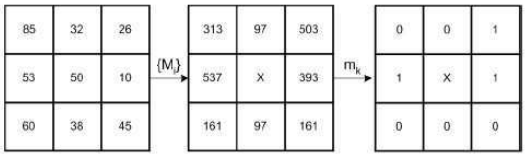


Fig. 5 LDP Code with k=3

Since edge responses are more stable than intensity values, LDP provides the same pattern value even if there is some presence of noise and non-monotonic illumination changes [17]. Stability of LDP vs. LBP on a small image patch is showed in Fig.6.

The input image of size $M \times N$ can be represented by an LDP histogram H using (3) after computing all the LDP code for each pixel (r,c), where i is the LDP code value.

$$H(i) = \sum_{r=1}^{M}\sum_{c=1}^{N} f(LDP_k(r,c),i),\ f(a,i) = \begin{cases} 1 & a = i, \\ 0, & a \neq i, \end{cases} \tag{3}$$

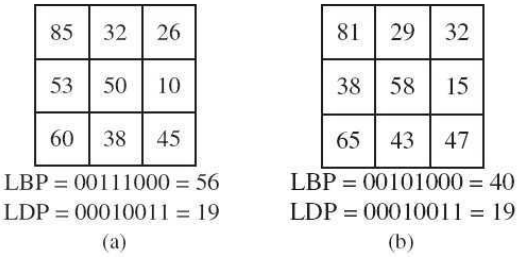


Fig. 6 Stability of LDP vs. LBP(a) Original Image with LBP and LDP code; (b) Image with noise with new LBP and LDP code

For a particular value *k,* there has C_8^k different number of bins for the histogram *H*. In essence, a resulting histogram vector size of $1 \times C_8^k$ is produced for the image.

LDP descriptor contains detail information of an image, such as edges, spots, corner, and other local textures [18]. Whereas computing LDP over the whole face image only considers the occurrences of micro-pattern without any information of their location and spatial relationship which usually represents the image content better. Hence, the image is divided into *g* regions $R_0, R_1, \hbar\ R_{g-1}$ as shown in Fig.7

when using LDP, so that there will be a LDP^i histogram for every region R_i. Consequently, the resulting LDP descriptor is obtained via concatenating all the LDP^i histograms.

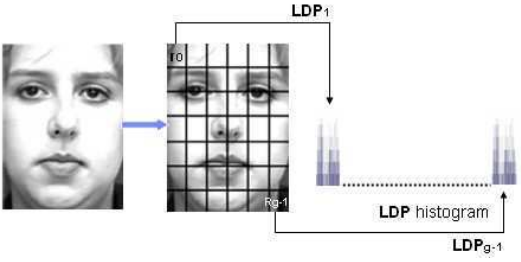


Fig. 7 Expression image is divided into small regions from which LDP histograms are extracted and concatenated into LDP descriptor

2.3 Features Fusion

Feature extraction is a key step in the whole process of facial expression recognition. Usually eyes and mouth have great influence on expression variation, while most existing appearance-based method only considers the whole face. The proposed method in this paper combines global features of the whole face with local features of eyes and mouth effectively in considering of their different contributions to expression variance.

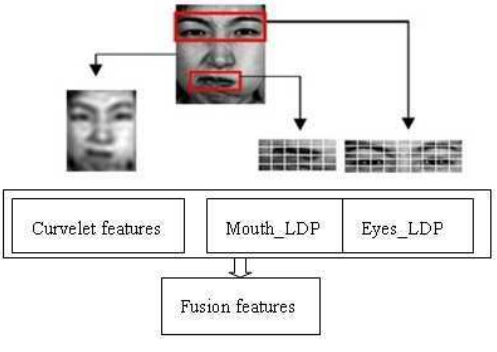


Fig. 8 The process of features fusion

First, we conduct the curvelet decomposition on expression image in four scales. The coefficients of the first scale after decomposition are low frequency, and others are every high frequency in sequence among which the energy reduces progressively [20]. Seeing that the low frequency has most energy and it can greatly reduce and represent the essential characteristics of a face image which is shown in [2][25], we

select the coefficients of the low frequency as global facial expression features. Second, LDP is used to describe the regions of eyes and mouth respectively. In order to contain more characteristic information of location and spatial relationship, these two regions are divided into appropriate sub-regions. Then we concatenate their LDP histograms as local facial expression features. Finally, we combine the global curvelet features with local LDP features as final facial expression features. The process of features fusion is shown in Fig.8.

2.4 Features via Dimensionality Reduction and Expression Classification

After combining these two different types of features, we get a group of features with high dimension. So, PCA is used to reduce the features dimension to ease the complicated computation. Here, in consideration of reserving more distinct characters of each global and local feature, we applied two-stage PCA reduction. That is, reducing global curvelet features and local LDP features separately first, and then used PCA again to reduce the concatenating reduced features. Because our essential is feature extraction, the NN classifier is used for recognition in classification stage. But the difference is that we applied LDA transform before classification which can further reduce and optimize the features and improve the performance of recognition.

3 Experiment and Result

3.1 Expression Images Acquisition

We evaluated the proposed method on two benchmark databases: the JAFFE databases [22] and the Cohn-Kanada databases [23]. The JAFFE database contains 213 gray images (256x256) of individual human subjects with a variety of facial expressions. In this database, 10 different Japanese female performed seven prototypical expressions: *anger, disgust, fear, joy, sadness, surprise* and *neutral*. We choose three samples per facial expression for each subject, and a total of 210 images among which every expression has 30 images. The Cohn-Kanada database includes video sequences of 97 subjects displaying distinct facial expression. We create a subset with 10 subjects for our experiments. All the subjects selected have six basic expressions: *anger, disgust, fear, joy, sadness* and *surprise*. From every sequence for each expression of a subject, we select the last four frames as static gray images (640x490). So there are 240 total images in all. After choosing the images, they were cropped from the original one using the position of two eyes and resized into 150×110 pixels [18]. The resulting expression images are shown in Fig.9.

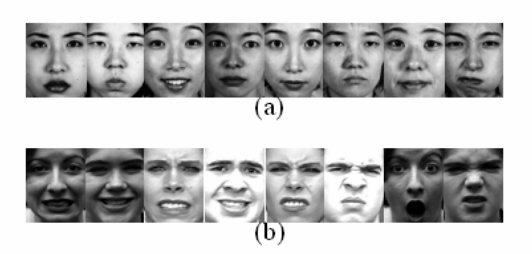


Fig. 9 Some expression samples from (a) JAFFE and;(b) Cohn-Kanade

In our experiment, in order to verify the effectiveness of various methods with different numbers of training samples,15,20 and 25 images per expression were selected randomly from JAFFE database for training and the rest images are used for testing respectively. In the same way, for the Cohn-Kanada databases we selected 10, 15 and 20 images per expression randomly for training and the rest images are used for testing respectively.

3.2 Experiments

1) Experiment 1

Normally the number of training sample exerts significant influence on the performance of one method. We compared several common appearance-based approaches with different number of training sample as shown in Table 1. Here, for every method the designed experiment is repeated 10 times, and the final recognition rate is the average accuracy with possible best dimension of PCA. For space limitations, we don't list all the experimental results with different parameters such as the value of k and the number of block partition. But, Just to be clear: for LDP and LBP the same parameters were assigned as [18].That's k=3 and blocks= 7×6 with which the performance is best. In addition, for the method based on curvelet features listed in Table 1 we only select the coefficients of low frequency after decomposition in four scales as facial features with different parameters such as the value of k and the number of block partition. But, Just to be clear: for LDP and LBP the same parameters were assigned as [18].That's k=3 and blocks= 7×6 with which the performance is best.

Table 1 The performance of various methods with different numbers of training samples

No. of training sample per expression	JAFFE			Cohn-Kanade		
	15	20	25	10	15	20
PCA	69.23%	78.43%	87.57%	60.57%	79.89%	88.5%
Curvelet	80.84%	86.43%	92.62%	86.23%	93.01%	96.79%
Standard LBP	75.05%	82.29%	89.14%	83.38%	90.89%	95.21%
Standard LDP	77.05%	84.43%	91.43%	84.92%	91.71%	95.79%
Proposed method	82.95%	88.71%	94.57%	87.57%	94.46%	98.03%

From the results in Table 1, one obvious observation is that the recognition rates of every method had improved along with the increasing of the training samples. More important, our proposed method has best performance in all cases.

Table 2 The performance of proposed method with different resolutions

Resolution	JAFFE			Cohn-Kanade		
	15	20	25	10	15	20
150x110	82.95%	88.71%	94.57%	87.57%	94.46%	98.03%
75x55	81.71%	87.00%	94.02%	86.43%	93.48%	97.29%
37x27	80.95%	85.86%	92.28%	84.02%	91.14%	95.35%

2) Experiment 2

In real environments, a good facial expression analysis system must be able to recognize expressions in face images with relatively lower resolution [24], which has attracted little attention in the existing literature. In this paper we explore the proposed method with different image resolutions, and the result is shown in Table 2. From the result we can confirm that reducing image resolution has little impact on recognition rate for our methods and a good recognition rate can still be obtained under a low resolution of 37×27.

4 Conclusion

Human facial expressions are so complex and subtle that only one type of features often can not obtain better results. Therefore in practical applications, we should combine some different powerful features and make full use of their superiority to describe facial expression so that a good performance will be achieved. The proposed method of features extraction in this paper exactly combined global curvelet features with local LDP features to represent facial expression. Experimentations have indicated that our method improves the correct rate of recognition comparing with other simple appearance-based methods. Furthermore, it is less sensitive to illumination changes and better performance can be received in a low image resolution, which is useful in practical application. However, there are many works we should do in future. One is the regions of eyes and mouth should be located automatically, and another is making deep studies on the influence of different parameters such as the no. of block partition.

Acknowledgement. This paper is supported by a Chunhui Planning project from China Education Committee with grant number Z2009-1-65001 and a Science and Technology Planning project in Yunnan Province with grant number2007F202M.

References

1. Niu, Z., Qiu, X.: Facial expression recognition based on weighted principal component analysis and support vector machines. In: Int. Conf. on Advanced Computer Theory and Engineering, IEEE, Los Alamitos (2010)
2. Mandal, T., Majumdar, A., Hu, J.: Face Recognition Via Curvelet Based Feature Extraction. In: International Conference on Image Analysis and Recognition, pp. 806–817 (2007)
3. Gritti, T., Shan, C., Jeanne, V., Braspenning, R.: Local Features based Facial Expression Recognition with Face Registration Errors. IEEE, Los Alamitos (2008), 978-1-4244-1/08/2008
4. Murthy, G.R.S., Jadon, R.S.: Effectiveness of Eigenspaces for Facial Expressions Recognition. International Journal of Computer Theory and Engineering 1(5) (December 2009)
5. Fasel, B., Luttin, J.: Automatic Facial Expression Analysis: a survey. Pattern Recognition 36(1), 259–275 (2003)
6. Zhang, Z., et al.: Comparition between Geometry-Based and Gobor-wavelet-based Facial Expression Recognition Using Multi-layer Perception. In: Proc. IEEE. Int. Conf. Auto. Face Gesture. Recog., pp. 454–459 (April 1998)
7. Guo, G.D., Dyer, C.R.: Learning from examoles in the small sample case:face expression recognition. IEEE Transaction on System,Man and Cybermetics-Part B, Special Issue on Learning in Computer Vision and Pattern Recognition 35(3), 477–488 (2005)
8. Weimin, X.: Facial Expression Recognition Based on GaborFilter and SVM. Chinese Journal of Electronics 15(4A) (2006)
9. Shan, C., Gong, S., McOwan, P.W., McOwan, P.W.: Facial expression recognition based on Local Binary Patterns:A comprehensive study. Image and Vision Computing 27, 803–816 (2009)
10. Valstar, M., Patic, I.: Facial action unit diction using probabilistic actively learned support vector machines on tracked facial point data. In: IEEE Conference on Computer Vision and Pattern Recognition Woprkshop, vol. 3, pp. 76–84 (2005)
11. Praseeda Lekshmi, V., Sasikumar, M.: Analysis of Facial Expression using Gabor and SVM. International Journal of Recent Trends in Engineering 1(2) (May 2009)
12. Dai, D.Q., Yan, H.: Wavelets and Face Recognition" in Face Recognition. In: Delac, K., Grgic, M. (eds.) I-TECH Education and Publishing, Vienna (2007)
13. Candes, E.J., Demanet, L., Donoho, D.L., et al.: Fast Discrete Curvelet Transforms. In: Applied and Computational Mathematics, pp. 1–43. California Institute of Technology, California (2005)
14. Ojala, T., Pietikainen, M.: Multiresolution Gray-Scale and Rotation with Local Binary Patterns and Linear Programming. IEEE Trans. Patter Anal. Mach. Intell. 29(6), 915–928 (2007)
15. Jabid, T., Kabir, M.H., Chae, O.S.: Local Directional Pattern (LDP) for Face Recognition. IEEE Int. Conf. Consum. Electron, 329–330 (2010)
16. Jabid, T., Kabir, M. H., Chae, O.S.: Local Directional Pattern(LDP)-A robust Descriptor for Object Recognition. In: IEEE Int. Conf. on AVSS 2010 (2010), 978-0-7695-4264-5/10
17. Jabid, T., Kabir, M. H., Chae, O.S.: Gender Classification using Directional Pattern (LDP). IEEE Pattern Recognition (2010)

18. Jabid, T., Kabir, M. H., Chae, O.S.: Robust Facial Expression Recognition Based on Local Directional Pattern. ETRI Jounal 32(5) (October 2010)
19. Candes, E.J., Donoho, D.L.: Curvelet surprisingly effective non-adaptive representation for objects with edges, Curve and Surface Fitting, pp. 105–120. Vanderbilt Vniversity Press, Saint-Malo (1999)
20. jingwen, Y., xiaobo, Q.: Analysis and Application of Super-Wavelet. National Defence Industry Press (2008)
21. Cand'es, E.J., Guo, F.: New multiscale transforms, minimum total variation synthesis: application to edge-preserving image reconstruction. Sig. Process., special issue on Image and Video Coding Beyond Standards 82, 1519–1543 (2002)
22. Jaffe dataset, http://www.kasrl.org/jaffe.html
23. Kanade, T., Cohn, J., Tian, Y.: Comprehensive Database for Facial Expression Analysis. In: Proc. IEEE Int'l Conf. Face and Gesture Recognition (AFGR 2000), pp. 46–53 (2000)
24. Tian, Y.-L., Brown, L., Hampapur, A., Pankanti, S., Senior, A., Bolle, R.: Real world real-time automatic recognition of facial expressions. In: Proceedings of IEEE Workshop on Performance Evaluation of Tracking and Surveillance, Graz, Austria (2003)
25. Jiulong, Z., Zhiyu, Z., Wei, H., Yanjun, L., Yinghui, W.: Face Recognition Based on Curvefaces. In: Third International Conference on Natural Computation, vol. 2, pp. 627–631 (2007)

Constructing Maximum-Lifetime Data Gathering Tree without Data Aggregation for Sensor Networks

Jinhui Yuan, Hongwei Zhou, and Hong Chen

Abstract. In this paper,we propose the approach constructing maximum lifetime data gathering tree without aggregation for sensor networks. In the scenario sensor nodes have the capability to adjust their transmission power with the transmission range, we approximately construct the maximum lifetime data gathering tree with the goal to balance the energy consumption among the sensor nodes to prolong the lifetime of the network. Our simulation shows that our approach is effective.

1 Introduction

Recently Wireless sensor networks(WSNs) have been widely used in military surveillance, traffic monitoring, habitat monitoring and object tracking etc. [1][2]. Such networks deploy lots of sensor nodes which have the capabilities of sensing, data processing and wireless communicating in the monitoring area. Generally, sensor nodes collect the sensing data of the monitoring area to sink, and help users to make decisions. However, sensor nodes are resource-constraint, and energy-efficient is an important goal in wireless sensor networks.Thus, energy-efficient data gathering techniques have become the hot researches in recent years.

We briefly lay out the design space of data gathering in the network explored by existing work. We have identified two categories of them by different user requirements. One is to make full use of data aggregation function, for example SUM, AVG, MAX etc., to meet the special query of the user. With data aggregation, each sensor node fuses the receiving data and its own data, and transmits the same size of

Jinhui Yuan · Hongwei Zhou
School of Information, Renmin University of China, Beijing, China and
Institute of Electronic Technology, Information Engineering University, Zhengzhou, China
e-mail: jcyjh@126.com, hong_wei_zhou@hotmail.com

Hong Chen
School of Information, Renmin University of China, Beijing, China
e-mail: chong@ruc.edu.cn

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 47–57.
springerlink.com

data to its parent. However, data aggregation is not always suitable in any situation. The other is to gather meaningful sensing data without aggregation to sink. It means that the size of the transmission data increases, which leads to more energy of the sensor node to be consumed.

Improving of routing technique is an promising approach to save the energy consumption on data transmission [3][4][5]. Usually, they select the minimum energy paths to transmit the data. However, the disadvantage of these approaches is that the energy of the sensor nodes in these paths will use up quickly, which maybe result in the disconnection of the sensor networks. Thus, lots of efforts have been made in recent years to construct maximum lifetime tree for data gathering. They propose some algorithms to balance the energy consumption among the sensor nodes in the monitoring area to prolong the lifetime of the network [6][7][8][9][10].

Wu et al. [6] prove that constructing the maximum lifetime data gathering tree with aggregation in sensor networks is an NP-complete problem and propose a near optimal algorithm as the solution. Moreover, they make further efforts to extend their solution to the case which there were multiple sinks in the networks, and create a maximum lifetime data gathering forest with data aggregation in polynomial time [10]. Lin et al. [7] believe the model used in [6] is not general, in which all sensor nodes have fixed and same transmission power. They construct the new model for the scenario which the transmission power levels of sensor nodes are heterogeneous and adjustable. However, they are only interested in constructing maximum lifetime data gathering tree with data aggregation, and data gathering tree without data aggregation is beyond the scope of their paper. Liang et al.[8] presents an approximation algorithm MITT to construct a min-max-weight spanning tree for data gathering without aggregation. However, they do not concern about the situation which the sensor nodes have adjustable power levels.

In this paper, we focus on constructing maximum lifetime data gathering tree without aggregation for sensor networks, in which sensor nodes can adjust their transmission power levels according to the transmission range. With adjustable energy levels, sensor nodes can send data consuming the least energy within its transmission range. Thus, it reduces the energy consumptions as much as possible. With the help of the tree built by our algorithm, the energy consumption among the sensor nodes is balanced. Thus, the lifetime of the networks are prolonged. Experiment results show that the algorithm MLTTA(Maximum Lifetime data gathering Tree wiThout aggregation for Adjustable-power algorithm) we propose is effective and efficient. To the best of our knowledge, no previous work aims to solve the problem as this paper.

The rest of this paper is organized as follows. Section 2 describes the system model of the network and the problem definition of constructing maximum lifetime data gathering tree. Our proposed algorithm is introduced in section 3. Section 4 describes the simulation results. Finally, section 5 concludes the paper.

2 System Model and Problem Definition

2.1 Sensor Network Model

Suppose that a sensor network is modeled as an undirected connected graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V}$ is the set of all sensor nodes (including n sensor nodes $v_1, ..., v_n$ and one sink v_0, denoted as $\mathbb{V} = V_n \cup v_0$), $\mathbb{E}$ is the set of the edges. Similar to [7], we assume that the energy of node v_i is $E(v_i)$ $(i = 1, ..., n)$, and all senor nodes can adjust their transmission energy levels according to their maximum transmission ranges. Sink is powered with infinite supply, the energy of which is set to ∞. An edge (v_i, v_j) is in $\mathbb{E}$, if and only if v_i and v_j are within each other's maximum transmission range. A data gathering tree T is a tree rooted at the sink v_0 spanning all of the sensor nodes V_n. Each sensor node adjusts the energy level of its transmitter just high enough to maintain the connectivity of the data gathering tree T.

Table 1 List of Symbols

Symbol	Meaning
N	number of nodes
T	data gathering tree without aggregation
$E(v_i)$	initial energy of node v_i
$E_T^t(v_i)$	current transmission energy for node v_i to transmit one bit message in T, which is just enough energy to reach all neighbors in T
$E_T^t(v_i)_{max}$	maximum transmission energy for node v_i to transmit one bit message in T
$E_T^t(v_i)_{min}$	minimum transmission energy for node v_i to transmit one bit message in T
E^r	energy consumption of receiving one bit message, it is fixed in the network
$L(T)$	lifetime of data gathering tree T
$L(v_i)$	lifetime of node v_i in T
$D_T(v_i)$	the number of descendants for node v_i in T
$W(T)$	inverse lifetime of data gathering tree T
$W(v_i)$	inverse lifetime of node v_i in T

This paper concerns about the sensor nodes deployed in the network with adjustable and heterogeneous transmission energy levels. Different sensor nodes maybe have different energy levels, with respect to different transmission energy consumptions. The symbols and their meanings used following are summarized in Table 1. The b-bit message can be delivered to its parent and sent to its children nodes as the acknowledgement messages. It is known that we focus on transmitting data without aggregation. Thus for node v_i, the size of the data sent from its children is $b * D_T(v_i)$, and the size of the data transmitting to its parent is $b * (D_T(v_i) + 1)$ after fusing its data and the receiving data.

The characteristics of the sensor networks discussed in this paper are as follows.

(1) The wireless sensor networks are static, and the topology does not change in the networks lifetime.

(2) There are many data gathering trees without aggregation in G, and we will find the maximum lifetime tree among them.

(3) The most energy consumptions of sensor nodes are transmitting and receiving data, and the consumption of processing data is ignored.

(4) For each sensor node in the networks, it has the capability of adjusting energy consumption while transmitting the data to its parent.

While sending the fused data to the parent, the sensor node sends the acknowledgement message to its children at the same time. Similar to [7], the transmission energy of sensor node is just sufficient enough to send data to its farthest neighbor node in the constructed data gathering tree. The transmission range of each node is similar to discussion above.

2.2 Problem Definition

In this paper, we define the lifetime of a sensor network is the time from the initial deployment of sensor nodes to the time the energy of any node is completely depleted. The energy consumption of node v_i in each round of given data gathering tree T is $C_T(v_i)$, $C_T(v_i) = D_T(v_i)bE^r + (D_T(v_i)+1)bE_T^t(v_i)$.

The lifetime $L(v_i)$ of node v_i is defined as the number of running rounds from initial deployment to its death, i.e. the result of dividing its initial energy by the energy consumption in each round.

$$L(v_i) = \frac{E(v_i)}{D_T(v_i)bE^r + (D_T(v_i)+1)bE_T^t(v_i)} \tag{1}$$

The lifetime $L(T)$ of a tree T is the lifetime of first dead node in the data gathering tree T.

$$L(T) = \min_{v_i \in V_n} L(v_i) \tag{2}$$

As we all know, there are a number of possible data gathering trees in G. Assume that $PT(G)$ is the set of all possible data gathering trees without aggregation in G. Our goal is to find the data gathering tree T which holds the maximum lifetime without data aggregation, i.e.,

$$\max_{T \in PT} L(T) = \max_{T \in PT} \min_{v_i \in V_n} L(v_i) \tag{3}$$

Equation (3) is equivalent to:

$$\max_{T \in PT} \min_{v_i \in V_n} \frac{E(v_i)}{D_T(v_i)bE^r + (D_T(v_i)+1)bE_T^t(v_i)} \tag{4}$$

In Equation (4), b and E^r are constants, and $E_T^t(v_i)$ and $D_T(v_i)$ are two variables. Thus, simplifying slightly, we obtain:

$$\max_{T \in PT} \min_{v_i \in V_n} \frac{E(v_i)}{D_T(v_i)(E^r + E_T^t(v_i)) + E_T^t(v_i)} \tag{5}$$

We define the inverse lifetime of node v_i as following, which is similar to [6].

$$W(v_i) = \frac{D_T(v_i)(E^r + E_T^t(v_i)) + E_T^t(v_i)}{E(v_i)} \tag{6}$$

$W(v_i)$ is the load of node v_i in T, which denotes the ratio of the energy consumption in each round to the initial energy of node v. Then,

$$(3) \Longleftrightarrow \min_{T \in PT} W(T) = \min_{T \in PT} \max_{v_i \in V_n} W(v_i) \tag{7}$$

In other words, to find the maximum lifetime of data gathering tree without aggregation is transformed to find the minimum load data gathering tree without aggregation. i.e.,

$$\min_{T \in PT} \max_{v_i \in V_n} \frac{D_T(v_i)(E^r + E_T^t(v_i)) + E_T^t(v_i)}{E(v_i)} \tag{8}$$

Equation (8) reveals that those nodes which hold more descendants and much farther communication range consume more energy in each round. As the result, those nodes with more initial energy should be the nodes holding more loads. By doing this, the network can achieve load balancing as much as possible. From previous works [7][6], we infer that this is an NP-complete problem and further propose an approximation algorithm.

3 The Proposed Algorithm

3.1 Basic Idea

In order to achieve the load balancing in the network, we divide the sensor nodes into three sets according to their loads. The three sets are high-load set V_h, medium-load set V_m and low-load set V_l.

For $W(T) = \max_{v_i \in V_n} W(v_i)$, $\varepsilon > 0$, we determine the set V_h, V_m and V_l as follows.

$$V_h = \{v_i | W(T) - \varepsilon < W(v_i) \leqslant W(T), v_i \in V_n\} \tag{9}$$

$$V_m = \{v_i | W(T) - \frac{E^r + 2\Delta^t(v_i)}{E(v_i)} < W(v_i) + \varepsilon \leqslant W(T), v_i \in V_n\} \tag{10}$$

where, $\Delta^t(v_i) = E_T^t(v_i)_{max} - E_T^t(v_i)_{min}$

$$V_l = V_n - V_h - V_m \tag{11}$$

Among them, the node in V_m becomes the high-load node if the number of its descendants increases one, i.e. Equation (12).

$$\frac{(D_T(v_i)+1)(E^r+E_T^{t'}(v_i))+E_T^{t'}(v_i)}{E(v_i)} > W(T)-\varepsilon \tag{12}$$

where, $E_T^{t'}(v_i)$ is the current transmission energy of node v_i after adding one descendent. It is obvious that $E_T^{t'}(v_i)-E_T^t(v_i) \leqslant \Delta^t(v_i)$, then we have:

$$W(v_i)+\frac{E^r+2\Delta^t(v_i)}{E(v_i)} > W(T)-\varepsilon \tag{13}$$

Thus, Equation (10) is get.

3.2 Detailed Algorithm

The sensor nodes in different load sets have different capability of holding other nodes. We define additional attribute *volume* to describe this character. The nodes in V_h have high loads and should transfer some of their descendants to other nodes, whose *volume* is assigned to be -1. The nodes in V_m have medium loads and can not add any descendant, whose *volume* is assigned to be 0. And the nodes in V_l, whose *volume* is determined by the amount of the nodes they can accept, have low loads and can accept other nodes as their descendants in the condition of not more than their receiving capability.

Formal representation is as follows.

$v \in V_h$: $v.volume = -1$,

$v \in V_m$; $v.volume = 0$,

$v \in V_l$:

$$v.volume = \frac{(W(T)-\varepsilon-W(v_i))*E(v_i)}{E^r+E_T^t(v_i)}. \tag{14}$$

Now, we prove Equation (14) below.

Assume sensor node v_i can additionally hold on *nodeno* descendants before it becomes the member of V_h, then we have:

$$\frac{(D_T(v_i)+nodeno)*(E^r+E_T^t(v_i))+E_T^t(v_i)}{E(v_i)} \leqslant W(T)-\varepsilon \tag{15}$$

$$(15) \iff W(v_i)+\frac{nodeno*(E^r+E_T^t(v_i))}{E(v_i)} \leqslant W(T)-\varepsilon \tag{16}$$

$$(16) \iff nodeno \leqslant \frac{(W(T)-\varepsilon-W(v_i))*E(v_i)}{E^r+E_T^t(v_i)} \tag{17}$$

Thus the Equation (14) is proved to be true.

The above is the initial assignment of the nodes for attribute *volume*. As we all know, if the ancestor of the nodes belongs to set V_h, its descendants can not add the node to its subtree any more. Consequently, the *volume* of the descendants is the minimum of its initial volume and the volumes of its ancestors.

The goal of our algorithm MLTTA is to reduce the loads of the nodes in set V_h as much as possible. Our approach is to transfer the nodes or their descendants to be the children of other neighbor nodes who belong to set V_l. The procedure of MLTTA is as follows.

(1) Build an initial data gathering tree H without aggregation.

   a. Make all of the neighbor nodes of sink as its children, and their levels are set to be 1.
   b. For each node v_i, if it is the neighbor of those nodes whose levels are 1, it selects those nodes as its candidate parents.
   c. Compute the load of each candidate parent according to the current number of the descendants and the initial energy of the node, v_i selects the node who holds the minimum load as its parent and its level is 2.
   d. If v_i is the neighbor of the nodes whose levels are 2, it selects those nodes as its candidate parents. Repeat (c) and (d) until all the nodes are in the data gathering tree, then the initial tree is constructed successfully.

(2) Determine which set every node belongs to.

   a. Using Equation (6), the load $W(v)$ of node v can be get, and further get the maximum load $W(T)$ of the tree.
   b. Which set does the nodes belong to? It is determined by our definitions. If the load of the node satisfies Equation (9), the node is in set V_h. If the load of the node satisfies Equation (10), the node is in set V_m. Otherwise, the node belongs to set V_l.

(3) Transfer the initial data gathering tree to reduce the load of V_h.

   a. Find the node whose load is the max in set V_h. Suppose the node is v_j.
   b. Sort the descendants of v_j according to the number of the descendants in ascending order. Check each descendant v_k of v_j according to the order, if v_k has the neighbor node who is in set V_l and its *volume* is more than the number of the descendants of v_k, v_k and its subtree are transferred to be the child of this neighbor. If none of such node, it is not allowed to be transferred.
   c. If the *volume* of all neighbor nodes are 0, it is not allowed to be transferred directly. Suppose the neighbor node of v_k is v_l. If v_l holds the neighbor node whose *volume* is more than the number of the descendants of v_l, it can transfer v_l and its subtree to be the child of the neighbor, and then transfer v_k and its subtree to be the child of v_l. This procedure maybe recursively perform.

(4) After transferring, update the load and *volume* of each node, and further determine the set it belongs to.
(5) Repeating step (3) and (4) until each of the high-load node which is possible to be transferred has been transferred successfully. Then, we obtain the maximum lifetime data gathering tree without aggregation.

4 Simulation Results

In our simulation, the parameters used in the experiments are listed in table 2. Among them, the initial energy of the sensor node is randomly range from 1 to 10J. Each sensor node can adjust their transmitting energy according to the farther transmission range of its neighbor nodes in the data gathering tree. Similar to [7], we use the data sheet in [12] to determine the different energy levels and transmit energy consumption of the nodes according to the transmission range. The detailed parameters are listed in table 3.

Table 2 Simulation Parameters values

Parameter	Value
Number of nodes	(10 100)
Field(m)	100×100
Position of sink(m)	(50,50),(0,50)
Initial energy(J)	U(1,10)
Data rate(kbps)	250
Receiving energy(mJ)	0.116
Transmitting energy(mJ)	Table 3
the length of sensing data(bit)	560
Algorithm parameter ε	0.000005

Table 3 Transmission Range and Energy Consumption

Energy level	Transmission range(m)	Energy consumption(mJ)
1	9.97	0.1016
2	18.78	0.1248
3	22.32	0.1449
4	25.04	0.1562
5	28.10	0.1618
6	29.76	0.1750
7	31.53	0.1800
8	33.40	0.1963
9	37.47	0.2107

The performance of the experiments is the lifetime of the network, i.e., the running rounds from the initial deployment to the first node depletes its energy. We compare MLTTA with the other two algorithms. These algorithms are listed as follows. (1) INITA: it constructs the tree according to the breadth first traversal and randomly selects the parent from the candidate parents. (2) NAIVE: it constructs the randomly generated tree in the beginning and does not make any improvement. The results of all the experiments are mean of 20 times of executions.

Figure 1 shows the comparison of the three algorithms while the sink locates at (50,50). It reveals that the lifetime of the data gathering tree constructed by MLTTA

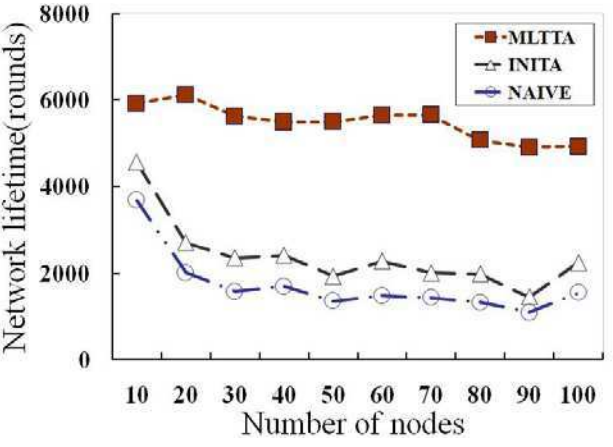


Fig. 1 Sensor network lifetime of different algorithms. Sink locates at (50,50).

is longer than other algorithms whatever the number of the nodes is. The reason is that MLTTA effectively reduces the loads of the nodes holding high loads and achieve the load balancing as possible as it can. Figure 1 also shows that INITA performs better than NAIVE. The reason is INITA randomly select parent from candidate parents, which balance the loads of the nodes to some extent. MLTTA useS the same method to select the parent, which also partly improve the performance of the algorithm.

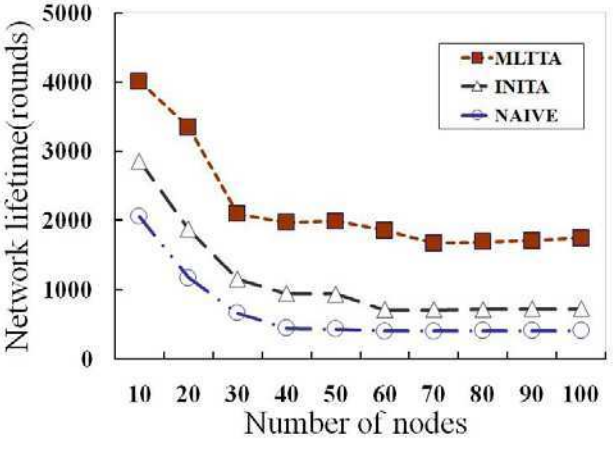


Fig. 2 Sensor network lifetime of different algorithms. Sink locates at (0,50).

Then we conduct the experiment to compare the lifetime of the three algorithms when the location of the sink is (0,50). From figure 2, we note that the lifetime of the

data gathering tree constructed by all of the three algorithms is much shorter than the scenario the sink locates at (50,50). And, it also shows that MLTTA achieves longer lifetime of the data gathering tree than the other algorithms.

5 Conclusion

We consider that the lifetime of the networks is the time from initial deployment to the first node depletes its energy. Ideally, we would like to have a balance that every node run out at almost the same time. However, this may not be possible in practice. Usually, some nodes consume their energy heavily, and it leads to the death of sensor network in short time. To prolong the lifetime of the sensor networks, the energy consumption among the nodes must be balanced as possible as it can.

In this paper, we propose the algorithm MLTTA to construct maximum lifetime data gathering tree without data aggregation to prolong the lifetime of the sensor network. In our supposing scenario, every node can adjust their transmission power to reduce the energy consumption. To balance the energy consumption in the networks, the maximum lifetime data gathering tree is created with our proposed algorithm, and our simulation shows that our approach is effective.

Acknowledgements. This research is partly supported by National High Technology Research and Development 863 Program of China Under Grant No.2008AA01Z133, the National Science Foundation of China(61070056), Key Program of Science Technology Research of MOE(106006), Program for New Century Excellent Talents in University, and Renmin University of China under grant no. 11XNH119.

References

1. Cruller, D., Estrin, D., Sivastava, M.: Overview of sensor networks. Computer 37(5), 41–49 (2004)
2. Kahn, J.M., Katz, H., Pister, K.S.J.: Next Century Challenges: Mobile Networking for 'Smart Dust'. In: Proceedings of the MOBICOM, pp. 271–278 (1999)
3. Heinzelman, W., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of HICSS (2000)
4. Singh, S., Woo, M., Raghavendra, C.S.: Power-aware routing in mobile ad hoc networks. In: Proceedings of the MOBICOM, pp. 181–190 (1998)
5. Rodoplu, V., Meng, T.H.: Minimum energy mobile wireless networks. IEEE Journal on Selected Areas in Communications 17(2), 1333–1344 (1999)
6. Wu, Y., Fahmy, S., Shroff, N.B.: On the Construction of a Maximum-lifetime Data Gathering Tree in Sensor Networks: NP-Completeness and Approximation Algorithm. In: Proceedings of the IEEE INFOCOM, pp. 356–360 (2008)
7. Lin, H.C., Li, F.J., Wang, K.Y.: Constructing Maximum-lifetime Data Gathering Trees in Sensor Networks with Data Aggregation. In: Proceedings of the IEEE ICC, pp. 1–6 (2010)
8. Liang, J.B., Wang, J.X., Cao, J.N., et al.: An Efficient Algorithm for Constructing Maximum lifetime Tree for Data Gathering Without Aggregation in Wireless Sensor Networks. In: Proceedings of the IEEE INFOCOM, pp. 1–5 (2010)

9. Kalpakis, K., Dasgupta, K., Namjoshi, P.: Maximum lifetime data gathering and aggregation in wireless sensor networks. In: Proceedings of IEEE NETWORKS, pp. 1–13 (2002)
10. Wu, Y., Mao, Z.J., Fahmy, S.: Constructing Maximum-Lifetime Data-Gathering Forests in Sensor Networks. IEEE/ACM Transactions on Networking 18(2), 1571–1584 (2010)
11. Tan, H.O., Korpeoglu, I.: Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks. SIGMOD Record 32(3), 66–71 (2003)
12. Texas Instruments. In: CC2520 Datasheet: 2.4 GHz IEEE 802.15.4/Zigbee RF Transceiver (2007), http://focus.ti.com/docs/prod/folders/print/cc2520.html

An Empirical Study of Programming Performance Based on Keystroke Characteristics

Dapeng Liu and Shaochun Xu

Abstract. This study investigates programming habits based on keystrokes, software quality, code format, and their correlation. We conducted an experiment by asked ten undergraduate students and one graduate student to complete a programming request in a controlled environment. We used a software tool to record the keystroke frequency, designed criteria to evaluation program quality, and conducted a survey after the experiment. The experiment results demonstrate that while novice programmers are diverse in terms of programming styles, good ones tend to control execution in finer granularity. Source code format can be a flag of programming performance. It seems that there is no direct correlation between the frequency of keystrokes and the quality of programs. We think monitoring low-level keystrokes could provide a way to study cognitive activity of programmers.

Keywords: Software Development, Programming Performance, Cognitive Activity, Keystroke Logging.

1 Introduction

Software is a human-intensive technology and the studies of cognitive processes in software engineering can shed light on many software engineering problems [5, 13]. Cognitive research on programming activities focuses on what programmers

Dapeng Liu
The Brain Technologies
Marine Del Rey
CA, USA
e-mail: dliu@thebrain.com

Shaochun Xu
Department of Computer Science
Algoma University
Sault Ste Marie, Canada
e-mail: simon.xu@algomau.ca

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 59–72.
springerlink.com

are doing and how they are doing. Any findings will benefit programmers themselves and might have implication for knowledge engineering.

As we know, programming is a challenging job and related activities are complex. Different studies have concentrated on different aspects of effects on high level and low level cognitions. There are a lot of researches with regards to the cognitive activities during software engineering process. However, most of them focus on high level activities such as programmer's behavior and mental activities [2][14][15]. There is little research on the keystrokes, one kind of low level activities. Thomas et al. (2005) was probably the only one who studied the correlation between keystroke speed and programming performance by conducting a case study [11]. However, they only measured quality of the subjects' programs in terms of completeness. A further study on this issue seems necessary.

Since we deem key strokes as an elementary indicator of the programmer's brain activity, in this study, we also adopted a low-level approach that monitors programmers' key strokes. We conducted an experiment with eleven students and observed the distribution of the key stroke frequency, the productivity of programmers, the code format, and their correlation to the quality of the final programs. Our experiment might help us to understand if the low level activity of programmers could contribute to the understanding programmer cognitive activity during programming. Therefore, in this study, we try to answer the research questions listed as below:

1. Is there any correlation between keystroke frequency and the quality of programs/expertise level of programmers?
2. Is the low level activity useful to study the cognitive process of programmers?

The rest of the paper is as follows: Section 2 describes related work. The case study setting is described in Section 3. Section 4 discusses the experiment results. The conclusions and the future work are presented in Section 5.

2 Related Work

There are a lot of researches with regards to the cognitive activities during software engineering process. For example, Davies did a systematical analysis on the programming strategy [2], and suggested that what is needed is an explanation of programming skill that integrates ideas about knowledge representation with a strategic model, enabling one to make predictions about how changes in knowledge representation might give rise to particular strategies and to the strategy changes associated with developing expertise. Most recently, the analogy between constructivist learning and incremental software development process has been recognized by Rajlich and Xu [15]. They identified four cognitive activities (absorption, denial, reorganization and expulsion) which correspond to incremental change, rejection of change request, refactoring and retraction, four programming activities.

Visser [12] conducted experiments on professional programmers and studied the strategies used during programming. He found that programmers used a

number of data sources and include sample program listings into them, so programmers may recall that a solution exists in a listing, find the listing, and then use the coded solution as an approach for the current problem. The programming knowledge was classified by Ye and Salvendy (1997) into a five level abstractions [16]. They also found that experts have better knowledge at an abstract level, and the novices tend to have concrete knowledge.

The coding activities by experts and novices was studied by Davies [2], in term of information externalization strategies. Davies found that experts tend to rely much more upon the use of external memory sources. He stated that the novices tend to focus on the key words in the problem statement rather the deep structure of the problem [3]. Petre and Blackwell studied the mental imagery of experts during software design [9]. They discovered that there are some common elements or principles all the experts applied.

Keystrokes are easily understood. However, there are only a few studies about the correlation between keystrokes and productivities in general. Brown and Gould (1987) studied experience in creating spreadsheets and found that experienced users spent a large percentage of time using cursor keys for moving the cursor around the spreadsheet and users did not spend a lot of time planning [1]. Sauto (2009) proposed to use composite operators for keystroke level modeling to measure the productivity [10] and a few researchers proposed to use keystroke characteristics as a way for identity verification ([4] [6] [8]).

Thomas et al. (2005) might be the only one who conducted an unsupervised experiment to study the relation between keystroke latency and quality of programs [11]. They found moderately strong negative correlations between speed and coding performance. However, their study is preliminary and they evaluated the quality of the programs based only on the completeness of the programs.

3 Experiment Setting

The purpose of the experiment is to test whether there is any correlation between keystroke frequency and the quality of programs/expertise level of programmers, and how the correlation is weighted against other factors, such as code format. Whether the low level activities of programmers are useful for cognitive process research is another purpose of this work.

We designed a Java program and asked subjects—a group of ten undergraduate students and one graduate student—to implement a sorting algorithm within a given program frame and to enhance its robustness. The assignment was not hard so that every student was expected to finish it within one hour. To stimulate undergraduate subjects to take the experiment seriously, two extra scores were allocated to them; on the other side, we neutralized the stimulus by telling them code quality was not accounted with the intention of observing their daily programming habits.

Subjects were allowed to bring their laptops to guarantee familiar programming environments and actually they all did it. During the one-hour test, we collected the key stroke counts in each minute, which sketchily depicted the subjects' behaviors.

3.1 Key Stroke Surveillance

We created a key stroke surveillance tool which is a simple windows keyboard hook. When the tool begins to run, key strokes of each minute in one hour will be counted and saved into a log file. At the end, the log file contains 60 lines of key stroke counts. Each line documents the number of keystrokes for that minute. As this is the first trial of such research, we did not keep track of other activities such as mouse clicks, although they might be important.

```
String inputLine = new BufferedReader(
  new InputStreamReader(System.in )).readLine();

String[] tokens = inputLine.split(" ");
int arrayLen = tokens.length;
int numbers[] = new int[arrayLen];

for( int i=0; i < arrayLen; i ++ ) {
  numbers[i] = Integer.parseInt(tokens[i]);
}
// replace the next statement with your own code
Arrays.sort(numbers);
```

Fig. 1 Code frame given to subjects for completion

3.2 Participants

The ten undergraduate participants are from the Department of Computer Science, Algoma University, which include nine sophomores who are in their second year of undergraduate study, one in upper year. Everyone had written 3,000—10,000 lines of code. All of them are male and taking the first "Data Structures" course. One graduate student has also joined the experiment and he has five years of Java programming experience.

3.3 Programming Assignment

The code being handled to the subjects consists of less than 50 lines, in which comments and blank lines are also included. The essence of the code given to the subjects is shown in Fig. 1.

The subjects were explicitly told that the expected input was a line of integers that were separated by spaces and then the program extracted the input and sorted the numbers.

There are two programming requirements for all the subjects about this piece of code:

- To replace the line *Arrays.sort(numbers)* with their own code. While students are allowed to freely choose any sorting algorithm and to consult any pseudo code, however, verbatim coping source code is prohibited.
- To make the program being robust for any input since in some situations the code might crash (the program throws an exception and quits abnormally).

The first requirement was intended to test whether subjects have fundamental programming skills. The second one tried to test whether they are better than entry-level. The statement *inputLine.split(" ")* would produce empty strings if the input string cannot be converted to integers since there should be there consecutive spaces between two numbers in the input line. An experienced programmer is expected to prevent such exceptions from disrupting the whole program.

3.4 Evaluation Criteria

We try to assign the scores to their individual programs. Since the main body of the subjects is a group of sophomores, correctness of code is their first priority. If their code cannot be compiled and executed on a list of correct inputs, we generously give them 50%. Additionally, while code style is considered as an important habit, we allocate 20% to it, including the line alignment, indentation, bracket positions, and consistency in the whole code. Each detailed criterion is worth 5%.

The code logic will be also investigated with the focus on the implementation details, how exceptions is handled, how prompt messages are fed back to users, etc. A set of test cases have been designed and run on the turned-in cod; which were literally [], [1], [4 3 1 2], [4 a 1 2], [1_ _ _ _], [_ _ _ _ _ 2], and [1_ _ _ _ _ _ _ _ _ _2], in which a pair of rectangular brackets ([]) delimit a line of input and '_' indicates a space.

3.5 Procedure

The participants were informed two weeks ahead that a simple experiment would be given during the class time. They were emailed the tool and have been asked to install it in their own laptops prior to coming to the classroom to conduct the experiment.

Right before the experiment was taken; all participants were emailed with the source code (code frame) and handled the requirements for the experiment. They were reminded by the mentor to run the tool before they started to work on the task. The graduate student who is the mentor has completed the experiment one week before.

After subjects completed the task, the mentor collected the modified code and the log file and put them into individual folders. Thereafter the subjects were handled with a questionnaire which was used to get their subjective opinions about programming and the experiment. The questions included how many years of programming experience, how hard the experiment was, what kind of programming environments they were using, etc. The information collected through the survey was compared with their performance.

One of the authors did code analysis and tested and ran the turned-in programs, evaluated their code style, summarized their characteristics.

4 Experimental Result and Analysis

Undergraduate subjects are labeled as s1~s10 and the graduate student is labeled as s11 for anonymity. Final scores are listed in Table 1 which is calculated based on the criteria give in Section 3.4.

Table 1 Final scores

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11
rank	11	5	5	4	5	3	10	2	5	9	1
score	0	80	80	85	80	90	70	95	80	75	100

Student s1 is an outlier since his code was far to being completed. Here we can see the graduate student completely finished the assignment and good undergraduate students were not far behind. In fact, s8, the best performing undergraduate student, shared similar key stroke-time distribution shape with s11. Dissection of the evaluation will be given later.

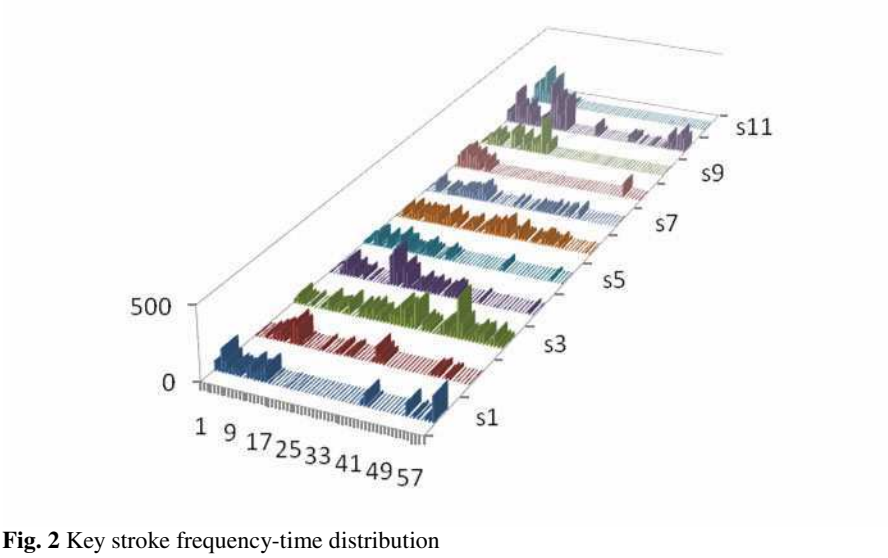


Fig. 2 Key stroke frequency-time distribution

4.1 Key Stroke Frequency-Time Distribution

While the experiment was expected to be completed in one hour, most of subjects did it, except s1. The key stroke data are shown in Fig 2. Although s1 did not

type for 23 minutes in the middle of the experiment, he tried again near the end. Most of Subjects who rested in the second 30 minutes—here we tolerated one segment of keyboard typing in this duration, s8, s9, and s11, scored 95, 80, and 100, respectively. While s8 and s11 were ranked as the top two, s9 had scrambled code format and imprudent exception handling. Based on this observation, we believe that good programmers can be identified in fast solvers with scrutiny. Nonetheless, fast completion should not be used as the only consideration because s3 and s6 spread their key strokes over long durations; especially s6 who is ranked at 3rd place.

Table 2 Maximum Key Stroke Frequency

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11
max ks	175	154	284	243	99	111	104	150	250	355	256
rank	6	7	2	5	11	9	10	8	4	1	3

We have identified the maximum key strokes counts for every subject. It seems that there is no direct correlation between the numbers of keystrokes and any factor of software quality. Maximum key stroke frequencies are listed in Table 2.

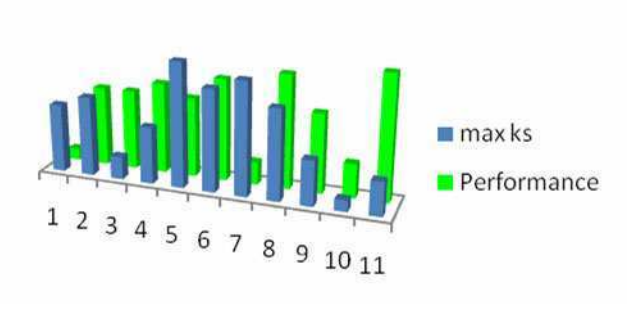


Fig. 3 Max key stroke frequency vs. performance

Fig 3 shows the comparison between the max keystroke speed and the performance rank. The performance is calculated by (*12 – rank position*), i.e., higher bars indicate better ranks. From figure 3, we can see the two ranks are totally orthogonal to each other.

4.2 The Productivity of Key Stroke/Character

Since there are many auxiliary key strokes, such as *alt* and *ctrl* and a programmer may also delete typed texts during programming, they might waste some key strokes.

The original code frame has 1427 bytes. The sizes of the turned-in code were measured and the added bytes were calculated. Fortunately, we didn't notice that any changes were made to the existing code besides those two programming requirements, so that the simple subtraction provides a precise measurement of volume of the added code. In addition, since the test code is pure ASCII, therefore, one byte is equal to one character.

Table 3 Key stroke, added bytes and productivity

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11
Ks	317	451	457	848	542	521	429	815	413	420	704
k/c	.21	.38	.15	.47	.55	.25	.36	.79	.28	.13	.33
c/k	4.6	2.6	6.7	2.1	1.8	3.9	2.8	1.3	3.5	7.9	3.0

In Table 3, row *ks* indicates the key stroke counts, row *k/c* lists the ratio of key strokes over characters, which is inversion of the next row, being labeled as *c/k*.

We can see s11 is right the median of the *c/k* values. The next best, s8, has the highest productivity. Subjects s5, s4, s2, and s7, who are in the upper section of the sorted *c/k* list, were ranked as 5, 4, 5, 10, respectively; s9, s6, s1, s3, and s10, who are less productive, were ranked as 5, 3, 11, 5, 9, respectively. It seems that different programmers have different programming styles and for such a basic test coding iteratively can still produce satisfying result.

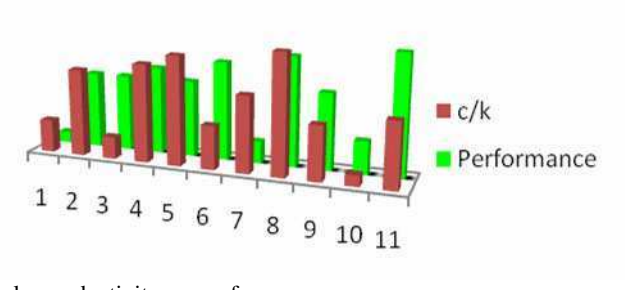


Fig. 4 Key stroke productivity vs. performance

Correlation between key stroke productivity and performance is shown in Fig 4. It seems that there is no obvious relationship between the two data series.

4.3 Dissection of Source Code

A single score cannot fully describe code quality and two equal scores do not imply that the two pieces of code are similar to each other. Code details have to be dissected in order to obtain interesting observations.

1) Code format. While it is widely accepted that code format is not dispensable in good programming habits, it is still a surprise to see the positive correlation between it and the final scores. Subjects, s6, s8, and s11 got full points in terms of code format; coincidently, they were ranked as top three. Subject s7 had too messy formatting problem to gain any point in this category, he was ranked as the second lowest. S2 and s3 got only 5 out of 20 points in terms of code format; though they performed okay well, their key strokes spreaded the duration, especially s3; therefore it was reasonable to doubt whether they could keep up when the assignments got harder.

   Evaluating the ability of a programmer solely based on his code format might be unreliable. However, this experiment demonstrates that code format is a reliable indicator of a programmer's performance .

2) Control precision. While some students, s5, s6, s8, s9, and s11, tried to precisely control the exceptions, most of them, except the graduate subject, permitted the subtle side effect of illegitimate input; i.e., they did not handle the situation but connive at the continuous execution. To be precise in this experiment, empty strings were converted to zero; as a result, ghost input was created. It seems that this is a dangerous programming habit and it is named as error permission to distinguish it from error tolerance which is considered as a merit.

   Ironically, the code which encapsulates the whole method body with *try…catch…* block has no chance to make a mistake, although it would certainly interrupt normal execution flow which may be unnecessary.

   In order to better control execution in a finer way, programmers might produce more errors along the learning process. This phenomenon is named here as *cost of growth*, which might discourage students from improving themselves.

   Comfortingly, no subjects who tried finer exception control got lower-than-average score.

In this experiment, extra spaces can be handled in an elegant way as shown in Figure 5:

```
String[] tokens = inputLine.split("( )*");
```

Fig. 5 An Elegant Way to Handle Consecutive Spaces

There was plenty of time for subjects to get themselves familiar with the APIs and to compose the elegant solution; especially for those who finished the assignment in less than half of the allocated time. Nonetheless, a robust implementation was missing in turned-in code.

4.4 Survey Result

The survey results are shown in Table 4 and Table 5. Table 4 includes the quantitative data and Table 5 includes the qualitative data.

1) Programmer confidence. Subjects were asked about whether they felt the assignment was hard. On a scale of 1 to 10 where 1 means very easy, the average answer and the median were 3, while the range was 1 to 6. We noticed that who thought the assignment was easy did not necessarily performed well while who performed well did think the assignment was easy. Who thought it was hard but scored okay had wide spread of key strokes. Those findings are not a surprise since this attribute can contribute much to the final effect.
2) Coding tools. It is a surprise to learn that 6 subjects used general text editors (textpad) in this experiment, especially with the consideration that they all worked on their own laptops. Three subjects used Eclipse, s4, s9, and s11; and two others used jgrasp, s3 and s10. It seems that using a convenient IDE does not necessarily indicate a high programming performance of the users. While it is merely a menu click for modern IDS, such as Eclipse, to format the code, such finding revealed that entry-level programmers have considerable potentials to improve their programming productivity.

Table 4 Quantitative data about the programmers' experience and experiment

	Questions	Answers
Personal Info and Java language	The number of Years in University	1.5
	How many years have you used?	1.5 (two with 5)
	Total lines of code you have written in Java (approximately)	1.5 years
Experiment	How hard do you think the problem you solved is (1-10) 1: very easy 10: very hard	3
	Are you aware of the meaning of the experiment result? If so, does such information match your expectation? 1: not at all 10: totally	5.5
	Do you often "jump" to other parts of the code to make some changes and then go back to continue your programming: 1: never 10: all the time, like type a line here, and type a line there…totally woven	4.1

Table 5 Qualitative data about personal programming habits statistics

	Questions	Answers
Experiment	Will you be interested in doing similar experiment in future?	All 11 students said yes
	What kind of improvement is there in your mind for this experiment?	• Yes, sorting method used (3) • Clearer about the question (1) • Separate room (3)
	What kind of compiler did you use?	• Jgrasp8 (2), • textpad (6) • Eclipse (3)
Opinions on yourself	Do you prefer to think over details before starting; or you always code-and-correct?	• Think-plan-code-fix (4) • Code-correct (3) • Sometime code-correct, sometime thinking in general (4)
	When you think, do you think in general (broad) or think one in deep?	• One-step in deep at a time (9) • Thinking in general (2)
In general (not limited to this experiment), about your own programming activities	Do you feel your programming activity is smooth, or often interrupted by other activities, such as referring to textbook, reading technical documents? If so, what kinds of activities were there? If so, how do you feel?	• Smooth sometimes (2) • Always not smooth, search google for methods, thinking the solution (1) • Often interrupted, referring textbook, internet (5), • Fairly smooth, some referring to textbook (2)
	Do you often "jump" to other parts of the code to make some changes and then go back to continue your programming: 1: never 10: all the time, like type a line here, and type a line there…totally woven	• Not often, just work on one part, (2) • When debugging, I hung around often to change variables. (1) • Yes(5) • Not often, just work on one block, but jump when finding an error (3)

3) Other programming habits. Based on the survey we noticed that most, 9 out of 11 subjects felt that programming was not a smooth flow-out of thinking. S4 simply claimed that programming was smooth while he thought this assignment was 6 out of 10 hard, his ks/c was 2.14, and he rarely jump among different locations of code when coding the program. So we conjecture that s4 had set-ready-go programming style. S9 thought programming was fairly smooth if no reference was needed. All other subjects claimed that programming was not a smooth production procedure.

This impression was validated by Fig 2 in which mostly key stroke frequency varies a great deal for continuous minutes. This pattern happens to various coding styles: 4 subjects claimed to plan before coding, 3 claimed coding first and then fixing errors, the rest 4 swung between the two styles. We think that the consistently varing key stroke was caused by the fact that 9 out of 11 subjects confessed that while programming they were often distracted to search through textbooks or online search engines.

4.5 Summary

First of all, our first hypothesis has not been supported by our observation and the correlation between key strokes and software quality may be unsymmetrical. While some of the subjects who wrote good code typed fast, some subjects who typed fast did not perform well. Please be advised that the program request is easy and key strokes were monitored in a controlled environment but not in daily programming. In realistic case the correlation may be harder to find. From the discussion, the low level activities, such as keystroke could provide us a way to study the cognitive activities of programmers.

The experiment shows that the fresh programmers present diverse key stroke distribution, which implies different recognition patterns and already bifurcates in terms of productivity and software quality. With the consideration that the most undergraduate participants have less than a couple years of similar education, the diversity of programmers is prominent, which in our experiment were represented by various key stroke distributions and key stroke productivity. Although good programmers may be fast in implementation, the converse is far from the fact.

In the survey, more than half participants reported that programming was not a smooth producing procedure. This may suggest that programming consists of iterative recognition processes instead of a single round of study-and-exercise. Such complex behaviors call for a suitable theoretic framework for recognition research and effective evaluation criteria.

Software code itself represents software quality in multiple ways, not limited to evaluation of how many tests the code can pass. One interesting observation is that comments are tightly correlated to test performance. Though this may be expected by many through intuition, the over-high positive correlation still needs to be highlighted.

It seems that better programmers may practically make more errors when trying to escort the program execution in finer ways which will make the code better handle input noises. While robustness is desirable for practical software, programmer should be encouraged to improve themselves, even through trials and errors.

5 Conclusion and Future Work

This paper presents a case study with eleven students by asking them to implement a simple sorting algorithm. We collected numbers of keystrokes per minute

and correlated them with the program quality and code format. The experiment result demonstrates that participants presented diverse programming habits and it seems that there is no direct correlation between the keystrokes characteristic and program quality/expertise of programmers. We think monitoring low-level keystrokes might help us studying the cognitive process of programmers.

Although key strokes are an undoubtedly important part of programming behavior, mouse operations which complement them are equally indispensable in such research, which, nonetheless, was missing at the first stage of research. This research will be extended by including more and diverse subjects, conceiving programming requests of various difficulty levels, unifying mouse operations with key strokes as a more complete indicator of programmer cognition.

Acknowledgements

The authors are grateful to the students at Algoma University who participated in this experiment. We also thank Rejosh Samuel, who was the mentor to assist this experiment. Shaochun Xu would like to acknowledge the support of Algoma University Travel and Research Fund, Canada and the National Science Foundation of China under Grant No. 60963007.

References

[1] Brown, P.S., Gould, J.D.: An experiment study of people creating spreadsheets. ACM Transactions on Information Systems 5(3), 258–272 (1987)
[2] Davies, S.P.: Models and theories of programming strategy. International Journal of Man-Machine Studies 39(2), 237–267 (1993)
[3] Davies, S.P.: Knowledge restructuring and the acquisition of programming expertise. International Journal of Human-Computer Studies 40, 703–725 (1994)
[4] Joyce, R., Gupta, P.: Identity authentication based on keystroke latencies. Communications of the ACM 33(1), 168–176 (1990)
[5] Kinsner, W., Zhang, D., Wang, Y., Tsai, J.: Proceedings of the 4th IEEE International Conference on Cognitive Informatics (ICCI 2005), UCI, California, USA. IEEE Computer Society Press, Los Alamitos (2005)
[6] Leggett, J., Williams, G., Usnick, M., Longnecker, M.: Dynamic identity verification via keystroke characteristics. International Journal of Man-Machine Studies 35(6), 859–870 (1991)
[7] Martin, R.C.: Agile Software Development, Principles, Patterns, and Practices. Addison Wesley, Massachusetts (2002)
[8] Monrose, F., Rubin, A.D.: Keystroke dynamics as a biometric authentication. Future Generation Computer Systems 16, 351–359 (2000)
[9] Petre, M., Blackwell, A.F.: A glimpse of expert programmers mental imagery. In: Proceedings of the 7th Workshop on Empirical Studies of Programmers, New York, pp. 109–123 (1997)
[10] Sauro, J.: Estimating Productivity: Composite operators for keystroke level modelling. HCI 1, 352–361 (2009)

[11] Thomas, R.C., Karahasanovic, A., Kennedy, G.E.: An investigation into keystroke latency metrics as an indicator of programming performance. In: Proceedings of the 7th Australasian Conference on Computing Education, vol. 42, pp. 127–134 (2005)
[12] Visser, W.: Strategies in programming programmable controllers: a field study on pro-fessional programmer. In: Olson, G.M., Sheppard, S., Soloway, E. (eds.) Empirical studies of programmers: second workshop, pp. 217–230. Ablex Publishing Corporation, Norwood (1987)
[13] Robillard, P.N., Kruchten, P., d'Astous, P.: Software Engineering Process with the UPEDU. Addison-Wesley, London (2002)
[14] Xu, S., Rajlich, V.: Dialog-based protocol: an empirical research method for cognitive activity in software engineering. In: Proceedings of the 4th ACM/IEEE International Symposium on Empirical Software Engineering, Noosa Heads, Queensland, November 17-18, pp. 397–406 (2005)
[15] Rajlich, V., Xu, S.: Analogy of Incremental Program Development and Constructivist Learning. In: Proceedings of the 2nd IEEE International Conference on Cognitive Informat-ics (ICCI 2003), pp. 142–150 (2003)
[16] Ye, N., Salvendy, G.: Expert-novice knowledge of computer programming at different levels of abstraction. Ergonomics 39(3), 461–481 (1996)

A Theory of Planned Behavior Perspective on Blog Service Switching

Kem Z.K. Zhang, Sesia J. Zhao, Matthew K.O. Lee, and Huaping Chen

Abstract. Blog has been one of the social technologies that greatly change many online users' daily lives. Bloggers not only can publish personal dairies on it, but also can develop relationships with other bloggers or blog visitors. Although many blog services are provided freely, its service switching behavior have attracted much attention from practitioners. Blog service switching is a specific instance of the broader social technology switching phenomenon. It differs from general usage behavior as it involves both membership attraction and retention. Thus, prior research on information technology usage may not fully account for the phenomenon. In this study, we adopt a theory of planned behavior perspective and build up a switching model to explain blog service switching behavior. We employ a survey to explain how two quality beliefs (service quality and quality of alternatives) and two types of costs (sunk costs and relationship costs) exert influence in determining bloggers' switching behavior. Discussions and implications are provided to better understand the switching behavior of blog and other social technologies.

Kem Z.K. Zhang
Suzhou Research Institute, City University of Hong Kong, 166 Ren'ai Road, Dushu Lake Higher Education Town, SIP, Suzhou, Jiangsu, China
e-mail: zikzhang@cityu.edu.hk

Sesia J. Zhao
USTC-CityU Joint Advanced Research Center, 166 Ren'ai Road, Dushu Lake Higher Education Town, SIP, Suzhou, Jiangsu, China
e-mail: sesiazj@mail.ustc.edu.cn

Matthew K.O. Lee
Department of Information Systems, City University of Hong Kong, Tai Chee Avenue, Kowloon, Hong Kong SAR, China
e-mail: ismatlee@cityu.edu.hk

Huaping Chen
School of Management, University of Science and Technology of China, Jinzhai Road, Hefei, Anhui, China
e-mail: hpchen@ustc.edu.cn

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 73–83.
springerlink.com

1 Introduction

Writing blogs, also known as blogging, has become a routine behavior for many bloggers around the world. As many other social technologies (e.g., virtual communities, microblogging, and social networking sites), blog provides a number of useful features for the purpose of personal and interpersonal usage. Bloggers can post personal dairies, share knowledge, express opinions in a reverse chronological order. They can also use it to interact with blog visitors and establish networks of online friends. The multi-functionality of blog technology quickly facilitates its diffusion on the Internet. Ever since the early 2000s, blog proliferation has been showing exponential growth. The amount of bloggers has reached into a huge number on the Internet. For instance, Technorati claims to index over 124 million blogs [1]. China Internet Network Information Center (CNNIC) indicates that there are over 300 million blogs in China [2].

For practitioners, blogs generally contain users' personal information, their interests or even product preferences, and their networks of online friends. Thus, it may be highly valuable to employ these resources for marketing purposes. Many online companies have spotted such opportunities. They provide easy-to-use and multi-personalized blogging features to users. Famous online companies include Google, Microsoft, Yahoo, and Xanga.

Although many blog services are provided freely, its service switching behavior have attracted much attention from practitioners. In 2007, a survey report from CNNIC indicated that 20% of bloggers had switched their blog services [3]. In 2009, Technorati further suggested that 59% of bloggers had more than two years of blogging experiences, and over 50% were using the 2nd or even the 8th blogs [4]. The switching behavior of blog services implies the massive migration of valuable user resources on the Internet.

Note that blog service switching is a specific instance of the broader social technology switching phenomenon. For academic researchers, it differs from general usage behavior as it involves both issues about membership attraction and retention. Thus, it may be insufficient to account for the phenomenon only deriving from previous studies on initial or continuous information technology adoption [e.g., 5, 6]. Therefore, this study attempts to raise attention of both researchers and practitioners and provide a solid perspective to understand the phenomenon. We focus on blog as typical of social technologies. To obtain a rich and practical understanding, we first discuss the data and relationship transfer of blog service switching.

1.1 Data Transfer

Bloggers mostly maintain their blogs through posting user-generated content. For those who have been blogging for a long time, there would be a large amount of

data created in blogs. The data may be documentation of daily lives, repository of professional knowledge, or useful information gathered from the Internet. It is highly important to realize the value of this type of user generated contention. For instance, marketers are likely to find helpful consumer information or product preferences on blogs. When considering switching decision, an important problem bloggers have to face is whether the data in old blog services can be fully transferred to new ones. Some perceptive online blog companies, such as Blogger.com and WordPress.com, have discerned bloggers' migration behavior. They provide "import tools" to facilitate the data transfer process and further enlarge their blogger membership.

1.2 Relationship Transfer

Another important blog feature is the capabilities of connecting with online friends. For instance, Livejournal and Xanga are two high community-oriented online blog companies. Their blog services promote bloggers to add others into some "friend lists". Bloggers can search for friends through their email contracts or meet new friends from interest groups within the communities. They can also reply to blog visitors' comments and foster ongoing relationships with them. Some survey reports have shown that the main reasons of blogging are sharing experiences or opinions with others, and connecting with like-mined people, friends and family members [7]. Thus, if a blogger wants to switch away from his/her current blog service, the established social relationships become a vital issue for the decision.

2 The Switching Model

To better understand blog service switching, a solid theoretical perspective is highly necessary. In this study, we develop a switching model based on the theory of planned behavior. We scrutinize bloggers' switching behavior from a rational angle. Given the concerns of both data and relationship transfer issues, bloggers are expected to make serious plans before making switching decision.

2.1 Theory of Planned Behavior as the Foundation

Theory of planned provides a framework to understand how people make "planned" decision. According to this theory [8], actual behavior is primarily predicted by behavioral intention, which is further determined by three factors: attitude, perceived behavioral control, and subjective norms. Attitude is again predicted by attitudinal beliefs. The followings are descriptions of five key constructs in the theory of planned behavior.

- Behavioral intention pertains to a user's likelihood and willingness of performing some behavior.

- Attitude refers to the overall evaluation of whether performing the behavior is favorable or not.
- Attitudinal beliefs are the related perceptions or assessments about the behavioral object or expected consequences of the behavior.
- Perceived behavior control describes the controllability, facilitation, or difficulty of performing the behavior.
- Subjective norms address the pressure from "important" others who believe the user should perform the behavior.

2.2 Hypotheses Development

In this study, behavioral intention is the dependent variable. Particularly, we focus on bloggers' switching intention, which is their subjective willingness of switching away from the current blog service to a new one. We further define attitude as the evaluation of whether performing the blog service switching behavior is favorable or not. Based on the link from attitude to intention in the theory of planned behavior, we first propose the following hypothesis:

H1: Attitude is positively associated with switching intention

We identify two attitudinal beliefs with respect to blog service switching behavior: current service quality and quality of alternative blog services. Service quality pertains to bloggers' overall judgment with respect to the excellence of the current blog service. Quality of alternatives refers to the positive perceptions of the expected or experienced quality of alternative blog services. The two quality-related beliefs have been proposed as salient perceptual factors in other switching contexts (e.g., in the offline context) [9]. The nature of switching decision builds on the assessments of both incumbent and substitute services. Bloggers are less likely to switch away their current blog service if they find the service is of high service quality. Similarly, if they observe there are other better blog services provided, the chance of switching becomes higher. According to the relationship between attitudinal beliefs to attitude in the theory of planned behavior, we propose that service quality will have a negative impact on attitude to switch, while quality of alternatives has a positive impact. The two following hypotheses are:

H2: Service quality is negatively associated with attitude
H3: Quality of alternatives is positively associated with attitude

We postulate that sunk costs and relationship costs are the perceived behavior control for bloggers' switching behavior. Sunk costs refer to the irrecoverable time or effort bloggers have put into using their blogs. Relationship costs address bloggers' uncomfortable perception of losing their online relationships with other bloggers or visitors. The two factors focus on the barriers that bloggers will encounter

when considering switching. High levels of sunk costs and relationship costs are likely to prevent bloggers' switching behavior. As mentioned earlier, data transfer and relationship transfer are critical issues that affect bloggers' switching decision. Data and relationships are highly important and take users much time and effort during the adoption of social technologies. Therefore, we posit that both sunk costs and relationship costs are important factors in the context of this study. Consistent with prior studies [e.g., 10], we propose that they will have negative impacts on switching intention. The two hypotheses are proposed:

H4: Sunk costs are negatively associated with switching intention
H5: Relationship costs are negatively associated with switching intention

Note that prior research suggests subjective norms may be more important in mandatory settings rather than in voluntary contexts [11]. Some scholars also find that subjective norms do not play an important role in affecting the usage of blogs [12]. Therefore, we do not consider the impact of subjective norms in the switching model. In sum, Figure 1 depicts the research model of this study.

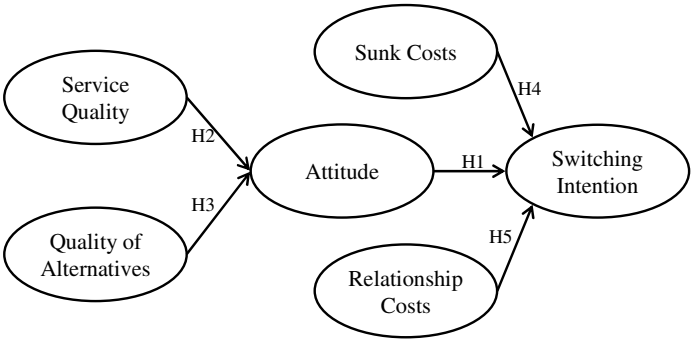


Fig. 1 The research model

3 Research Method and Data Analysis

We employed a survey method to test the research model. The constructs in the model were measured with multi-item scales, all adapted from the prior literature [9, 10, 13, 14]. Details about measures of constructs are listed in Table 1.

Table 1 Measures of constructs

Construct	Items
Service Quality [9]	1. Overall, I consider my current blog service to be excellent
	2. I believe that the general quality of my current blog service is high
	3. The quality of my current blog service is generally (7-point semantic differential scale, from -3=very poor to 3=excellent)
Quality of Alternatives [14]	1. I know that there are alternative blog services I can switch to. *
	2. There are other blog services that provide high service quality.
	3. There are blog services I find more attractive than the one I am using.
Attitude [9]	For me, switching from my current blog service to a new blog service would be (7-point semantic differential scales, from -3 to 3)
	1. A bad idea…A good idea
	2. Useless…Useful
	3. Harmful…Beneficial
	4. Foolish…Wise
	5. Unpleasant…Pleasant
	6. Undesirable…Desirable
Sunk Costs [13]	1. A lot of energy, time, and effort has gone into using my blog service.
	2. Overall, I have invested a lot in using my blog service.
	3. All things considered, I have put a lot into previous use of my blog service.
	4. I have spent a lot of time and effort on my blog.
	5. I have invested much into using my blog service.
Relationship Costs [10]	1. I would lose some of the visitors of my current blog if I switched to another blog service.
	2. I am more comfortable interacting with the visitors of my current blog than I would be if I switched blog services. *
	3. The visitors of my current blog matter to me
	4. I enjoy interacting with the visitors of my current blog.
Switching Intention [14]	1. I am considering switching from my current blog service.
	2. The likelihood of me switching to another blog service is high.
	3. I am determined to switch to another blog service.

Note: 1) If not specified, items use 7-point Likert scales, from 1=strongly disagree to 7=strongly agree; 2) * denotes that the item was deleted due to low factor loading.

We built up an online questionnaire to collect data from bloggers. We sent URL to members of many blog communities, as well as local forums in Hong Kong. To increase the sample size, we also provided lucky draw prizes as incentives. In total, 299 usable responses were collected. There were 169 female and 130 male bloggers.

The sample consisted of many highly educated bloggers (University or above). Nearly 90% of respondents aged from 19 to 30, and 258 respondents had more than one year of blogging experiences. Xanga.com was the blog service that occupied more than half of the respondents in our sample.

We adopted Partial Least Squares, a structural modeling approach, to analyze the data. We followed the two-step analytical procedure [15]: the measurement model and structural model. Table 2 and Table 3 illustrate results of convergent and discriminant validities for the measurement model. We found that all values of composite reliability exceeded 0.7. All values of average extracted variance were greater than 0.5 (in Table 2). In addition, all squared roots of averaged variance extracted surpassed any correlations of corresponding constructs (in Table 3). According to prior criteria on convergent and discriminant validities [16], we postulate that the measurement model was sufficient for this study.

Table 2 Convergent validity of constructs

Construct	Item	Loading
Service Quality (SQ)	SQ1	0.87
Composite Reliability=0.91	SQ2	0.90
Average Variance Extracted=0.78	SQ3	0.88
Quality of Alternatives (QA)	QA1	0.79
Composite Reliability=0.86	QA2	0.95
Average Variance Extracted=0.76		
Attitude (ATT)	ATT1	0.84
Composite Reliability=0.96	ATT2	0.89
Average Variance Extracted=0.78	ATT3	0.87
	ATT4	0.90
	ATT5	0.91
	ATT6	0.89
Sunk Costs (SC)	SC1	0.84
Composite Reliability=0.93	SC2	0.85
Average Variance Extracted=0.73	SC3	0.84
	SC4	0.87
	SC5	0.85
Relationship Costs (RC)	RC1	0.65
Composite Reliability=0.83	RC2	0.72
Average Variance Extracted=0.62	RC3	0.96
Switching Intention (SI)	SI1	0.93
Composite Reliability=0.94	SI2	0.91
Average Variance Extracted=0.85	SI3	0.93

Table 3 Discriminant validity and correlations of constructs

	SQ	QA	ATT	SC	RC	SI
SQ	**0.88**					
QA	-0.09	**0.87**				
ATT	-0.24	0.44	**0.88**			
SC	-0.36	-0.06	-0.04	**0.85**		
RC	0.36	0.01	-0.08	-0.37	**0.79**	
SI	-0.20	0.31	0.63	-0.17	-0.17	**0.92**

Note: 1) SQ=Service Quality, QA=Quality of Alternatives, ATT=Attitude, SC=Sunk Costs, RC=Relationship Costs, SI=Switching Intention; 2) The bold diagonal elements are squared roots of AVEs.

Next, we performed the structural model. As shown in Table 4, all proposed hypotheses were supported by the data. Service quality had a negative, while quality of alternatives had a positive impact on attitude. In turn, attitude positively affected switching intention. Sunk costs and relationship costs exhibited negative impacts on switching intention. In total, the research model explained 44.8% of variance for switching intention.

Table 4 Results of hypothesis testing

Hypothesis	Path coefficient	T-value
Service quality→attitude	-0.20	3.53***
Quality of alternatives→attitude	0.42	7.74***
Attitude→switching intention	0.60	14.65***
Sunk costs→switching intention	-0.22	4.57***
Relationship costs→switching intention	-0.20	2.69**

Note: **=p<0.01, ***=p<0.001.

4 Discussion and Conclusion

In this study, we discuss blog service switching in the blogosphere. We propose the switching model based on the theory of planned behavior. The model is further tested through an empirical survey. The results fit much with the proposed hypotheses. We find that:

- Two quality beliefs (i.e., current blog service quality and quality of alternative blog services) show important impacts on bloggers' attitude of performing switching behavior. It explains the cognitive process of quality comparison between different blog services.
- Two types of costs (i.e., sunk costs and relationship costs) reveal bloggers' perceived barriers when considering the switching decision. These two barriers are

likely to be created when switching behavior takes place in the context of social technologies. The findings confirm that they can directly prohibit switching intention.

- Finally, attitude is found to positively affect switching intention. Along with the significant impacts from two quality beliefs and two types of costs, the theory of planned behavior appears to be a solid perspective that can provide good explanatory power on blog service switching.

4.1 Implications

This study establishes a theory of planned behavior based switching model. It extends our understandings on the usage behavior of information technologies. The switching behavior clearly involves aspects of both membership augmentation and maintenance. Moreover, this research enriches prior studies on understanding the substantial social technology: blog [e.g., 12, 17]. In practice, the switching model can offer immediate guidelines to online blog companies through two dimensions.

To attract new bloggers

Lower sunk costs and relationship costs would increase the intention of switching. Therefore, it is easier to attract new bloggers to a new blog service when they have not invested much on using the old one. Online blog companies could also provide helpful functions to mitigate the impacts from two types of costs. Provision of "import tools" is the option that can facilitate the switching process. Further, community-oriented features or friend invitation tools can help the blogger to reestablish their relationships with other bloggers or visitors. Another key strategy would be providing better blog services. Bloggers will compare the current blog service with the new one in terms of their quality assessments. Thoughtful companies, which provide attractive and stable blogging functions to meet bloggers' need, are more likely to gain larger market share.

To maintain current bloggers

We recommend online blog companies to promote bloggers' participation level. The more they use blog services, the more likely that their sunk costs and relationship costs will become higher. In particular, blogging composition competitions and online community activities are two possible approaches to increase sunk costs and relationship costs respectively. To better maintain bloggers, online blog companies could provide easy-to-use tools to facilitate their blogging or knowledge sharing activities on bogs. For instance, ScribeFire Blog Editor [18] is a well-known add-on for Firefox Browser. It integrates well with the browser and easily helps bloggers to post blog entries when they find interesting information on the Internet. In addition, online blog companies could advocate community-oriented features for their blog services. Provisions of helpful friend invitation tools and friend management functions would be good options as well.

Apart from the practical implications for online blog companies, this study can also contribute to the fields of other social technologies, such as virtual communities, microblogging, and social networking sites. Practitioners in these fields are recommended to consider the aforementioned two dimensions, and further obtain sizable membership.

4.2 Limitation and Future Research

In light of our exploratory work, this research still has some limitations. First, we collected data through a convenient sample (i.e., in Hong Kong). We recognize that a convenient sample may affect the generalization of the switching model. A bigger sample or comparison of samples from different regions could provide more convincing results. Second, this study adopts a cross-sectional survey approach and use switching intention to capture bloggers' subjective willingness of switching away from current blog services. To further extend this line of study, we recommend that scholars could conduct longitudinal survey studies and to investigate whether bloggers with higher switching intention would actually change their blog service to a new one after a period of time. Finally, the switching model explains 44.8% of variance for switching intention. Although it is regarded as of big prediction power [19], there may be other important independent variables as well. For instance, it is possible that bloggers will switch their blog services because they think the new blog service would bring them with more valuable relationships with others. We welcome future researchers' efforts on including more insightful variables into the switching model.

References

1. Technorati. Blog Directory - Technorati (November 4, 2010,), http://technorati.com/blogs/directory/
2. CNNIC, Chinese blog market and blogging behavior report in 2008-2009 (2009)
3. CNNIC, Chinese blog market report in 2007 (2007)
4. White, D.: Day 1: Who Are the Bloggers? - Technorati Blogging, P. 2 (November 4, 2009), http://technorati.com/blogging/article/day-1-who-are-the-bloggers/page-2/
5. Bhattacherjee, A.: Understanding Information Systems Continuance: An Expectation-Confirmation Model. MIS Quarterly 25, 351–370 (2007)
6. Venkatesh, V., et al.: User Acceptance of Information Technology: Toward a Unified View. MIS Quarterly 27, 425–478 (2003)
7. Lenhart, A., Fox, S.: Bloggers: A Portrait of the Internet's New Storytellers. In: Pew Internet & American Life Project, Washington D.C (July 19, 2006)
8. Ajzen, I.: The Theory of Planned Behavior. Organizational Behavior and Human Decision Processes 50, 179–211 (1991)
9. Bansal, H.S., et al.: Migrating to New Service Providers: Toward a Unifying Framework of Consumers' Switching Behaviors. Journal of the Academy of Marketing Science 33, 96–115 (2005)

10. Burnham, T.A., et al.: Consumer Switching Costs: A Typology, Antecedents, and Consequences. Journal of the Academy of Marketing Science 31, 109–126 (2003)
11. Komiak, S.Y.X., Benbasat, I.: The Effects of Personalization and Familiarity on Trust and Adoption of Recommendation Agents. MIS Quarterly 30, 941–960 (2006)
12. Hsu, C.-L., Lin, J.C.-C.: Acceptance of blog usage: The roles of technology acceptance, social influence and knowledge sharing motivation. Information & Management 45, 65–74 (2008)
13. Jones, M.A., et al.: Why Customers Stay: Measuring the Underlying Dimensions of Services Switching Costs and Managing Their Differential Strategic Outcomes. Journal of Business Research 55, 441–450 (2002)
14. Kim, G., et al.: A Study of Factors that Affect User Intentions toward Email Service Switching. Information & Management 43, 884–893 (2006)
15. Hair, J.F., et al.: Multivariate Data Analysis, 5th edn. Prentice Hall, Englewood Cliffs (1998)
16. Fornell, C., Larcker, D.F.: Structural Equation Models with Unobservable Variables and Measurement Errors. Journal of Marketing Research 18, 382–388 (1981)
17. Zhang, K.Z.K., et al.: Understanding the role of gender in bloggers' switching behavior. Decision Support Systems 47, 540–546 (2009)
18. ScribeFire. ScribeFire: Fire up your blogging (November 5, 2010), http://www.scribefire.com/
19. Cohen, J.: Statistical Power Analysis for the Behavioral Science, 2nd edn. Lawrence Erlbaum Associates, Hillsdale, NJ (1988)

User Contribution and IT-Enabled Features of Online Review Platforms: A Preliminary Study

Kem Z.K. Zhang, Sesia J. Zhao, Matthew K.O. Lee, and Huaping Chen

Abstract. Online review platforms are increasing popular websites where consumers can easily find others' evaluations, opinions, or comments on many products or services. These review messages become important online information sources affecting consumers' purchase decision, and thus are essential assets for online review platforms. The survival of platforms largely depends on a number of users' voluntary contribution of online reviews. In this study, we shed light on the user contribution of online review platforms. In particular, we attempt to understand how information technology (IT) enabled features can facilitate users' online review contribution. To achieve this objective, we conduct a preliminary study on a Chinese online review platform. The findings confirm that social networking technology and virtual community technology provide helpful IT-enabled features to attain a high level of user contribution on the platform. Implications for both researchers and practitioners are discussed.

Kem Z.K. Zhang
Suzhou Research Institute, City University of Hong Kong, 166 Ren'ai Road, Dushu Lake Higher Education Town, SIP, Suzhou, Jiangsu, China
e-mail: zikzhang@cityu.edu.hk

Sesia J. Zhao
USTC-CityU Joint Advanced Research Center, 166 Ren'ai Road, Dushu Lake Higher Education Town, SIP, Suzhou, Jiangsu, China
e-mail: sesiazj@mail.ustc.edu.cn

Matthew K.O. Lee
Department of Information Systems, City University of Hong Kong, Tai Chee Avenue, Kowloon, Hong Kong SAR, China
e-mail: ismatlee@cityu.edu.hk

Huaping Chen
School of Management, University of Science and Technology of China, Jinzhai Road, Hefei, Anhui, China
e-mail: hpchen@ustc.edu.cn

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 85–94.
springerlink.com

1 Introduction

Recent worldwide diffusion of Web 2.0 technologies has provided many opportunities for users to generate online content. In the domain of marketing and consumer behavior, one type of common user-generated content (UGC) is online consumer reviews. These online reviews contain evaluations, opinions, or comments on various products or services. They have been found in many Internet media, including brand websites, blogs, virtual communities, and social networking sites. In particular, online review platforms arise as a type of websites that systematically collect a large volume of online reviews from many users. They provide a channel for users to contribute review messages on a broad range of products. Popular examples of online review platforms include Epinions.com, Tripadvisor.com, Yelp.com, and Dianping.com.

Online consumer reviews have become highly adopted content on the Internet. A survey report from Pew Internet & American Life Project revealed that many online users tend to read online reviews before making purchase decision [1]. These review messages have become important online information sources that can affect consumers' purchase behavior [2]. Therefore, they are the most important assets for online review platforms. A platform with sufficient reviews on various products is more likely to be adopted by consumers. The survival of these platforms largely depends on a number of users' voluntary contribution of online reviews.

Although it is imperative to understand reasons behind user contribution on the platform, prior research on this area is still limited [3]. Some scholars employ social exchange theory to understand the benefits and costs of users' contribution behavior [4]. On the other hand, the role of information technology (IT) and how IT-enabled features on the platform can promote user contribution are less explored. Among the utilities that may motivate users to articulate online reviews, Hennig-Thurau [3] initially suggested that platform assistance could help users contribute more. Following this line of research, this study attempts to understand how certain IT-enabled features are beneficial to user contribution on online review platforms. The ITs that we are interested in are social networking technology and virtual community technology.

Social networking technology has gained much attention in recent years. Facebook and Myspace are well-known emerging social networking websites with a huge number of users. The power of social networking technology attracts great interests from many practitioners, including famous IT companies (e.g., Microsoft, Google, and Apple). Previous studies postulate that social networking technology has great potential to create and maintain interpersonal relationships [5]. It may have positive influence on the usage of Internet media [6].

In addition, virtual community technology has been widely applied and studied in the past few years [e.g., 7, 8, 9]. In an online discussion forum, members can collectively discuss subjects of similar interests. It helps the members to foster social relationships and establish a community with rich knowledge [7].

For online review platforms, some pioneers have been attempting to take credit for the two ITs. For instance, Yelp.com allows users to build up personal social networks. Dianping.com has been adopting both social networking technology and

virtual community technology. Users on Dianping.com can add other users as online friends. They can also join forums hosted by the platform and interact with others.

In sum, it would be practically relevant and important to examine whether the two IT-enabled features could function well on online review platforms. In the following sections, this study draws on the literature of social network perspective and social capital theory. The theoretical background further guides us to develop hypotheses regarding to several IT-enabled factors. Next, to test these hypotheses, we conducted a preliminary study on the Chinese online review platform: Dianping.com. Finally, we conclude this study with a discussion of implications based on the findings.

2 Theoretical Background

2.1 Social Network Perspective

Social network perspective is based on several structural components, including nodes, ties, and networks [10]. Nodes are the entities studied, which can be individuals, groups, or organizations. Ties are the linkage between two nodes. They reveal relationships among the entities. Nodes and ties compose a network. There are two types of social networks: socio-centric network and ego-centric network [11]. The former network pertains to a whole social network. On the other hand, the latter refers to a focal node and other nodes that have direct relationships with it.

In the context of this study, we focus on ego-centric networks. Each user on online review platforms can be regarded as a focal node. First, a user and his/her online friends compose a general sense of social network on the platform. In advance, we extend the user's ego-centric network by adding forums, which are affiliated to the platform, as another type of nodes. Thus, two types of nodes can link to a focal user on the platform. One is the user's online friends and the other is forums s/he enrolls in (See Figure 1).

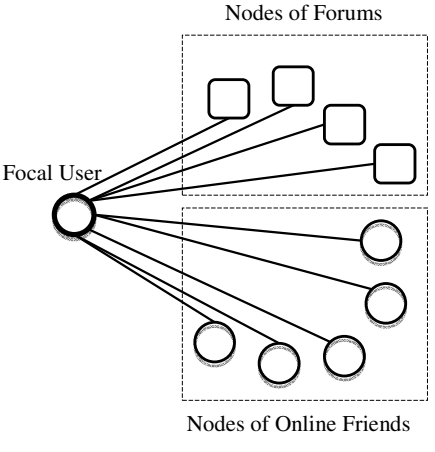


Fig. 1 Ego-centric network of a user on online review platforms

2.2 Social Capital Theory

In general, social capital refers to "resources embedded in a social structure that are accessed and/or mobilized in purposive action" [12, p. 29]. Nahapiet and Ghoshal [13] is one of the founding studies that apply social capital theory at the organizational level. They investigated intellectual capital creation in organizations and advanced the framework of social capital into three dimensions: cognitive, structural, and relational capital. The cognitive dimension of social capital pertains to the shared understandings and common language of members in organizations. Structural capital refers to the network ties and network configuration for the overall organization. Finally, the relational dimension addresses the norms, identification, and commitment among relationships in organizations.

The recent study from Wasko and Faraj [14] extends social capital theory to explain individuals' knowledge contribution behavior. They conceptualized the three dimensions of social capital at the individual level. It is proposed that individuals' knowledge creation and contribution can benefit from the social capital embedded among their relationships with others [14]. In this study, we follow the perspective from Wasko and Faraj's research. With concerns of IT-enabled features on online review platforms, we investigate the impacts of some structural and relational social capital factors on the review contribution behavior.

3 Hypotheses Development

3.1 Social Networking Technology

Websites equipped with social networking technology can allow users to "(1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system" [5, p. 211]. With the implementation of social networking technology, users on online review platforms can add various online friends into their ego-centric networks. In this study, we discuss two kinds of variables arising from this IT-enabled feature on the platform: friend network size and social support.

Friend network size captures the size of online friends in the ego-centric network. It is regarded as a structural capital factor. A large friend network size indicates that the focal user possesses many relationships with others on online review platforms. Hence, the online reviews s/he contributed could be instantly distributed to all online friends in the network. It increases the possibility for these reviews to help more others to make better purchase decision. High perceived utilities of helping others are more likely to promote users' review contribution [3]. In addition, users with a large friend network size imply a high level of centrality, conducing them to recognize and comply with norms of review contribution [14]. The contribution behavior of online friends is also more likely

to influence the focal user to contribute more. Therefore, we propose the following hypothesis.

H1: Friend network size is positively related to review contribution.

Social support is defined as the support a user received regarding to his/her review contribution behavior. The support may come from both explicit online friends in the network or from other uses on the platform. In this study, we refer it as a relational capital factor.

Social support reflects others' norms about online review contribution. It conveys the acknowledgment and compliments to the focal user. In addition, users with a high level of support are more likely to realize the social benefits and self-enhancement from contributing online reviews on the platform. These positive reinforcements may lead to their further contribution behavior [3]. Hence, the following hypothesis is provided:

H2: Social support is positively related to review contribution.

Moreover, prior researchers indicate that network size may have a high correlation with social support [15]. In a large friend network, the focal user has many connections with others. It greatly increases the chance of receiving social support from online friends after s/he contributes reviews on the platform. Therefore, we provide the following hypothesis:

H3: Friend network size is positively related to social support.

3.2 Virtual Community Technology

Virtual communities provide opportunities for online users of similar interests to share knowledge and communicate with each other [8]. This technology allows online review platforms to operate many subordinate forums on a range of subjects. Thus, users can join some of the forums and communicate with others. In this study, we shed light on two factors pertaining to virtual community technology: forum network size and topic initiation.

Forum network size refers to the forums a user subscribes to. It is viewed as a structural capital factor. Although the forums can cover various subjects, all subjects should be related to the central theme of the platform. If the focal user joins as a member of many forums, then s/he is likely to share core values of the platform. In addition, the user's online reviews are more likely to be aware of by many others. Hence, similar to friend network size, we propose that:

H4: Forum network size is positively related to review contribution.

In this study, we refer to topic initiation as the number of messages a user posts to the forums. The user may explicitly join these forums or may not. The intensity of topic initiation can describe the relationships between the user and forums. Compared to message browsing or replying behavior, message posting behavior generally requires more time and efforts from the user. A high level of topic initiation may closely relate to the user's identification and commitment to the forums,

which in turn can affect their review contribution behavior [7, 14]. Therefore, we conceive topic initiation as a relational capital factor. The following hypothesis is proposed:

H5: Topic initiation is positively related to review contribution.

In addition, we contend that there may be a positive relationship between forum network size and topic initiation. A user joins a forum implies s/he shares common interests with it and has a sense of membership. If the user has participated in many forums, then s/he may augment these senses of membership, which further increases the chance of initiating topics [16]. Thus, the final hypothesis is:

H6: Forum network size is positively related to topic initiation.

In sum, the research model of this study is illustrated in Figure 2.

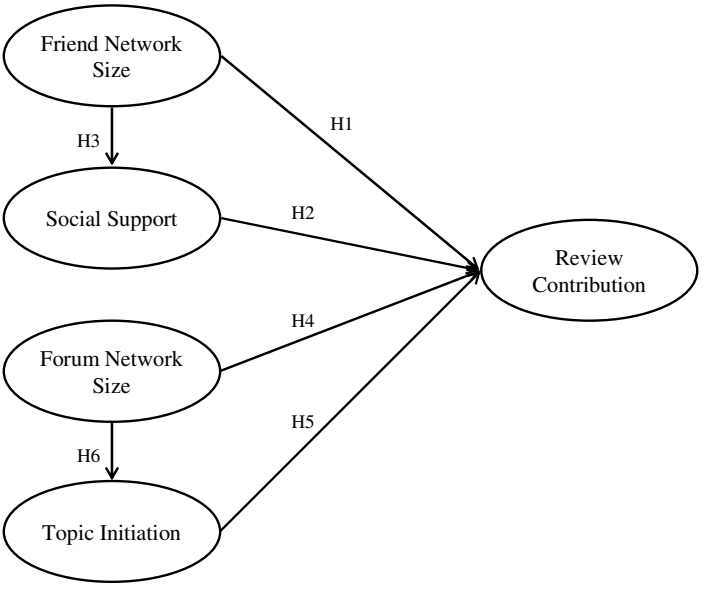


Fig. 2 The research model

4 Research Method

To test the six hypotheses, we conducted an empirical study on an online review platform. The Chinese platform, Dianping.com, is adopted as our research site. Along with the fast development of e-business in China, Dianping.com has become one of the most popular online review platforms in the country. In addition, the platform has adopted both social networking technology and virtual community technology for a few years. Many registered users of the platform are familiar

with the two IT-enabled features. Thus, it is suitable to test the proposed hypotheses in this context.

We adopted real observational data from the platform. Review contribution was measured by the number of reviews contributed by a user. Friend network size was operated as the number of his/her online friends. Social support was calculated based on the total numbers of compliments the user received. On Dianping.com, other users can send "virtual flowers" to the user as a way of expressing compliments. Some "virtual flowers" express that "your review is great", some is about "your review is very helpful", and some refers to "I support your review". Further, forum network size was estimated by the number of forums the user subscribed to. Finally, we use the number of forum messages the user posted to measure topic initiation.

Given the preliminary nature of this study, we randomly chose eighty users on Dianping.com. Each user's personal page on the platform was carefully examined to obtain the data of five factors. To ensure the data accuracy, their personal pages were examined for two to three times. The details of factors, measures, and descriptive statistics are listed in Table 1.

Table 1 Factor's measure, mean, medium, and standard deviation

Factor	Measure	Mean	Med.	SD
Review contribution	Number of online reviews	142.5	108.5	128.9
Friend network size	Number of online friends	114.3	70.0	126.9
Social support	Number of "virtual flowers" received	3595.9	674.5	4900.6
Forum network size	Number of forums subscribed to	15.1	10.0	16.9
Topic initiation	Number of forum messages posted	1487.6	356.5	3287.6

5 Results

We analyzed the data with a structural equation modeling approach, using the partial least square (PLS) algorithm. The method can estimate multi-stage models and have few strict requirements on sample size and sample distribution [17]. Thus, it was deemed to be appropriate for this study.

As shown in Figure 3, the results indicated that most of the hypotheses were supported. Friend network size (β=0.627, t=7.54) and social support (β=0.342, t=4.76) positively affected review contribution. Social support was also determined by friend network size (β=0.620, t=8.77). Forum network size had positive impacts on both topic initiation (β=0.517, t=4.64) and review contribution (β=0.148, t=2.42). Topic initiation has no significant impact on review contribution (β=-0.105, t=1.06). The variances explained to social support and topic initiation were 38.5% and 26.7%, respectively. In total, review contribution was explained to 81.8% of variances by all the four antecedents.

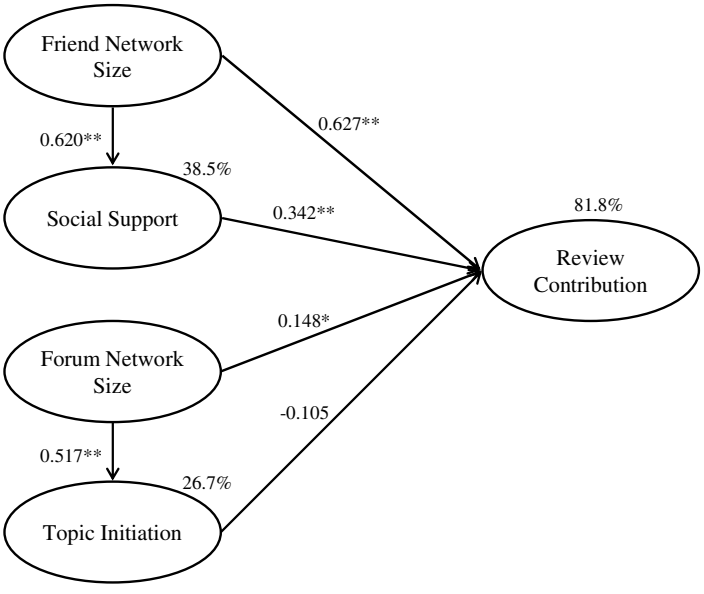


Fig. 3 PLS results of the research hypotheses (*=p<0.05, **=p<0.01)

6 Discussion and Conclusion

User contribution is highly important for the success of online review platforms. In this study, we contend that social networking technology and virtual community technology are the two potential helpful IT-enabled features that could help online review platforms attain a high level of user contribution.

6.1 Implications

This study is one of the earliest ones that shed light on the facilitating effects of IT-enabled features: social networking technology and virtual community technology. The findings indicate that friend network size has the most significant impact, followed by social support and forum network size. Surprisingly, topic initiation does not exhibit positive influence on review contribution. It further implies that the facilitating effects from social network technology may be relatively stronger than that from virtual community technology.

We derive the literature of social network perspective and social capital theory. The two IT-enabled features could generate ego-centric network for each registered user on the platform. Hence, users may develop their social capital in the structural and relational dimensions, which in turn may promote them to contribute more online reviews on the platform.

In practice, designers may implement social networking technology and virtual community technology into their online review platforms. Thus, registered users can have the opportunities to build up personal social networks and discuss subjects of

interests within forums. As demonstrated from the research model, platform designers may try to foster users' social and relational capital. In particular, designers may need to pay more attention to social networking technology. The power of this technology may be relatively stronger and have higher influence on user contribution on the platform.

6.2 Limitations and Future Study

This research attempts to address user contribution and IT-enabled features on online review platforms. A few of limitations should be recognized. First, as a preliminary research, this study adopted a relatively small sample size and a single research site. Thus, to increase the generalizability of findings, future research will employ a larger sample size or more online review platforms to test the hypotheses.

In addition, research may also consider possible impacts from culture. The findings of the research model may vary in different cultures. For instance, eastern cultures may have more collective elements than western cultures. From this perspective, the effects of social variables may differ across some countries.

Third, we employed a cross-sectional approach to collect data. A longitudinal data collection may further provide insightful findings to understand the dynamics of review contribution.

Finally, user contribution may be affected by other factors, such as motivational factors (extrinsic and intrinsic) and also factors in the cognitive dimension of social capital. Therefore, scholars are encouraged to incorporate more factors and improve our understandings on user contribution on the platform.

References

1. Horrigan, J.B.: Online Shopping: Internet users like the convenience but worry about the security of their financial information. Pew Internet & American Life Project (2008)
2. Zhang, K.Z.K., et al.: Understanding the Informational Social Influence of Online Review Platforms. In: Proceedings of the International Conference on Information Systems (ICIS), St. Louis, Missouri, USA (2010)
3. Hennig-Thurau, T., et al.: Electronic word-of-mouth via consumer-opinion platforms: What motivates consumers to articulate themselves on the Internet. Journal of Interactive Marketing 18, 38–52 (2004)
4. Tong, Y., et al.: Understanding the Intention of Information Contribution to Online Feedback Systems from Social Exchange and Motivation Crowding Perspectives. In: Proceedings of the 40th Hawaii International Conference on System Sciences, Waikoloa, Big Island, Hawaii, USA (2007)
5. Boyd, D.M., Ellison, N.B.: Social Network Sites: Definition, History, and Scholarship. Journal of Computer-Mediated Communication 13, 210–230 (2007)
6. Haythornthwaite, C.: Social networks and Internet connectivity effects. Information, Communication & Society 8, 125–147 (2005)

7. Chiu, C.-M., et al.: Understanding knowledge sharing in virtual communities: An integration of social capital and social cognitive theories. Decision Support Systems 42, 1872–1888 (2006)
8. Bagozzi, R.P., Dholakia, U.M.: Intentional social action in virtual communities. Journal of Interactive Marketing 16, 2–21 (2002)
9. Hsu, M.-H., et al.: Knowledge sharing behavior in virtual communities: The relationship between trust, self-efficacy, and outcome expectations. International Journal of Human-Computer Studies 65, 153–169 (2007)
10. Borgatti, S.P., Cross, R.: A relational view of information seeking and learning in social networks. Management Science 49, 432–445 (2003)
11. Scott, J.: Social network analysis: a handbook. Sage Publications, London (2000)
12. Lin, N.: Social Capital: A Theory of Social Structure and Action. Cambridge University Press, Cambridge (2001)
13. Nahapiet, J., Ghoshal, S.: Social Capital, Intellectual Capital, and the Organizational Advantage. Academy of Management Review 23, 242–266 (1998)
14. Wasko, M.M., Faraj, S.: Why should I share? Examining social capital and knowledge contribution in electronic networks of practice. Management Information Systems Quarterly 29, 35–57 (2005)
15. Schaefer, C., et al.: The Health-Related Functions of Social Support. Journal of Behavioral Medicine 4, 381–406 (1981)
16. Yoo, W.-S., et al.: Exploring the Factors Enhancing Member Participation in Virtual Communities. Journal of Global Information Management 10, 55–71 (2002)
17. Arazy, O., Nov, O.: Determinants of Wikipedia Quality: the Roles of Global and Local Contribution Inequality. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work, Savannah, GA (2010)

Implementation and Performance Analysis of Fuzzy Replica Replacement Algorithm in Data Grid

Toukir Imam and Rashedur M. Rahman

Abstract. Replication is a technique used in Data Grid environments that reduces access latency and network bandwidth utilization by creating multiple copies of data in different locations. Replication increases data availability thereby enhancing system reliability. One of the challenges for replication is to select the candidate sites to host replica. Other challenge is to select replica(s) to replace when storage capacity of a site is full. Our current research presents a replica replacement algorithm based on Fuzzy logic. We build a Fuzzy rule-base called Fuzzy12 algorithm and implement it on GridSim simulator. The study of our replica replacement algorithms is carried out using a model of the European Data Grid (EDG) Testbed 1 [2] sites and their associated network geometry. This paper contains a detail description of the Fuzzy12 algorithm, its implementation and the set-up of the EDG simulation. Our simulation results demonstrate that the Fuzzy algorithm outperforms the LRU replacement algorithm with different perspective, for example, hit ratio, byte hit ratio, miss rate etc.

1 Introduction

In May 1999, David Gedye first used a large number of Internet connected computers as a supercomputer for searching external intelligence, the Search for Extraterrestrial Intelligence(SETI) project marks the beginning for the grid computing. The SETI@HOME project gained access to 62 Teraflop/s in 2004 which is double that of the most powerful super computer at present [3]. The use of the grid computing has since been remarkable with BOINC reaching a tremendous 5.128 Peta-Flops in Apr 24th 2010 [19]. Grid computing have advantages over traditional computers, for example, ability to make better use of computational resources,

Toukir Imam · Rashedur M. Rahman
Department of Electrical Engineering and Computer Science
North South University, Bashundhara, Dhaka, Bangladesh
e-mail: mtoukir@gmail.com, rashedur@northsouth.edu

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 95–110.
springerlink.com

ability to solve problems that could only be solved by enormous amount of computing power, and ability to synergistically harnesses and manage the resources of a large amount of computers towards a common objective[12].

Grid computers can be of two types, computational grids and data grids. The Enabling Grids for E-science E project(EGEE), which is based in the European Union and includes sites in Asia and the United States, is a follow-up project to the European Data Grid (EDG) and is arguably the largest Data Grid on the planet. This, along with the LHC Computing Grid (LCG), has been developed to support the experiments using the CERN Large Hadron Collider. The LCG project is driven by CERN's need to handle huge amounts of data, where storage rates of several gigabytes per second (10 petabytes per year) are required [4].

As Data Grid technology is developed to permit data sharing across many organizations in different geographically disperse locations, data replication becomes critical because data must be cached close to users [9]. The data grid envisioned by GriPhyN [7] is hierarchical in nature. It consists of multiple tiers with all data generating at Tier 0; Tier 1 consists of national centers; and below that there are regional centers. Each tier has its own storage capacity, which varies from tier to tier. Using the storage capacity at each tier, replicas can be placed at each tier to increase the data availability among different sites. The general idea of replication is to store copies of data in different locations so that data can be easily recovered if one copy at one location is lost or unavailable. Moreover, if data can be kept close to users via replication, data access performance can be improved dramatically. However, as it is not possible to provide unlimited storage capacity, a replica replacement policy is required. Our current research present a dynamic replica replacement algorithm based on Fuzzy logic.

Replica replacement algorithms must be tested thoroughly before deploying them in real Data Grid environments. One way to achieve a realistic evaluation of various strategies is through simulation that carefully reflects real Data Grids. To evaluate our approach we use a simulation package called GridSim [4,18]. The GridSim toolkit provides a way of simulating heterogeneous resources, users, applications, resource brokers and schedulers. It is designed over the SimJava2: Java discrete event simulation library(and uses modified SimJava2 since version 5.2). The DataGrid, an extension of the GridSim simulator, is an excellent tool for simulating data grids. In our paper we use data grid extension to simulate the European Data Grid and test the Fuzzy replication algorithm on it. The study of our replica replacement algorithms is carried out using a model of the EU Data Grid Testbed 1 [2] sites and their associated network geometry.

The rest of the paper is organized as follows: Section 2 presents the related work on data replication in grids. Section 3 provides a detailed description of our algorithm, starting by the class view of the classes for Fuzzy 12 algorithm, a step by step walk through towards the implementation of the Fuzzy12 algorithm. Section 3 outlines the design of the EDG, the mapping of the replication algorithm in the datagrid, specifies the entity characteristics and provides outline of the classes

used in the construction of the simulation. Section 4 presents and analyses the simulation results. Finally, Section 5 concludes and gives direction of future research.

2 Related Work

Over the last few years, "the Grid" in the context of resource sharing in distributed environments has gained a lot of attentions from the academic, government and commercial researchers. The term "Grid" refers to systems and applications that integrate and manage resources and services distributed across multiple control domains [6]. The Grid can accommodate very diverse resource types including storage devices, software, databases, objects, CPU power, files and others across many organizations in different geographical locations. Besides Grid computing aims to provide control of resource sharing and problem-solving collaboration with flexibility and security. The flexibility allows dynamic membership of a Grid in which Grid components can join and leave at will. Grid computing provides services with very large scale datasets and resource sharing in a global scale with heterogeneous systems.

GridFTP is a data transfer protocol which is an extended version of FTP to provide secure, reliable and effective data transport of Grid data [1]. GridFTP supports GSI and Kereberos authentication with user controlled setting of various levels of data integrity and confidentiality. It also makes a third party to initiate, monitor, and control file transfer between two other sites. It supports parallel data transfer through FTP command extensions and data channel extension. Moreover GridFTP can initiate stripped data transfer as well as it supports partial file transfer between two sites. Allcock *et al.* [1] develop a replica management service using the building blocks of the Globus Toolkit [6]. The Replica Management infrastructure includes Replica Catalog and Replica Management Services to manage multiple copies of shared data sets. The Replica Catalog maintains mappings between logical files and physical locations as well as allows users to register files with a logical filename(s) or logical collection(s). The replica catalog was implemented as a Lightweight Directory Access Protocol (LDAP) [10] directory. The replica management service by Globus does not implement the full replica management functionality and does not enforce any replication semantics.

In our earlier work [14,16] we used *p*-median, *p*-center and multi-objective model for replica placement in Grids. To relocate replicas we also use a dynamic replica maintenance algorithm. We [15] also proposed replica selection algorithm by *k*-nearest neighbour rule that uses local transfer log rather getting information from replica catalogue.

Caching is the strategy used to reduce bandwidth and server load by storing copies of document in servers closer to the user(i.e. proxy servers), so that subsequent request can be met from the proxy server[8]. Web caching is now a very popular method for reducing server load and improving user satisfaction by providing services much faster than the End-to-End model. Caching can be of two

types. Browser caching is when the web browser stores documents locally. It is implemented by most of the major web browsers. Our focus is however on the type of web caching, proxy caching. Proxy caching is done at the network level [5]. In proxy caching, a proxy server that is located between the user and destination, copies the data that passes through it. Proxy caching can be done by the ISP. Also, by web servers with demand may use regional proxies to serve the request of a particular area [8].

In case of proxy caching, the user's request for web document first goes to the proxy server. The server checks its content to see if the request can be served. If the proxy server does not contain the requested document, it forwards the request to the original destination. Once the proxy server receives the requested file from the network, it keeps a copy of the file before forwarding the result to the user. The proxy server retains the file for a certain time. If the user requests the file again within the time limit the request is served from the Proxy server thus reducing the latency.

Data Grids provide a geographically distributed resources for large-scale data intensive applications that generate large data set[11]. Since Data Grids generate and share huge data sets the time latency of the Internet and WAN can be a serious drawback for the Grid architecture. With the data that needs to be shared currently being on the scale of Terabytes and soon expected to reach Petabyte[11,17], ensuring efficient access time is a huge challenge for the grid designers.

With its successful application in the web, caching clearly has application in large scale Data Grids. In large scale Data Grids such as European Data Grid(EDG) or EGEE, caching is done by replication of the master files to several locations. Replication decisions are made based on a cost model that evaluates data access costs and performance gains of creating each replica. When the cache is full, replacement is done by evaluating frequency of use, size and last access time of the already cached files.

When the proxy server becomes full, a replacement policy is needed to replace a existing file for the new one. The decision of replacement is made by a caching algorithm. The performance of the proxy caches depends directly on the replacement algorithm. As a result caching algorithms are one of the most extensively studied subjects in data replication literature. Many caching algorithms have been proposed. Among them some of the prominent algorithms are Least Recently Used(LRU), Least Frequently Used(LFU) ,Greedy Dual-Size(GDS) Algorithm and Adaptive Replacement Algorithm(ARA). LRU evicts the file with the longest last accession time, while LFU evicts the least used file. ARA takes into account both time factor and the frequency factor. GDS, on the other hand, incorporates short term temporal locality and long term popularity of web request streams[18].

In this paper, we present the Fuzzy Caching Algorithm[5] which takes into account last access time, frequency of use and size of the files. This algorithm has been tested in web proxies with success [5]. We intend to test this algorithm in Data Grid environment. We describe the algorithm and its implementation in next two sections. Next section describes the classes and tools developed for implementing fuzzy

algorithm. Section 4 describes the implementation of the algorithm in the GridSim simulator.

3 Fuzzy Algorithm Description

We design a package called **FuzzyController** that contains four classes. Another class called **Fuzzy12** that contains the implementation of the algorithm is declared outside of the package. The class view of the package is described below:

3.1 Class View of Fuzzy Controller Package

The package FuzzyController contains four classes:

1. AbstractMF
2. FuzzyValue
3. FuzzyVariable
4. TriangularMF

The detailed description with functions of those classes is presented below:

1. *AbstractMF :* the abstract class for all membership functions. Any membership function(i.e., triangular or wave shaped) must extend this class and implement the abstract methods ***getMidOfMax(double), getMembership(double)***, ***get COS(double).***
2. *FuzzyVallue:* encapsulates a fuzzy value with name value pair.
3. *TriangularMF:* An implementation of the AbstractMF. We will only implement those methods of the Abstract MF that we need.
4. *FuzzyVarible:* It contains one or more membership function(i.e., ***TriangularMF***). The method ***getValue(double x)*** returns the FuzzyValue at a particular point ***x***. The method ***getMid(string s,double m)*** returns the mid of the membership function named ***s*** when cut at point ***m.***

3.2 The Implementation of the Fuzzy12 Algorithm

The fuzzy12 algorithm is implemented in the class ***Fuzzy12*** .The algorithm takes crisp values as input, fuzzifies them, uses fuzzy logic to infer an output and then defuzzifies it to return a crisp output. The first step is to identify the inputs. In this paper we used 3 inputs. These input variables describes a file in terms of size(SIZE),access frequency(FREQUENCY), i.e., the number of accesses, and access recency (TIME), i.e., time elapsed since last access . We have used triangular Membership Functions (MF) to describe these variables. Variable size and frequency have three MFs: LOW, MEDIUM and HIGH. Variable time has five MFs, VERY LOW, LOW, MEDIUM, HIGH, VERY HIGH. Those membership functions are depicted through Fig. 1- Fig 3.

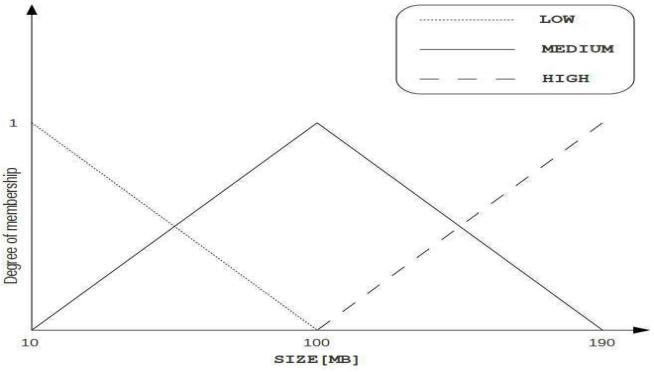


Fig. 1 Membership function of variable SIZE

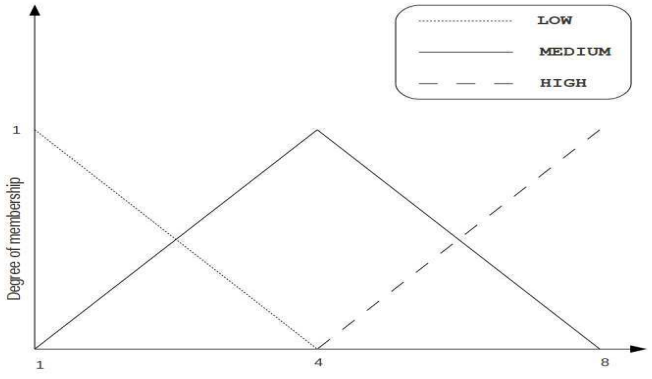


Fig. 2 Membership function of variable FREQUENCY

Having defined the membership functions, we now describe the fuzzy algorithm:

1. Measure the input data from the grid resource;
2. Fuzzification of the crisp input data into fuzzy sets;
3. Make inference from fuzzy rules;
4. Aggregation across the rules and defuzzification of the fuzzy output into a non fuzzy control action.

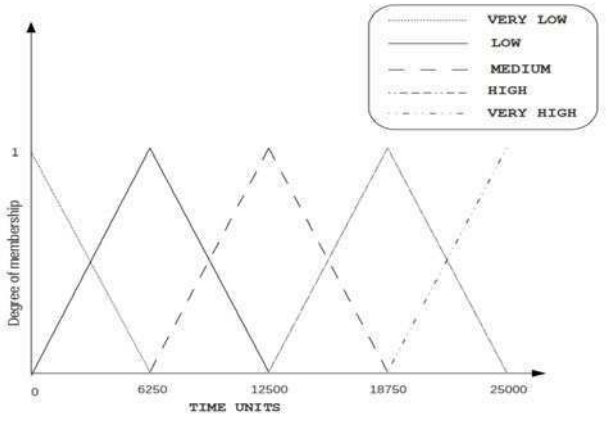


Fig. 3 Membership function of variable TIME

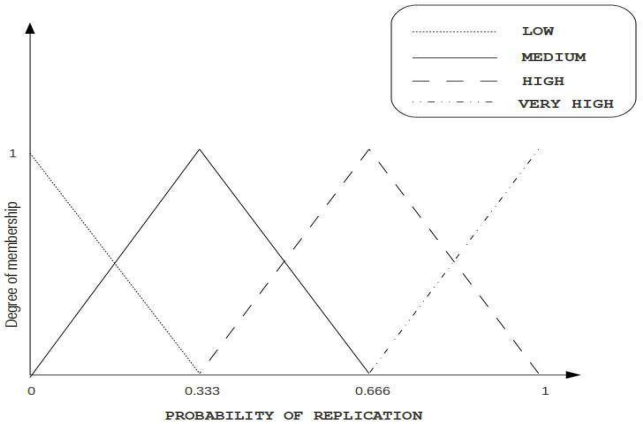


Fig. 4 Membership function of variable RP

To fuzzify the crisp data we used the **`getValue(double c)`** method of the **`FuzzyVariable`**, which checks the membership of MFs of the fuzzy variable and returns a **`FuzzyValue`** object.

From these fuzzy values, we have to use fuzzy rules to infer a fuzzy output. The first step is to use an implication relation to get a fuzzy relation from the three fuzzy sets SIZE, FREQUENCY and TIME. Here we used the mamdani min implication operator to get the relation.

```
inf = mamdaniMin(amem,bmem,cmem)
```

Finally, we use the fuzzy12 rules to find the corresponding cut and the mid values of the membership functions of the variable RP (replication probability) for the rules that fires (i.e. `inf !=0`).

If (Frequency is LOW) and (Time is VHI) and (Size is MED) then (RP is VHI)
If (Frequency is LOW) and (Time is HIG) and (Size is HIG) then (RP is VHI)
If (Frequency is MED) and (Time is VHI) and (Size is HIG) then (RP is VHI)
If (Frequency is LOW) and (Time is VHI) and (Size is HIG) then (RP is VHI)
If (Frequency is LOW) and (Time is HIG) and (Size is LOW) then (RP is HIG)
If (Frequency is MED) and (Time is HIG) and (Size is LOW) then (RP is MED)
If (Frequency is MED) and (Time is VHI) and (Size is MED) then (RP is HIG)
If (Frequency is MED) and (Time is HIG) and (Size is HIG) then (RP is HIG)
If (Frequency is HIG) and (Time is VHI and (Size is HIG) then (RP is LOW)
If (Frequency is HIG) and (Time is HIG) and (Size is HIG) then (RP is LOW)
If (Frequency is LOW) and (Time is MED) and (Size is HIG) then (RP is HIG)
If (Frequency is MED) and (Time is HIG) and (Size is MED) then (RP is MED)

Fig. 5 Fuzzy12 rules.

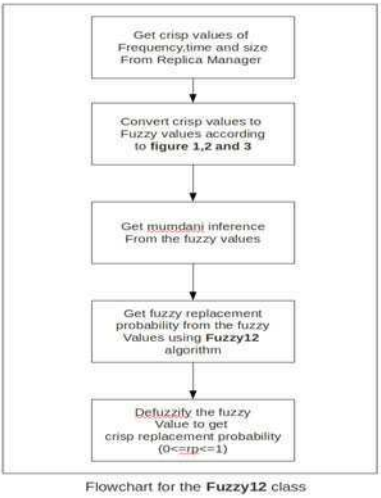


Fig. 6 Flowchart of Fuzzy12 class

For the purpose of defuzzyfication we use Centre of Sum (COS) method. The defuzzified value is between 0 and 1 which represents the probability of replacement for a file. The replacement probabilities of all files are found out and the file with maximum RP is deleted to make space for the new file. The membership function of RP is presented in Fig. 4.

The process of generating a crisp replication probability between 0 and 1 from crisp inputs using fuzzy algorithm is described by following flowchart represented in Fig.6.

4 Simulation Setup

The GridSim Package or the DataGrid extention does not have any direct facility to implement Caching. Moreover not all kinds of Network are suitable for caching. For caching to take place a user must be connected to the LocalRC instead of a RegionalRC. Because we are evaluating a caching algorithm, we are only interested in the processing of the file request by different Grid entities.

4.1 Mapping Fuzzy Algorithm to Data Grid

A diagram of the steps to process a file request and the interaction of the classes is given in Fig. 8.

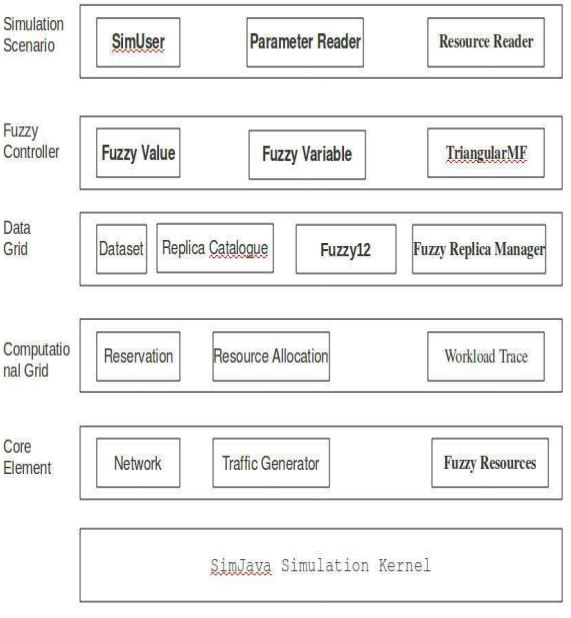


Fig. 7 Block Diagram of GridSim with Fuzzy Controller

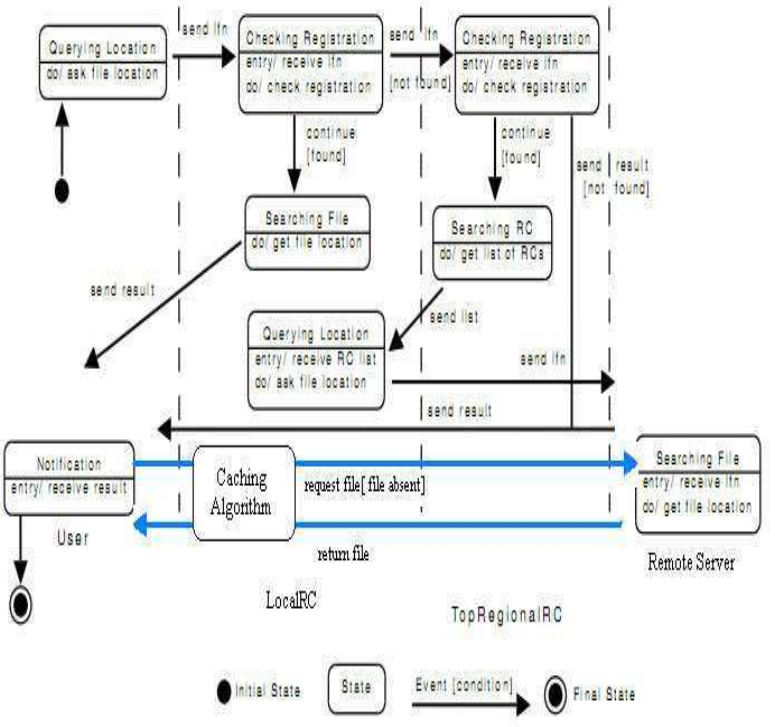


Fig. 8 A State Chart Diagram for Getting a File

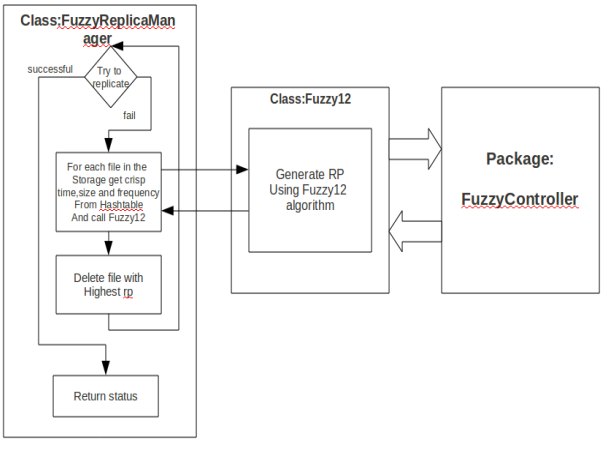


Fig. 9 Relation between FuzzyReplicaManager and Fuzzy12 class

The upper part represented in black describes the process of getting file location and the lower part(in blue) presents the process of getting the file and caching in the process. A modified user broker and replica manager class handles the replication of a file. A block diagram of the Fuzzy Controller Package and its relationship with the GridSim is given in Fig. 8 where the modified or new classes are shown in bold. Most important of all is the modification of the FuzzyReplicaManager class so that it interacts with the Fuzzy12 class. This in turn uses the Fuzzy Controller package.

Finally, Fig.9 shows the details of Replication process and the relation between Fuzzy12 class and the ReplicaManager class.

4.2 Entity Specifications

The study of our replica placement and selection algorithms was carried out using a model of the EU Data Grid Testbed 1 [2] sites and their associated network geometry. One of the sites is considered as CERN (European Organization for Nuclear Research) that will hold all the master files for simulation. A master file contains the original copy of some data samples and can not be deleted.

Each Testbed site in Fig.10 is represented by a computer terminal where a rectangle represents a router. Each link between two sites shows the available network bandwidth. The network bandwidth is expressed in Mbits/sec (M) or Gbits/sec (G). Initially all files are placed on CERN storage element. Jobs are based on the CDF use-case as described in [2]. We use the same job configuration file used by Bell et al [2]. There are six jobs with no overlapping between the set of files each job accessed. The total size of each file accessed by any job type was estimated in [18]; they are summarized in Table 2.

Resource: The specification of the grid resources are given in Table 1.

Table 1 Resource specification [18]

Resource Name(Location)	Storage(TB)	CPU rating	#Of users	Policy
RAL(UK)	2.75	49000	7	Space-Shared
Imperial College(UK)	1.8	62000	11	Space-Shared
NorduGrid (Norway)	1	20000	3	Space-Shared
NIKHEF (Netherlands)	0.5	21000	10	Space-Shared
Lyon (France)	1.35	14000	8	Space-Shared
CERN (Switzrland)	2.5	70000	6	Space-Shared
Milano (Italy)	0.35	7000	6	Space-Shared
Torino (Italy)	0.1	3000	4	Time-Shared
Rome (Italy)	0.25	6000	6	Space-Shared
Padova (Italy)	0.05	1000	4	Time-shared
Bologna (Italy)	5	80000	4	Space-Shared

However we scaled the storage capacities by 100 because to simulate with the real storage capacity will need more than 2GB of RAM.

Files: For the simulation we have created 100 files in the range of 10 to 200 MB (it is also scaled down by 100 to get results in a reasonable time) in size. At the start of the simulation the files are at the CERN server. As the simulation progresses, the files are copied to other locations.

Jobs: We defined 6 types of jobs. Each contains varying number of files. Each job has a probability of being called. The jobs and their probability is given in the Job Table:

Table 2 Estimated sizes of CDF secondary data sets (from [2])

Data Sample	Total Size (GB)	Probability
Central J/ψ	12	0.16
High p_t leptons	2	0.2
Inclusive electrons	5	0.25
Inclusive muons	14	0.13
Inclusive E_t photons	58	0.8
$Z^0 \longrightarrow b\bar{b}$	6	0.18

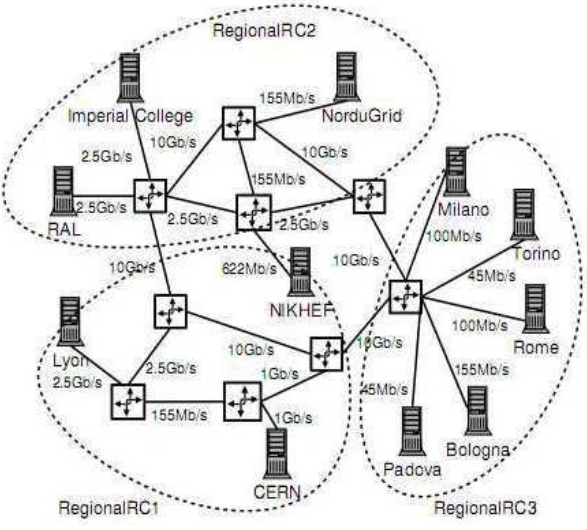


Fig. 10 EU DataGrid TestBed 1

Users: For the purpose of the simulation created 70 users. The distribution of users is given in Table 1. Each user requests for around 11 files. Users start to submit their jobs every around 5 minutes.

Replication Strategy: Each time a user asks for a file in the grid, the server keeps a copy of the file in its cache. When the cache is full it uses a replacement algorithm to determine which file to delete. In this simulation we used Fuzzy12 algorithm as replacement algorithm. We also ran the same simulation using LRU to compare it with Fuzzy12.

Network: We modelled our simulation according to the EU DataGrid TestBed 1, the network topology of the testbed is shown in Fig. 10.

4.3 Class View

Classes used in the design of the simulation can be divided into three types. We modified or extended classes of the DataGrid package to better meet our demand. There are 4 classes of this type. A short description of them is given here:

FuzzyDataGridResource: extends the **GridResource** and is actually a modification of the **DataGridResource** class. This class is modified to accommodate the **FuzzyLocalRC** and **FuzzyReplicaManager** classes, which are necessary for the simulation. Also these classes contain codes for statistical purpose.

FuzzyReplicaManager: modifies the **replicaManager** class. This class which is used to access the storage of a resource, implements the replacement policy used for caching with the help of **fuzzy12** class.

FuzzyLocalRC: extension of the **AbstractRC** class. It is modified to function with the **fuzzyReplicaManager** class. Previous classes are used to implement the fuzzy12 replacement algorithm.

FilesReader and *ResourceReader* read the file specification, storage capacity, network bandwidth from files. **UserReader** generates random grid tasks and assigns them to different users. Finally, **dataGridSim** class initiates and ends the simulation.

5 Performance Evaluations

To evaluate the efficiency of our fuzzy replica replacement algorithm we use two performance metrics, known as hit rate and byte hit rate. Hit rate is the fraction of the page requests satisfied by its local cache. One of the drawbacks of this metric is that it does not account file size into consideration. Byte hit rate overcomes it by taking non-homogeneity of file size into consideration. So, byte hit rate is the fraction of files (in bytes) satisfied over total requests (in bytes). Fig. 11.(a)-(b) depict the hit rate and byte hit rate between Fuzzy12 algorithm and the LRU. The resources are arranged from smallest storage size to largest. The resources where no replacement occurs due to their large storage size are omitted.

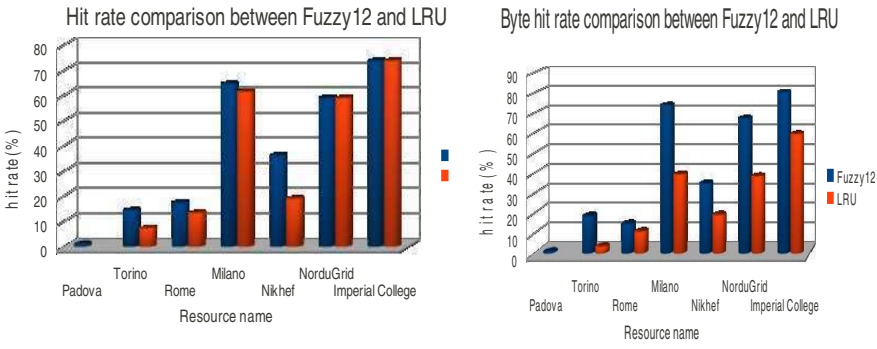


Fig. 11 a) Hit rate comparison of replica replacement algorithms, b) Byte hit ratio comparison for replica replacement algorithms.

We first note that in all scenario Fuzzy12 performed as good as LRU. With hit rate, the difference is most noticeable when the storage size is small. With large storage size, where replacement occurs rarely, Fuzzy12 and LRU evens out. In case of byte hit rate, the Fuzzy12 always performs better than the LRU. This is due to the fact that Fuzzy12 takes into consideration the file size that the LRU does not.

In this research, we consider a hierarchical replica catalogue (RC) model [18], three regional RC entities are considered, i.e., RegionalRC1, RegionalRC2, and RegionalRC3. RegionalRC1 is responsible for mapping master file locations and communicating with CERN, Lyon and NIKHEF. RegionalRC2 is responsible for NorduGrid, RAL and Imperial College, and RegionalRC3 is responsible for Padova, Bologna, Rome, Torino andMilano. Finally, TopRC oversees all three regional RCs.

As CERN is the master server (where all the master copies are initially kept) the load on CERN resource is high. A good replication strategy should reduce the load on CERN. Because we designed a hierarchical replica catalogue, if a resource in RegionalRC3 does not have certain file, it will first look within that regional RC before requesting that file to CERN. Figure 12 a) presents the number of times each resource in the regionalRC3 requests any file to CERN. When using Fuzzy12 replacement algorithm, the number of requests are significantly lower than when using LRU. This means even if a file is not in the local storage, it is nearby (within that regionalRC).

Figure 12 b) presents a comparison graph of total Mega Bytes (MBs) transferred from CERN by the resources under the RegionalRC3. The graph clearly shows a performance improvement of Fuzzy algorithm over the LRU algorithm. Finally, Figure 12 c) plots the requests (in MBs) satisfied by the local storage site by NIKHEF over different time periods. During the third time period, when sudden burst of requests are coming, most of them are satisfied by the local storage (3288 MBs) whereas LRU only satisfies (1822 MBs) requests. It clearly shows the dominance of Fuzzy algorithm over LRU even when sudden burst of request arrive which is very common to Grid domain.

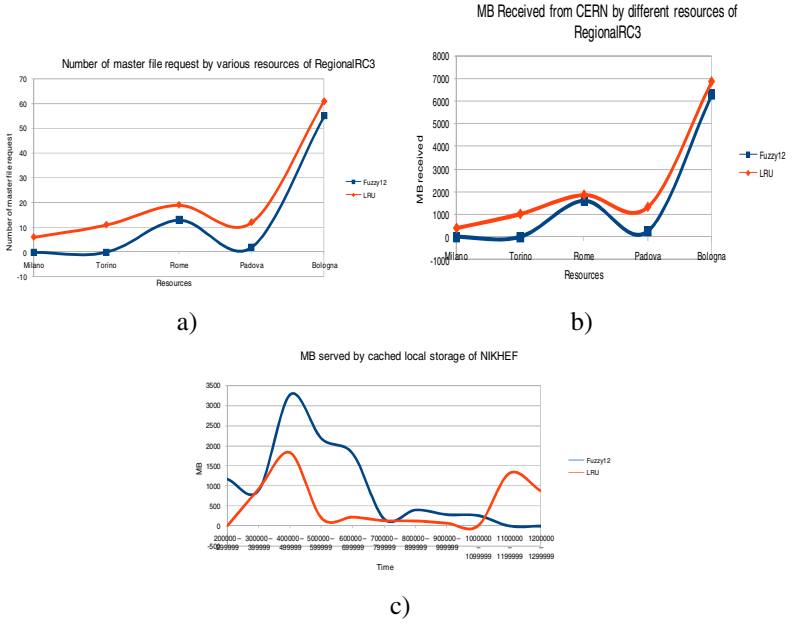


Fig. 12 a) RegionalRC3 File Requests by Replacement Algorithms, b) MB transferred from sites under RegionalRC3, c) Request in MB served by NIKHEF over time.

6 Conclusions and Future Work

In our current research we propose and implement a replica replacement algorithm by Fuzzy logic and compare the performance of our algorithm with a traditional and wide known caching algorithm known as LRU algorithm. We deploy and test our algorithm on GridSim simulator with EU Testbed1 sites. For most of the cases, our fuzzy replica replacement algorithm outperforms the LRU algorithm. We get better performance with respect to hit rate and byte hit rate. As a future work we plan to increase the number of users and other testbed sites. We also plan to compare our Fuzzy12 algorithm with other caching algorithms such as LFU and Greedy Dual.

Acknowledgements

We specially thank Himadree Shekhor Das for his help and support. We also thank Anthony Sulistio for his examples on DataGrid and Maria C. Calzarossa and Giacomo Valli for their work on Fuzzy web caching algorithm.

References

1. Allcock, B., Bester, J., Bresnahan, J., Chervenak, A.L., Foster, I., Kesselman, C., Meder, S., Nefedova, V., Quesnal, D., Tuecke, S.: Secure, Efficient Data Transport and Replica Management for High Performance Data-Intensive Computing. In: IEEE Mass Storage Conference (2001)
2. Bell, W., Cameron, D.G., Capozza, L., Millar, A.P., Stockinger, K., Zini, F.: OptorSim - A Grid Simulator for Studying Dynamic Data Replication Strategies. International Journal of High Performance Computing Applications 17(4) (2003)
3. BOINCstats – BOINC combined credit overview (retrieved August 20, 2010)
4. Buyya, R., Murshed, M.: GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing. The Journal of Concurrency and Computation: Practice and Experience (CCPE) 14(13-15) (November-December, 2002)
5. Calzarossa, M.C., Valli, G.: A Fuzzy Algorithm for Web Caching. Simulation Series Journal 35(4), 630–636 (2003)
6. Foster, I., Kesselman, C.: Globus: A Metacomputing Infrastructure Toolkit. Intl. Journal of Supercomputer Applications 11(2), 115–128 (1997)
7. High Energy Physics Experiment Website, http://www.hep.net
8. Huston, G.: Web Caching. Cisco. The Internet Protocol Journal 2(3) (2010), http://www.cisco.com/web/about/ac123/ac147/ac174/ac199/about_cisco_ipj_archive_article09186a00800c8903.html (retrieved August 20, 2010)
9. Kavitha, R., Foster, I.: Design and Evaluation of Replication Strategies for a High Performance Data Grid. In: CHEP 2001 Conference (2001)
10. Howes T.A ,Smith M.: A scalable, deployable directory service framework for the internet. Technical report, Center for Information Technology Integration,University of Michigan
11. Lamehamedi, H., Szymanski, B., Shentu, Z.: Data Replication Strategies in Grid Environments. In: Proceedings of Fifth International Conference on Algorithms and Architectures for Parallel Processing (2002)
12. LCG website, http://lcg.web.cern.ch/LCG/
13. Magoulès, F., Pan, J., Tan, K.-A., Kumar, A.: Introduction to Grid Computing. Taylor & Francis Inc., Bosa Roca (2009)
14. Rahman, et al.: Replica placement strategies in Data Grid. Journal of Grid Computing 6(1), 103–123 (2008)
15. Rahman, et al.: Replica selection strategies in Data Grid. Journal of Parallel and Distributed Computing 68(12), 1561-157 (2008)
16. Rahman, R.M., Barker, K., Alhajj, R.: Study of different replica placement and maintenance strategies in Data Grid. In: Proceeding of IEEE International Conference on Cluster Computing and Grid (CCGrid), pp. 171–178 (2007)
17. Stockinger, H., Samar, A., Allcock, B., Foster, I., Holtman, K., Tierney, B.: File and Object Replication in Data Grids. Cluster Computing 5(3), 305–314
18. Sulistio, A., Cibej, U., Venugopal, S., Robic, B., Buyya, R.: A toolkit for modelling and simulating Data Grids: An extension to GridSim. In: CCPE, vol. 20(13), pp. 1591–1609 (2008)
19. Techtarget, http://searchdatacenter.techtarget.com/sDefinition/sid80_gci773157,00.html. (retrieved on February 22, 2011)

Integrating Business Process Analysis and Complex Event Processing

Natália C. Silva, Cecília L. Sabat, César A.L. Oliveira, and Ricardo M.F. Lima

Abstract. Despite the advances on business rules theory and the increasing number of large enterprises doing efforts to model their business rules, there is still a lack for a meaningful integration between business analysis and process modeling activities. The event-driven paradigm has been shown to be an effective solution for the implementation of process rules. However, the connection between the business side of the rules and their software implementation has been made in an ad-hoc, unstructured manner. In this paper, we propose a methodology to tackle such a problem by naturally moving from informal business rules toward the implementation of a business process using complex event processing. The methodology allows for the active participation of business people at all stages of the refinement process. This is important to guarantee the correct alignment between information systems and business needs. Throughout the paper, we present an example to illustrate the application of the methodology. The methodology was applied to implement a real process of a building company.

1 Introduction

The construction of enterprise information systems is a complex task, involving several domains of knowledge and a variety of technologies and methodologies. Recent advances on information systems development have been directed towards the separation of business logic from software abstractions [12], while trying to maintain a seamless integration between these two concerns. The objective is to enable the construction of *flexible* information systems. The flexibility allows organizations to quickly adapt to environmental changes in order to capture maximum value from new oportunities [10].

Natália C. Silva · Cecília L. Sabat · César A.L. Oliveira · Ricardo M.F. Lima
Center for Informatics, Federal University of Pernambuco, Brazil
e-mail: {ncs,cls,calo,rmfl}@cin.ufpe.br

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 111–126.
springerlink.com

In the beginning of the nineties, workflow management became the major driver of these efforts and business process management became a frequent topic at academic and industrial discussion forums [14][10]. In the years that followed, several technologies were developed to further improve the capabilities of enterprises to manage their business processes, which, in turn, keeps becoming more complex. Examples of such technologies are the Service-Oriented Architecture (SOA), Business Rules [1], and Event-driven architectures [7].

The many techniques and methodologies in use today encompass a large variety of concepts and semantics (ranging from computer networks to business modeling). They approach different points of view and act on different levels of abstraction. For example, while the IT manager may be concerned with server capacity and network traffic, process analysts are concerned with activity coordination and finance managers are concerned with policy enforcements.

Enterprises often need to integrate these many frameworks in an ad-hoc manner. According to Therani [9], this can introduce a series of semantic mismatches and information loss when making correspondence from one framework to the other. Furthermore, changes at one level of abstraction can possibly not be adequately propagated to other levels due to semantic incompatibilities. Therefore, in order to achieve the sought-after flexibility, organizations need systematic means to reduce the semantic gap between the many technologies used to implement their business processes.

In this paper, we propose a systematic analysis technique for integrating business rules analysis and event-driven process implementation.

In practical terms, the approach allows for the extraction of *events* and *activities* from *informal* business rules. It aims at supporting the development of event-driven information systems directly from the business rules definitions, in a way that business people can *understand* and *actively participate*. Literature shows that this is a critical demand of process modeling methodologies today [3][6][9][11]. It is then a framework to be shared by both business and IT communities inside the organization.

Our methodology consists of a series of derivation steps by which the events and activities are modeled on the basis of a set of business rules provided. We argue that the active participation of business people throughout the whole development process may significantly reduce the semantic gap between the business rules and the derived process implementation. As an additional contribution of this paper, we presents an ontology to describe event-driven systems.

1.1 Structure of the Paper

This paper is structured as follows: The background is presented in Section 2. Section 3 discusses related works in the area. Section 4 presents an ontology for event-driven systems. Section 5 presents the phases of the refinement process that support the methodology proposed in this paper. An example is shown in Section 6

in order to illustrate the application of the methodology. Finally, Section 7 discusses the conclusions of the paper.

2 Background

Business rules technology has provided a way for modern organizations to obtain high flexibility, enabling them to rapidly adapt their business processes to changes in environment and business needs. The main idea is to remove the business decisions from the business process level.

Event-driven technologies have been acknowledge to be an approach that provides great flexibility for implementing the coordination of activities [13][4][5]. Event-Condition-Action (ECA) frameworks, for example, enable the design of business rules by means of IF/THEN rules that are triggered by events and, in turn, generate new events. Complex Event Processing (CEP) [7] provide languages and algorithms for recognizing complex patterns of events from large, distributed sets of events at real-time.

Although ECA and CEP are rich technologies for the implementation of business rules, the integration between business rules on the business level and the corresponding event-driven rules is still ad-hoc and made by enterprises on the basis of proprietary technologies [5][9]. Furthermore, the integration between business process engines and business rules engines present limitations on integrating both abstractions [13].

3 Related Works

Although many works have investigated the integration of business rules approaches to process modeling activities, there is still a lack of a definitive solution that encompass all requirements for flexibility and integration in the development of enterprise systems.

The issue on this context is to bridge the gap between the rules as understood by business executives and the actual process implementation.

Therani [9] proposes an ontology for designing flexible business processes. The work proposes a two-layer framework for the description of business processes that aims at bridging the communication between domain abstractions and software abstractions. The first layer corresponds to domain semantics and is mapped from the real world, from the point of view of the user. The second layer corresponds to the technology-specif abstractions, from a developer's point of view. The authors argue that managing the relationship between these two layers in a consistent manner is the key to developing reliable process-management systems. Nevertheless, no methodology for performing such management is presented. The mapping from tasks, states, and agents to software objects is still ad-hoc. Also, no methodology for defining the tasks, states, and agents from a business analysis is provided.

Knolmayer et al. [3] propose an approach for modeling workflow based on business rules. These rules are represented by Events, Conditions and Actions (ECA). The work proposes that textual business rules should be formatted in the form of Event, Condition and Action descriptions and that they should be incrementally detailed in order to achieve an executable process specification. This work do not present any methodology for modeling the ECA rules. Also, the refinement to low-level implementation is ad-hoc.

Kovacic et al. [6] discuss *business renovation*, which is the effort for redesigning business processes and information systems on the basis of a critical examination of current business policies and practices. They state that business rules should be described in *natural language* and business processes should be modeled *only* at the level of detail that is sufficient to achieve the rules' objectives. They also propose that the textual rules should be incrementally detailed into lower-level abstractions. Yet, they do not provide any framework for methodologically deriving software models from business rules descriptions. They argue that, in small cases, the manual revision is more economic than the use of current tools.

Several other works propose the use of event-driven rules to implement flexible business processes [7][13][4]. However, none of these works provide means for integrating the *business-side rules* to the software abstractions provided by these event-driven frameworks.

In this paper, we aim to overcome the limitations found on these related works by defining an analysis technique for integrating the concepts of business rules as seen from the business side to the concepts of complex event processing, and show a refinement process for moving from the business rules to software implementation.

4 Ontology of Events

This section defines an introductory ontology regarding event-driven systems concepts. The need for an ontological framework for the integration of enterprise systems development is discussed by Therani [9].

The fundamental concepts regarding event-driven systems are defined below.

event any change in the enterprise state that has a relevant business meaning for the business operations or for management;

activity any action that significantly changes the state of the enterprise (generating events, in turn);

constraint a restriction on the states that can be assumed by the system;

rule a statement that govern the occurrence of events and relate them to the execution of activities;

agent the application/person responsible for executing activities when events are recognized.

An enterprise's events, activities, and rules build up the enterprise's event model. Two features must be provided by an event model. It must be *complete* and it must be *effective*. These concepts are defined below.

Definition 1 (Event Model). A set of events, activities, agents and relationships between them (rules and constraints) that interact for building up the behavior of a dynamical system.

Definition 2 (Complete Event Model). An event model is complete if, for a set of events and activities defined, the following affirmations are true:

1. there is no activity that generates events outside of the model, and there is no event related to the execution of activities outside of the model;
2. all structural constraints that affect these events and activities in the real world are present in the model.

Definition 3 (Effective Event Model). An event model is effective if it correctly reproduce the real world behavior, regarding the events that are present in the model.

An event model is a representation of an enterprise in terms of events and activities (Def. 1). The criteria for its *completeness* and *effectiveness*, described by Def. 2 and Def. 3, respectively, assure the correctness of the model.

We can consider three levels of abstraction at which the event model is realized:

- *Business level*: events and activities are informally present on business rules, business documents and on the executives' vocabulary;
- *Abstract level*: events and activities are identified and well-documented, but not mapped to software code;
- *Software level*: the event model is implemented into software objects.

Our approach is composed of a sequence of phases into which the event model is refined from the highest, business level, to the software level. These phases are described in Section 5. The *business level* is the field of discussion of the business community. The *software level* is the field of discussion of the IT community. It is the *abstract level* that enables the communication between them. Both business and IT staff are involved in elaborating the event model at the abstract level.

5 Systematic Refinement of Event Models

In this section we present our refinement method for implementing event-driven business processes. We use as examples a fictious *Pet Shop* company in which clipping and bath services are provided. It give us simple and intuitive rules, yet it is illustrative enough for our objective.

5.1 Analysis Phase

- **Input:** *Natural language business rules*.
- **Output:** *Business Rules Analysis (BRA)*.
- **Level:** *Business*.

At this phase, a document called *Business Rules Analysis (BRA)* must be constructed. The BRA contains all relevant information extracted from the rules at the business level.

The construction of the BRA is made by answering the following questions for each individual business rule:

1. Which **product or service** of the enterprise the rule affects or contributes to?
2. Which organizational **roles** are affected by the rule?
3. Which **resources** are mentioned or used to evaluate the rule?
4. What **conditions** must be verified to evaluate the rule?
5. What **events** are mentioned by the rule (directly or indirectly)?
6. What events are produced by trigerring the rule?
7. Which **activities** are required by the rule or are necessary to evaluate the rule?

To reduce bias on the answers, these questions must not rely on existing implementations of the system.

An example of rule is given b elow:

> **Rule** - CONTACT_CUSTOMER: *The customer must be notified if any of the following affirmations is true: 1) A clipping requested by the customer has finished; 2) A bath requested by the customer has finished.*

Answers should be objectively filled in a *Rule Description Form (RDF)*. The BRA is composed of a set of RDFs. Table 1 displays an example of RDF, filled with information for the Rule CONTACT_CUSTOMER.

Table 1 Example of Rule Description Form (RDF)

Rule	CONTACT_CUSTOMER
Product/Service	Clipping and Bath
Roles affected	Receptionist
Resources	Telephone
Pre-conditions	Clipping or bath concluded
Events observed	Clipping or bath conclusion
Events produced	Customer notification request
Activities	Customer notification

5.2 Event Modeling Phase

- **Input:** *Business Rules Analysis(BRA)*.
- **Output:** *Event Definitions (ED)*.
- **Level:** *Abstract*.

Notice that events are mentioned in the BRA in an informal manner. The purpose of this phase is to identify and register all events found during the previous phase. Each event found receives a unique name and is registered in a document called *Event Definitions (ED)*. The *ED* is a glossary of event names. As such, it may include a brief description of the event, synonymous names and, when necessary, examples of situations when it occurs. This phase is where we begin to move from the business level to the abstract level of the event model.

During this phase, we classify events according to their causal relations regarding the phenomena that created them in the real world. Two situations are recognized:

Event After Action (EAA) the event indicates the occurrence of a past fact;
Event Before Action (EBA) the event indicates that something is about to occur.

This classification is important because the implementation of each type of event is different. EAA events are not controllable, since they indicate events that already occurred in the real world. EBA events, on the other hand, are controllable. Therefore, one can define rules to impose restrictions on its execution.

Once event-driven frameworks are naturally reactive, in the sense that they can only process an event after its occurrence, it is difficult to define rules for prohibiting the occurrence of an event. Therefore, how can a prohibition rule be effectively triggered?

The solution we provide for handling such rules is as follows: Any EBA event *Ev* must be issued in two steps. Firstly, an advice event (*EvAttempt*) is generated. Rules that could prohibit *Ev* are triggered on the occurrence of this advice event. If a condition that prohibits the event is found, a denying event (*EvDenied*) is generated. Otherwise, *Ev* is issued normally. Only one entity in the system is responsible for doing such verification and only it is allowed to issue the *Ev* event.

5.3 Pattern Definition Phase

- **Input:** *Business Rules Analysis (BRA), Event Definition (ED).*
- **Output:** *Pattern Definition (PD).*
- **Level:** *Abstract/Software*.

This phase has the purpose of identifying the conditions that should be observed in order to trigger a given rule. These conditions are recognized by the definition of event patterns. Patterns are defined by data, environment, and time constraints that recognize the states when the conditions hold. In this phase, IT people and business people must be involved in order to formally define the rules in a rules language.

Patterns identified are registered in a document called Pattern Definition (PD). Each pattern is associated with a rule in the BRA. The PD also classify the rules according to three generic classes:

Reactive Rule defines that some events must be triggered when a give condition is satisfied;
Prohibition Rule defines that certain events can not happen if a give condition is satisfied;

Communication Rule specifies that someone must be notified when a given condition holds.

The objective of this classification is to assure better comprehension on how the rules should be modeled during the implementation phase. An example of pattern can be found in the illustrating example given in Section 6.

5.4 Activity Modeling Phase

- **Input:** *Business Rules Analysis (BRA), Events Definition (ED).*
- **Output:** *Activities Definition (AD), Events Definition (ED) (updated).*
- **Level:** *Abstract.*

Business rules are the main source for discovering what activities are executed by the enterprise. Rules either require the execution of activities or affect how these activities are executed. There are two aspects concerning what the rule requests to be done:

1. *implicit activities* - some action must be executed to verify the applicability of the rule;
2. *explicit activities* - the rule may explicitly require some action to be taken.

For example, a rule that states "*If a package arrives with defect, it can not be accepted.*" does not explicitly require any action. However, to discover if the package has any defect, an activity *Verify Package* must be executed. On the other hand, a rule that states "*Any package arriving must be verified before being accepted.*" explicitly defines the action to be taken before accepting the package.

The activities are registered in a document named *Activity Definitions (AD)*, which contains unique names for each activity and their descriptions. It also describes which events are related to the activity, which roles are responsible for executing it, and what resources are necessary.

Also, for each activity, we include events to indicate the activity start (*Do Something Started*), finalization (*Do Something Finished*), cancellation (*Do Something Canceled*), and an event that requests its execution (*Do Something Request*). Notice that the *ED* must be updated with these new events.

5.5 Implementation Phase

This section describes an architecture by which the event model can be implemented. The fundamental components are presented. Naturally, several supporting technologies must be employed, such as databases, network and graphical interfaces. We abstract the necessity of these technologies and focus on the specific concerns of the event-driven part of the system.

The objective is to allow the flexibility and modularity in the implementation of event models, improving the ability to add and exclude rules during the system's lifetime without compromising its execution.

The architecture has six components: Data Model, Event Objects, CEP Engine, Agents, Worklist Handler, and Rules Manager.

Figure 1 displays a graphical view of the architecture elements and their relationship.

5.5.1 Data Model

This component contains all classes that represent business entities. It corresponds, for example, to the model layer in a Model-View-Controller (MVC) architecture.

In the Pet Shop event model, the business model contains classes such as *Customer*, *Pet* and *Order*.

5.5.2 Event Objects

The ED document provides the definition of all events that must be present in the event model. Every event described in this document is modeled by class that contains data about its occurrence. For example, the "*Cage Reserved*" event is modeled by a class `CageReserved` which contains fields such as date/time reserved, pet, order etc.

5.5.3 CEP Engine

The responsibility of the CEP Engine is to process the events generated by the application. Traditional engine implementations can handle millions of events, recognize complex patterns of events and trigger actions from the application when necessary. Each implementation of CEP uses different approaches for achieving this and different languages for expressing the event patterns.

We chose EsperTech's Esper framework [2] for implementing the Pet Shop event model. However, any framework that has the necessary features could be used as well. The main requirement is that the language provided by the engine is powerful enough to express the conditions necessary for recognizing the business rules that will be implemented.

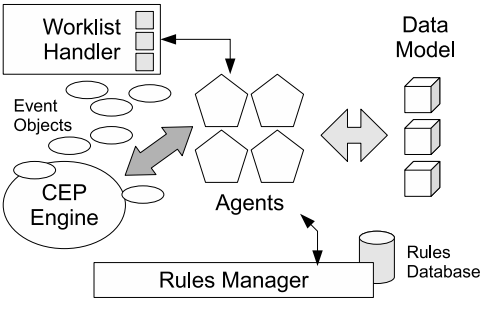


Fig. 1 Graphic view of the architecture

5.5.4 Agents

Agents are objects responsible for taking actions when patterns are recognized. Each agent detects a set of patterns that have common characteristics. Different agents run in parallel, processing different sets of events, therefore improving the performance and scalability of CEP [7].

The structure of an agent is composed by the set of patterns that it recognizes and the code that it will execute for each pattern. These patterns are retrieved from the Patterns Definition (PD) document. Possible actions taken by the agents can be the execution of automated business operations, database updates, issueing of new events or interaction with other applications. This behavior must be codified by the IT staff according to the definition of the rules.

5.5.5 Worklist Handler

Worklist is a list of tasks that are assigned to an employee. The goal of the worklist handler is to manage the interaction between the system and the employees, by submitting tasks to the employee's worklist and getting the notification of completion when the task is concluded. An agent makes the bridge between the CEP engine and the worklist handler by converting action request events to tasks in the worklist and notifications of completion to conclusion events.

5.5.6 Rules Manager

As the system evolves, it is necessary to manage its execution, change, add, or remove rules. The rules manager provides the funcionality for managing the business rules stored in the system. It enables the communication between the user, the rules database and the CEP engine.

6 Implementing the Pet Shop's Event Model

In this section we present the implementation of the Pet Shop's event model.

As we explain in Section 5.5, we chose EsperTech's Esper framework [2] to implement the CEP Engine. This is a well documented framework that runs under Java and has support for integration with most technologies available for enterprise information systems, such as Web Services, XML, XPath and Java Messaging Service (JMS).

In order to explain how the Pet Shop is implemented, we detail the implementation of two rules. For each rule, we present the events, the agents, how the pattern is described and what the agent will process. These aspects are the essential characteristics to describe the rule implementation. The implementation of all rules follow the same principles.

Consider Rules 1 and 2, described below.

Rule 1 - PET_ENTER_PROHIBITION: *The customer can not order a new treatment if any of the following affirmations is true: 1) There is a pending payment for that customer; 2) There is no free cage at the moment for the customer's animal.*

Rule 2 - FEED_PET: *The pet must be fed if any of the following affirmations is true: 1)The pet has been in the cage for six hours; 2) The pet was fed six hours ago and it is still in the cage.*

In the first rule, we want that, when the receptionist attempts to register the entering of the animal, the conditions of the rule are verified by the system. The system must block the registration of the animal and notify the receptionist when necessary. In the second rule, the objective is to alarm when an animal in the cage needs to be fed.

6.1 Phase 1: Analysis

The first step for implementing theses rules is filling the *Rule Description Form* (RDF). Tables 2 and 3 display these forms.

6.2 Phase 2: Event Definition

The next step is the events description. Observing the RDF of each rule, we can define the following events:

Table 2 RDF for Rule 1

Rule	PET_ENTER_PROHIBITION
Product/Service	Clipping and Bath
Roles affected	Receptionist
Resources	none
Pre-conditions	Customer in the store
Events observed	Before entering a pet
Events produced	Block the pet entering, admit the pet
Activities	Check payments, Check cages

Table 3 RDF for Rule 2

Rule	FEED_PET
Product/Service	Clipping and Bath
Roles affected	Cage maintainer
Resources	food
Pre-conditions	Pet in the store
Events observed	Put the pet into the cage, take the pet out of the cage, Pet was fed
Events produced	Pet Feeding Demanded, Pet Fed, Payment Pending
Activities	Feed Pet

- *PetEnterAttempt*: when the receptionist attempts to register the animal entrance;
- *PetEnterDenied*: when the pet entrance is blocked;
- *PetEnter*: when the pet entrance is registered;
- *PetCaged*: when the cage maintainer put the pet into the cage;
- *PetOutofCage*: when the cage maintainer take the pet out of the cage;
- *PetFeedingDemanded*: when the system notifies that the pet needs to be fed;
- *PetFed*: when the pet is fed;
- *PaymentPending*: when any bill is added to the Customer account (e.g. after feeding the animal, the customer is charged for the food).

6.3 Phase 3: Patterns Definition

Once the events are defined, it is necessary to define the patterns that identify each rule. In the case of Rule 1, whenever the receptionist attempts to register the pet, the conditions must be verified. Therefore, the pattern monitored is every occurrence of the *PetEnterAttempt* event.

In Esper's pattern language (EPL), this pattern is expressed by:
"`select * from PetEnterAttempt;`"

For Rule 2, we have two patterns to be recognized. The first pattern corresponds to six hours after the moment the pet was caged in the case it has not been removed from the cage in the mean time. This is the first condition mentioned by the rule. In EPL, it is expressed by:

```
    select * from pattern
[every (a = PetCaged -> (timer:interval(6 hour) and not
PetOutOfCage(order.orderNumber = a.order.orderNumber)))];
```

Notice that, in order to assure that the *PetCaged* and *PetOutOfCage* events correspond to the same order, we check whether the order numbers are the same.

The second pattern corresponds to the situation where the pet is still in the cage six hours after the last time it was fed. This expression in EPL is only slighly different from the previous one:

```
    select * from pattern
[every (a = PetFed -> (timer:interval(6 hour) and not
PetOutOfCage(order.orderNumber = a.order.orderNumber)))];
```

6.4 Phase 4: Activity Modeling

After the patterns are modeled, we need to define what actions must be taken when each pattern is recognized. We can observe that the first rule is a *prohibition rule* and the second is a *communication rule*. The following activities can be defined from the rules description:

- *Check Payments* - this is an activity that can easily be implemented in the system to be performed automatically;
- *Check Free Cage* - this is also an automatic activity chosen to be implemented in the system;
- *Feed Pet* - this is an ordinary manual activity that issues events externally from the system. However, this activity can be considered atomic, since we do not need to follow each step of its execution. As such, it does not need to have the started and finished events, but just an event indicating that it was done. We chose the *PetFed* event to indicate that.

The first two activities are implemented as methods that are called when the patterns are recognized. On the case of the third activity, once it is manual, it is required that an event requesting its execution is issued. The *PetFeedingDemanded* event is used with that purpose.

6.5 Phase 5: Implementation

Firstly, the event objects must be modeled through the definition of *classes*. It is necessary to define what data each event will store. On the case of the Pet Shop, we observed that for most events, only the order identification is enough for processing them. Therefore, all these events have a similar structure as exemplified below:

```
class PetEnterAttempt {
 Order order;
}
```

The Order is a class of the Data Model that stores all information about the Customer order, as shown below.

```
public class Order {
  private int orderNumber;
  private Customer customer;
  private ServiceType service;
  private double bill =0.0;
  ...
}
```

Next, several agents are implemented. A good criteria for defining agents is to group rules that correspond to the same organizational role and create an agent for each role. So, the Pet Shop model has the following agent objects: *ReceptionistAgent, ClippingAgent, PaymentAgent, CageAgent, CustomerRelationshipAgent*, amongst others.

Considering the Rules 1 and 2 above, the *ReceptionistAgent* is responsible for, at every time the *PetEnterAttempt* occurs, verifying if the customer account is free from debit and that there are free cages for accepting the animal. The event data will bring all information necessary for doing this verification, such as customer name, customer identification and animal race and size.

The *CageAgent* listens to the two patterns defined previously. Every time one of the patterns is recognized by the Esper engine, a method will generate the *PetFeedingDemanded* event.

We implemented simple simulation features in order to test the correctness of the Pet Shop implementation. The simulator generates customer arrivals with exponential or normally distributed inter-arrival times and the order data is retrieved from a collection of typical orders stored in a database.

In all simulations the system presented the output expected regarding the rules that were used as input, which includes rules for billing and payment processing, customer relationship, and treatment coordination.

7 Conclusions

Despite the advances on business rules theory and the increasing number of large enterprises doing efforts to model their business rules, there is still a lack for a meaningful integration between business analysis and process modeling activities. Much of this deficiency is due to the low support of traditional workflow modeling techniques [8].

Aiming at overcoming these difficulties, we proposed a new approach for the implementation of *event-driven* business processes by employing a stepwise refinement from the business level to the software level. We show how a set of business rules described in natural language can be used as the specification for the development of an event-driven management system in a way that business people can actively participate.

This methodology provides the following contributions:

- allows for the traceability between business needs and software artifacts;
- keeps business people and stakeholders involved along the modeling process;
- provides a number of documents that help in the system specification, maintaining a vision towards *events* since the analysis phase;
- allows for the coordination of activities based on rules that can be easily modified according to business needs, instead of relying in hard structures of workflow designs.

The possibility of changing the behavior of the system simply by changing business rules is essential for achieving the desired flexibility and business alignment. By proposing the use of *event models*, we allow for the implementation of *rule-directed* business processes (as oposed to *workflows*) using technologies that already showed their value and effectiveness for implementing real world applications [7][1].

We validated our approach by implementing the scenario of a fictitious company. The experiments showed that the event model is an effictive approach for implementing business processes, providing the desired flexibility while maintaining robustness and improving manageability.

The approach is also currently in use for the development of a management system for a real company. Several rules have been successfully implemented. A prototype of this system is available at `http://www.cin.ufpe.br/~ncs/xbdngco`.

As future works, we intend to implement tools for automating the generation of the event model by following a model-driven approach and for the simulation and performance evaluation of the system.

Acknowledgements

The authors would like to thank CNPq for supporting this research.

References

1. Debevoise, T.: Business Process Management with a Business Rules Approach: Implementing The Service Oriented Architecture. BookSurge Publishing (2007)
2. EsperTech. Esper - complex event processing (March 2010)
3. Endl, R., Knolmayer, G., Pfaher, M.: Modeling processes and workflows by business rules. Business Process Management: Model, Techniques and Empirical Studies, 16–29 (2000)
4. Gong, Y., Janssen, M., Overbeek, S., Zuurmond, A.: Enabling flexible processes by eca orchestration architecture. In: ICEGOV 2009: Proceedings of the 3rd International Conference on Theory and Practice of Electronic Governance, pp. 19–26. ACM, New York (2009)
5. Graml, T., Bracht, R., Spies, M.: Patterns of Business Rules to Enable Agile Business Processes. In: 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007), pp. 365–365 (October 2007)

6. Kovacic, A., Groznik, A.: Business renovation: From business process modelling to information systems modelling. In: Proceedings of the 24th International Conference on Information Technology Interfaces, pp. 405–409 (2002)
7. Luckham, D.: The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems. Addison-Wesley Professional, Reading (2002)
8. Indulska, M., zur Muehlen, M., Kamp, G.: Business process and business rules modeling: A representational analysis. In: Proceedings of the Eleventh International IEEE EDOC Conference Workshop, pp. 189–196 (2007)
9. Madhusudan, T.: Ontology development for designing and managing dynamic business process networks. IEEE Trans. Industrial Informatics 3(2), 173–185 (2007)
10. Nurcan, S.: A survey on the flexibility requirements related to business processes and modeling artifacts. In: HICSS 2008: Proceedings of the 41st Annual Hawaii International Conference on System Sciences, p. 378. IEEE Computer Society, Washington, DC, USA (2008)
11. Object Management Group. Business Motivation Model (BMM), v1.0. Technical report, Object Management Group (2008)
12. Shao, J., Pound, C.J.: Extracting business rules from information systems. BT Technology Journal 17(4), 179–186 (1999)
13. van Eijndhoven, T., Iacob, M.E., Ponisio, M.L.: Achieving business process flexibility with business rules. In: Proceedings of the 12th International IEEE Enterprise Distributed Object Computing Conference, EDOC 2008, pp. 95–104. IEEE Computer Society Press, Los Alamitos (2008)
14. zur Muehlen, M.: Workflow-Based Process Controlling: Foundation, Design, and Application of Workflow-driven Process Information Systems. Logos Verlag, Berlin (2002)

3D NAT Scheme for Realizing Seamless End-to-End Connectivity and Addressing Multilevel Nested Network Address Translation Issues

Hartinder Singh Johal, Balraj Singh, Amandeep Nagpal, and Kewal Krishan

Abstract. Network address translation model (NAT) has been viewed as clear transgression of end-to-end connectivity principle of internet. Since local area network (LAN) based internal addresses are all disguised behind one publicly accessible NATenabled router, it is virtually impossible for public hosts to initiate a connection to a specific private host. This in turn adversely affects the NAT enabled network's ability to support VOIP, Video conferencing and other peer-to-peer applications. The situation is acutely compounded when we have multiple levels and nesting of Static and Dynamic NATs. The situation is tempting when we hope to retain the benefits of NAT but may still aspire to achieve the principle of end-to-end connectivity. This manuscript tends to realize the above situation by proposing a three dimensional network address translation scheme with a tentative capability to support end-to-end connectivity based applications and at the same time retaining the benefits of conventional NAT model.

Keywords: NAT, NAT Exemption, Static NAT, Dynamic NAT, Multi-level Nested address translation.

1 Introduction

Conventional NAT scheme offers a wide array of benefits major being address space preservation, security and resolution of overlapping addresses [1]. Since the IP translation life is limited by the span of connection, a specific user may not keep the same IP address after the translation dies out [2]. Hosts on the destination network therefore are unable to consistently initiate a connection to a host inside private network that uses Dynamic translation and this is true even when we have

Hartinder Singh Johal · Balraj Singh · Amandeep Nagpal · Kewal Krishan
Department of Computer Science and Engineering, Lovely Professional University 144402 Phagwara, India
e-mail: {hs.johal,balraj.13075,amandeep.nagpal, kewal.krishan}@lpu.co.in

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 127–141.
springerlink.com

configured access control lists (ACL) for incoming traffic from same destination host [3]. We can address it to some extent by using Static translation but there are inherent disadvantages of this option [4]. NAT exemption can address the situation as it allows both local and external IPs to initiate connections [5].

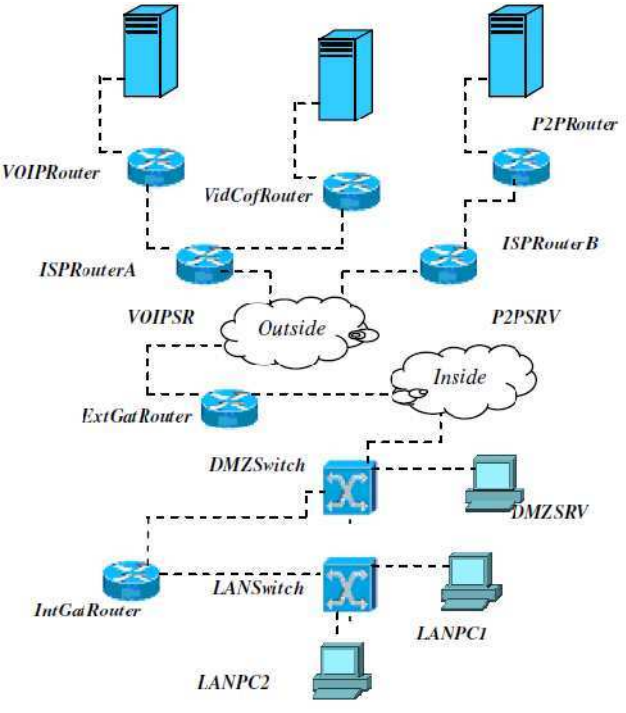


Fig. 1 Simulated Network used for NAT analysis.

But we cannot dynamically oscillate between Static, Dynamic and exempted NAT scenarios [6] [7]. The situation further deteriorates when we have multiple levels of nested Static and Dynamic NATs configured on precincts of private and public network [8]. The need is to incorporate the third dimension of NAT exemption along with ability to seamlessly manage the issue of multilevel and nested Static and Dynamic NATs [9]. We propose 3D NAT scheme, where we pursue efforts to incorporate third dimension of NAT exemption together with addressing the issues of multilevel nested Static and Dynamic NATs. Initially, we perform analysis of multilevel nested Static and Dynamic NATs by taking each translation scenario mutually exclusively and then concurrently. We quantify the computational complexity required to override pre-configured translations. As a consequence, it is observed that computational complexity rises exponentially in correlation to multilevel and nesting factor for translation. We then subsequently propose a 3D NAT scheme and demonstrate its ability to tackle issue of multilevel nested translations while incorporating NAT exemption.

A simulated network (fig 1) is used for analysing the effect of nested Static and Dynamic translations [10]. Parameters are assumed to quantify the computational complexity involved during configuration and management of nested Static and Dynamic network address translations [11]. We then make an effort to quantify the effort of improvement we anticipate to accomplish by this new scheme.

2 Implementing Nested Static and Dynamic Address Translations

For configuring and implementing multiple levels and nesting of Static and Dynamic Address Translations, we use network in fig 1. Participating devices in network are configured as per interface configuration and IP address allocation matrix shown in Table 1.

Table 1 Interface and IP Matrix

Device (Interface)	IP Address	Device (Interface)	IP Address
IntGatRouter's Interface/IP		P2PRouter's Interface/IP	
DMZSwitch (0/0)	192.168.50.1	ISPRouterA(0/0)	210.10.10.2
LANSwitch (1/0)	192.168.0.1	P2PSRV(1/0)	230.10.10.1
VidCofRouter's Interface/IP		VOIPRouter's Interface/IP	
ISPRouterA(0/0)	220.10.10.2	ISPRouterA(0/0)	210.10.10.2
VCSRV(1/0)	235.10.10.1	VOIPSR(1/0)	230.10.10.1
ISPRouterB's Interface/IP		ISPRouterA's Interface/IP	
ExtGatRouter(0/0)	240.10.10.2	ExtGatRouter(0/0)	200.10.10.2
P2PRouter(1/0)	245.10.10.1	VOIPRouter(1/0)	210.10.10.1
ExtGatRouter's Interface/IP		VidCofRouter(2/0)	220.10.10.1
DMZSwitch (0/0)	192.168.50.2	DMZSwitch's Interaface	
ISPRouterA(1/0)	200.10.10.1	IntGatRouter(0/0)	192.168.50.1
ISPRouterB(2/0)	240.10.10.1	ExtGatRouter (0/0)	192.168.50.2

In the fig 1, IntGatRouter and ExtGatRouter serve as internal and external gateways for private network. Demilitarized zone (DMZ) boundaries are created by placing DMZSwitch between the internal and external gateway router. It will protect the private network from external attacks and also hosts on the internal network from servers in the DMZ in the event that DMZSRV is compromised. There are two ISP Routers (ISPRouterA & ISPRouterB) connected to external gateway router. Nested Static and Dynamic address translations will be implemented at Internal and external router gateways. The ultimate objective is to allow Voice over IP server (VOIPSR), Video Conferencing Server (VCSRV) and Peer-to-Peer Applications (P2PSRV) server to initiate connections for hosts (LANPC1 & LANPC2) inside the private network in event when we have multiple levels and nesting of address translations. RIP is used to create a seamless network access for all external network devices extending up to the external interfaces

(fa1/0 for ISPRouterA & fa2/0 for ISPRouterB) of ExtGatRouter. Routing table contents of VOIPRouter are displayed to ensure that devices are behaving as anticipated for this case.

VOIPRouter#show arp
Protocol Address Age (min) Hardware Addr Type Interface
Internet 210.10.10.2 - 000C.2187.9744 ARPA FastEthernet0/0
Internet 230.10.10.1 - 000C.6516.6802 ARPA FastEthernet1/0
Internet 210.10.10.1 0 000C.8172.9045 ARPA FastEthernet0/0
Internet 240.10.10.1 2 000C.8172.9045 ARPA FastEthernet0/0
Internet 230.10.10.2 4 000C.5016.2460 ARPA FastEthernet1/0
Internet 200.10.10.1 7 000C.8172.9045 ARPA FastEthernet0/0
Internet 220.10.10.1 7 000C.8172.9045 ARPA FastEthernet0/0
Internet 235.10.10.1 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 245.10.10.1 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 245.10.10.2 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 250.10.10.2 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 250.10.10.1 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 235.10.10.2 8 000C.8172.9045 ARPA FastEthernet0/0
Internet 220.10.10.2 9 000C.8172.9045 ARPA FastEthernet0/0

We are able to trace the packet route from VOIPSR till external interface of ExtGatRouter
C:#trace 240.10.10.1
"Type escape sequence to abort."

Tracing the route to 240.10.10.1
1 230.10.10.1 0 msec 16 msec 0 msec
2 210.10.10.1 20 msec 16 msec 16 msec
3 200.10.10.1 20 msec 16 msec *

2.1 Multi-level Static Address Translation

Configuring first level of Static Network Address Translation at IntGatRouter for LANPC2 (192.168.0.3)

IntGatRouter(config)#int fa1/0
IntGatRouter(config-if)#ip nat inside
IntGatRouter(config-if)#ip nat outside
IntGatRouter(config-if)#exit
IntGatRouter(config)#ip nat inside source Static 192.168.0.3 192.168.100.3
IntGatRouter(config)#end
IntGatRouter#show ip nat translations

Pro Inside global Inside local Outside local Outside global
--- 192.168.100.3 192.168.0.3 --- ---
IntGatRouter#show ip nat translations
Pro Inside global Inside local Outside local Outside global

icmp:9392 192.168.50.1:9392 192.168.50.2:9392 192.168.50.2:9392
icmp:9393 192.168.50.1:9393 192.168.50.2:9393 192.168.50.2:9393
icmp:9394 192.168.50.1:9394 192.168.50.2:9394 192.168.50.2:9394
icmp:9395 192.168.50.1:9395 192.168.50.2:9395 192.168.50.2:9395

icmp:9396 192.168.50.1:9396 192.168.50.2:9396 192.168.50.2:9396
icmp:9392 :9392 192.168.50.1:9392 192.168.50.1:9392
icmp:9393 :9393 192.168.50.1:9393 192.168.50.1:9393
icmp:9394 :9394 192.168.50.1:9394 192.168.50.1:9394
icmp:9395 :9395 192.168.50.1:9395 192.168.50.1:9395
icmp:9396 :9396 192.168.50.1:9396 192.168.50.1:9396
icmp:9392 192.168.0.1:9392 192.168.50.2:9392 192.168.50.2:9392
icmp:9393 192.168.0.1:9393 192.168.50.2:9393 192.168.50.2:9393
icmp:9394 192.168.0.1:9394 192.168.50.2:9394 192.168.50.2:9394
icmp:9395 192.168.0.1:9395 192.168.50.2:9395 192.168.50.2:9395
icmp:9396 192.168.0.1:9396 192.168.50.2:9396 192.168.50.2:9396
icmp:9392 :9392 192.168.0.1:9392 192.168.0.1:9392
icmp:9393 :9393 192.168.0.1:9393 192.168.0.1:9393
icmp:9394 :9394 192.168.0.1:9394 192.168.0.1:9394
icmp:9395 :9395 192.168.0.1:9395 192.168.0.1:9395
icmp:9396 :9396 192.168.0.1:9396 192.168.0.1:9396

After mounting first level of Static translation at IntGatRouter we subsequently configure ExtGatRouter for second level of Static translation, for translated IP (192.168.100.3) to LANPC2 (192.168.0.3).

```
ExtGatRouter(config)#int fa0/0
ExtGatRouter(config-if)#ip nat inside
ExtGatRouter(config-if)#end
……
ExtGatRouter(config)#int fa2/0
ExtGatRouter(config-if)#ip nat outside
ExtGatRouter(config-if)#exit
ExtGatRouter(config)#ip nat inside source Static 192.168.100.3 251.100.100.3
   ExtGatRouter(config)#end
ExtGatRouter#show ip nat translations
Pro Inside global Inside local Outside local Outside global
--- 251.100.100.3 192.168.100.3 --- ---
```

IP of LANPC2 (192.168.0.3) is translated to 192.168.100.3 at IntGatRouter for fa0/0 interface and at ExtGatRouter it is translated to 251.100.100.3 for fa2/0 and to 251.100.100.4 for fa1/0 interfaces, as in fig 2. For allowing P2PSRV to initiate connection for LANPC2 we need to configure ACL for allowing incoming traffic from P2PSRV.P2PSRV have no clue of LANPC2's original IP for initiating connection as there are multiple levels of Static address translations. The obvious way-out of this problem is to use NAT exemption. For using NAT exemption we need to manually override Static NAT configurations at IntGatRouter and ExtGatRouter respectively. This effort of removing Static NAT configurations can be represented as $\delta(s)$.

$$\delta(s) = \sum_{i=0}^{n} \eta^i (\eta + i)^i i)^i$$

where η is computation required to remove i^{th} Static address translation from each router n and $\eta + i$ is the associated complexity required to reconfigure ACL, configured for specific incoming traffic scenario. Graphically this effort can be represented in fig 3.

Best-Fit Plane for Fig3: z = A*x + B*y + C,
Coefficients A = 1.14741, B = -0.0647594, C = 0.222989, Correlation coefficient is 0.868634
Peak Values: Minimum Z = 0 at X = 0.1, Y = 0, Maximum Z = 2 at X = 1, Y = 1,
Max - Min Z = 2, Mean = 0.764317, Standard deviation = 0.4201231

2.2 *Nested Static and Dynamic Address Translation*

Concurrent to previous case we configure Static address translation at IntGatRouter and pursue Dynamic address translation for this statically translated address at ExtGatRouter as in fig 4

IntGatRouter(config)#int fa1/0
IntGatRouter(config-if)#ip nat inside
IntGatRouter(config-if)#ip nat outside
IntGatRouter(config-if)#exit
IntGatRouter(config)#ip nat inside source Static 192.168.0.3 192.168.100.3
IntGatRouter(config)#end
IntGatRouter#show ip nat translations
Pro Inside global Inside local Outside local Outside global
--- 192.168.100.3 192.168.0.3 --- ---
……
ExtGatRouter(config)#ip nat pool johalpool3 249.100.100.1 249.100.100.50 netmask 255.255.255.0
ExtGatRouter(config)#ip nat pool johalpool2 251.100.100.1 251.100.100.50 netmask 255.255.255.0
ExtGatRouter(config)#ip nat inside source list johallist2 pool johalpool2
ExtGatRouter(config)#access-list johallist2 permit 192.168.100.0 0.0.0.255
ExtGatRouter(config)#end

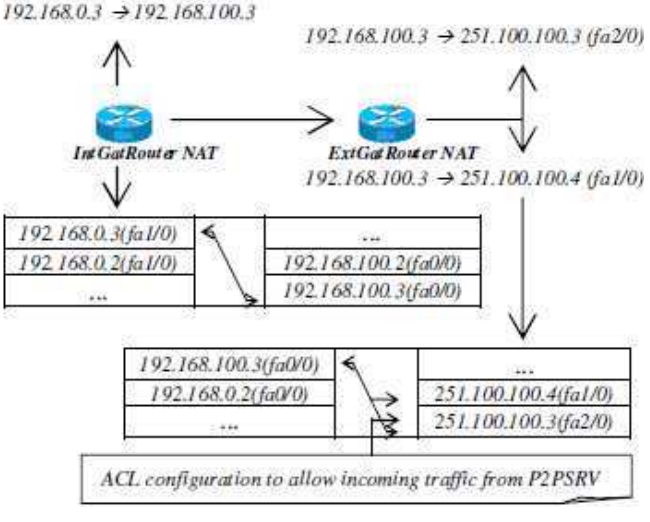


Fig. 2 Multi-level Static NAT.

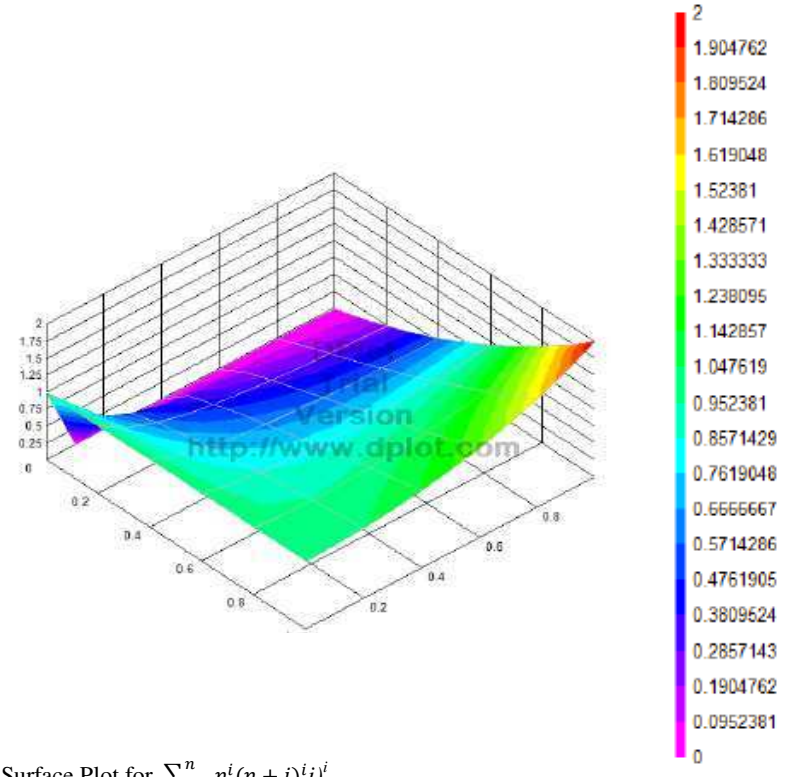


Fig. 3 Surface Plot for $\sum_{i=0}^{n} \eta^{i}(\eta + i)^{i} i)^{i}$

P2PSRV have no clue of LANPC2's original IP for initiating connection as there is nesting of Static and Dynamic address translations at IntGatRouter and ExtGatRouter. The obvious way-out of this problem is to use NAT exemption. For using NAT exemption we need to manually override Static NAT configurations at IntGatRouter and Dynamic NAT configuration at ExtGatRouter respectively. This effort of removing nesting of Static and Dynamic NAT configurations can be represented as δ *(sd).*

$$\delta(sd) = \sum_{i=0}^{n} \eta^{i+1}(\eta + i)^{i+1} i^{+1} i$$

where η is computation required to remove i^{th} Static address or Dynamic translation from each router n and $\eta + i$ is the associated complexity required to reconfigure ACL, configured for specific incoming traffic scenario. Graphically this effort can be represented as in fig 5.

Best-Fit Plane for Fig 5: z = A*x + B*y + C, Coefficients

A = 1.25927, B = 1.06822, C = -0.767441, Correlation coefficient is 0.763071
Peak Values: Minimum Z = 0 at X = 0, Y = 0, Maximum Z = 4 at X = 1, Y = 1
Max - Min Z = 4, Mean = 0.3963049, Standard deviation = 0.6871763

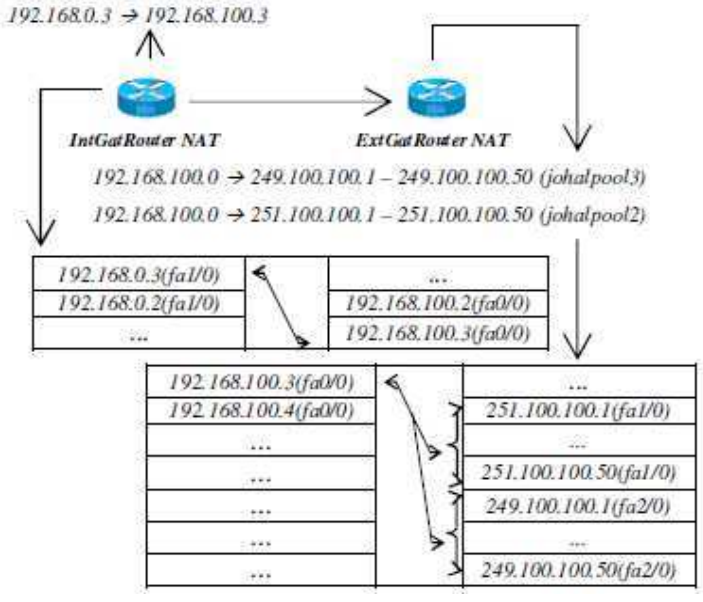


Fig. 4 Nested Static and Dynamic NAT.

2.3 Nested Static and Dynamic Address Translation

Next we create first level of Dynamic network address translation pool (johalpool1) at IntGatRouter for accessing 192.168.0.0 network devices.

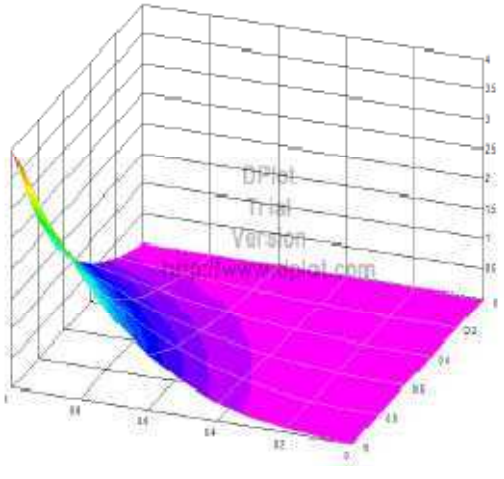


Fig. 5 Surface Plot for $\delta(sd) = \sum_{i=0}^{n} \eta^{i+1}(\eta + i)^{i+1} i^{+1} i$

```
IntGatRouter(config)#ip nat pool johalpool1 192.168.100.1 192.168.100.50 netmask
255.255.255.0
IntGatRouter(config)#ip nat inside source list johallist1 pool johalpool1
IntGatRouter(config)#access-list johallist1 permit 192.168.0.0 0.0.0.255
IntGatRouter(config)#end
```

At ExtGatRouter we create second level Dynamic NAT pool (johalpool2) to configure nested translation for johalpool1.

```
ExtGatRouter(config)#ip nat pool johalpool3 249.100.100.1 249.100.100.50 netmask
255.255.255.0
ExtGatRouter(config)#ip nat pool johalpool2 251.100.100.1 251.100.100.50 netmask
255.255.255.0
ExtGatRouter(config)#ip nat inside source list johallist2 pool johalpool2
ExtGatRouter(config)#access-list johallist2 permit 192.168.100.0 0.0.0.255
ExtGatRouter(config)#end
```

P2PSRV have no clue of LANPC2's original IP for initiating connection as there is multi-level Dynamic address translations at IntGatRouter and ExtGatRouter. The obvious way-out of this problem is to use NAT exemption. For using NAT exemption we need to manually override first and second level of Dynamic NAT configurations at IntGatRouter and Dynamic NAT configuration at ExtGatRouter respectively. This effort of removing multiple levels of Dynamic NAT configurations can be represented as $\delta\,(d)$.

$$\delta(d) = \sum_{i=0}^{n} \eta^i(\eta + i)^i i)i^i$$

where η is computation required to remove i^{th} Dynamic translation from each router n and $\eta + i$ is the associated complexity required to reconfigure ACL, configured for specific incoming traffic scenario. Graphically this effort can be represented as in fig 6.

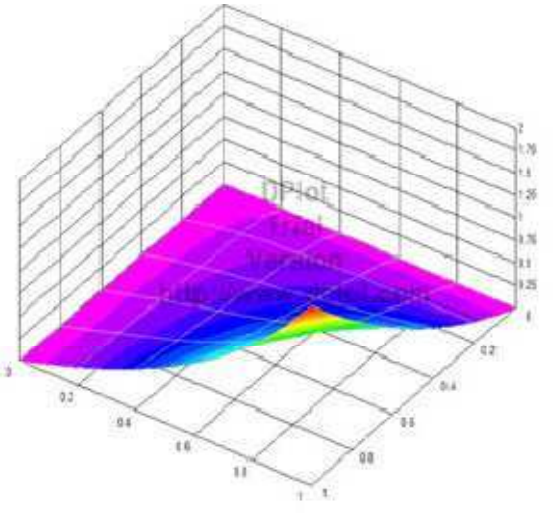


Fig. 6 Surface Plot for $\delta(d) = \sum_{i=0}^{n} \eta^i(\eta + i)^i i)i^i$

Best-Fit Plane for Fig 7: z = A*x + B*y + C, Coefficients
A = 0.737862, B = 0.805471, C = -0.395983, Correlation coefficient is 0.860056
Peak Values: Minimum Z = 0 at X = 0, Y = 0, Maximum Z = 2 at X = 1, Y = 1
Max - Min Z = 2, Mean = 0.3756826, Standard deviation = 0.4033071

3 3D NAT Scheme

In order to incorporate option of NAT exemption for selected applications and to make seamless transition between Static, Dynamic and exempted NAT we add third dimension to NAT table. This third dimension will keep track of nesting level information of Static and Dynamic address translations for a specific private network together with configuration setting to instantly implement NAT exemption as and when required. This 3DNAT table will be automatically generated by extracting Static and Dynamic address translation information from NAT tables of routers (like IntGatRouter in fig 1) participating in private network and will be maintained at outermost external gateway router (like IntGatRouter in fig 1).Here we assume that the third dimension will have information about level of Static and Dynamic address translation in form of α and β respectively, further π will have information for NAT exemption to be implemented for a specific entry in the table as in Fig 7,

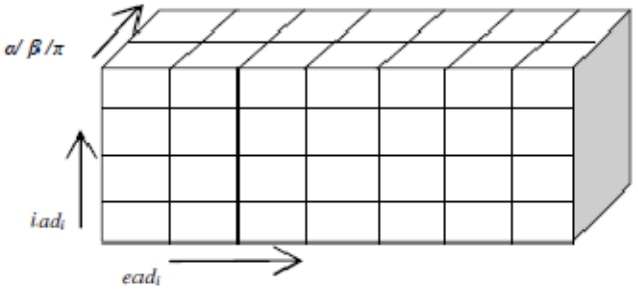
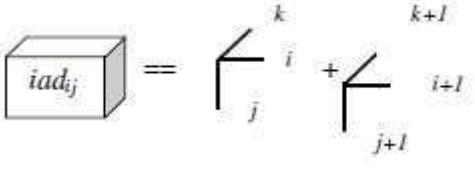


Fig. 7 3D NAT table structure.

where iad_i & ead_i represent the nested internal and external address mappings for i^{th} translation instance. Working of this 3D scheme can be interpreted in form of matrix.



i and j represent matrix indices for addresses translation mappings and k represent the type of translation or exemption. Matrix representation for 3DNAT implementation for multilevel Static address translation only.

	ead_{11}	ead_{21}	ead_{31}	ead_{41}	ead_{51}	ead_{ij}
iad_{11}	$\alpha+x$	β	β	β	β	.....
iad_{21}	β	$\alpha + x$ -1	β	β	β	.....
iad_{31}	β	β	$\alpha + x$ -2	β	β	.....
iad_{41}	β	β	β	$\alpha + x$ -3	β	.....
iad_{51}	β	β	β	β	$\alpha + x$ -4	.....
iad_{ij}	.....	.....	.....	.....	.....	.....

where *x* represent the level of nesting for Static address translation mappings, governed by number of intermediate routers between local host and destination host.Matrix representation for 3DNAT implementation for multilevel Dynamic address translation only.

	ead_{11}	ead_{21}	ead_{31}	ead_{41}	ead_{51}	ead_{ij}
iad_{11}	$\beta+x$	α	α	α	α	.....
iad_{21}	α	$\beta + x$ -1	α	α	α	.....
iad_{31}	α	α	$\beta + x$ -2	α	α	.....
iad_{41}	α	α	α	$\beta + x$ -3	α	.....
iad_{51}	α	α	α	α	$\beta + x$ -4	.....
iad_{ij}	.....	.....	.....	.....	.....	.....

where *x* represent the level of nesting for Dynamic address translation mappings, governed by number of intermediate routers between local host and destination host. Matrix representation for 3DNAT implementation for multilevel nested Static and Dynamic address translations.

	ead_{11}	ead_{21}	ead_{31}	ead_{41}	ead_{ij}
iad_{11}	$\alpha+x$	β	β	β	.....
iad_{21}	β	$\alpha + x$ -1	β	β	.....
iad_{31}	β	β	$\beta +(x-1)$ -1	β	.....
iad_{41}	β	β	β	$\alpha + ((x-1) - 1)$ - 1	.....
iad_{51}	β	β	β	β	.....
iad_{ij}	.....	.....	.....	.....	.....

where *x* represent the level of nesting for Static address translation mappings, governed by number of intermediate routers between local host and destination host.

Matrix representation for 3DNAT implementation for NAT exemption. NAT exemption can be nested with different levels of Static and Dynamic translation.

	ead_{11}	ead_{21}	ead_{31}	ead_{41}	ead_{51}	ead_{ij}
iad_{11}	π	β	β	β	β	……
iad_{21}	β	π	β	β	β	……
iad_{31}	β	β	π	β	β	……
iad_{41}	β	β	β	π	β	……
iad_{51}	β	β	β	β	π	……
iad_{ij}	……	……	……	……	……	……

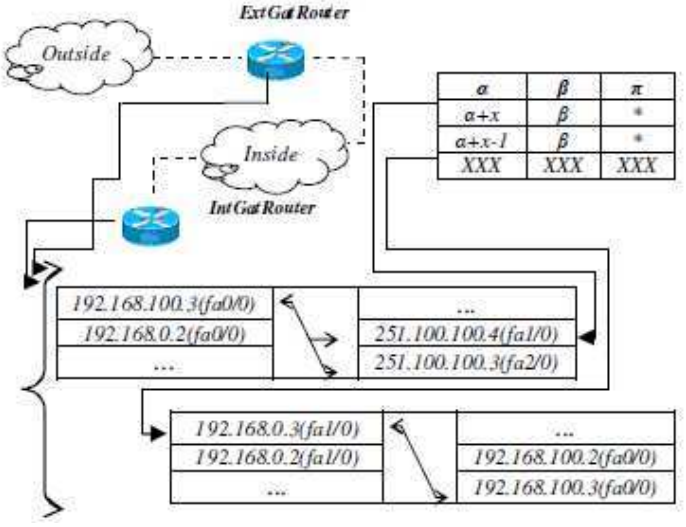


Fig. 8 3DNAT implementation for Multilevel Static NAT.

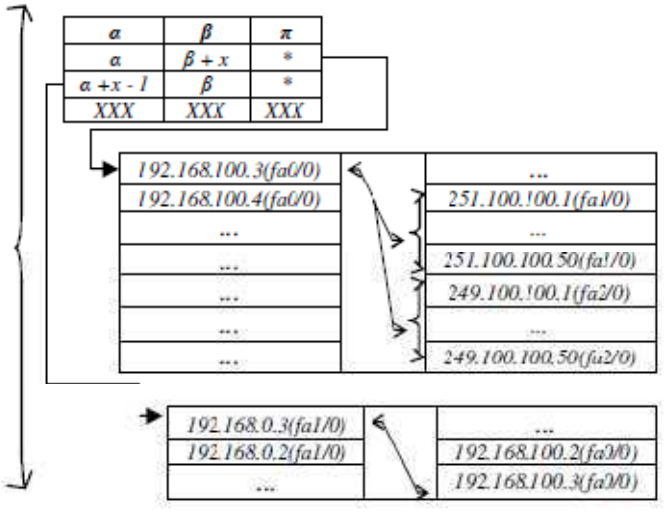


Fig. 9 3DNAT implementation for Nested Static & Dynamic NAT.

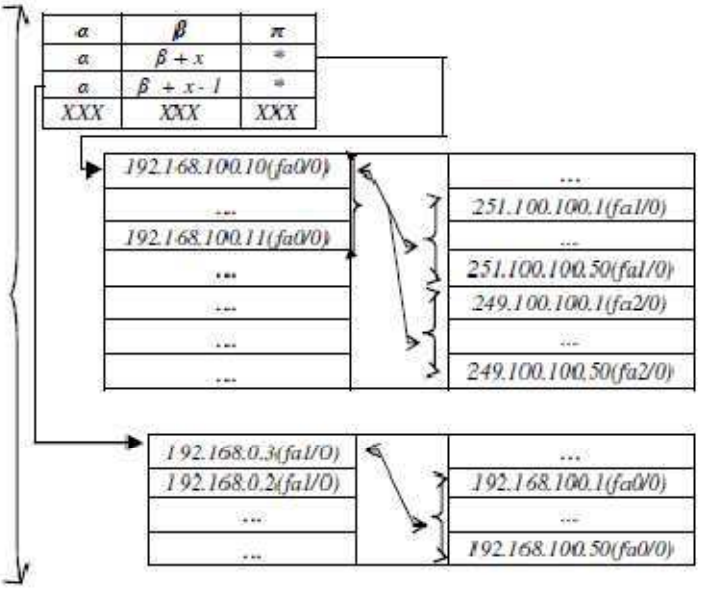


Fig. 10 3DNAT implementation for Multilevel Dynamic NAT.

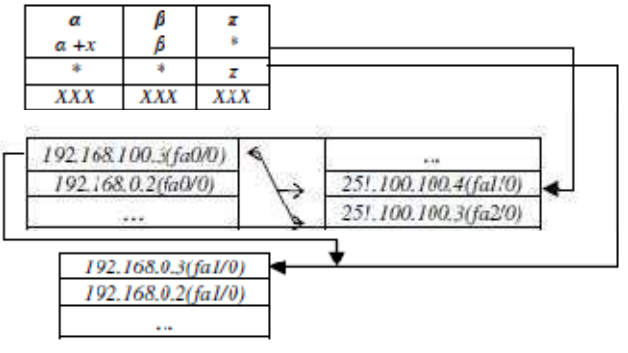


Fig. 11 3DNAT exemption with multilevel Static NAT.

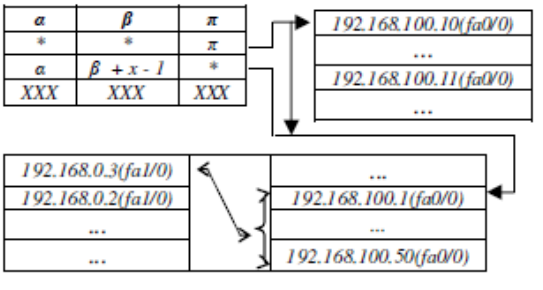


Fig. 12 3DNAT exemption with multilevel Dynamic NAT.

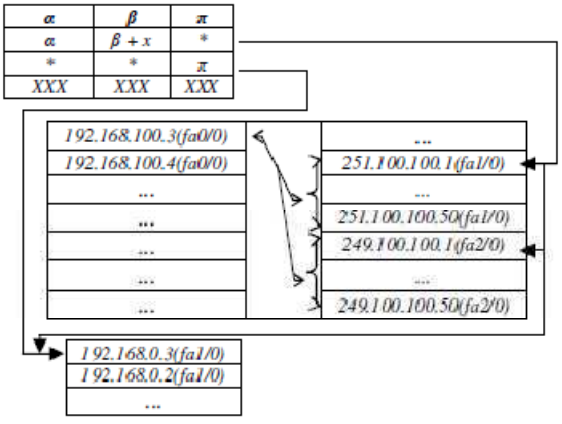


Fig. 13 3DNAT implementation with nested Static & Dynamic NAT.

4 Conclusion

With the proposed 3D NAT scheme there is no need to use 'clear xlate' command to implement changed NAT configurations. With 3DNAT this can be achieved seamlessly and moreover the table is able to retain all connections [12]. While traditionally we are unable to initiate connections from contexts shared interfaces when we apply NAT exemption for destination address because the packets can only be assigned by classifier to a shared interface context when we configure Static translation for destination address [13]. But with this scheme we can share outside interface even when NAT exemption is activated on inside interface and there is no effect on outside traffic reaching inside addresses.

References

1. Baker, F.: Requirements for IP Version 4 routers. RFC 1812, Internet Engineering Task Force (June 1995)
2. Cisco Systems. Cisco IOS Network Address Translation (NAT). Technical report, as of (December 2000)
3. Cohen, A., Rangarajan, S.: A programming interface for supporting IP traffic processing. In: Proc. Of IWAN 1999 (1999)
4. Hain, T.: Architectural implications of NAT. RFC 2993, Internet Engineering Task Force (November 2000)
5. Hasenstein, M.: IP network address translation. Diplomarbeit, Technische Universite at Chemnitz, Chemnitz, Germany, as of (December 1, 2000)
6. Srisuresh, P., Gan, D.: Load sharing using IP Network Address Translation (LSNAT). RFC 2391, Internet Engineering Task Force (August 1998)
7. Srisuresh, P., Holdrege, M.: IP Network Address Translator (NAT) terminology and considerations. RFC 2663, Internet Engineering Task Force (August. 1999)

8. Srisuresh, P., Tsirtsis, G., Akkiraju, P., Hefferman, A.: DNS extensions to Network Address Translators (DNS ALG). RFC 2694, Internet Engineering Task Force (September 1999)
9. Tsirtsis, G., Srisuresh, P.: Network Address Translation| Protocol Translation (NAT-PT). RFC 2766, Internet Engineering Task Force (February 2000)
10. Braden, R.: RFC 1122: Requirements for Internet Hosts — Communication Layers (October 1989)
11. Eppinger, J. L.: TCP Connections for P2P Apps: A Software Approach to Solving the NAT Problem. Tech. Rep. CMU-ISRI-05-104, Carnegie Mellon University, Pittsburgh, PA (January 2005)
12. Ford, B., Srisuresh, P., Kegel, D.: Peer-to-peer communication across network address translators. In: Proceedings of the 2005 USENIX Annual Technical Conference, Anaheim, CA (April 2005)
13. Audet, F., Jennings, C.: (text). RFC 4787 Network Address Translation (NAT) Behavioral Requirements for Unicast UDP (January 2007)

Maximizing Coverage Degree Based on Event Patterns in Wireless Sensor Networks

Majid Rafigh and Maghsoud Abbaspour

Abstract. A wireless sensor network is composed of a large number of sensors that are deployed in a field of interest to monitor specific phenomena such as temperature, sound, vibration, light, humidity, etc. Spanning health, home, environmental and military areas are some potential applications of WSNs. In several situations, events sensed by nodes occur on special geographical pattern. This paper proposed a new routing algorithm schema based on event occurrence pattern to satisfying k-coverage of event paths and maintaining degree of coverage in maximum level as more as possible. This method improves the network lifetime by shifting the routing responsibility from covering nodes to communication nodes, while maximizing the degree of coverage in the main path of events.

1 Introduction

Recent advances in low-cost and low-power circuit design and micro-electronics, and wireless communications have made possible the realization of low-cost miniaturized sensors to collect information throughout a sensing field and to send out this information to a data sink for additional processing[1,2].

Ensuring a steady monitoring of the whole surveillance area such that important events do not go undetected determines the wireless sensor network efficiency in many applications. In several critical applications, such as object tracking and intruder detection, the sensors need to be deployed in a field in a way that every point in this field is sensed by at least one sensor, where the events sensed and covered by at least k sensors. Such setup is called a K-Coverage WSN. High coverage degree helps achieve higher sensing accuracy and stronger robustness

Majid Rafigh · Maghsoud Abbaspour
Electrical and Computer Engineering Department
Shahid Beheshti University
Tehran, Iran
e-mail: m.rafigh@mail.sbu.ac.ir,
maghsoud@sbu.ac.ir

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 143–154.
springerlink.com

against sensor failures [3].In K-Coverage application redundant sensing and coverage is the one of most important QoS parameters [4].

In some environments, object movement follows a predictable geographical pattern, in contrast to applications where the object movement is completely random. This geographical pattern determines an event path, across which the events in the network happen by a high frequency. In such environments, events are sensed only by a subset of the networks' nodes, which are deployed close enough to the network's event path, called Covering Nodes. For example in road car tracking or in the environments containing natural obstacles (like valleys and rocks) object movement is restrained by the environmental features as shown in Fig. 1.

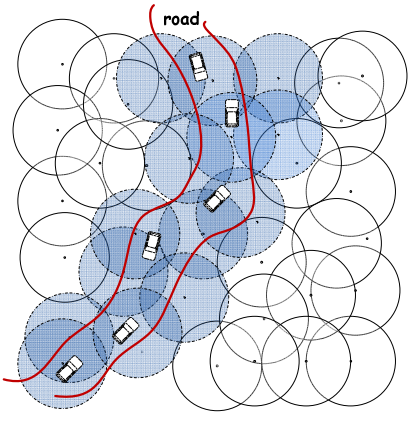


Fig. 1 Event Occurrence Pattern Sample

In such situations the lifetime of the Covering Nodes which collect the important data about desired objects, defines the networks' effective lifetime. Thus reducing the energy consumption of these nodes, which leads to a longer coverage of the event path, is of a great importance. By shifting the data routing functionality from Covering Nodes to other nodes of the network, namely Communication Nodes, increased event path coverage time with higher degrees could be achieved while keeping an effective communication of the sensor nodes to the sink.

The rest of paper has been organized as follows. Section 2 provides an overview of related work on k-Coverage techniques and coverage-aware routing algorithms for wireless sensor network. In Section 3, we introduce several parameters that are used in our model. The proposed algorithm outlines in Section 4. Section 5 provides details on the simulation setup and the simulation results, examining the different parameters for next node selection in multi-hop routing approach. Finally Section 6 concludes the paper.

2 Related Work

Work in the WSN routing is quite extensive, with energy efficiency and scalability being the main focus of many of the protocols proposed so far. Similarly, a lot work have been done on sensor coverage protocols, which focus on selecting a subset of the active sensor nodes that are sufficient to satisfy the network's coverage requirements, while allowing the remainder of the sensors to conserve their energy by entering the sleep mode. In this section, we discuss the related studies that have been conducted in both areas.

2.1 K-Coverage Algorithms

The problem of K-Coverage networks can be classified into scheduling algorithms that select minimum set of node to satisfy degree of K for every point in filed and verifying network to determine network is K-Covered or not, and study of connectivity and coverage problem.

The problem of verifying k-coverage is studied in [5]. Each sensor is modeled as a disk and it is proved that the area is properly k-covered if the perimeter of all disks is k-covered. The running time of the algorithm is $O(n^2 \log n)$ in the worst case for a set of n sensors. An improved modeling is presented in [6], where the authors use the concept of order-k Voronoi diagrams [7] to build a verifier algorithm. They show that if all vertices of a bounded Voronoi diagram are sufficiently covered, then the whole area is covered.

Xing et al. [8] analyzed the correlation between the coverage and connectivity properties of a network using a geometrical approach and developed a coverage configuration protocol (CCP) to provide any degree of coverage specified by the user. CCP is based on a proof that if the intersection points between all sensors are k-covered, the whole area is k-covered.

Kim et al. [9] presented a randomly ordered activation and layering (ROAL) scheme which solved the k-coverage problem in a distributed manner without using GPS information by performing a dynamic reconfiguration of the sensor network. The underlying principle of ROAL is to select k-disjoint subsets from deployed sensors to construct k layers, with each layer providing 1-coverage.Hefeeda and Bagheri [4] showed that the problem of selecting the minimum number of sensors required to achieve k-coverage is NP-hard and developed a randomized coverage scheme based on local information and low complexity messages to achieve a near optimal solution. The authors modeled the problem as a set system for which an optimal hitting set[10] corresponds to an optimal solution for coverage and used an approximation algorithm to find the optimal hitting set.

2.2 Coverage Preserving Routing Algorithms

Many routing algorithms have been proposed to satisfy the requirement of sensor networks. Several papers deal with the design of routing methods for the case of coverage preserving protocols of wireless sensor networks. The distributed activation with predetermined routes (DAPR) protocol proposed in [11] is the first routing protocol designed to avoid routing of data through areas sparsely covered by the sensor nodes. To accomplish this goal, the importance of every sensor node for the coverage-preserving task is quantified by a coverage-aware cost metrics. Salzmann et al[12] propose a routing algorithm which takes into account information concerning coverage and energy and using the advantages of scale free networks [13]. The method detects and exploits transmission and sensing redundancy in sensor networks with Geography Adaptive Fidelity algorithm (GAF)[14]. The redundant nodes will be shut down for energy saving purpose. Minget al. [15] proposed an energy-aware routing protocol (EAP) for a long-lived sensor network. EAP introduces a new clustering parameter for cluster head election, which can better handle the heterogeneous energy capacities. EAP uses the average residual energy of the clustering range and the residual energy of nodes for cluster head election. Also it introduces intra-cluster coverage to cope with the area coverage problem. Each cluster head selects some active nodes within clusters while maintaining coverage expectation of the cluster.

3 Cost Metric Parameters for Algorithm

In our proposed method the nodes are divided into two categories: Covering and Communication, each having a different functionality. Each node in the network is assigned to either of the two categories based on a function that takes various parameters into account to assess the importance of the node in terms of event path coverage. In this section these parameters and some basic definitions are provided.

Definition 1 *(sensing neighbor set):* For every node S_i, we define a group of neighboring nodes *SN(i)* that includes all nodes with sensing areas either partially or fully overlapped with the sensing area. The $\left|\xi_j - \xi_i\right|$is the Euclidean distance between the two sensor nodes.

$$SN(i) = \left\{S_j : \left|\xi_j - \xi_i\right| \leq 2r_s\right\} \tag{1}$$

Definition 2 *(coverage):* A point ρ covered by a sensor S_i and event can be detectable if event take palaces in sensing range of node S_i. Formally, the coverage *Cov(ρ,Si)* of under the deterministic sensing model is defined as follows:

$$Cov(\rho, S_i) = \begin{cases} 1 \; if \; \left|\xi_\rho - \xi_i\right| \leq r_s \\ 0 \end{cases} \tag{2}$$

Definition 3 *(k-coverage):* A point ρis k-covered simultaneously covered by at least k nodes. Degree of coverage for a point defines with $Cov^d(\rho)$. As consequence point ρ is k-covered if $Cov^d(\rho) \geq k$.

$$Cov^d(\rho) = \sum_j Cov(\rho, S_j) \tag{3}$$

Definition 4*(communication neighbor set):* The communication range of a sensor S_i is a region, every node in which can communicate with S_i. The communication neighbor set of a sensor S_i, denoted by *CN(i)*, is a set of all the sensors that are located in its communication range r_c.

$$CN(i) = \left\{S_j : \left|\xi_j - \xi_i\right| \leq 2r_c\right\} \tag{4}$$

3.1 Event History of Node

To determine the importance of a node in a coverage area event history of node, representing the number of events that has been sensed by node recently, is used. If a node has larger number of detected events, it is more important to this coverage area. *NCI(i)*define the coverage importance of node i in coverage of area points.

$$NCI(i) = \sum_j^{detected\ events} Cov\left(\rho_j, S_i\right) \tag{5}$$

In equation 5, $\rho_j$is the position of the detected event. When in an event path, detected events follow a static pattern, *NCI(i)* represents the importance of the node i in network coverage, but if events have dynamic pattern changing during network life, NCI(i) is not sufficient.

We use a time window to count recent activity of node i in recent Δt of process. As will be described later, this method operates in rounds. In the equation, $EH(i)$is event history of node *i.*

$$EH(i) = NCI(i)\ in\ last\ round \tag{6}$$

3.2 Node Effective Coverage

The number of the detected events by each node is not adequate to find the importance of a node in terms of network coverage, since there might be some nodes

that cover few events but cover a large area of the event path. These nodes have a significant responsibility to cover the event path. The relative position of the events covered by each node can form a minimal covering sector (MCS), defined by angle β and angle. MCS is the smallest sector of the nodes sensing circle that contains all the events covered by the node.

$$MCS(i) = \frac{(\beta - \alpha)}{360} \tag{7}$$

3.3 Redundant Coverage Impact

In redundant sensor deployment, a point will be covered by more than one sensor and if the each point of field covers by at least *K* sensor the network will be K-Coverage. Common Covering Set (CCS) of node Si and point p located in the covering area of node Si, is a subset of the S_i's neighbors that p is also located inside their covering area.

We define a parameter as *point cover* parameter to detect K-Coverage effect of a node. This parameter has been used to improve routing algorithm.

$$\mathrm{CCS}(S_i, \rho) = \{S_j : Cov(\rho, S_j) = 1 \, , S_j \in SN(i)\} \tag{8}$$

If members of CCS of point ρ and node *i* are more than *K-1*, then ρ will be K-Covered. For those points that have greater degree of coverage than *K*, members of common covering set has a greater probability of being selected as a communication node. $RedundantCov(i, \rho_{event})$ shows number of nodes that can be used as next hop node for sensor node S_j in $\mathrm{CCS}(S_i, \rho)$members.

$$RedundantCov(i, \rho_{event}) = \begin{cases} |\mathrm{CCS}(S_i, \rho_{event})| - k - 1 & if \ |\mathrm{CCS}(S_i, \rho_{event})| > k - 1 \\ 0 & otherwise \end{cases} \tag{9}$$

Average redundancy coverage for node *i* is defined as below:

$$\overline{RedundantCov(\imath)} = \frac{\sum_{e}^{detected\ events} RedundantCov(i, e)}{number\ of\ detected\ events} \tag{10}$$

$\overline{RedundantCov(\imath)}$ represents the average value the degree of coverage of the events' locations that has been covered by node i. The value of $\overline{RedundantCov(\imath)}$ is proportional to the coverage density of point *i*. The higher the density of a point is, the better are the nodes around that point for the purpose of communication.

3.4 Remaining Energy and Communication Cost

Energy consumed by a node depends on the size of the data packet that has been sent, the distance between the transmitter receiver, and another constant, called transmitter amplifier.

When sensor node i transmits data to sensor node j, the power consumed by transmission portion is [16]:

$$E_{tx}(S_i, S_j) = E_{amp} r_d + \varepsilon_{fs} \left| \xi_j - \xi_i \right|^2 r_d \quad (11)$$

Where r_d denotes the data rate, E_{amp}denotes the electronics energy expended in transmitting one bit of data, $\varepsilon_{fs} > 0$ is a constant related to transmission amplifier energy consumption, $\left| \xi_j - \xi_i \right|$is the Euclidean distance between the two sensor nodes.

In communication node selection process, only the distance between current node and router (next hop) node is variable and other components are stable, thus we employ distance in next hope evaluation function.

4 Proposed Algorithm

In previous section we introduced all the factors that have been used in proposed routing algorithm. Our schema works as an overlay on a distributed K-Covered sensor network. We assume that all nodes have same sensing and communication range, also are aware of their own location.

Our schema consists of two phases: information update, next hope selection, as described below.

4.1 Phase I: Information Update

The first phase of the proposed method is updating routing parameters information. Each sensor node broadcasts an update packet with information about its parameters that has changed during last round to all its neighbors in the $2 \cdot r_c$. Each node needs to maintain a neighborhood table to store the information of its neighbors. Unlike similar methods, the proposed algorithm does not require every node to broadcast its routing information. The nodes evaluate their importance in coverage on the basis of their history of the parameters introduced in previous section. Nodes with high importance in coverage do not prefer to be a communication node and therefore do not need to broadcast their routing information, which leads to decreased energy consumption.

Equation 12 defines coverage efficiency function that evaluates the coverage importance for a node. Base on coverage efficiency of node, each node decides to broadcast information or not. If $CoverageEff(i)$ is greater than a sensing efficiency threshold, then node i will send information to its neighborhoods and can be a communication node in multi-hope communications in network.

$$CoverageEff(i) = \theta_0 \cdot EH(i) + \theta_1 \cdot MCS(i) + \theta_2 \cdot \overline{RedundantCov}(S_i) \quad (12)$$

$Where\ \theta_0$, θ_1 and $\theta_1$are values to calibrate parameters in $CoverageEff$ equation.

4.2 Phase II: Next Hope Selection

Upon receiving the update information from all neighbors, each node calculates its neighbor's connectivity efficiency to select the next hop in its routing table, as described in equation 13. $E(j)$ Shows residual power of node j and the function $D_{sink}(j)$ is used to estimate the distance between node and the sink.

$$ConectivityEff(j) = \frac{E(j)}{\left(\left|\xi_j - \xi_i\right|^2\right) \times D_{sink}(j)} \tag{13}$$

5 Experimental Results

This section demonstrates the experimental results of the proposed algorithm. To evaluate the performance of the proposed scheme, some experiments were conducted on various methods. We compare our routing method with the EAP protocol as a representative of the energy-aware and coverage persevering protocols. EAP has been compared with classic clustering algorithms, HEED and LEACH, and make a good balance between network parameters such as connectivity, coverage and power consuming.

The nodes are deployed randomly and the locations of the sensor nodes are randomly chosen based on a uniform distribution. The data sink is fixed and located at the position$(50{,}200)$. The simulation parameters are summarized in Table 1.

To simulate the event pattern we assume a static movement model for events. In this model, objects move across network field from one side to other **side.**

Table 1 Simulation Parameters

Parameter	Value
Network Grid	(0,0) to (500,500)
Sink Position	(50,200)
Number of Sensors	1000, 2000
r_c	30 m
r_s	10 m
Initial energy	2 J
E_{amp}	50 nJ /bit
ε_{fs}	10 pJ /bit /m^2
Data packet size	40 Byte
Broadcast packet size	20 Byte

5.1 Network Coverage

To show network behavior for coverage we calculate average degree of coverage, number of sensor that sensed per moving object and number of live sensors that located in movement path and have sensed event in event history.

Fig. 2 shows the degree of coverage when an event sensed during the network life. As shown, at beginning of network life EAP and proposed method have same manner in coverage degree but as time pass EAP lost the movement path coverage and degree of coverage decreased.

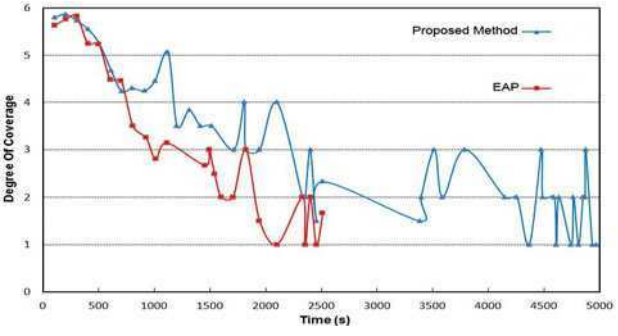


Fig. 2 Degree of coverage for 2000 sensor nodes

In case of covering the event path, proposed technique increase the covering time about 50% in compare with EAP. Also the degree of sensed events during network is more than EAP and by time passing it preserve the degree of coverage in high level as more as possible.

Figure 3 shows the number of live sensor that detected event and event history of a node is not empty. In fact this type of nodes located in the event path and if they have more power they will sense more events in network lifetime. In our algorithm the network energy remain in monitoring region more than EAP and the number of sensing nodes still greater than EAP. Fig. 3 depicts that number of nodes that located in event path in proposed method, lives 50% more than EAP and are 5 times greater than EAP.

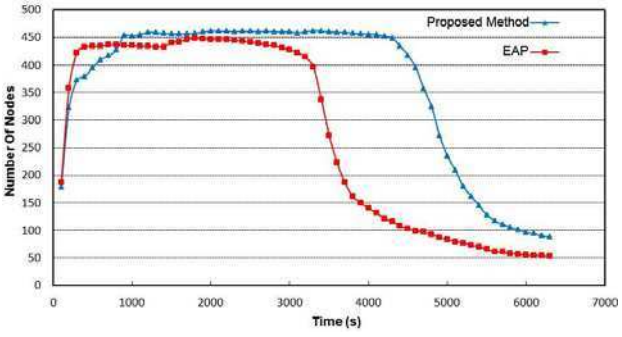


Fig. 3 Number of live nodes in sensing region that not have clear event history

Another parameter that we used to show network coverage behavior is the number of sense count per moving object. In our simulation, objects cross the network field during the simulation time. Each object has an ID related by arrival time. When a moving object passes from field, some of sensor nodes detect the object. Number of detection that concourse by different sensors has been shown in Fig. 4.

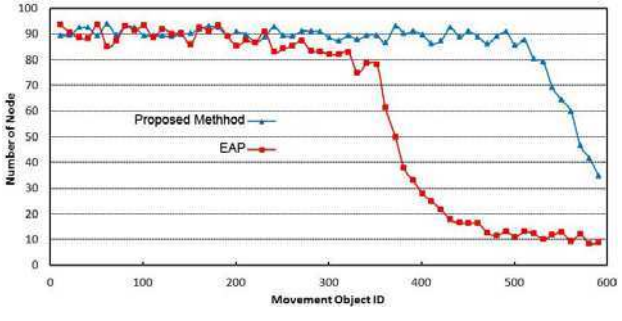


Fig. 4 Sense count by different sensor for moving object

In Figure 4 there are 600 moving objects in 2000 node deployed network. As like as degree of coverage, in proposed method the objects sensed with more sensor node in network life time. As shown in Figure 4, in our algorithm, moving objects have been detected about 5 times more than EAP in last 30% of network life in 2000 node deployment and about 2 times in 1000 node deployment scenario.

5.2 Power Consumption

As shown in information update section, in proposed algorithm only the routing nodes broadcast their information, so nodes consume their energy more efficient

and the network life time increased. We calculate the total energy of network for 2000 nodes with the initial power of $1J$ the result has been shown in **Fig. 5**.

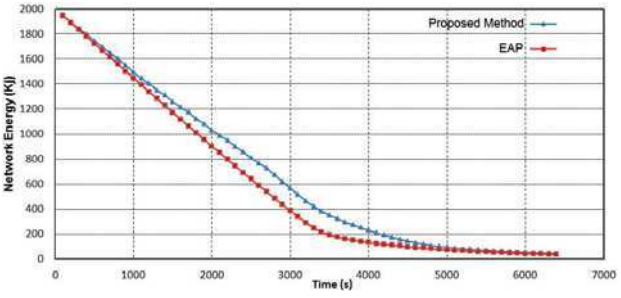


Fig. 5 Power consumption during network life

6 Conclusions

In this paper a novel solution for satisfying k-coverage of event paths in WSNs was proposed. This method improves the network lifetime by shifting the routing responsibility from covering nodes to communication nodes, while maximizing the degree of coverage in the main path of events. Thus the lifetime of main path coverage is increased dramatically as compared to previous methods.

Experimental results show the effectiveness of this approach in terms of increasing WSN lifetime and improving coverage along the main path of events.

Acknowledgments. This work is supported by Iranian Education and Research Institute for Information and Communication Technology (ERICT) under grant 8974/500.

References

[1] Akyildiz, I.F., et al.: A survey on sensor networks. IEEE Communications Magazine 40(8), 104–112 (2002)

[2] Culler, D., Estrin, D., Srivastava, M.: Guest Editors' Introduction: Overview of Sensor Networks. Computer 37(8), 41–49 (2004)

[3] Ammari, H.M., Das, S.K.: Joint k-Coverage, Duty-Cycling, and Geographic Forwarding in Wireless Sensor Networks. In: Proceedings of The Fourteenth IEEE Symposium on Computers and Communications, IEEE ISCC 2009 (2009)

[4] Hefeeda, M., Bagheri, M.: Randomized k-Coverage Algorithms For Dense Sensor Networks. In: Proceedings of 26th IEEE International Conference on Computer Communications, pp. 2376–2380 (2007)

[5] Huang, Y., Tseng, Y.: The coverage problem in a wireless sensor network. In: Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, pp. 115–121 (2003)

[6] So, A.M., Ye, Y.: On solving coverage problems in a wireless sensor network using voronoi diagrams. In: Proceedings of Workshop on Internet and Network Economics (2005)
[7] Okabe, T., Boots, B., Sugihara, K., Chiu, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, 2nd edn. John Wiley, Chichester (2000)
[8] Xing, G., Wang, X., Zhang, Y., Lu, C., Pless, R., Gill, C.: Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks. ACM Transactions on Sensor Networks (TOSN) 1(1) (2005)
[9] Kim, H., Kim, E.J., Yum, K.H.: ROAL: a randomly ordered activation and layering pro-tocol for ensuring k-coverage in wireless sensor networks. Journal of Networks (JNW) 3(1), 43–52 (2008)
[10] Bronnimann, H., Goodrich, M.: Almost optimal set covers in finite VC-dimension. Discrete and Computational Geometry 14(4), 463–479 (1995)
[11] Perillo, M., Heinzelman, W.: DAPR: a protocol for wireless sensor networks utilizing an application-based routing cost. In: Proceedings of IEEE Wireless Communications & Networking Conference - WCNC (2004)
[12] Salzmann, J., Kubisch, S., Reichenbach, F., Timmermann, D.: Energy and Coverage Aware Routing Algorithm in Self Organized Sensor Networks. In: Proceedings of Fourth International Conference Networked Sensing Systems, INSS (2007)
[13] Albert, R., Barabás, A.L.: Statistical Mechanics of Complex Networks. Reviews of Modern Physics 74(1), 47–97 (2002)
[14] Xu, Y., Heidemann, J., Estrin, D.: Geography-informed Energy conservation for Ad Hoc Routing. In: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, Rome (2001)
[15] Liu, M., Cao, J., Chen, G., Wang, X.: An Energy-Aware Routing Protocol in Wireless Sensor Networks. Sensors 9(1), 445–462 (2009)
[16] Wang, X., Ma, J.J., Wang, S., Bi, D.W.: Prediction-based Dynamic Energy Management in Wireless Sensor Networks. Sensors 7(1), 251–266 (2007)

Formal Specification and Implementation of Priority Queue with Starvation Handling

Jin Zhang, Gongzhu Hu, and Roger Lee

Abstract. Formal specification of software components, as a core research area in software engineering, has been widely studied in decades. Although quite a few formal models have been proposed for this purpose, specification of concrete software components is still a challenging task due to the complexity of the functionalities of the components. In this paper, we use the stream function model to specify the behavior of priority queue, a commonly used software component. This specification formally defines the regular behavior and fault tolerance behavior of priority queue. In particular, a priority-concatenation operator is defined to handle the ordering of data items to ensure the highest-priority item is removed first. A finite state machine as an implementation is built based on this specification. In addition, we also discuss a priority upgrading approach to handle possible starvation situation of low-priority data items in the priority queue.

Keywords: stream function, priority-concatenation, state transition machine, starvation prevention.

1 Introduction

Over the last decade, software development has a tendency to build on models or components. From the software engineering point of view, model-based or

Jin Zhang
Department of Computer Science, Hainan University, Haikou, Hainan 570228, China
e-mail: zj001_cn@163.com

Gongzhu Hu
Department of Computer Science, Central Michigan University,
Mt. Pleasant, MI 48859, USA
e-mail: hu1g@cmich.edu

Roger Lee
Department of Computer Science, Software Engineering & Information Technology Institute,
Central Michigan University, Mt. Pleasant, MI 48859, USA
e-mail: lee1ry@cmich.edu

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 155–167.
springerlink.com

component-based approaches provide an efficient way to improve the software development process in each stage of the life cycle of a software package.

The first and utmost critical aspect of component-based software development is the *specification* that describes the functionality and behavior of the component. There are three levels of component specifications — informal, self-formal and formal. *Informal specification* describes the component's behavior in natural language that needs to be translated to code by individual programmer. *Semi-formal specification* uses some descriptive language or other representation to specify the component so that the interpretation of the component's behavior is less ambiguous (if not all precise). Most specification standards today are semi-formal. For example, UML, the industry-standard specification language, uses graphical notation, case notation and component notation [9]. The most desirable approach is *formal specification* by which the behavior or functionality of a software component is described in a formal language (algebraic, for example) to ensures the correctness of the software component when implemented according to the specification.

Several formal specification methods have been proposed in the past, such as meta-modeling framework [10], object-oriented paradigm [8], Object-Z specification language [11], algebraic semantics [1], and *stream function* [12]. With stream function, a software component is considered a "black box" interacting with its operational environment and its functionality is described as a mapping between the input and output streams. Such mapping can lead directly to an implementation of the component [2].

Several software components, such as stack and queue that are widely used data structures, have been formally specified as stream functions [3, 7]. The items entered into these software components are simply data values (or objects) and identified by their positions (ordering) in the input stream. In this paper, we give a formal specification of priority queue as a mapping of stream function where the items are identified by their priorities, and by their positions in the input stream for those that have the same priority.

The main contribution of this work is an introduction of the *priority-concatenation* operator $\&_p$ that altered the ordering of the items of two streams being concatenated based on the priorities of the items. This is an enhancement of the regular concatenation operator for all previous software component specifications using streams.

2 Stream Function

First, we briefly review the basic concept of stream function as formal model for component specification.

Given an alphabet $\mathscr{A}$, the set $\mathscr{A}^\star$ comprises all *streams* $A = \langle a_1, a_2, \ldots, a_k \rangle$ of *length* $|A| = k$ with elements $a_i \in \mathscr{A}$ $(i \in [1,k], k \geq 0)$. Several operations can be applied to communication streams, one of the basic operator is *concatenation* defined as

$$\& : \mathscr{A}^\star \times \mathscr{A}^\star \rightarrow \mathscr{A}^\star$$

of two streams $A = \langle a_1, a_2, \ldots, a_k \rangle$ and $B = \langle b_1, b_2, \ldots, b_l \rangle$, that yields the stream

$$A \,\&\, B = \langle a_1, a_2, \ldots, a_k, b_1, b_2, \ldots, b_l \rangle. \tag{1}$$

A stream function $f : \mathscr{A}^\star \to \mathscr{B}^\star$ maps an input stream to an output stream [12].

Let *Input* and *Output*, or simply I and O, be the input and output alphabets, respectively. For an interactive software component C, its behavior can be specified as a stream function:

$$C : I^\star \to O^\star$$

where the domains of I and O depend on the characteristics of the component C.

3 Priority Queue

Priority queue is a data structure for objects that are entered and removed based on the *priority* associated with the objects. That is, the object removed from the priority queue is the one with the highest priority. We shall give a formal specification of priority queue in this section.

3.1 Basic Definition

Data items and priorities. let $D = \{d_1, d_2, \ldots\}$ be the set of items storable in a priority queue, $P = \{p_1, p_2, \ldots, p_m\}$ be a set of priority values with m being the maximum priority associated with items in D. The data stream of the priority queue can be represented as a sequence of ordered pairs

$$\langle (d_1, p_1), (d_2, p_2), \ldots, (d_i, p_i), \ldots \rangle$$

where d_i is the data value of the item and p_i is the priority associated with d_i. We also denote the set of items with the same priority as $D_{pi} \subseteq D$. That is, the item set D is partitioned into equal-priority subsets as

$$D = \{D_{pm}, D_{p(m-1)}, \ldots, D_{p1}\}. \tag{2}$$

where $D_{pi}, i = m, m-1, \ldots, 1$ is the sequence of items with the same priority p_i.

Input and output streams. Let $enq(d, p)$ be an enter-queue command that enters an item $d \in D$ with priority p into the priority queue, deq be a de-queue command that removes the first item with the highest priority from the priority queue, and $exc \notin D$ represents an exception. An input to the priority queue is

$$I \;=\; enq(D,P) \,\cup\, \{deq\}$$

and an output of the priority queue is

$$O \;=\; D \,\cup\, \{exc\}$$

The priority queue is a mapping

$$pqueue: \ I^\star \ \rightarrow \ O^\star$$

We use Enq and Deq to represent zero or more enq and deq commands, respectively:

$$Enq \in enq(D,P)^\star, \ Deq \in deq^\star$$

We also use enq^n to denote n enq commands and deq^n to denote n deq commands. The data items in the $enq(d,p)$ commands of an input stream $X \in I^\star$ is denoted as X_D. That is,

$$X_D = \langle \ldots, (d_i, p_i), \ldots \rangle \tag{3}$$

where $(d_i, p_i) \in D$ and $Enq(D,P) \subseteq X$.

Priority-concatenation operator. The more important factor within the ordered pairs of the data stream is the priority. To facilitate the use of priority in the specification, we introduce a second concatenation operator, called *priority-concatenation* denoted as $\&_p$. This operator plays the key role in the formal specification of priority queue. It is applied to two situations: concatenation of two data streams and concatenation of two input streams (note that an input stream is a sequence of enq and deq commands).

Data stream case — The $\&_p$ operator for two data streams is a mapping

$$\&_p : D^\star \times D^\star \rightarrow D^\star \text{ for } (d_i, p_i) \in D$$

and is defined recursively. Let $A, B \in D^*$ be two data streams and represented as a sequence of equal-priority subsequences:

$$A = \langle A_{pm}, \ldots, A_{pi}, \ldots, Ap1 \rangle$$
$$B = \langle B_{pm}, \ldots, B_{pi}, \ldots, Bp1 \rangle$$

where A_{pi} is the concatenation of items in A with priority pi; same for B. Note that some of the subsequences may be empty.

First, the initial conditions are:

$$A \,\&_p \langle\rangle = A \tag{4}$$
$$\langle\rangle \,\&_p B = B \tag{5}$$
$$\langle (a,p) \rangle \,\&_p \langle (b,q) \rangle = \begin{cases} \langle (a,p), (b,q) \rangle \text{ if } p \geq q \\ \langle (b,q), (a,p) \rangle \text{ if } p < q \end{cases} \tag{6}$$

Concatenation of A with an empty stream is A itself, as defined in Equations (4) and (5). If both A and B are single-item streams, $A \,\&_p B$ yields a 2-item stream with the two items sorted on their priorities, or b is appended after a if they have the same priority, as given in Equation (6).

The recursive relation of the $\&_p$ operator is defined by Equations (7)–(9).

$$A \,\&_p \langle (b,p) \rangle = \langle A_{pm}, \ldots, A_p \,\&\, (b,p), \ldots, A_{p1} \rangle \tag{7}$$

$$\langle (a,p) \rangle \,\&_p B = \langle B_{pm}, \ldots, (a,p) \,\&\, B_p, \ldots, B_{p1} \rangle \tag{8}$$

and

$$A \,\&_p B = \langle A_{pm} \,\&_p B_{pm}, \ldots, A_{pi} \,\&_p B_{pi}, \ldots, A_{p1} \,\&_p B_{p1} \rangle \tag{9}$$

A single-item stream of priority p is appended at the end or inserted at the front of the p-subsequence in the other stream, as indicated in Equations (7) and (8). Relation (9) says that priority-concatenation of A and B results in a sequence of items $(d,p) \in A \cap B$ sorted on p, and items in A precede items in B within each equal-priority subsequence.

Input stream case — We now extend the $\&_p$ operator to input streams. It is basically the priority-concatenation of the data streams in the $enq(d,p)$ commands in the input stream:

$$\&_p : I^\star \times I^\star \to I^\star$$

in such a way that for $X, Y \in I^\star$,

$$X \,\&_p Y = Z \tag{10}$$

where $Z_D = X_D \,\&_p Y_D$ and the command $c_1 \in Z$ precedes $c_2 \in Z$, denoted as $c_1 \prec c_2$, if

$$c_1, c_2 \in X \text{ or } c_1, c_2 \in Y, \text{ or} \tag{11}$$

$$c_1 = enq(d_1, p_1), c_2 = enq(d_2, p_2), p_1 \geq p_2 \tag{12}$$

Condition (11) ensures the ordering of commands (regardless of *enq* or *deq*) in X and in Y remains no change in Z, while condition (12) indicates the sorted ordering of *enq* commands on priority.

Example. Let

$$X = \langle enq(a,5), enq(b,5), enq(c,4), deq \rangle$$
$$Y = \langle enq(r,6), deq, enq(s,5), enq(t,3), deq \rangle$$

$X \,\&_p Y$ would be

$$\langle enq(r,6), enq(a,5), enq(b,5), deq, enq(s,5), enq(c,4), deq, enq(t,3), deq \rangle.$$

3.2 Regular Behavior of Priority Queue

Theoretically, a priority queue is unbounded, i.e. no limit on the size of the queue. But in practice, it is often bounded with a size limit. This limitation makes it harder

to specify than the unbounded case simply because it introduces an additional exception to handle when entering an item to a full priority queue. In this paper, we deal with bounded priority queue. First, we define several notations. Let

- m be the highest priority,
- $X \in I^\star$ be an input stream,
- N be the capacity of the priority queue,
- $\langle d_{i1}, d_{i2}, \ldots \rangle$ be a sequence of items of the same priority i.

The regular behavior of a priority queue is defined by the following equations.

$$pqueue(X \,\&\, enq(d_{pi}, p_i)) = pqueue(X \,\&_p\, enq(d_{pi}, p_i)) \tag{13}$$

$$pqueue(deq \,\&\, X) = \langle d_{m1} \rangle \,\&\, pqueue(X \setminus (d_{m1}, p_m)) \tag{14}$$

where $X \setminus (d_{m1}, p_m)$ is X with (d_{m1}, p_m) removed from X_D.

Equation (13) defines the normal *enq* operation on the priority queue. It does not generate output but the *enq* command is priority-concatenated to X. Normal *deq* operation is given in Equation (14). This operation produces the item d_{m1} to the output stream, which is the first item among those with the highest priority m from the priority queue, and it is consumed.

3.3 Irregular Behavior of Priority Queue

There are two exceptions to consider for a bounded priority queue: *enq* to a full queue and *deq* from an empty queue. There are several different ways handling the exceptions. The simplest way is *fault sensitive* that treats an exception as an error. Or, the *fault tolerance* approach "catches" the exception and produces a special symbol (i.e. *exc*) to the output stream. A more complicated approach is *fault correcting* that delays the exception-causing operation until a later time when the operation can be performed normally. These variations make the behavior of the priority queue "irregular." Similar specification for irregular stacks was given in [3].

We shall give formal specifications for fault sensitive and fault tolerance variations of priority queue. In the following, let k be the number of items in the queue.

Fault sensitive. The exceptions are considered an error, and the priority queue will produce no output for the remaining input once the exception is raised.

$$pqueue(X(k = N) \,\&\, enq(d_p, p)) = \langle\rangle \tag{15}$$

$$pqueue(deq \,\&\, \langle\rangle \,\&\, X) = \langle\rangle \tag{16}$$

That is, the priority queue will behave as if the operations are halted.

Fault tolerance. An exception is "caught" and an *exc* is produced to the output, the priority queue will consume and skip the operation and continue to process the rest of the input.

$$pqueue(X(k = N) \,\&\, enq(d_p, p)) = \langle exc \rangle \,\&\, pqueue(X) \tag{17}$$

$$pqueue(deq \,\&\, X(k = 0)) = \langle exc \rangle \,\&\, pqueue(X) \tag{18}$$

An *enq* operation on a full queue will produce an exception to the output stream and continue to process the remaining of the input stream with the the error-causing object d_{pi} ignored, as specified in Equation (17). Likewise, a *deq* operation on an empty queue produces an exception to the output stream shown in Equation (18).

Fault correcting. This approach is to feed the exception-causing command back to the input stream to execute at a later time. That is, the queue delays the command until it can be performed.

$$pqueue(X(k = N) \,\&\, enq(d_p, p) \,\&\, Y) = pqueue(X \,\&\, deq_1 \,\&\, enq(d_p, p) \,\&_p\, (Y \setminus deq_1)$$

where deq_1 is the first deq-queue commands in Y, and

$$pqueue(deq \,\&\, \langle\rangle \,\&\, X) = pqueue(enq(d_1, p_1) \,\&\, deq \,\&_p\, (X \setminus enq(d_1, p_1)))$$

where $enq(d_1, p_1)$ is the first (with highest priority) enter-queue command in X.

4 Implementation

The specification given above is from an abstract point of view and independent of the way the priority queue is implemented. It can be implemented in different ways, but an elegant and sound approach is to derive a *state transition machine* (STM) from the formal specification so that the implementation is guaranteed to satisfy the specified behavior of the software component. Here we shall only describe the implementation of fault-tolerance priority queue.

4.1 State Transition Machine

A *state transition machine with input and output* is a 6-tuple

$$M = (S, I, O, \delta, \phi, q_0)$$

where

- S is a non-empty finite set of *states*,
- I is a non-empty finite set of *input data*,
- O is a non-empty finite set of *output data*,
- δ is *state transition function* $\delta : S \times I \to S$,
- ϕ is *output function* $\phi : S \times I \to O^{\star}$, and
- s_0 is an *initial state* $s_0 \in S$.

Recall that the priority queue is defined as a mapping

$$pqueue: I^\star \rightarrow O^\star$$

The output depends not only on the input stream but also the size $0 \leq k \leq N \in \mathbb{N}$ of the priority queue, and k varies to the limit N. A state s in the STM describes the state of the priority queue that contains k data items. Hence a *state* is defined as

$$s = [D, k] \in S$$

where D is the sequence of data items stored in the priority queue and $k \in \mathbb{N}$ is its length. The initial state is that the priority queue contains no data with length 0:

$$s_0 = [\langle\rangle, 0]$$

The state transition function δ and output function ϕ are defined as function of the next command in the input stream and the queue length k. These functions are constructed according to the formal specifications given before.

4.2 Regular Behavior

Regular behavior of a priority queue is when enter-queue on a non-full queue or de-queue on a non-empty queue.

Transition on $enq(d,p)$**.** According to the specification equations (13), an $enq(d,p)$ command on a non-full priority queue of size $0 \leq k < N$ enters (d,p) into the priority queue and does not produce output:

$$\delta([D,k], enq(d,p)) = [D' \&_p (d,p), k+1] \tag{19}$$

$$\phi([D,k], enq(d,p)) = \langle\rangle \tag{20}$$

Transition on deq**.** Equation (14) specifies the regular behavior upon a deq command on non-empty priority queue of length $k \geq 1$. Accordingly, the transition function δ moves to a state in which the first item (d, p_m) among those with the highest priority m is removed and the queue length is decreased by 1, and the function ϕ produces (d, p_m) as the output:

$$\delta([D,k], deq) = [D \setminus (d,p_m), k-1] \tag{21}$$

$$\phi([D,k], deq) = \langle (d,p_m) \rangle \tag{22}$$

The state transition machine is shown in Figure 1, where the label on each edge is a triple (command, k, output).

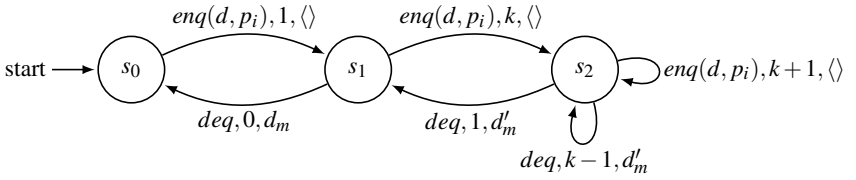


Fig. 1 State transition machine for regular behavior of priority queue.

The data items in each state in the figure are:

s_0: $[\langle\rangle, 0]$
s_1: $[\langle d_m\rangle, 1]$
s_2: $[\langle d'_m, \ldots, d_{pi}, \ldots, d_1\rangle, 1 < (k+1) \leq N]$

4.3 Fault Tolerance Behavior

A fault tolerance priority queue behaves the same as regular case, plus that it handles two exceptions: entering an item to the priority queue that is full or removing an item when it is empty.

Transition on $enq(d,p)$**.** When the priority queue reaches its capacity N, an $enq(d,p)$ command raises an exception that is caught and an *exc* is generated to the output stream, as stated in Equation (17). The state transition function δ and output function ϕ are given below.

$$\delta([D,N], enq(d,p)) = [D,N] \tag{23}$$
$$\phi([D,N], enq(d,p)) = \langle exc\rangle \tag{24}$$

Transition on deq**.** When the priority queue is empty, a deq command raises an exception that is caught and an *exc* is generated to the output stream. From Equation (18) that specifies this situation, we have

$$\delta([\langle\rangle,0], deq) = [\langle\rangle,0] \tag{25}$$
$$\phi([\langle\rangle,0], deq) = \langle exc\rangle \tag{26}$$

From these functions (Equations (19) – (26)), the state transition machine of fault tolerance priority queue is given Table 1.

Table 1 State transition table $(D,\ 0 \le k \le N)$

state	s	$Input$	$s' = \delta(s, Input)$	$Ouput = \phi(s, Input)$	Comment
0	$[\langle\rangle, 0]$	$enq(d,p)$	$[\langle(d,p)\rangle, 1]$	$\langle\rangle$	normal *enq*
0	$[\langle\rangle, 0]$	deq	$[\langle\rangle, 0]$	$\langle exc\rangle$	underflow
1	$[\langle c\rangle, 1]$	$enq(d,p)$	$[\langle c\rangle \&_p (d,p), 1 < k < N]$	$\langle\rangle$	normal *enq*
1	$[\langle c\rangle, 1]$	deq	$[\langle\rangle, 0]$	$\langle d_m\rangle$	normal *deq*
2	$[D,k]$	$enq(d,p)$	$[D \&_p (d,p), k+1]$	$\langle\rangle$	normal *enq*
2	$[D,k]$	deq	$[D \setminus d_m, k-1]$	$\langle d'_m\rangle$	normal *deq*
2	$[D,k]$	deq	$[D \setminus d_m, 1]$	$\langle d'_m\rangle$	normal *deq*; 1 item left
3	$[D,N-1]$	$enq(d,p)$	$[D \&_p d, N]$	$\langle\rangle$	normal *enq*; last item
3	$[D,N-1]$	deq	$[D \setminus d''_m, k]$	$\langle d''_m\rangle$	normal *deq*
4	$[D,N]$	$enq(d,p)$	$[D,N]$	$\langle exc\rangle$	overflow
4	$[D,N]$	deq	$[D \setminus d'''_m, k]$	$\langle d'''_m\rangle$	normal *deq*

The corresponding diagram is shown in Figure 2.

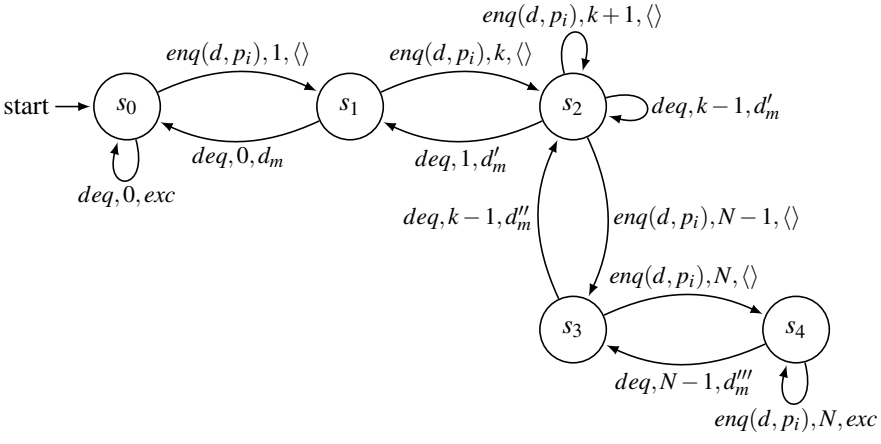


Fig. 2 State transition machine for fault tolerance behavior of priority queue.

The data items in each state in the figure are:

s_0: $[\langle\rangle, 0]$
s_1: $[\langle d_m\rangle, 1]$
s_2: $[\langle d'_m, \ldots, d_{pi}, \ldots, d'_1\rangle, 1 < k < N]$
s_3: $[\langle d''_m, \ldots, d_{pi}, \ldots, d''_1\rangle, N-1]$
s_4: $[\langle d'''_m, \ldots, d_{pi}, \ldots, d'''_1\rangle, N]$

5 Handling of Starvation

One practical problem with priority queue is that some items with low priority may not ever be selected to depart from the priority queue because higher-priority items keep entering. This situation is commonly called *starvation*. Several strategies have been proposed to handle the starvation situation, such as time-out and multiple queues (one queue for each priority).

We will formally introduce a *priority upgrading* approach to handle possible starvations. This approach does not prevent the starvation situation from occurring, but it will reduce the chance for a low-priority item to stay in the queue forever. The basic idea is to have a set Θ of thresholds as the parameter to the priority queue:

$$\Theta = \{t_m, \ldots, t_1\}$$

where $t_i, i \in [1,m]$ is a threshold value for the number of items of priority i in the queue. Let the data items in the priority queue be partitioned based on their priorities as given in Equation (2):

$$D = \langle D_{pm}, \ldots, D_{pi}, D_{p(i-1)}, \ldots, D_{p1} \rangle$$

and $D_{p(i-1)} = \langle d^1_{p(i-1)}, d^2_{p(i-1))}, \ldots \rangle$. That is, $d^1_{p(i-1)}$ is the first item in $D_{p(i-1)}$. Then, the priority upgrading is defined below if $|D_{pi}| \geq t_i$ when the next input command is $enq(d,p)$:

$$D_{pi} \quad \leftarrow D_{pi} \,\&\, d^1_{p(i-1)} \,\&\, d \tag{27}$$

$$D_{p(i-1)} \leftarrow D_{p(i-1)} \backslash \langle d^1_{p(i-1)} \rangle \tag{28}$$

That is, when the number of items in D_{pi} has reached its threshold t_i, the first item in the next priority level, $d^1_{p(i-1)}$, is promoted to priority level i, and the newly arrived item d will be appended after it.

Example. Assume that the current data items in the priority queue are

$$\langle (a,p_5), (b,p_5), (c,p_5), (d,p_4), (e,p_4), \ldots \rangle$$

and $t_5 = 3$. The next input command $enq(f,p_5))$ will cause (d,p_4), the first item of priority 4, to be promoted to priority 5 because $|D_{p5}| = 3 \geq t_5$. This upgrading yields this in the priority queue:

$$\langle (a,p_5), (b,p_5), (c,p_5), \mathbf{(d,p_5)}, (f,p_5), (e,p_4), \ldots \rangle.$$

To include priority upgrading in the implementation, we can extend the state in STM to include a counter vector C to record the lengths of the equal-priority subsequences:

$$s = [D,C,k] \in S$$

where D and k are the same as before, and

$$C = (c_{pm}, \ldots, c_{pi}, \ldots, c_{p1})$$

where $c_{pi} = |D_{pi}|$ is the length of the subsequence of items with priority i.

The state transition function δ is extended according to the priority upgrading formula (27) and (28):

$$\delta([D,C,k], enq(d,p_i)) = \begin{cases} [D \,\&_p\, (d,p_i), C, k+1], & \text{if } c_{pi} < |D_{pi}|, \\ [\langle \ldots, D_{pi} \,\&\, d^1_{p(i-1)} \,\&\, (d,p_i), D_{p(i-1)} \backslash d^1_{p(i-1)}, \ldots \rangle, \\ \quad (\ldots, c_{pi}+1, c_{p(i-1)}-1, \ldots), k+1], & \text{otherwise.} \end{cases}$$

The other parts of the STM remain the same as before except that C is included in the states and c_{pi} is increased by 1 each time an $enq(d,p_i)$ command is encountered and c_{pm} is decreased by 1 for each deq.

6 Related Work

A lot of work has been done on formal methods for software components and their designs [6, 8, 10]. Most of the work does not treat a component as an interactive device, neither do they define a component as a "black box" to work with the input/output as streams flowing through the box.

Study on specifications of regular queue as a mapping from input stream to output stream has been reported, such as [4]. In these studies, the behaviors of the queue in both normal mode and error mode were considered. In particular, the operation of a deq command on an empty unbounded queue was defined.

State transition machines, as a formal mechanism for the implementation of interactive components was also studied. In [5], a formal method was introduced to transform a stream function into a state transition machine using abstraction of input histories.

7 Conclusion

Many software components are interactive that receive continuous input streams and produce output streams, and can be defined using stream functions. Although the fundamental concept is the same, specification of an individual component is a nontrivial task because of the distinct characteristics of the component.

In this paper, we defined the stream functions for the specification of priority queue and created a state transition machine as an implementation for the regular behavior and fault tolerant behavior of the priority queue based on the formal specification. The distinct characteristic of priority queue is the order in which the data items are removed, that is very different from a regular queue. We defined the priority-concatenation operator to address this unique behavior of this software component.

In addition, we proposed a priority upgrading approach to prevent low-priority items from starvation.

References

1. Bidoit, M., Hennicker, R.: An algebraic semantics for contract-based software components. In: Bevilacqua, V., Roşu, G. (eds.) AMAST 2008. LNCS, vol. 5140, pp. 216–231. Springer, Heidelberg (2008)
2. Breitling, M., Philipps, J.: Diagrams for dataflow. In: Proceedings of Formale Beschreibungstechniken für verteilte Systeme (FBT), pp. 101–110. Verlag Shaker (2000)
3. Dosch, W., Hu, G.: On irregular behaviours of interactive stacks. In: Proceedings of 4th International Conference on Information Technology: New Generations, pp. 693–700. IEEE Computer Society, Los Alamitos (2007)
4. Dosch, W., Ruanthong, W.: On history-sensitive models of interactive queues. In: Proceedings of the 5th International Conference on Computer and Information Science, pp. 271–278. IEEE Computer Society, Los Alamitos (2006)
5. Dosch, W., Stümpel, A.: Transforming stream processing functions into state transition machines. In: Dosch, W., Lee, R.Y., Wu, C. (eds.) SERA 2004. LNCS, vol. 3647, pp. 1–18. Springer, Heidelberg (2006)
6. Horst, J., Messina, E., Kramer, T., Huang, H.M.: Precise definition of software component specifications. In: Proceedings of the 7th Symposium on ComputerAided Control System Design, pp. 145–150 (1997)
7. Hu, G.: Formal specification of bounded buffer using stream functions. In: Proceedings of the IEEE International Conference on Information Reuse and Integration, pp. 230–235. IEEE System, Man, and Cybernetics Society, Las Vegas (2009)
8. Lau, K.-K., Ornaghi, M.: A formal approach to software component specification. In: Proceedings of Specification and Verification of Component-based Systems Workshop at OOPSLA 2001 (2001)
9. Object Management Group.: Unified modeling language: infrustrcture, version 2.0. Tech. rep., OMG (2003)
10. Övergaard, G.: Formal specification of object-oriented meta-modelling. In: FASE 2000. LNCS, vol. 1783, pp. 193–207. Springer, Heidelberg (2000)
11. Smith, G.: The Object-Z specification language. In: Advances in Formal Methods. Kluwer Academic, Dordrecht (2000)
12. Stephens, R.: A survey of stream processing. Acta-Informatica 34(7), 491–541 (1997)

Brain Functional Network for Chewing of Gum

Ming Ke, Hui Shen, Zongtan Zhou, Xiaolin Zhou, Dewen Hu, and Xuhui Chen*

Abstract. Recent studies showed that gum-chewing induced significant increases in cerebral blood flow and blood-oxygenation level in the widespread brain regions. However, little is known about the underlying mechanism of chewing-induced regional interconnection and interaction within the brain. In this study, we investigated the human brain functional network during chewing of gum by using functional magnetic resonance imaging and complex network theory. Adjacency matrix of the network was constructed by the active voxels of chewing-related. The global statistical properties of the network revealed the brain functional network for chewing of gum had small-world effect and scale-free property. Computing the degree and betweenness which belong to the centrality indices, we found that the neocortical hubs of the network were distributed in the sense and motor cortex, and the nodes in the thalamus and lentiform nucleus held the largest betweenness. The sense and motor cortices as well as thalamus and lentiform nucleus have the important roles in dispatch and transfer information of network.

Keywords: Functional network; Gum-chewing; Small-world network; Functional magnetic resonance imaging.

Ming Ke · Xuhui Chen
College of Computer and Communication, Lanzhou University of Technology, Lanzhou, Gansu, 730050
e-mail: xhchen@lut.cn

Hui Shen · Zongtan Zhou · Dewen Hu
College of Mechatronics and Automation, National University of Defense Technology, Changsha, Hunan, 410073

Zongtan Zhou
Center for Brain and Cognitive Sciences, and Department of Psychology, Peking University, Beijing, 100871

* Corresponding author.

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 169–178.
springerlink.com

1 Introduction

Mastication is an essential physiological function of the human neural system. Some positron emission tomography (PET) study showed increased blood flow during chewing of gum, demonstrating that chewing activated widespread regions of the brain [1, 2]. Recent functional magnetic resonance imaging (fMRI) study found that activated brain regions associated with chewing include the primary sensorimotor cortex, premotor cortex, supplementary motor area, insula, thalamus, and cerebellum [3]. Using a conjunction analysis of gum chewing and sham chewing, Takada and Miyamoto found that some prefrontal and parietal cortex areas showed activity in chewing of gum, not in sham chewing. They speculated that a fronto-parietal network for mastication exists and may contribute to higher cognitive information processing [4]. A recent study reported that chewing may produce an enhancing effect on cognitive performance related to memory by using n-back tasks [5].

A number of studies have identified the widespread regions of the brain involved in chewing function. However, little is known about the underlying mechanism of chewing-induced regional interconnection and interaction within the brain. The brain is a complex dynamic system, in which complex function were organized and reshaped by transferring and integrating information between regions [6]. Recently studies have indicated that the functional and structural networks of the mammalian brain are between the regular network and the random network [7]. The brain network presents the distinctive combination of high clustering of a lattice graph and the short path length of a random graph, which characterizes the small-world property [8]. The human brain functional networks have been reported had the small-world topology during behavior and even at rest [9].

In the present study, we used the activation-detected methods and the complex network theory. After identifying the activation of the brain in the gum chewing task, functional connectivity between the pair of the significant activity voxels was estimated and the adjacency matrix of the network was constructed. The statistical properties of the chewing brain network indicated the network had small-world effect and free-scale property. The degree and betweenness centrality of network nodes reflect, to some extent, of the functional organization pattern within brain.

2 Materials and Methods

2.1 The Data Acquisition and Experiment Design

Sample. Sixty neurologically healthy college volunteers participated in this study (age range, 18-35 years; 28 females and 32 males). All subjects had normal mastication function and were right-handed. None had taking medication, abusing alcohol or illicit drugs. The subjects were instructed to minimize head movements during jaw movement, and data from participants where the heads were evaluated to have moved more than 0.75 *mm* would be discarded. Finally, the data of 38 participants were included in following data analysis.

Experiment design. Each subject performed the following two tasks: chewing gum and rest. The experiment were designed in a block manner (each block of 25*s* duration, alternated for a total scanning times of 400*s*). The scanner is in the acquisition mode for 8*s* before each series to achieve steady-state transverse magnetization(Fig. 1).

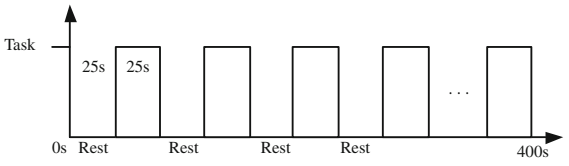


Fig. 1 Diagram of the task pattern for each participant. For the series, participants alter between 25s of rest (off) and 25s of task (on) for 400s. The scanner is in the acquisition mode for 8s before each series to achieve steady-state transverse magnetization.

fMRI data acquisition. The fMRI experiments were performed on a 3.0T Siemens Trio system in Beijing MRI Center for Brain Research. The head was positioned in a standard head-coil and fixed with cushions. All subjects were instructed to minimize head movements and follow the instruction word (chew/rest) on the project screen. Earplugs were provided to avoid auditory discomfort noises during the measurements. Prior to the fMRI examination, high-resolution sagittal T1-weighted magnetic resonance (MR) images were obtained to determine the imaging plane parallel to the AC (anterior commissures)-PC (posterior commissures) line. The functional MR image data were acquired by the gradient-echo echo-planar imaging (EPI) sequence. The EPI sequence was used with the following parameters: Repetition time (TR)=3000*ms*, Repetition time (TE)=30*ms*, Flip angle(FA)=90°, Field of view (FOV)=240*mm*, matrix=64×64, slice thickness=3.8*mm*, gap=0.2*mm*. 32 transversal slices of functional images covered the whole cortices and the cerebellum. It has been considered difficult to obtain sufficient fMRI data during jaw movement because the associated head motion created artifacts on images. To avoid these artifacts, larger voxel ($3.75\times3.75\times4mm^3$) were used, allowing some head motion of the participants. Functional images with 135 volumes were acquired for this experiment.

2.2 fMRI Data Analysis

The procession of constructing the functional network for chewing of gum and analyzing the network properties as follows: fMRI data preprocessing; acquisition of the brain activity during chewing;construction of adjacency matrix by Pearson's correlation; topology analysis of complex network.

Data preprocessing and activation-detected. All data set was preprocessed initially by using SPM2 software (Wellcome Department of Cognitive Neurology, Institute of Neurology, London, UK). T1 anatomical images were coregistered to the

mean of the functional scans and aligned to SPM T1 template. The calculated nonlinear transformation would be applied to all functional images for spatial normalization. The functional images were normalized and resampled with the voxel size of $4\times4\times4mm^3$. Subsequently, the functional images were spatially smoothed with a Gaussian kernel of $8mm$ full-width half-maximum. Normalized anatomical images of every subject was then segmented into the gray matter, white matter, and cerebrospinal fluid. Averaging all the individual gray-matter images across subjects generated a mean image to create a mask. The smoothed functional images were processed by the mask, and the voxels within the mask would be further analyzed. The intention of this process was to reserve the gray component as possible and remove the white matter, cerebrospinal fluid and skull. To minimize effects of physiological noise, a high pass filter of $80s$ and a low pass filter of $5s$ were applied within the design matrix. Moreover, the 6 movement parameters of spatial transformation obtained from motion correction were also used. Specific effects were tested by applying the general linear model to parameter estimates. Then, random-effects analysis was performed to compute group activation. Group results were obtained with the statistical threshold of $P<0.05$ corrected for multiple comparisons controlling FEW for per voxel and the activated clusters over 10 voxels.

We created a mask in terms of the t-statistical map in the result of statistical analysis within the group. In this mask image, a single voxel, which was exceed the threshold based on a $P=0.05$ (corrected) level of significance, would be set 1 instead of the original t value from that t-statistical image, while the remains would set 0. After removed the regions outside the mask, the new data were obtained by averaging the no-smoothed fMRI time series over the corresponding voxels in the brain across all the subjects. Then, the volumes were segmented to 92 regions using the anatomically labeled template image . This parcellation divided the cerebra into 90 regions (45 anatomical regions in each cerebral hemisphere); while each cerebellum hemisphere were look as a whole in this study. The anatomically labeled template was reported by Tzourio-Mazoyer, and had been used in several previous studies [10].

Construction of brain Network for gum chewing. We defined the voxel as the node in the brain network during gum chewing, and that the link between pair of nodes exist was determined by the functional connectivity between two voxels. The Pearson's correlation coefficient measures the strength of the linear association between a single variable with the other variable. Here the variable in Pearson's correlation coefficient was denoted by the activity of a voxel within time series. Correlations between the time course of pairs of voxels were computed by using Pearson's correlation coefficients method:

$$cc=\frac{\sum(r_i-\overline{r_i})(r_j-\overline{r_j})}{\sqrt{\sum(r_i-\overline{r_i})^2}\sqrt{\sum(r_j-\overline{r_j})^2}} \tag{1}$$

where r_i and r_j were the time series of i and j, respectively. We defined whether existed functional connectivity between two voxels in terms of the correlation coefficient exceeded the predetermined threshold T, regardless of their anatomical connectivity [11]. After estimated the pairwise of voxels correlations, we obtained

a result correlation matrix, in which the arrangement of elements was according to the cerebellum, subcortical nuclei, insula, the limbic lobe, the occipital lobe, the parietal lobe, the temporal lobe, central region, and the frontal lobe.

Analysis of functional network for gum chewing. The functional brain network for mastication could be described as the graph with a number of nodes or vertices, N, and a number of undirected edges connecting pairs of nodes, E. The key statistical characteristics of complex network are the degree $< k >$, the clustering coefficient C_{real}, and the characteristic path length L_{real}. The degree of the i node was the number of its connection with the rest nodes of the graph. The degree of the graph, $< k >$, was the average number of edges per node. The cluster coefficient measured the local network structure, and C_i was the fraction of the numbers of existing connections between the neighbors of the i-th node divided by the maximum possible connections. The cluster coefficient ranged from 0 to 1; larger cluster coefficient of the node implied that the neighbors of that node were also nearest neighbors of each other. The average cluster coefficient C_{real} for the whole graph was given by the sum of all cluster coefficient divided by the number of voxels. The path was expressed by the minimum number of distinct connections which linked the source node i to the target node j; and the characteristic path length L_{real} was given by the global mean of the minimum path length between any pair of nodes in the undirected graph. We also calculated the cluster coefficient C_{rand} and the characteristic path length L_{rand} in random networks with the same number of nodes and edges for comparing with the metrics of the functional network in this study. When

$$\gamma = C_{real}/C_{rand} \gg 1, \lambda = L_{real}/L_{rand} \sim 1 \tag{2}$$

the brain network in this study would be a small-world network [8].

To analyze the brain functional network for chewing of gum, we chose a series of threshold T to convert the full correlation matrix to a sparely binary graph. When the value of threshold T was increased, fewer edges existed in the network and the graph became sparse. Since small-world properties would not be estimate under too high threshold, the maximum values of threshold T in this study was not exceed 0.91 based on the circumstance that the mean degree $< k >$ of the network would be more than the log of the number of nodes ($ln(1858) = 7.53$) [8]. The degree and betweenness are the centrality indices of complex networks, and the centrality measures of network come of social networks[12], for example, the centrality of the person according to their position and status in the context of social networks. Here, the hubs of the network are the nodes with the largest degrees, connected with most of nodes in the network, and are the importance of nodes in the network. Betweenness of the i was calculated as the number of the shortest path length between pairs of nodes that passed through the i node [12]. The nodes that have biggest nodes have to display the important roles in the existence of paths between any two nodes in the network. We computed and sought the nodes that had the largest degree and the nodes with the largest betweenness. Consequently, we would find the important regions that dispatch and transfer the information of sense and cognition.

3 Results

3.1 Statistical Results of Chewing-Related Activation Pattern

Surface projections of chewing-related contrast maps are shown in Fig. 2. Significantly greater activations to the chewing task were found in many brain regions, including the primary sensorimotor cortex extending down into the insula, the bilateral premotor cortex and supplementary motor area, the parietal cortex, the templet cortex, the frontal cortex, the subcortex and cerebellum bilaterally ($P < 0.05$, corrected). Table 1 showed the coordination of significant regions.

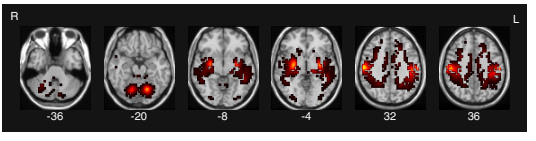


Fig. 2 Activated regions superimposed on MNI standard brain ($P < 0.05$, corrected for multiple comparisons).

Table 1 Coordination of significant regions for the gum chewing tasks.

Anatomical structure	brain hemisphere	Maximal t value	Talairach coordinates			BA area
			x	y	z	
Precentral gyrus	L	20.28	-55	-18	34	BA4
Postcentral gyrus	L	16.13	-48	-22	31	
Lentiform nucleus	L	16.03	-28	-12	-3	
Precentral gyrus	R	17.80	59	-14	30	
Postcentral gyrus	R	17.09	51	-18	34	
Lentiform nucleus	R	16.30	28	-12	-3	
Cerebellum, posterior lobe	L	16.01	-16	-67	-13	
Cerebellum, anterior lobe	R	13.35	32	-56	-27	
Cerebellum, anterior lobe	R	5.15	24	-48	-28	

Note. (L), left hemisphere; (R), right hemisphere.

3.2 Small-World Properties of Functional Network

The statistical properties of chewing-gum functional network with different correlation threshold from 0.6 to 0.91 were list in table 2. When the correlation threshold was increased, the ratio γ became monotonically increased, which indicated greater clustering at higher thresholds in the brain networks. While the ratio λ did not change greatly at the different threshold and showing values very closed to one. Our results demonstrated that the brain functional network for chewing of gum was a small-world network.

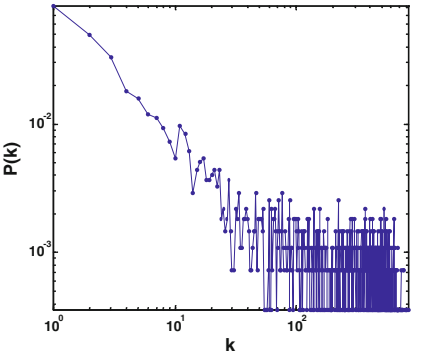


Fig. 3 The degree distribution of the adjacency matrix at a series of threshold.

Table 2 The statistical properties of chewing-gum functional network with different correlation threshold.

T	N	$< k >$	C	L	C_{rand}	L_{rand}	γ	λ
0.6	1841	695.4338	0.75229	1.7941	0.3745	1.6246	2.0088	1.1043
0.7	1760	341.1593	0.64829	2.2622	0.18377	1.8153	3.5278	1.2462
0.8	1389	98.8676	0.44596	2.7179	0.053186	1.9506	8.3850	1.3934
0.9	643	12.1313	0.16958	4.1787	0.0058424	3.2909	29.0263	1.2698
0.91	546	9.2971	0.13954	4.5536	0.0049672	3.622	28.0923	1.2572

Note. T denotes the threshold of correlation coefficient; N denotes the number of nodes that have at least one degree in the network; $< k >$ denotes the average number of edges per node; C denotes the mean clustering coefficient; L denotes the mean path length; C_{rand} denotes the mean clustering coefficient in random network; L_{rand} denotes the mean path length in random network; $\gamma = C_{real}/C_{rand}$, $\lambda = L_{real}/L_{rand}$.

Increased the threshold T, this functional network became spare with fewer edges survived as well as the decreased mean degree of the network. Fig. 3 showed the degree distributions of the chewing-gum functional network at the series of thresholds. The skewed distributions of links with a heavy-tailed were displayed.

To find out the significant nodes in functional network for chewing of gum, we ordered the degree and betweenness of the nodes in this functional network from big to small. The top hundred nodes of highest degrees were corresponding to the left precentral gyrus(BA4 and BA6, 27 nodes), right precentral gyrus (BA4 and BA6, 44 nodes), left postcentral gyrus (BA2,BA3 and BA43, 14 nodes), right postcentral gyrus (BA2,BA3 and BA43, 12 nodes), left insula (3 nodes). The top hundred nodes of largest betweenness centralities were location in the cortex regions, including the subcortex (lentiform nucleus, 14 nodes; thalamus, 6 nodes; insula, 6 nodes; total 26 nodes), bilateral precentral gyrus (left, 9 nodes; right, 14 nodes; total 23 nodes), bilateral postcentral gyrus (left, 10 nodes; right, 11 nodes; total 21 nodes), bilateral inferior parietal lobule (left, 4 nodes; right, 5 nodes; total 9 nodes), bilateral superior

temporal gyrus and transverse temporal gyrus (left, 7 nodes; right, 6 nodes; total 13 nodes), right cingulate gyrus (2 nodes), bilateral cerebellum (left, 3 nodes; right, 2 nodes; total 5 nodes), right inferior frontal gyrus (1 nodes).

4 Discussion and Conclusion

We found the brain activity related to mastication in the precentral gyrus, postcentral gyrus, frontal lobe, temporal lobe, parietal lobe, the subcortex and cerebellum by using the detecting-activated measure. Our results were in keeping with the previous PET [1, 2] and fMRI [3, 4]studies. Then we used the complex network method to analyze the functional network of chewing and analyzed the topology features. The results showed that the cluster coefficient remain one order of magnitude larger than C_{rand} and the characteristic path length was similar with L_{rand} of random network. This study was consistent with the previous reports about anatomical networks and functional networks[9, 7], and the results indicated that the functional network for chewing of gum was a small-world network, in which the topology is highly clustered and short path length. Furthermore, this network was a scale-free network, in which the degree distribution followed a power law. We also found that The nodes which had larger degree had larger clustering and had mean short path closed to three. These results indicated the functional network could transmit sense and cognitive information much more quickly and effectively between the regions.

After computed and sorted the degree and betweenness, we found the nodes in motor and sense cortex had largest degrees and connected widely to many nodes in other regions, and these nodes were the hubs in the network. The previous study reported that chewing of moderately hard gum leaded to a significant change in blood oxygenation level-dependent signals in the motor and sense cortex compared with the chewing of hard gum [3]. Another study showed the signal increases in motor and sense cortex were age-dependent [13]. These studies indicated that the motor and sense cortex had an important status in the brain functional network for mastication. They could distribute the resource of information rapidly and have wide influence. We also found the nodes which had the largest betweenness, located in the thalamus and lentiform nucleus, following in motor and sense cortex. It is indicated that these brain regions were in the pivotal position during communications between nodes in the brain functional network. The previous studies already confirmed that thalamus receives the projection from the cortex, and projects to the cortex or the cerebellum [14]. The lentiform nucleus is involved in corpus striatum, which is part of motor integration center, receiving inputs from the cortex and thalamus and connecting widely with the reticular formation and red nucleus [15]. Thus, it is reasonable to say that those nodes in subcortex play indispensable role in the shortest path from one community (brain region) to other community (brain region). These results indicated adequately that the motor and sense cortex as well as thalamus and lentiform nucleus are the important brain regions in transferring and dispatching the information of the brain functional network.

In this study, the human brain during chewing gum emerged a sparse, scale-free small-world functional network. The highly connected hubs distributed in sense and motor cortices; whilst the pivotal betweenness distributed in the thalamus and lentiform nucleus, following in motor and sense cortices.

Acknowledgment

The work was partially supported by the National Science Foundation of China (61065007, 61003202, 60835005, 90820304), and the National Science Foundation of Gansu of China (0916RJZA020).

References

1. Momose, I., Nishikawa, J., Watanabe, T., Sasaki, Y., Senda, M., Kubota, K., Sato, Y., Funakoshi, M., Minakuchi, S.: Effector of mastication on regional cerebral blood flow in humans examined by positron-emission tomography with 15O-labelled water and magnetic resonance imaging. Arch. Oral. Biol. 42(1), 57–61 (1997)
2. Kubota, K., Momose, T., Abe, A., Narita, N., Ohtomo, K., Minaguchi, S., Funakoshi, M., Sasaki, Y., Kojima, Y.: Nuclear medical PET-study in the causal relationship between mastication and brain function in human evolutionary and developmental processes. Ann. Anat. 185, 565–569 (2003)
3. Onozuka, M., Fujita, M., Watanabe, K., Hirano, Y., Niwa, M., Nishiyama, K., Saito, S.: Mapping Brain Region Activity during Chewing: A Functional Magnetic Resonance Imaging Study. J. Dent. Res. 81(11), 743–746 (2002)
4. Takada, T., Miyamoto, T.: A fronto-parietal network for chewing of gum: a study on human participants with functional magnetic resonance imaging. Neurosciece 360, 137–140 (2004)
5. Hirano, Y., Obata, T., Kashikura, K., Nonaka, H., Tachibana, A., Ikehira, H., Onozuka, M.: Effects of chewing in working memory processing. Neurosci. Lett. 436(2), 189–192 (2008)
6. Sporns, O., Chialvo, D.R., Kaiser, M., Hilgetag, C.C.: Organization, development and function of complex brain networks. Trends Cogn. Sci. 8, 418–425 (2004)
7. Salvador, R., Suckling, J., Schwarzbauer, C., Bullmore, E.: Undirected graphs of frequency-dependent functional connectivity in whole brain networks. Phil. Trans. R. Soc. B 360, 937–946 (2005)
8. Watts, D.J., Strogatz, S.H.: Collective dynamics of ”small-world” network”. Nature 393, 440–442 (1998)
9. Achard, S., Salvador, R., Whitcher, B., Suckling, J., Bullmore, E.D.: A Resilient, Low-Frequency,Small-World Human Brain Functional Network with Highly Connected Associaition Cortical Hubs. The Journal of Neursoscience 26, 63–72 (2006)
10. Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. NeuroImage 15, 273–289 (2002)
11. Eguíluz, V.M., Chialvo, D.R., Cecchi, G.A., Baliki, M., Apkarian, A.V.: Scale-free brain functional networks. Physical Review Letters 94, 018102 (2005)

12. Newman, M.E.J.: A measure of betweenness centrality based on random walks. Social Networks 27, 39–54 (2005)
13. Onozuka, M., Fujita, M., Watanabe, K., Hirano, Y., Niwa, M., Nishiyama, K., Saito, S.: Ages-related changes in brain regional activity during Chewing: A Functional Magnetic Resonance Imaging Study. J. Dent. Res. 82, 657–660 (2003)
14. Fallon, J.H., Opole, I.O., Potkin, S.G.: The neuroanatomy of schizophrenia: circuitry and neurotransmitter systems. J. Clinical Neuroscience Research 3, 77–107 (2003)
15. Kandel, E.R., Schwartz, J.H., Jessell, T.M.: Principles of neural science, 4th edn., New York, vol. 3, pp. 77–107 (2003)

Effects of Value-Based Mechanism in Online Advertisement Auction

Yosuke Motoki, Satoshi Takahashi, Yoshihito Saito, and Tokuro Matsuo

Abstract. In recent years, the listing service is widely used in search site such as Yahoo!, Google, and MSN. In the service, advertising fee and advertising order are decided by the auction that is called Generalized Second Price Auction (GSP) and the auction is actually employed in a lot of search service sites. There are a lot of researches on GSP in order to analyze and clarify its feature and advantages. However, in those researches, the advertisement is mutually independent. Additionally, the value of advertisement is not considered. In this paper, we propose a new mechanism based on GSP that is used in advertisement auctions. Each advertisement has some value, because users click the advertisement when it may be useful for them. We analyze the auctioneer's profit in comparison between normal GSP, normal VCG (Vickrey-Clarke-Groves Mechanism) and our proposed mechanism. The contribution of our research includes to clarify the features and advantages of advertisement auctions and effects to website owner's profit rate.

1 Introduction

Agent-based electronic commerce is one of promising techniques to enhance effectiveness and performance of trading. In this paper, we give an analysis of agent-based advertisement auction, which is displayed on a webpage.

Advertisements on the web pages provide good opportunity to get new customers. In recent years, a lot of web pages providing a search service have advertisements, which are related with searched word by user. Paying some money to the search engine company, the company normally has a space to show their advertisements. As same as items trading in the Internet auctions, a displayed advertisement

Yosuke Motoki · Yoshihito Saito · Tokuro Matsuo
Graduate School of Science and Engineering, Yamagata University, Yonezawa, Japan

Satoshi Takahashi
Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba, Japan

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 179–189.
springerlink.com

on web page is also based on the auction, called the online advertisement auction. Online advertisement auction is employed in some search engines such as Yahoo! and Google[1][2]. When users search for some words on the search engine, an advertisement related with the searched keywords is displayed with result of search [3]. The order of advertisements to be displayed is determined based on bid value in an auction. Advertisement owners can set up the interval and period to display the advertisement as a time slot. The payment amount is determined based on the Generalized Second Price Auction, which is known higher revenues than the Generalized Vickrey Auction [4]. Winner in the auction gets a space to display their advertisement and the web page owner allocates time and position in the web page to show the advertisement. There are a lot of contributions about GSP(Generalized Second Price Auction) researches in electronic commerce research. In this auction, bidding and winner determination are conducted multiple time. Advertiser agent can change his/her bid value because the auction is continued with repetition. When agents try to bid in an auction, they bid on their strategy. However, GSP has an envy free equilibrium and webpage owner providing advertisement space can get larger benefit compared with VCG (Vickrey-Clark-Groves) Mechanism.

Figure 1 shows the interface of searched result of ”travel” in Google. In right side of the interface, advertisements are shown. These advertisement is related with the searched words. When the advertisement is clicked by end-users, advertisement fee is paid from advertisement companies to the website owner (Google). Generally, possibility of click is high order of display. This means that the advertisement fee of top-displayed advertisement is more expensive than lower advertisements. Google earned about 520 USD by this advertisement system in 2008.

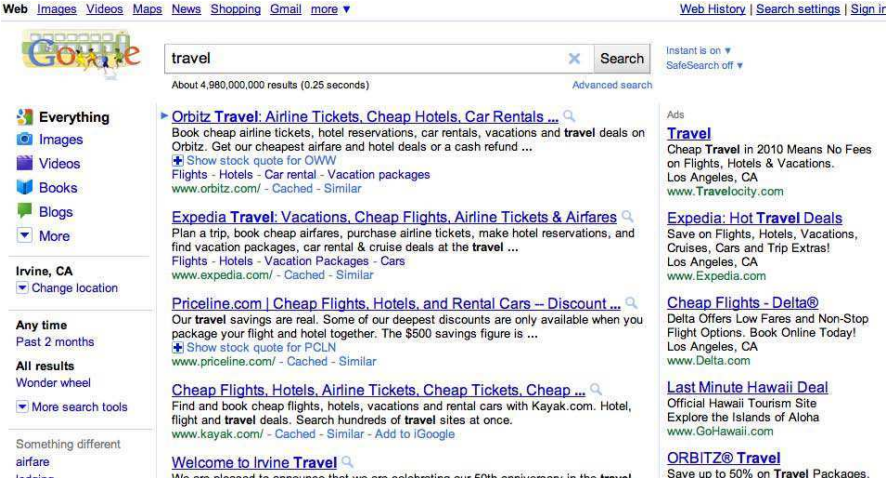


Fig. 1 Google Adwords

In existing research, the value of advertisement is assumed as independent with each other. Otherwise, some of their researches do not refer the value of the advertisement. However, each advertisement has a certain value for users. When same or similar item is soled in two e-commerce sites, the price on the advertisement is different from another one. If buyer considers the price is important attribute to choose item, the advertisement selling items at low price has more value for the buyer. For example, a shop A gives an advertisement to sell an item for \$100. When a shop B gives the advertisement to sell the same item for a shop \$80, its value of the advertisement is higher than shop A's value if the condition of item and other situations between shop A and B. In this paper, we focus on such situation and simulate the revenue of advertisement owner. Also, we analyze a result of simulation of online advertisement auction with relationship between value of each advertisement.

The rest of this paper consists of the following three parts. In Section 2, we show preliminaries on several terms and concepts of auctions. In Section 3, we propose our value-based GSP and simulate in some conditions. Finally, we present our concluding remarks and future work.

2 Preliminaries

In this section, we introduce GSP protocol and VCG mechanism employed by Google Adwords auction. The GSP protocol is used by many company such as Google Adwords, Yahoo!, and mixi. The mixi is one of social networking service (SNS) companies in Japan, who has most largest share in Japan. There is a few difference of protocol between each company, however, this paper shows Google Adwords's protocol. Suppose that there are n agents as advertisers and k slots. A slot is a place of advertisement on a web page. Let c_i be a click-through-count of advertisement placed on the slot i. We employ a following rule about each c_i.

$$c_{i-1} \geq c_i,\ 2 \leq i \leq k$$

This rule means that a click-through count of slot i is fewer than the slot $i-1$ for $2 \leq i \leq k$. When an agent bids b as cost per click to use a slot i, a payment of the agent is defined by $b \cdot c_i$.

We assume every following auction satisfies Nash equilibrium. The Nash equilibrium shows that a strategy $S_A^\star$ is a best strategy for agent A if every agent without agent A chooses an optimal strategy $S^\star$.

We review some proposed auction protocol or mechanism of advertise auctions.

2.1 Vickrey Auction

Vickrey auction is an auction protocol which deals single item as same as second price sealed bid auction. In this protocol, every agent bids own value for an auctioneer agent, which their bids do not be opened. A winner of the auction is the highest valued bidder, and he/she pays a second highest value of the auction. The Vickrey

auction has week dominant strategy in which every agent bids own truth value. It is well known that the English and Dutch auction has also the same week dominant strategy[5].

2.2 VCG (Vickrey-Clark-Groves) Mechanism

VCG mechanism is generalized from Vickrey auction, which has dominant strategy as truthful bidding. Each agent j bids own value per click for auctioneer. The auctioneer allocates a slot for the agent by descending order of bids. Suppose b_i is a value per click of the agent allocated a slot i, we define a payment per click p of the agent as follows,

$$p = \sum_{j=1}^{k+1} b_j - \sum_{j=1}^{k} b_j - 2 \cdot b_i.$$

VCG mechanism satisfies incentive compatibility and Pareto efficiency. The Incentive compatibility (Strategyproofness) means that each agent (bidder) choice an optimal strategy without influence of other agents. The Pareto efficiency means a total utilities of each agent and auctioneer[4].

We show an example, suppose that there are two slots and three agents. Agents 1, 2 and 3 bids \$300, \$200 and \$100 per click, respectively. In this case, an auctioneer allocates slot 1 and 2 to agent 1 and 2, and agent 1 and 2 pays \$100 and \$100 per click, respectively. Also, the auctioneer's gain is $(\$100 + \$100) = \$200$.

2.3 GFP (Generalized First Price Auction) Protocol

GFP protocol had been employed by Overture (Yahoo! Searching Marketing) before GSP proposed. This protocol is nearly single item first price auction, that is an agent who is a winner of the auction pays own value. The GFP protocol has dominant strategy for each agent. This protocol gives a highest utility for an agent, when the agent bids a lowest value he/she win. We consider an auction of one slot A and two agents. If agent a and b bids \$300 and \$200, respectively, then the agent a get a slot A. However, if the agent a bids \$201, then also the agent wins. Therefore, GFP protocol has an incentive that every agent try to decrease own value. This means that the more increasing a number of agents, the more decreasing an auctioneer's gain.

2.4 GSP (Generalized Second Price Auction) Protocol

GSP protocol is an auction protocol which is natural extended form second price auction. The auctioneer sorts all bided values by descending order, and allocates slot i to i-th highest valued agent for all slots. The agent who is allocated slot i pays b_{i+1} per click for the auctioneer.

It is known that GSP protocol does not satisfies incentive compatibility. Therefore, the truthful bidding is not dominant strategy in GSP. On the other hands, GSP converges on Locally Envy Free equilibrium[6]. The auction is Locally Envy Free

equilibrium, if an agent who gets a slot i does not increase own utility neither getting a slot $i-1$ nor getting a slot $i+1$. Hence, the slot i is an optimal position which maximizes the agent's utility.

We consider the same example in VCG. Suppose that there are three agents and two slots, and agent 1, 2 and 3 bids \$300, \$200 and \$100 per click. In this case, the agent 1 and 2 gets the slot 1 and 2, and pays \$200 and \$100 per click, respectively. The gain of auctioneer is $\$200+\$100=\$300$. Therefore, the GSP protocol is better than VCG mechanism in the advertise auction, since the auctioneer gains \$200 on the VCG mechanism. Note that if there is one slot, then the result of auction is the same on both GSP protocol and VCG mechanism.

2.5 Google Adwords

Google Adwords is an auction protocol similar to GSP protocol. Google Adwords employs CTR (Click-Through-Rate) and QS (Quality-Score). CTR is a ratio of click denoted by

$$\text{CTR} = \frac{\text{Click-through-count of an advertisement}}{\text{Number of page view of an advertisement}}$$

Quality score is decided by Google from CTR and relationship between text of the advertisement and searching keyword. Also, Google sets a minimum bidding value. An allocation of slots are based on descending order of multiplying the value by quality score, called *evaluation score*. It means that if high quality score has a possible to get a good position of slot by cheap payment. Google requires all advertisements positioned on upper slots must have a certain quality score level. Let q be a quality score of an agent allocated on a slot i, and b_i $(b_1 > b_2 > \cdots > b_i > \cdots > b_k)$ be a evaluation score. A payment p per click is denoted by

$$p = \frac{b_{i+1}}{q} + 1$$

It is known that CTR is proportional to order of slots. N. Brooks[7] say that there is a strong correlation between CTR and order of slots. The report also shows the ratios of CTR when a first ordered CTR is 100%. in this result, a second ordered is 77.4%, and third is 66.6%. However, Google suggests there is an exception. For example, some famous companies positioned lower slots has larger CTR than some upper positioned companies, since the famous companies get many click-through-counts even lower position.

On the other hand, Sponsored search which is derived by Yahoo! Search Marketing has technique similar to Google Adwords, but, there is a difference that order of slots is descending order of only bidding value.

We consider the same example in VCG. Suppose that there are three agents and two slots, and agent 1, 2 and 3 bids \$300, \$200 and \$100 per click. Also agent 1, 2 and 3's quality score is 2, 1.5 and 1, respectively. In this case, the evaluate

scores are 600, 300 and 100, respectively. The agent 1 and 2 gets the slot 1 and 2, and pays $\$300/2 = \$150$ and $\$100/1.5 = \$66$ per click. The gain of auctioneer is $\$150 + \$66 = \$216$.

2.6 Proposed Mechanism

Each advertisement has a co-dependent value and it is expressed by linear to be evaluated. When a value of company i's advertisement changes A_{after} from A_{before}, co-dependent value of other advertisement with company i is shown B_{before} and B_{after}. B_{before} is changed a value effected by i's advertisement to B_{after}.

$$B_{after} = B_{before} + \alpha(A_{after} - A_{before})$$

The condition of the above equation is given as $0 \leq \alpha \leq 1$. Quality score in the GSP auction protocol used in Google Adwords is placed a value in which we have defined above definition in the simulation . Figure 2 is an example of the model of our proposed mechanism. When end-user clicks the link of the advertisement, its value is increased. Relatively, other advertisement's value becomes going down. When low-ranked advertisement is clicked by users, the advertisement is regarded as valuable comparing with high-ranked advertisement. In the Figure 2, we assume all of agents participate to bid for the first time. After bidding, the winners are

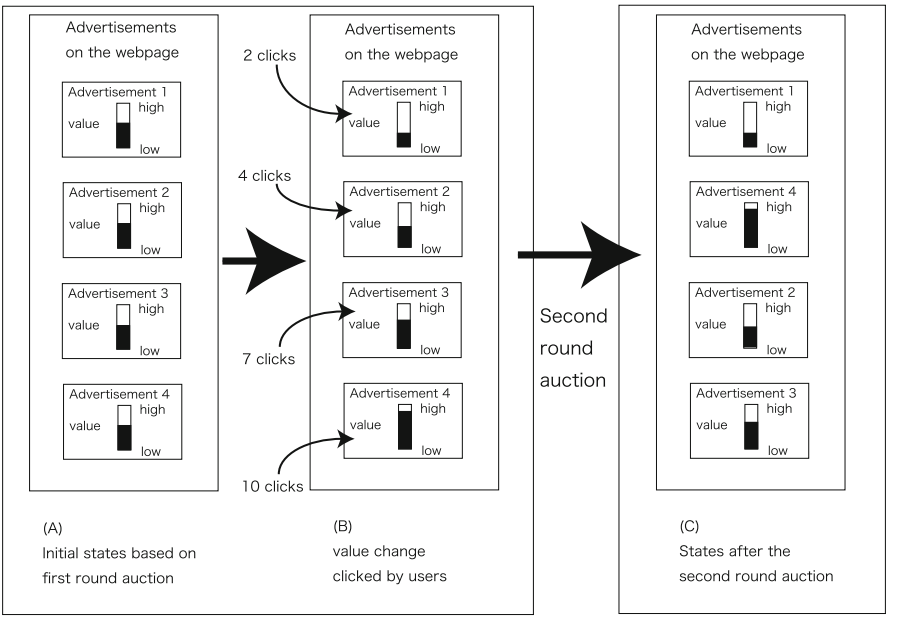


Fig. 2 Concept of the proposed mechanism

determined be the auction. Then, each advertisement is displayed at the website as (A). Users click the advantages and the value of each advertisement changes based on number of click as (B). After one period passes, agents bid at second round auction to keep their advertisement in the website. We also assume all agent bid same price comparing with first round auction. The order of advertisement is changed based on both bid price and advertisement's value. In this case, although advertisement 1's value decreased in (B), position of advertisement 1 is kept at top because bid price is very high as (C). Because advertisement 4's value is quite high in (B), the rank of advertisement in (C) becomes second although bid price is the lowest in other three agents.

3 Simulation

3.1 Condition

We set 3-10 slots to be put advertisements and 10-50 agents (companies to join in the auction) who bid to get a space for their advertisement. The lowest bid price in the auction is set $10 and agent's bid value is defined a uniform distribution between $10 and $100. Initial value of each advertisement is defined on a uniform distribution between 0.2 and 2.2. Number of clicking by end-user is assumed on a uniform distribution between 1 and 100 in a time slot.

3.2 Procedure of Trade

The following is procedures to simulate.

1. Web page owner decides number of slot for advertisement.
2. Number of click in a period is decided.
3. Bid value for each agent and advertisement value are decided.
4. Each slot is allocated based on large order of a valuation that is multiplied by bid price and value of advertisement.
5. Payment amount and benefit of each agent are calculated.
6. Value of advertisement of a certain agent is changed.
7. New advertisement's value is computed based on above change.
8. Procedure (4) and (5) are conducted based on new advertisement's value and bid price.

The simulation is conducted 100,000 trials.

3.3 Results

Table 1 shows result of simulation in which value of advertisement is changed. There are 20 advertiser agents and value of a certain advertisement is reduced and it effects other values of advertisement. When number of slot is changed from 4 to 10

and a value of one advertisement is reduced, 54,000 auctions make whole profit in the market increase in 100,000 trial. Average of the increased profit is $21.38. We discuss result of simulation from Table 1.

Table 1 A value of one advertisement is reduced.

Number of slot	Increase (%)	Decrease (%)	Average of increased /decreased profit
4	54.1	45.9	$22.38
6	55.3	44.7	$25.92
8	55.7	44.3	$27.56
10	56.3	43.7	$30.22

1. Averages of profit is normally increased and the profit increases when number of slots increases.
2. Possibility of profit increase is increased when number of slot increases.

This feature is apparent because the curve in Table 1 is monotonic increase.

As same as the above, table 2 shows the case where 20 agents join in the auction and value of one agent's advertisement is increased. The number of slot is changed from 4 to 10 in each trial. We discuss result of simulation from Table 2.

Table 2 A value of one advertisement is increased.

Number of slot	Increase (%)	Decrease (%)	Average of increased /decreased profit
4	43.8	56.2	-$32.85
6	43.0	57.0	-$36.63
8	42.2	57.8	-$38.81
10	42.0	58.0	-$40.31

1. Averages of profit is normally decreased and the profit decreases when number of slots increases.
2. Possibility of profit increase is decreased when number of slot increases.

This feature is also apparent because the curve in Table 1 is monotonic decrease.

Table 3 is a result where number of slot is fixed as 5 and one agent changes value of his/her advertisement. The number of agent is changed 10 to 50 in each trial. Rate of increase/decrease of value of advertisement is assumed by uniform distribution. The result shows a comparison of profits between non-affective and affective.

1. Averages of profit is normally decreased and the profit decreases when number of slots increases.
2. Possibility of profit increase is decreased when number of slot increases.

Table 3 Number of slot is fixed as 5.

Number of agents	Increase (%)	Decrease (%)	Average of increased /decreased profit
10	49.8	50.2	-$2.81
20	49.1	50.9	-$4.87
30	48.7	51.3	-$5.17
40	48.3	51.7	-$6.27
50	48.1	51.9	-$6.85

Average of profit is negative because possibility that the profit decreases is large. The figure 3 shows the graphical result of above simulation. This shows the monotonic decrease of webpage owner's profit.

From above simulation and analysis, we find out the following features. First, total profit of webpage owner reduces when each advertisement has co-dependence between its value. Second, when the size of auction becomes large, average of profit is decreased.

3.4 Comparison to VCG

Table 4 shows the result of simulation when the number of agents is 20 and number of slots are changed from 4 to 10 in each trial. When number of agent increases, our proposed GSP mechanism makes large profit comparing with general VCG mechanism.

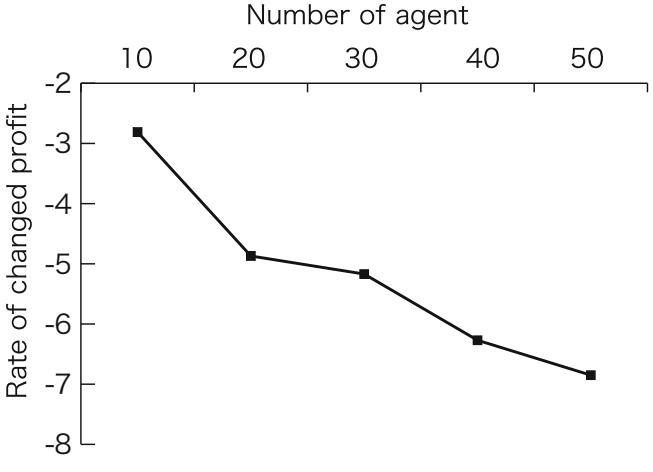


Fig. 3 Result of simulation 1

Table 4 A value of one advertisement is increased.

Number of slot	Increase (%)	Decrease (%)
4	80.3	19.7
6	83.6	16.4
8	83.7	16.3
10	83.8	16.2

Table 5 Number of slot is fixed as 5.

Number of agents	Increase (%)	Decrease (%)
5	51.6	48.4
10	62.3	37.7
20	82.9	17.1
30	89.5	10.5
40	92.8	7.2
50	94.5	5.5

Table 5 shows the result of simulation when the number of slot is fixed as 5 in comparison between our proposed GSP and general VCG mechanism. Our proposed GSP makes larger profit compared with normal VCG with monotonic increase when the number of agents increases. When number of agents is not many, the increase rate is high. After number of agents is 30, increase rate becomes less and it seems to become convergence.

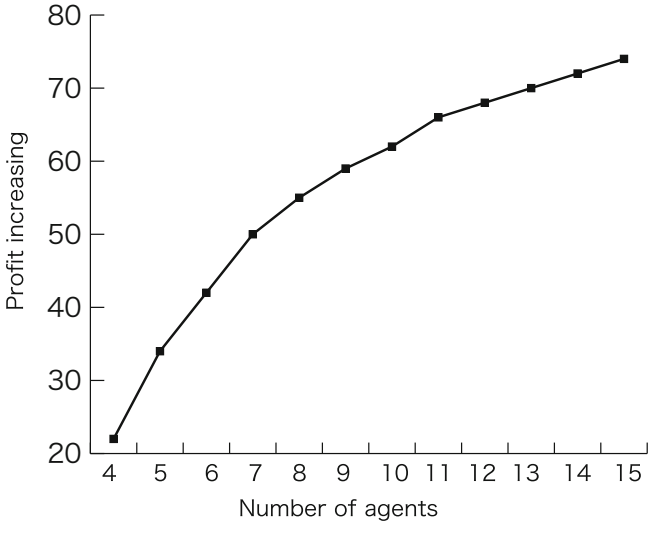


Fig. 4 Result of simulation 2

To analyze more special case, we try a simulation when the number of slots are fixed as 3. The figure 4 shows the result of simulation, which is compared total profit between VCG and our mechanism. We find out the following two features from the simulation. First, when number of agent increases, our GSP provides larger profit than VCG. Second, rate of increase becomes small when number of agents decreases.

4 Conclusion

In this paper, we proposed value-based GSP mechanism in advertisement auctions based on multi-agents. Our analysis shows that total profit changes in different auctions mechanism GSP, VCG, and our proposed mechanism. Particularly, from the analysis, auctioneer changes the auction protocol based on his/her estimate profit. However, our auction protocol has an advantage where the website provide more useful advertisement for users because the order of allocation is based on both price and value. Our future work includes the analysis of profit and expected utility for agents in the mixed type of normal GSP, VCG, and our protocol.

References

1. http://www.yahoo.com
2. http://www.google.com
3. Edelman, B., Ostrivsky, M., Schwarz, M.: Internet Advertising and the Generalized Second-Price Auction Selling of Dollars Worth of Keywords. The American Economic Review (2005)
4. Vickrey, W.: Couterspeculation, Auctions, and Competitive Sealed Tenders. Journal of Finance (1961)
5. Krishna, V.: Auction Theory. Academic Press, London (2002)
6. Edelman, B., Ostrovsky, M., Schwarz, M.: Internet Advertising and the Generalized Second Price Auction: Selling Billions of Dollars Worth of Keywords. American Economic Review 9(1), 242–259 (2007)
7. Brooks, N.: The Atlas Rank Report: How Search Engine Rank Impact Traffic. Insights, Atlas Institute Digital Marketing (2004)

Research on Dynamic Optimized Approach of Value Chain in Tourist Destinations*

Li Yunpeng, Xie Yongqiu, Ni Min, Hao Yu, and Qi Lina

Abstract. Up to the present, the research on the value chain of tourism business is simply focused on the level of business organization or in the industry. Only a small part of the analysis and exploration is on the value chain of tourism destination. Yet, the destination works as a comprehensive system of value actualization in the tourism industry. Therefore, the research on the relevant issues is of great theoretical and practical significance for improving the management and marketing of tourism enterprises, further, greatly improving the industry's operation efficiency and business achievement. This dissertation, through introducing the conception of SOA framework, has studied the construction technology of soft value chain platform which involves tourism enterprises, the tourism industrial administrative institutes in destinations and ordinary tourists. The platform is compatible to various available tourism information systems and can be seamlessly connected to the information systems in a dynamic way. Consequently, a successively optimized operating mechanism of value chain and a management, service and marketing system can be constructed. As a theoretical creation, the research explores the coordinating pattern of value chain and has constructed the

Li Yunpeng
School of Government, Beijing University, Beijing, 100871
College of Business Administration, Capital University of Economics and Business, Beijing, 100070

Xie Yongqiu · Hao Yu
College of Business Administration, Capital University of Economics and Business, Beijing, 100070

Ni Min
Beijing Yongyou Software Company, Beijing, 100094

Qi Lina
College of Urban and Environmental Sciences, Beijing University, Beijing, 100871

* Funded by Qualified Teacher Enhanced Plan of Beijing Municipal Education Commission, 2010.

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 191–199.
springerlink.com

model of the value chain platform in tourism destinations and has proposed relevant theories. That is supporting the destination value chain operate effectively with dynamic integrated technology of tourism information system construction; the data searching and analyzing approaches to efficiently process the feedbacks from data users ; studying the satisfaction level of tourism guests.

Keywords: Tourism destination; Value chain; Dynamic optimization; SOA； Data mining.

1 Introduction

With the brisk growth of market demand from tourists and the constant expansion of tourism economy, tourism industry is ceaselessly upgraded to a newer level. The value chain of tourism industry has greatly promoted tourism industry, which made the exploration on the tourism value chain become increasingly active and intense than before. Due to the instability of individual travelling stages and other factors, the problem of how to manage tourism chain value arises. This dissertation intends to explore the integrated management of tourism value chain with SOA technology, which aims at contributing new research methodology and thinking approaches to the issue of how to manage the value chain in China's tourism industry.

2 Research Background

The purpose of value chain management is to achieve competition advantages for business organizations by analyzing the value increase in individual business links. The theory holds that the impact of specific operational activities within an enterprise, such as product design, production, research and development, marketing, transport and other independent areas, is the key for ultimately promoting competition advantages. Usually, a value chain performs and produces effect within an industrial section, especially an enterprise and tourism industry dose not possess the internal value chain process as most other ordinary industries. Zhang jie(2005)concluded that relatively less research has been conducted on the value chain of tourism industry (or product) [1]. Yilmaz(2006)Pointed that there is no attempt in the tourism management literature proposing frameworks or models, which can assist the tourism companies, evaluate and control the overall tourism value chain [2].

With traditional operation pattern of tourism value chain, a tourist product supplier (tourist resorts, hotels, etc.) transmits relevant information to tourist product intermediates (travel agencies) and intermediates designs tourist products (travel routes) by integrating various tourism resources and then transmits the designed products to traveling demanders. In such a situation the product supplier cannot transmit first hand and complete information about the tourist product to tourism demanders. Within the value chain, once a certain link is broken and the

regular operation blocked, the operational efficiency of the complete chain will be reduced, even, the normal operation of the chain cannot remain as it does.

Meanwhile, due to the unsymmetrical information supply, the chain members cannot collaborate with each other smoothly. The stiffened value chain makes the chain members focus their emphasis only on the value and interests of their own and tends to neglect the value and interests of the value chain as a whole. Besides, tourism consumers do not know adequately about the tourism enterprise and so does the enterprise about consumers. The value of the tourists cannot be actualized. Therefore, the operational cost of value chain under this pattern is high, but the efficiency is low.

Thirdly, the value chain under this pattern is easily controlled by intermediates. Major intermediates in tourism business circle take the advantage of their developed selling channel and the advantage of large scale purchase. At the same time they exploit the supplier's drawbacks of enabling to communicate directly with the tourists so as to control the complete value chain. [3].

The ultimate need of a tourist is to achieve an enjoyable traveling experience of high quality service, but with limited time and money cost. However, with the present operational pattern and work efficiency, the value chain is greatly difficult to offer a high satisfaction to the tourists. This problem brings about a bottleneck to tourist service enterprises, including tourist service websites in their regular business and places obstacles for the deep level development of tourism industry. The research needs to be conducted is how to effectively incorporate different parts of a tour destination and build a tourism industrial value chain in a dynamic way. In providing a one-stop service to the travelers the concerned business circle reaches the target of utilizing the destination resources with high efficiency.

In this consideration, the relevant researchers put forth an important research theme of great practical significance, which is ***how to optimize the value chain in a tourist destination*.** Most of the research on the value chain in tourism industry is focused on that of tourist service enterprises and the industry proper, and less is the analysis and study on the value chain in destinations. But point concerning this topic is a tourist destination serves as a "resources collection" to achieve the value of the complete industry in a particular region. So the research on the value chain of a tourist destination is greatly significant for improving the management and marketing of the local business organizations in the area.

3 Research Achievement and How the Issue Is Proposed

Having analyzed the respective functions performed by tourism administration institutes, tourist service intermediates and the service suppliers in the marketing process in a particular tourist destination, Miss Wang guixia and other researchers makes an exploration on the principal marketing pattern of the domestic tourist destination in the E-commerce era [4]; Zhang mengcai and other researchers have studied the value chain transferring process of the business entity in tourism industry under the condition of E-commerce[5]. Liu renhuai proposed the

measures of consolidating tourism value chain and explored on the management of tourism value chain[6]; Liu chaohua applied the analysis framework of ***target—object—structure—function*** to the research on the regional integrating function of a destination's marketing system[7]; Zou rong, from the perspective of information service, made the analysis on how to construct Network marketing system in the tourist destination[8] ; Lu ke (in 2006) proposed that, in a new situation, the target of reconstructing the model of a tourism value chain should be ***redefining the coral nodes of a value chain.*** He also established the model of tourism value chain which takes tourist sites as the core [9].

Although E-commerce technology has optimized the traditional tourism value chain, each of the links in the tourism value chain works in its own way, which brings numerous troubles to the guest travelers. Consequently, the optimized buying on the Network cannot be achieved, and in the situation of which a series of problems arises, such as disordered management in destinations, etc. The six major elements of tourism industry, diet, accommodation, transport, shopping and entertainment depends that it is actually an industry of resources sharing. Obviously, a tourism value chain in which each link does in its own way is not a desirable one[10]. Nowadays, more and more travelers are keen on enjoying personalized traveling experience, which cannot be offered in the present value chain of tourism industry.

Some scholars have detected these problems and further put forth that an alliance should be set up among value chain members. They gradually perceived the importance of the alliance for value chain integration and its dynamic optimization. (Palmer et al, 1995) They believe that the following three reasons make it necessary to establish marketing alliances in tourism destinations. First, the resources of an individual stakeholder is very limited and his marketing behavior cannot produce adequate influence on potential traveling guests; Secondly, the present market mechanism cannot ensure that all the concerned stakeholders would support the collective marketing in a particular destination and share the marketing achievements. The third reason is that in the process of marketing planning the stakeholders could be aware of that the inter-dependence relationship is more helpful for fulfilling the target of each [11]. (Wang et al, 2006) analyzed the issue of how to construct the marketing alliance in destinations in three-dimensions of economy, society and environment. He believes that crisis, competition, organizational support and technological support are the preconditions for forming the alliance. When examining the purpose for which the stakeholders join the marketing alliance the scholars found out the stakeholders have five purposes as follows: purposes in strategy selection, transaction cost measuring, learning, competition power accumulation and community responsibilities [12]. Chen et al, 2005 also have similar research achievements. They believe the major impetus for different enterprises to establish marketing alliance among themselves in tourism industry is to ***diversify promotion channel, reduce operational, cost consolidate one's own market standing*** and ***improve the company's business achievements.*** [13]

The present research achievements on value chain integration have laid a foundation for this project, but further research on how to complete the integration has not yet been conducted. However, the real problem is that a number of defects still exist in the present information system between different enterprises. Popularly, the typical problems are as follows: Some enterprises are isolated to each other and their information systems are not compatible to those of others, so that they cannot communicate smoothly and extensively with other companies. Information cannot be exchanged directly between them as well.

So information cannot be directly exchanged between these business organizations. If a new uniformed structural system is constructed excessive financial input will have to be made unnecessarily. Therefore, the research on how to construct an optimized value chain which is integrated in a dynamic way with relatively less input is of great theoretical significance and will produce some practical usefulness. Hence, the research objective of this project is to enable travelers freely select their own traveling package products in a uniformed information platform so as to achieve the dynamic optimization of the value chain.

4 Research Objective and Approaches

A. Research objective

Specifically, the research objectives are constructing tourist destination value chain system, offering quality services to travelers and enabling destination tourism administration, tourist service enterprises and other relevant organizations fully exploit various local resources. As a result, the local tourism industry can be greatly developed and the management and customer's satisfaction well improved.

Specifically this research aims at perfecting the local government regulation and supervision in tourist destinations. Travelers will save more time and trouble in their traveling. And the industry-level management over tourism business will become more standardized and regulated Furthermore, dynamic optimization of the value chain will be achieved. The introduction of SOA offers a soft platform for the tourism enterprises to establish an organizational alliance, on the basis of which a destination's tourism industry will be able to achieve its dynamic optimization. The achieved research results and the relatively perfected application of SOA in other industries have laid a foundation for the present research of this project. But further study on how to use SOA in the dynamic integration of tourism industry and how to optimize the value chain in tourist destinations still need to be conducted. In the process mass of quantitative analysis and model system survey will be adopted to carry forward the research on the value chain of tourism industry so that the research will become more scientific. The research achievements of this project are beneficial for both improving the management of the local tourism business in certain destinations and being applicable to extend the chain of tourism industry. Therefore, the research is of deep theoretical significance and practical value.

B. Research approaches and feasibility

The research plans to combine quantitative analysis with the qualitative one. Specific studying approaches are as follows: customer satisfaction comment and assessment, SOA analysis and design and the method of dynamic integration of value chain[14]. Thus, it is of deeply realistic and practical significance. The introduction of SOA system enables the members of a tourism value chain to contact each other not only in a crisscross way, but also in a link-cross way, instead of contacting a certain number of business partners. They are competing with each other as well as cooperating with them. [15] Based on constructing information technology managing platform, the new value chain of tourism industry will help to formulate a seamless connection between different chain members in the space of information communication [15], thus we hope to realize the dynamic integration and optimization of tourism value chain with the conception of SOA framework. This framework is widely accepted by the business circle in China's software design and production industry, but is still lacking of dynamic integration application.

Besides, the institutional procurement in tourist destinations is usually conducted in the pattern of governmental guidance and regulation[16]. SOA structural system provides a technological feasibility for the local government's real-time supervision over the business operation of tourism industry in these areas.

5 Design of Tourism Value Chain Research

From the perspective of technological service, the research on the value chain in tourism industry is categorized into tourist-oriented and manager-oriented. Although the ultimate value of government-oriented technological service is directed by tourists' value orientation, its direct service object is still tourism business managers when examined in the science of management. And the concrete service is concerned with tourism business management and operation as well. The vertical integration and networking of a value chain is a dynamic process, which is a dynamic combination process based on traveler's wants and needs.

The value chain of tourist serving business is derived from the continuity of tourist needs and wants, thus concerning information technology support, a relationship between input and output does exist in the value chain of tourism industry [17], and the construction of which is apparently complicated and comprehensive. Within the industry there is no steady and continuous process of value chain flow. In the construction of tourism industry value chain tourists' traveling consumption activities serve as the major clue [18]. The value chain is much more dynamic and desirous of integration [19], so we should develop a management and measurement framework that would allow various players to communicate and coordinate their processes and activities in a more mature manner. Therefore, it becomes critical to measure and manage the overall efficiency and effectiveness of the tourism product and services from a value chain management perspective[2].

One research content of this project is based on the basic technology of SOA platform of the value chain in tourist destinations. The platform has laid a solid foundation and gives necessary technological support for the normal operation of each of the value chain links. Meanwhile, it is capable of effectively expanding the available service. Because different travelers come into a particular link of the value chain from different access, the researchers need to study the destination's value chain optimization from different perspectives. And the present business should be executed and integrated in a dynamic way, including the management of distribution channels.

In addition, the value chain platform in tourist destinations is able to give assistance to the decision-making of the local management institutions, (such as making comments on platform users' satisfaction, collecting information from various channels in tourism administration's net work, conducting user's research and investigation and giving feedback to guests' complaints.) The functions of trade management that the platform performs are as follows:

A. Actualizing relevant business support (including management function), proposing business designing model (the business that needs to be integrated for different participants and the model of cooperation)

B. The dynamic integration of SOA (system framework, design, and analysis methodology) and the standardized routine interface provided by SOA can ensure the dynamic information interchange between different closed information systems of different suppliers.

The unique property of loose coupling of the value chain platform offers a solution to the problem of non-standardized touring routes and that of the diversified needs from guests. That enables the platform to provide large varieties of prompt and flexible services. Additionally, SOA provides better supports to management divisions in their decision-making process, such as making comments on customers' satisfaction and supervising over tourism market. With the application of SOA in tourism industry, suppliers stored in different data systems can be incorporated into a uniformed system so that the benefit of economy of scale as a whole will be improved and the enterprise's operational cost reduced as well. Once the integration and dynamic optimization of the value chain is completed, tourism management with information technology will be applied more extensively and upgraded to a new level. That will help the chain of tourism industry and the regional environment develop in collaboration.

C. Making comments on user's satisfaction for making optimized decision (data-searching and storing technology and commenting model)

6 Conclusion and Prospect

With the new pattern of value chain, all its members are able to collaborate with each other with more flexibility. Horizontally, tourism industry is comprehensive

and is composed of by quite a few number of trades that need close coordination in routine operation. Enterprises of Dieting, accommodation, transporting, sightseeing, shopping and entertainment commonly constitute all the supplier links [20]. Tourism service corporation groups, travel service companies and travel agencies all serve as the intermediates. When observed vertically, all the enterprises in the overall value chain of tourism industry have realized information transmitting from suppliers to consumers on the tourism e-commerce platform which takes the Network and e-commerce technology as the major communication medium. This type of business integration saves transactional cost and trading time for the complete value chain.

Presently, the application and the Network and e-commerce platform makes the contact between the chain members become more accessible. Information sharing and transactional automation have greatly improved the operational efficiency of tourism value chain. All the business stages, from need prediction, product research and development, product wrapping and packaging, publicity and sales promotion, guest resources generating and organizing to customer service, are being operated in co-ordination and smoothness. The situation has created a multi- win condition for all the chain members.

The application of the network and e-commerce platform gives a remedy for the shortcomings of the traditional value chain of tourism industry and offers more flexibility.

The contact between different members of the value chain is not as fixed as it was, it could be crisscrossing as well as link-spanning. For example, a tourist product supplier can not only cooperate with a number of intermediates, but sell his products directly to travelers by skipping over the intermediates. With more types of value chain created, the chain members will have more alternatives to choose from for optimizing the integration of effective resources. As a result, the value chain efficiency will be greatly promoted.

Bibliography

[1] Zhang, J., Zhang, J., Liu, J.: Research on Combination Pattern of tourism industry based on the value chain theory and relevant technology. Science of Tourism (1) (2005) in Chinese

[2] Yılmaz, Y., Bititci, U.S.: Performance measurement in tourism: a value chain model. Journal of Contemporary Hospitality Management 18(4) (2006)

[3] Li, Y., Wanf, J., Xu, S.: Reconstruction of value chain of tourism industry in the Internet times and research on the optimization model. Issue of Forest Economy (3) (2007) (in Chinese)

[4] Wang, G., Qiu, Y.: Pattern of principal marketing for domestic tourist destinations, exploration based on e-commerce times. Market weekly Research edition (9) (2005) (in Chinese)

[5] Zhang, M., Tian, Y.: Research on value chain transferring process in E-commerce business. Shenyang Industry News (29) (2007) (in Chinese)

[6] Liu, R., Yuan, G.: Exploration on management of China's tourism value chain. Ecological Economy (12) (2007) (in Chinese)
[7] Liu, S., Luzi: On regional integrating function of tourist destination marketing system. Tourism Tribune (2) (2004) (in Chinese)
[8] Zou, R.: Construction of marketing network in tourist destination based on information service. Finance and Trade Economics (2) (2005) (in Chinese)
[9] Lu, K.: Primary exploration on new model of supply chain in tourism industry. Tourism Tribute (3) (2006) (in Chinese)
[10] Zhao, J., Lu, R.: Research on tourism resources integration based on value chain of tourism resources. Social scientist (10) (2005) (in Chinese)
[11] Palmer, A., Bejou, D.: Tourism destination marketing alliances. Annals of Tourism Research 22 (1995)
[12] Wang, Y.C., Fesnmaier, D.R.: Collaborative destination marketing, a case study of Elkhart county, Indiana. Tourism Management (2006) (Article in press)
[13] Chen, H.M., Tseng, C.H.: The performance of marketing alliances between tourism industry and credit card issuing banks in Taiwan. Tourism Management (26) (2005)
[14] Hao, X., Li, R.: Research on e-commerce system based on SOA. Knowlege Economy 11 (2007) (in Chinese)
[15] Lao, B., Yang, L., Li, X., Chen, H.: Reconstruction of tourism value chain in condition of e-commerce. Business Times (23) (2005) (in Chinese)
[16] Li, D., May, U.: Research on model of tourism value chain n the background of tourism management by IT. Value Engineering (11) (2006) (in Chinese)
[17] Gao, J.: Current situation and prospect of domestic research on tourist destination marketing. Journal of Beijing International Studies University (11) (2008) (in Chinese)
[18] Ma, M.: Analysis on e-commerce product and service on China's tourism websites. Tourism Tribune (6) (2003) (in Chinese)
[19] Fang, C., Xue, H., Huan, J.: Exploration on work flow process management of tourism industry in experience economy times. Market Forum (7) (2007) (in Chinese)
[20] Zhang, J.: Science, technology and tourism development. Tourism Tribune (6) (2004) (in Chinese)

Analysis and Quantitative Calculation on Switching Costs: Taking 2002-2006 China Wireless Telecommunications Market as an Example

Ge Zhu, Jianhua Dai, and Shan Ao

Summary. Much research has already examined the formation and influence of switching costs on the consumer's repeat purchase intentions but little research focused on quantitative measurement of the switching cost itself. This paper constructs a new algorithm by Nash-Bertrand model which consists of observed variables such as profits, yields and the changing market shares of two firms, in order to deduce unknown switching costs. The complete Nash-Bertrand model considers price compensation and transport costs in order to accurately estimate consumer switching costs in a duopoly. Based on 2002-2006 data from China's wireless communication industry, a multi-period model is applied to calculate the consumer annual average switching costs of the only two companies licensed to operate wireless communication in 2002-2006 in China: China Mobile and China Unicom. The result shows that China Mobile users' switching costs are significantly higher than those for customers of China Unicom, and the gap was increasing generally. The quantitative analysis demonstrates that reducing of consumer switching costs will relatively benefit small operators and intensify competition.

1 Introduction

When consumers think that a product/service is not worth changing, they may have perceived switching costs as arising from their search costs, transaction costs, learning costs, loss of loyal customer discounts, customer habit, emotional cost, and cognitive effort. Together with the perceived financial, social, and psychological risks on the part of the buyer, these costs constitute barriers to switching [1]. Generally, it is impossible to directly calculate the entire switching cost because these are difficult to exhaustively classify. However, an accurate estimation of switching

Ge Zhu · Jianhua Dai · Shan Ao
School of Information Management, Beijing Information Science & Technology University
No.12 Xiaoyingqiao, Haidian Dist. 100192 Beijing, China
e-mail: Zhuge01@gmail.com

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 201 – 212.
springerlink.com

costs is significant for research on corporate strategy and regulator policy, as well as on consumer behavior.

Enterprises increase switching costs for profits by locking-in existing customers and by providing them with value-added products or services. Thus a firm's existing customer based market share is an important determinant of its existing and future profitability [2]. Due to network externalities, in information and open competition economies, market share is important to trigger positive feedback effects and form a winner-takes-all market share. When customer-based market share has been recognized as an asset, firms will even provide price compensation to offset their switching costs in order to attract competitors' customers [3]. Therefore, the existence of switching costs leads to vigorous marketing competition for market share before customers have attached themselves to a supplier to positively influence their pre-purchase switching costs assessments [4]. In such markets, it is not surprising that firms supply new products or services free or even with a negative price. For example, electronic game firms pay customers to play their new network game in order to form an initial customer base. In other words, if firms acquire information suggesting that future demand is likely to be high, they will price aggressively, sacrificing current profits for a higher future market share and the expectation of higher future profits [5]. More examples are provided by Shapiro and Varian to suggest the impact of switching costs on market behavior and market structure in information economies [3].

Higher network externality leads to higher intrinsic switching costs and in such markets it is easier to formulate a natural monopoly because a customer's product/service valuation increases with the number of other users who adopt the same product/service [6-8]. At the same time, companies increase switching costs by furthering upon extrinsic barriers. Based on this rationale, similar to macroeconomic theories, it can be argued that switching costs tend to reduce competition and initiate a monopoly especially in a mature market [9]. In an oligopoly market, a company, in particular the market leader, has the incentive to increase barriers for consumers who might otherwise consider switching supplier [10]. It is in a firm's interest to move toward monopoly or at least maintain a duopoly. Because of these competitive effects, even inefficient incompatible competition is often more profitable than compatible competition especially for dominant firms [11]. Thus firms are likely to seek incompatibility too often; for example, the telecommunications industry, where established operators are unwilling to interconnect networks with the small operators; and mobile phone manufacturers who have larger market shares have little incentive to unify the phone charger standards.

Nevertheless, lower switching costs and greater standardization are advisable for social welfare [12]. In addition, according to Klemperer's two-period model [4], switching costs do not necessarily make firms better off overall because of greater competition in the early stages of the market's development. Accordingly, government regulation and market rules or laws hope to provide checks and balances to increased switching costs, and optimize the market structure from the point of view of benefiting consumers. For example, in 1999, the Hong Kong government issued a regulatory policy called wireless number portability (WNP) to reduce switching costs for mobile phone users. Shi, Chiang and Rhee [13] have discussed what

implications it has brought to the market structure. They argue that the reduction of switching costs can not optimize market structure and be conducive for the development of small operators. On the other hand, WNP creates a market condition conducive for larger networks to gain more market share. However, this conclusion is not consistent with common sense and is contrary to the original intention of the government's policy. This paper will present a reasonable interpretation of the relation between switching costs and market structure, and provide a theoretical reference for the formulation of market regulation.

The large theoretical literature has demonstrated the impact of switching costs on pricing and industrial structure in a variety of markets (banking, insurance, retail, telecommunication etc.), of which Klemperer [9] and Farrell and Klemperer [2] present an excellent literature review. In contrast, there are only a limited number of empirical analyses on the measurement of switching costs. A direct measure of switching costs is difficult to obtain because switching barriers are industry-specific as well as consumer-specific, and they are not directly observed by the economists. However, some empirical studies are still significant for quantitative research on switching costs. Schlesinger and Schulenburg study an insurance market with established insurers and new entrants by modeling a Hotelling-type of spatial equilibrium [14]. Borenstein empirically studies the market for gasoline and indicates that price discrimination is possible because of differences in the willingness of customers to switch gas stations [15]. Nissen proposes a multiperiod duopoly model to examine the effects of changing two switching costs: transaction costs and learning costs [16]. To posit the idea that market share in one period affects profits and welfare in future periods and builds up a two-period model of oligopolistic competition with switching costs [17]. Knittel using a panel dataset of rates, empirically tests for the influence of switching costs on the price-cost margin and notes that switching costs have provided operators with market power [18]. Bakos empirically studies the impact of electronic markets on search costs and the results show that firms may prefer to increase search costs and evaluation costs, these sunk costs make potential consumers likely to choose which he has paid more attention [19]. Chen and Hitt study the determinants of customer retention in Internet-enabled businesses; e.g. the online brokerage industry, and thus measure switching costs [20]. Gabrielsen and Vagstad introduce consumer heterogeneity to discuss second-degree price discrimination with switching costs [21]. Kim, Kliger and Vale present an empirical model of company behavior to estimate the magnitude and significance of switching costs in the market for bank loans [22]. Maria studies the behavior-based price discrimination in the presence of switching costs by a two-period model [23]. Israel develops a behavioral model of consumer-company relationships to estimate switching costs in the auto insurance industry [24]. Recently, Lee et al. provide a conjoint analysis to illustrate that number portability does partially reduce phone users' switching costs [12]. Shi et al. discuss the impact of number portability on market structure [13].

In this paper, we propose a complete Nash-Bertrand model to compute the period of switching costs, by observed variables of profits, yield and the change of market share, in order to study the influence of switching costs on market structure in quantitative analysis. To get the relative and real switching costs, we integrate the

Hotelling model to reflect consumer preference, and consider price compensation and the increase of new users. The model's results give not only switching costs but also the customer transfer rates, equilibrium price and compensation. In this paper, China's mobile telecommunication market in 2002-2006 will be taken as a demonstration case.

2 A Quantitative Calculation Model

Consider an oligopoly of two firms competing in a multiple-period price (Bertrand) competition. The goods sold by the firms are not storable. To focus on the customer's decision from which firm to purchase the good, the customer is assumed to have an inelastic demand. Specifically, each customer purchases a single unit of the good at each one of infinitely many discrete periods. The customer behavior described here yields probabilities of switching between firms. We call these probabilities 'transition probabilities'. Transition probabilities are functions of the price and switching costs. Aggregation of transition probabilities yields the demand faced by each firm. The specific hypothesis and signification of parameters are presented as fellows:

Consider an oligopoly of two firms i (i=1, 2) competing in a multiple-period price (Bertrand) competition in a specific industry (this paper relates to the telecommunication industry). The good sold by the firms is homogeneous and unstorable. We assume that the fixed and marginal costs are zero. Both of the firms aim to maximize the profits.

The initial user number is G, and the market share of firm i is respectively $\sigma_\alpha>0$ (i=1) , $\sigma_\beta>0$ (i=2), $\sum\sigma=1$, in which, $\sigma>0$ denotes the two firms are not new entrants. A consumer who has purchased firm 1's products is assumed to be an α type customer. Similarly, the consumer who has purchased firm 2's products is assumed to be a β type customer.

There are always new customers who enter during the process of competitive marketing. Assuming the number of new entrants is N, the new consumer is defined as an N type consumer.

Consumer preference satisfies the Hotelling model. The consumers are uniformly distributed on the interval [0, 1]. Firm 1 is located at the leftmost of the line so that the consumer who wants to purchase product 1 will spend a transport fee $t\times x$. On the other hand, firm 2 is located at rightmost of the line and the consumer who attempts to purchase product 2 will pay for the transport cost $t\times(1-x)$, where $t>0$ denotes the unit transportation cost, and $0<x<1$ denotes the distance between consumer and firm 1.

Whether incumbent users or new entrants, consumers are assumed to purchase a single unit product service at one time. U_α denotes the utility of the α type consumer. Similarly, U_β and U_n denote the utility of the β and N type consumer respectively.

The customer maximizes her utility by deciding from which firm to purchase, given the prices charged by each firm. When the utility of purchasing one firm's product becomes less than the other, the consumer will switch to the other firm. She

will need to pay for certain switching costs S_i (i=1, 2). In this model, the switching costs will always be average switching costs within a period except when specially explained. The magnitude of switching costs is seen as a whole for the individual consumer so that she can make a rational decision.

The capability of production is infinite for firms. A firm's yield (i.e. the number of consumers) is determined by its own and its rival's price. Every firm sets its price P_i (i=1, 2) independently at same time. This paper does not consider collusion between firms.

A firm can distinguish the new customers from incumbent ones. The firms will give uniform compensation A_i (i=1, 2) to the new customers which are made up of the N type consumer and the consumers switching from its rival. Comparatively, the incumbent consumers are treated to a discriminatory price. It is obviously that $A_i \leq S_i$ (i=1, 2) in order to assure that the compensation will not damage the original customer base.

The utility function of three consumer types derived from the next purchase is respectively given by:

$$u_\alpha(x) = \begin{cases} -P_1 - tx & keep\ \ 1 \\ -P_2 - S_1 - t(1-x) + A_2 & Switch\ \ to\ \ 2 \end{cases} \tag{1}$$

$$u_\beta(x) = \begin{cases} -P_1 - S_2 - tx + A_1 & Switch\ \ to\ \ 1 \\ -P_2 - t(1-x) & Keep\ \ 2 \end{cases} \tag{2}$$

$$u_n(x) = \begin{cases} -P_1 - tx + A_1 & Purchase\ \ 1 \\ -P_2 - t(1-x) + A_2 & Purchase\ \ 2 \end{cases} \tag{3}$$

Based on the simultaneous equations and observed eight variables ($\pi_1, \pi_2, G, \sigma_\alpha, \sigma_\beta, N, Q_1, Q_2$), the model can obtain the unknown seven variables ($S_1, S_2, P_1^*, P_2^*, A_1^*, A_2^*, t$). According to the above results the consumer retention rates ($x_\alpha, 1-x_\beta, x_n$) can be computed.

3 A Case Study of China Wireless Telecommunication Market

From 2002 to 2007, China's wireless mobile telecommunication market is a typical duopoly market. Although China has six telecommunication licenses, only four operators are valuable, which are delimited as fixed-line operators and mobile operators. China Mobile and China Unicom are the only two mobile phone carriers in China at that time. China Mobile resulted from a former government enterprise and is an established operator. China Unicom is a later entrant, also supported by the government, to create market competition. Although all operators are administered by the Chinese State Assets Committee as government enterprises, they belong to

different interest groups. Market competition is effective and China Unicom's entry has triggered a vigorous price competition and the China wireless mobile telecommunication industry has developed noticeably overall. From 2004 to 2007, users' average year growth rate reached 19%. Up to July 2007, there were 500 million mobile subscribers and the penetration rate had reached more than 37.7% (www.mii.gov.cn).

China Mobile operates a GSM network and had 316.2 million subscribers as of the end of March 2007. China Unicom operates both GSM and CDMA networks. At the same time, China Unicom had a total of 109.58 million GSM mobile phone subscribers and 37.724 million CDMA mobile subscribers. CDMA, 3rd generation (3G) mobile communication technology, has been launched in China by China Unicom. However, 3G is still in its initial period owing to a shortage of skilled service applications and insufficient market demand. Therefore, in this paper we treat the mobile telecommunication services as homogeneous voice service products before purchasing.

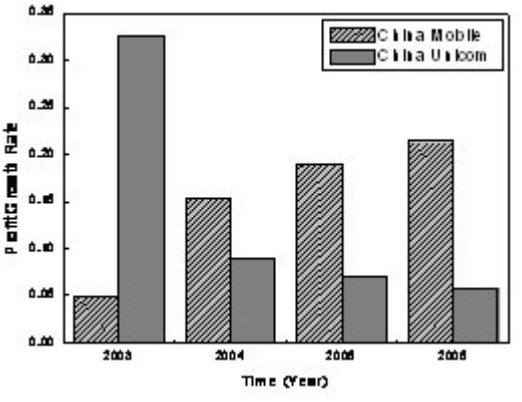


Fig. 1 China Mobile and China Unicom: the annual profit growth rate in 2003-2006

According to the 2002-2006 annual report on China's telecommunication from the China Ministry of Industry and Information Technology (formerly Ministry of Information Industry, MII), the profit growth of China Mobile is increasing, while China Unicom's annual growth rate is decreasing. As shown in Fig. 1, It implies that the development of China Unicom had been shrinking over these years.

Table 1 shows the number of initial users G, initial market shares σ_α and σ_β, new entrants N, final user of every operator Q1 and Q2, and profits $\pi 1$, $\pi 2$ respectively in 2002, as well as 2003-2006's data in which, the number of initial users is the number of final users in the previous year. It is interpreted as margin costs trend to zero when the fixed network costs have been launched**.**

Table 1 China's Wireless Mobile Telecommunication Market In 2002-2006

Year	Observed Variables	Market Total (millions)	China Mobile Share (millions)	Proportion (%)	China Unicom Share (millions)	Proportion (%)
2002	Initial users	G=145	104	σ_α=71.72	41	σ_β=28.28
	New entrants	N=61	34	55.74	27	44.26
	Final users	206	Q_1=138	66.99	Q_2=68	33.01
	Profits (RMB)	214060	π_1=163730	76.49	π_2=50330	23.51
2003	New entrants	64	40	62.50	24	37.50
	Final users	270	178	65.93	92	34.07
	Profits (RMB)	238575	171870	72.04	66705	27.96
2004	New entrants	64	43	67.19	21	32.81
	Final users	334	221	66.17	113	33.83
	Profits (RMB)	270920	198300	73.20	72620	26.80
2005	New entrants	59	43	72.88	16	27.12
	Final users	393	264	67.18	129	32.82
	Profits (RMB)	313500	235800	75.24	77700	24.78
2006	New entrants	68	53	77.94	15	22.06
	Final users	461	317	68.76	144	31.24
	Profits (RMB)	368440	286300	77.71	82140	22.29

Source: People's Republic of China Ministry of Information Industry, 2002-2006 Annual Report of China telecommunication, by Beijing: Post & Telecommunication Press.

Substituting the data for the number of subscribers, market shares, new subscribers and profits into equations yields the switching costs, equilibrium prices, compensation and transport costs of China Mobile and China Unicom as shown in Table 2. There are two sets of solutions and one set has been deleted because switching costs are negative.

Table 2 China's Wireless Mobile Telecommunication Market In 2002-2006

Year	Switching Costs		Equilibrium Prices		Equilibrium Compensation		Transport Costs
	S_1	S_2	P_1^*	P_2^*	A_1^*	A_2^*	t
2002	1867.94	1270.82	1405.60	1173.00	1007.10	906.66	676.85
2003	1563.90	1200.31	1114.33	979.47	875.98	835.11	520.36
2004	1419.82	967.64	1034.75	862.06	783.58	717.82	505.35
2005	1374.40	816.37	1025.11	812.39	738.92	658.99	522.27
2006	1390.53	728.48	1035.01	780.76	728.30	627.53	526.31

The result indicates the total values of switching costs of China Mobile and China Unicom have greatly decreased in the five years. This implies that the telecommunication industry's natural monopoly is being weakened. By calculation, the market average switching costs (S1×Q1+S2×Q2)/(G+N) are decreasing year after year, while the market capacity is still booming. In other words, the lock-in of mobile phone users is decreasing between the five years. In our opinion, many low-cost new communication technologies such as Internet communication had been influencing the traditional telecommunication industry's technical and economic basis for a monopoly. On the other hand, the booming market mainly comes from the rapid increase of new entrants, which can partially lower the whole switching costs.

As shown in Fig. 2, switching costs for China Mobile consumers are higher than those for China Unicom every year, which implies the well-established status of China Mobile. The switching costs ratio of China Mobile and China Unicom are from 1.31 to 1.91. Generally, switching costs are digressive and China Unicom has a more distinct decline than China Mobile. It shows that the degree of market power of China Mobile is obviously higher than that of China Unicom and there was a notable gap in 2006. It matches the fact that China Unicom's operational mistakes led to the stickiness of its brand decline.

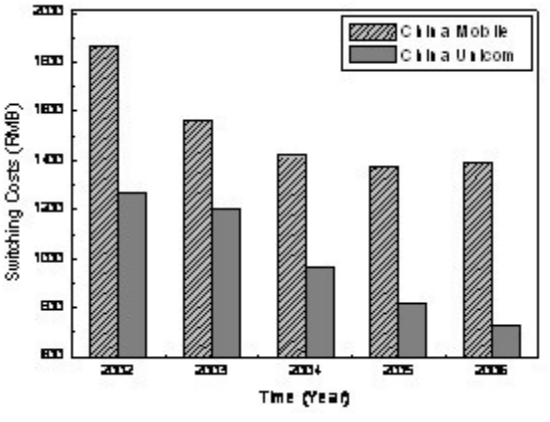


Fig. 2 The technology framework of Ubiquitous Network Society

Both operators' switching and transport costs are important factors in the decision on prices. The prices for new entrants and incumbent users decrease with the decline of switching costs. Especially in the last two years, China Unicom's switching costs have had a significant reduction compared with China Mobile so that the price difference becomes larger. Switching costs un-ambiguously relax price competition in equilibrium but, on the contrary, they may make tacit collusion more difficult to sustain although the government always tries to inhibit the fierce price competition.

Laffont, Rey and Tirole show that in both the mature and the entry stages of the industry, the nature of competition is substantially affected by price discrimination [10]. Since telephony is one of the most highly personalized type of goods, identifying individuals with telephone numbers, this naturally invites suppliers to take advantage of various sorts of discrimination such as quantity, location or time [26]. The most simple and general discrimination in prices are based on distinguishing between new entrants and incumbent subscribers. These non-linear prices can maximize the revenue of suppliers, and in the author's opinion, non-linear prices come from the recognition of switching costs of different consumers.

Through observed market results, we have obtained other variables apart from switching costs, as shown in Table II. The equilibrium price as well as the price compensation from China Mobile is higher than that of China Unicom. It is obvious that China Mobile is more expensive than China Unicom whether for new entrants or incumbent users. It matches the consumer's perceived price for the two operators. Although the price compensation of China Mobile is higher than China Unicom, the latter's price for new entrants (RMB163.23 in 2006) is still lower than that of China Mobile (RMB254.25 in 2006).

Owing to the existence of compensation, the price discrimination between incumbent and new entrants is prominent. While price compensation comes from the existence of switching costs, in order to steal the rival's users, operators offer some compensation so that the switching costs can be partially counteracted. However, in a growing market the switching users can not be differentiated from new entrants. For an operator, switching users and new entrants are all treated as new users and enjoy discounted prices, although even with the existence of price discrimination, most users still retain their loyalty to their original provider. Price competition is fierce between the two mobile operators though the regulator set a lowest price per minute (0.4Yuan/minute) from 1999. Operators provide a wide array of tariff packages in order to avoid government control of prices. The booming mobile market and battle for market share causes aggressive price cutting that is well below levels permitted under the state-controlled tariff regime. In 2003, the two-way charging fees were changed to become quasi-one-way (i.e., manned free) for the respective networks although the regulator has not approved it as yet. Manned free significantly decreases the prices for all of the new entrant and incumbent users, as shown in Fig. 3. However, what influence has it created on switching costs and market structure? From the viewpoint of business, the quasi-one-way charging fee within the respective networks is conducive for the small operator. Because it increases the power of the network effect in a larger network more than in a small network, therefore users of China Mobile benefit more than users of China Unicom. Accordingly, the loss for a large operator is larger than for a small operator. Furthermore, the switching costs of China Unicom had a comparably smaller decrease than China Mobile in 2003, as shown in Fig. 3.

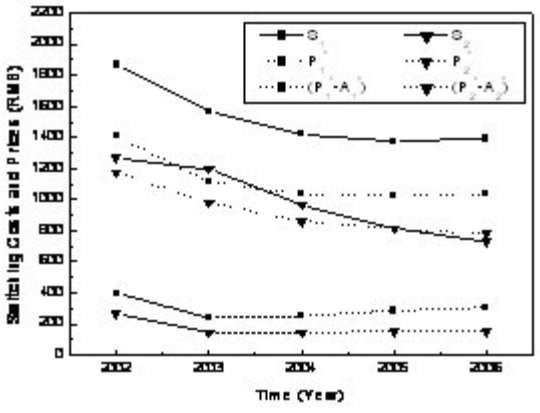


Fig. 3 The switching costs and price discrimination in 2002-2006

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

4 Conclusions

The model is valuable in calculating unseen switching costs and studying the impact of switching costs on market structure, especially for a duopoly in telecommunication. According to the profits and yields as well as the difference between market shares, this paper provides a complete Nash-Bertrand model and algorithm to calculate the switching costs and the other variables. By analysis, we find that switching costs have a crucial influence on the change of market structure in competition. Higher switching costs aggravate the gap between market shares. Firms can snatch extra profits by lowering the switching costs. For regulators, reducing switching costs does benefit small competitors and breaks the market monopoly, although sometimes it is not obvious. Through the model and its reduced model, we can analyze and interpret the WNP policy of the Hong Kong wireless telecommunication market in 1999 and reach a conclusion which the reducing of switching costs is not the reason of market differentiation [13]. On the contrary, the reducing of switching cost slow down the increasing of market differentiation. In addition, empirical research on China's wireless telecommunication market shows that China Mobile has a durable potential advantage whether its market share decreased or increased from 2002 to 2006. The impact of quasi-one-way charging fees on market structure is discussed, as well as the impact of WNP if it is ever implemented in China's telecommunication market. By virtue of the existence of

switching costs, relative to new entrants, firms impose discriminatory pricing on incumbent users, while the prices for new entrants are considerably lower. Therefore, price compensation is significant for the building of a complete model that reflects the real market.

The model is quite different from Shy's although his work is very instructive for our modeling [25]. Integrating with the wireless mobile telecommunication industry of China, the model provides a good interpretation and quantitative analysis for market structure and change. Results show that the present model is better than Shy's model in accurately calculating switching costs in a duopoly.

In order to simplify the model, we did not consider the fixed costs and marginal costs. We applied the Hotelling model in a simple form rather than in non-linear form so that the equilibrium prices are decided only by linear functions. It should be mentioned that switching costs are a function of time because they are different in different periods and for different individuals. However, in this paper the switching cost is an average cost of all the users within one year. In our empirical research, the companies' incomes taken as profits will influence the veracity of switching costs calculation. In our basic hypothesis, the consumers are very clear about their switching costs and services so that they can choose to purchase according to maximization of utility. The present model has not considered incomplete and imperfect information. In fact, competitive firms are apt to deliberately conceal information and consumers can not completely recognize or forecast the switching costs before switching behavior. However, this paper effectively avoids the question by using observed market variables to trace the switching costs. This model is fit to calculate actual switching costs in a duopoly rather than in the more oligopolistic market although there are three major operators in wireless telecommunication market now. Nevertheless, it is still a significant approach to investigating the impact of switching costs on market structure even in multi-game model as long as it is able to be simplified into a duopoly.

Acknowledgment

Project sponsored by National Nature Science Foundation of China (NSFC) (No.71001009) and Beijing Municipal Education Commission Scientific & Technological Development Plan Foundation (KM201110772008), Beijing Municipal Scholar Innovation Team (PHR201106133).

References

1. Fornell, C.: A national customer satisfaction barometer: The Swedish experience. Journal of Marketing 56(1), 6–21 (1992)
2. Farrell, J., Klemperer, P.: Coordination and lock-in: Competition with switching costs and Network effects. In: Handbook of Industrial Organization, vol. 3, ch. 3, pp. 1967–2072 (2007)
3. Shapiro, C., Varian, H.: Information rules: A strategic guide to the Network economy. Harvard Business School Press, Cambridge (1998)
4. Klemperer, P.: Markets with consumer switching costs. Quarterly Journal of Economics 102, 375–394 (1987)

5. Elder, E., To, T.: Consumer switching costs and private information. Economics Letters 63(3), 369–375 (1999)
6. Farrell, J., Shapiro, C.: Dynamic competition with switching costs. The Rand Journal of Economics 19, 123–137 (1988)
7. Katz, M.L., Shapiro, C.: Network externalities, competition, and compatibility. American Economic Review 75(3), 424–440 (1985)
8. Katz, M.L., Shapiro, C.: Technology adoption in the presence of network externalities. Journal of Political Economic 94(4), 823–841 (1986)
9. Klemperer, P.: Competition when consumers have switching costs. The Rank Journal of Economic Studies 62(4), 515–539 (1995)
10. Laffont, J.J., Rey, P., Tirole, J.: Network competition: II. Price discrimination. The Rand Journal of Economics 29(1), 38–56 (1998)
11. Waterson, M.: The role of consumers in competition and competition policy. International Journal of Industrial Organization 21(2), 129–150 (2003)
12. Lee, J., Kim, Y., Lee, J.D., Park, Y.: Estimating the extent of potential competition in the Korean mobile telecommunication market: Switching costs and number portability. Journal of Industrial Organization 24(1), 107–124 (2006)
13. Shi, M., Chiang, J., Rhee, B.D.: Price competition with reduced consumer switching costs: The case of 'wireless number portability' in the cellular phone industry. Management Science 52(1), 27–38 (2006)
14. Schlesinger, H., Schulenburg, J.M.: Search costs, switching costs and product heterogeneity in an insurance market. Journal of Risk & Insurance 58, 109–119 (1991)
15. Borenstein, S.: Selling costs and switching costs: explaining retail gasoline margins. The RAND Journal of Economics 22, 354–369 (1991)
16. Nilssen, T.: Two kinds of consumer switching costs. The RAND Journal of Economics 23(4), 579–589 (1992)
17. To, T.: Export subsidies and oligopoly with switching costs. Journal of International Economics 37(1-2), 97–110 (1994)
18. Knittel, C.R.: Interstate long distance rate: Search costs, switching costs, and market power. Review of Industrial Organization 12, 519–536 (1997)
19. Bakos, J.Y.: Reducing buyer search costs: Implications for electronic marketplaces. Management Science 43(12), 1676–1692 (1997)
20. Chen, P.Y., Hitt, L.M.: Measuring switching costs and the determinants of customer retention in Internet-enabled businesses: A study of the online brokerage industry. Information Systems Research 13(3), 255–274 (2002)
21. Gabrielsen, T.S., Vagstad, S.: Consumer heterogeneity, incomplete information and pricing in a duopoly with switching costs. Information Economics and Policy 15(3), 384–401 (2003)
22. Kim, M., Kliger, D., Vale, B.: Estimating switching costs: The case of banking. Journal of Financial Intermediation 12(1), 25–56 (2003)
23. Maria, A.: Behavior-based price discrimination and consumer switching. Advances in Applied Microeconomics 9, 149–171 (2000)
24. Israel, M.: Tenure dependence in consumer-firm relationships: An empirical analysis of consumer departures from automobile insurance firms. The Rand Journal of Economics 36(1), 165–193 (2005)
25. Shy, O.: A quick-and-easy method for estimating switching costs. International Journal of Industrial Organization 20(1), 71–87 (2002)
26. Blonski, M.: Network externalities and two-part tariffs in telecommunication markets. Information Economics and Policy 14(1), 95–109 (2002)

An Empirical Study of Network Topology Inference

Hui Zhou, Wencai Du, Shaochun Xu, and Qinling Xin

Summary. Understanding network topology is important for evaluating the performance of network protocols, for detecting large-scale denial-of-service or malicious intrusion, improving the design of resource provisioning, or studying the scalability of multicast. Usually, to infer the topology of a large-scale network, researchers use multiple vantage points to conduct extensive traceroute-based measurements. However, an inferred topology is often incomplete or inaccurate because the traceroute technique on which we heavily rely has inherent limitations. Furthermore, the Internet is so complicated and dynamic that discovering an immediate snapshot of its topology to be a very challenging task. In our experiments, to identify a large ISP cloud, we spread vantage points inside the cloud and over the world, and collect topology information by probing a fixed list of IP addresses which consists of more than 25,000 routers and 36,000 links. Data analysis shows that sampling bias, if undetected, could significantly undermine the conclusions drawn from the inferred topologies.

1 Introduction

Understanding the structural properties of the Internet has been proved to be a challenging task. There is no single place from which one can obtain a complete picture of its topology since the Internet is a collection of thousands of smaller networks, each under its own administrative control. Moreover, because the design of network does not provide explicit support for direct inspection, the task of

Hui Zhou · Wencai Du
Hainan University, Renmin Ave. No. 58, 570228, Haikou, China
e-mail: wencai@hainu.edu.cn

Shaochun Xu
Algoma University, Sault Ste, Marie, Ontario, P6A2G4, Canada

Qinling Xin
Central China University of Technology, Wuhan, China

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 213–225.
springerlink.com

"obtaining" the Internet's topology has been left to researchers who develop more or less sophisticated methods to infer this topology from a large volume of network measurement data. Because of the elaborate nature of the network protocol suite, there are many measurement approaches, each having its own strengths, weaknesses, and assumptions, and each resulting in a distinct view of target topology.

In the last 15 years, researchers have inferred five basic categories of network topologies. They are the graphs of connections between autonomous systems (ASs) [1], the point-of-presence (POP) topologies that interprets the structure of backbone using geography information [2], the IP-level topologies whose nodes are IP addresses and whose links are connections between the IP addresses [3, 4] the router-level topologies that resolve IP aliases and group the IP addresses in the unit of router [5], and the connectivity of physical components, including routers, switches, and bridges [6]. In particular, the router-level topology has attracted more attention than the others because it establishes the basis of AS-level and POP-level topologies, gives a more operational picture than the IP-level topology, and hides some unnecessary details of physical connectivity.

The study of topology inference is successful in that it has collected invaluable topology information of the Internet. For example, Pansiot and Grad detected 3,888 nodes and 4,857 links in 1995 [7]. Govindan and Tangmunarunkit developed the Mercator program and used it to map the Internet; their work resulted in a topology consisting of 228,263 nodes and 320,149 links in 1999 [5]. After that, Spring et al. applied Rocketfuel to discover the topologies of ten diverse Internet service providers (ISPs) using about 750 public traceroute servers [2]. Moreover, an ongoing project, Skitter, has been scanning the whole Internet for several years with tens of commercial network hosts, and it has released extensive graphs of the Internet's IP-level topologies [8].

As more and more topology information is available, researchers have been interested in finding significant features of the topologies. Faloutsos et al. proposed several empirical power laws that can characterize both the router-level and the AS-level topologies [9]. This finding not only spurs a large body of work in identifying and validating the properties of large-scale networks [10], but also stimulates researchers to create better topology generators in order to produce virtual networks that exhibit the structural and statistical features of Internet [11].

However, there is a growing agreement on topology inference that the inference is not complete or accurate [12, 13]. First, the traceroute technique [7] that almost all inference methods heavily rely on has inherent limitations. For example, traceroute does not see backup links in a network, and it does not expose link-layer dependency or redundancy (multiple IP links over the same fiber). Furthermore, the Internet is so large, complicated, and dynamic that inferring an instantaneous snapshot of its topology seems impossible. Existing systems measure target network in a period ranging from days to months. Therefore, an inferred topology won't be very complete or accurate since the Internet tends to undergo considerable changes during measurement. More seriously, there isn't an efficient technique for validating the

fidelity of the inferred topologies; it is especially the case when researchers try to map a large-scale network such as Abilene [14].

Therefore, it is very necessary to study the topology coverage, and to address questions such as "what is the cause of sampling bias?", "what does the sampled information tell us about the real network?", "how to capture an accurate topology of target network with as few measurements as possible?", and "what is an accurate topology, anyway?"

We introduce our inference approach in Section 2, perform an experiment to infer the topology of a large-scale network in Section 3, analyze the topology coverage in Section 4, and discuss our findings in Section 5. Finally, Section 6 concludes the paper.

2 Approach

To study the sampling bias of topology inference, the key challenge that we face is to obtain the real topology of target network. Intuitively, by comparing the real topology with the inferred one, we can identify both the common and the different of them so as to characterize the bias. Our solution is to capture an almost complete topology of a given network using as many vantage points as possible. Since each vantage point, which is a host used to probe the network for topology information, views the network from its own perspective, we combine the views of a random set of vantage points to infer various topologies, and analyze how these topologies are different from the almost complete one.

This approach has four rigorous requirements. First, the target network can't be too large or too small. If a network is too large for a given set of vantage points to measure in a short period (e.g., three days), the inferred topology may not be an accurate snapshot of the network since many end-to-end routes are only stable over time scales of days to weeks [15]. In contrast, if a network is too small, it may not include some salient features of the Internet, so the analysis of its topology can't be applied to the study of other networks. Therefore, we choose to measure an ISP cloud, i.e. a number of interconnected ISP networks without any other networks among them.

Second, the inferred topology should be almost complete, that is, it should include at least 95% routers and 95% links of the ISP cloud. Meanwhile, the inferred topology should not include routers and links outside the ISP cloud.

Third, vantage points should spread over the ISP cloud. Generally, an inferred topology is constructed by merging the information of nodes and links collected by vantage points, viewing a network using vantage points at different positions would possibly result in different interpretation of the network. Therefore, to study the sampling bias from a comprehensive perspective, the vantage points should not be placed in only a few locations. Instead, there ought to be a large number of vantage points, and they must be located in a wide range of network positions.

Finally, each vantage point should be configured to probe all the possible IP addresses of the ISP cloud. This is an arguable requirement since some existing methods assign each vantage point a partial list of IP addresses (or IP prefixes) so as to speed up the measurement and to decrease network overhead. But we argue that this requirement is necessary because it enables vantage points to gain possibly the most complete view of the real topology. In addition, this probing strategy and those of the other inference methods share the common nature: mapping the target network in a best-effort manner.

The main advantage of our approach is that it allows us to examine the sampling bias in a realistic manner. So this approach is superior to those depending on topologies produced by topology generators because the generators are usually built on assumptions and abstractions about the Internet [16].

However, to carry out the approach, we have to solve quite a few problems that have already been addressed and unsolved by early work. The reason is that though a few topology inference programs are publicly available, most of them do not open their source code, and therefore can't be modified to fit our requirements. Particularly, we develop our own program and install it on every vantage point to probe the ISP cloud, and we also build software to manage the information of nodes and links collected by all vantage points. Though our work also suffers from the problems faced by other work on topology discovery, e.g. the limitations of traceroute, we argue that our result can be applied to most of the topology data collected by existing systems.

3 Mapping an ISP Cloud

With the help from China Internet Network Information Center and three national bandwidth supervision corporations (e.g., China Telecom) that set up the majority of ISPs in the east of China, we identify an ISP cloud. The ISP cloud connects 21 campus networks and seven city networks, and is part of China Education and Research Network [17].

3.1 Basic Information

Specifically, the ISP cloud can be regarded as a heuristically optimal network like Abilene [14]. In the ISP cloud, the core is a loose mesh of high speed, low connectivity routers which carry heavily aggregated traffic, and this core also is supported by a tree-like structure at the edges in order to aggregate traffic through high connectivity. The ISP cloud consists of a large number of diverse networks, covering 598 separate IP address space (totally 245,878 potential IP addresses), and carries 3.1% of all traffic in China. Fig.1 gives the structure of the ISP cloud.

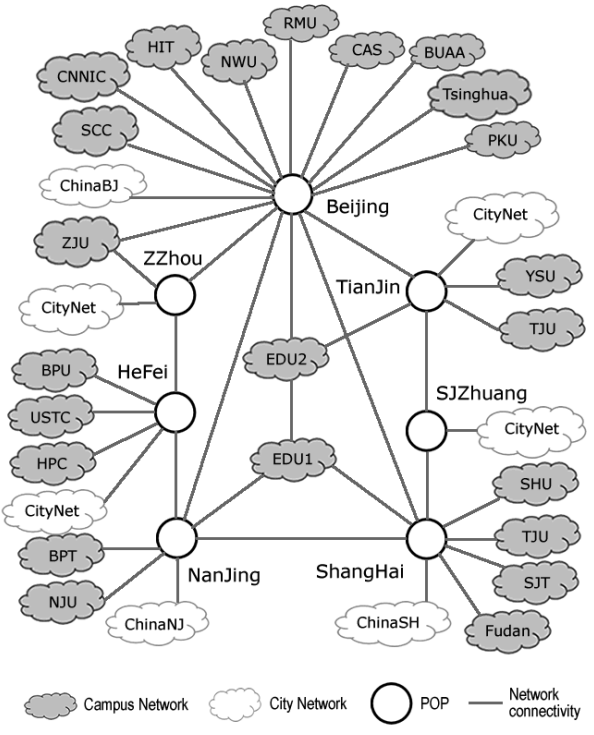


Fig. 1 The structure of the ISP cloud.

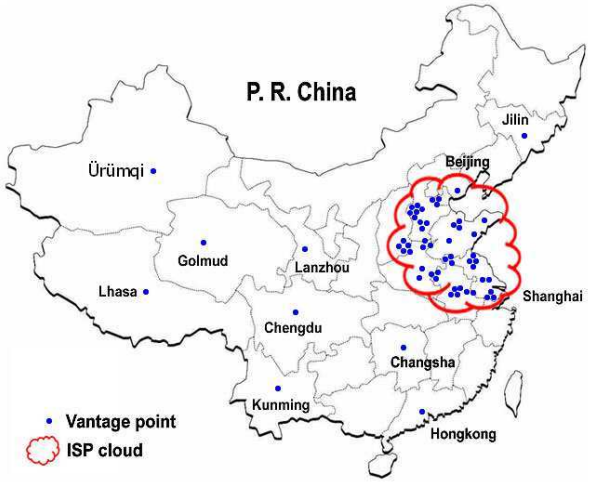


Fig. 2 The approximate location of ISP cloud S_1 and S_2.

More importantly, because of policy reason, the ISP cloud is specially configured so that all traffic traverses between two nodes in the cloud will not go through networks outside the cloud. In other words, the ISP cloud can be regarded as a large-scale, complicated, but independent AS. We are not allowed to give more details since they are regarded as confidential (e.g., the IP address space of Tier-2 and Tier-3 ISPs).

3.2 Vantage Points

Three sets of vantage points are deployed around the world to probe the ISP cloud for its topology. The first set (S_1) contains 49 vantage points, which are spread inside the ISP cloud evenly. The second set (S_2) consists of nine vantage points that are located outside the cloud but inside China. All the vantage points in S_1 and S_2 are volunteer computers in universities and organizations. Fig. 2 plots the geographical location of these vantage points and the approximate scope of the cloud. The last set (S_3) consists of nine vantage points that are outside China. Four of them are public traceroute servers [7] in Asia, and the others are hosts in universities of North American.

We do not use some public measurement frameworks used by RON [18] and PlanetLab [19]. The most of their nodes are outside the ISP cloud in our experiment. We believe S_1, S_2 and S_3 are enough. In this case S_1 is used to infer the topology of the ISP cloud, while S_2 and S_3 are used as an additional facility for checking whether or not S_1 misses any router or link.

Accordingly, S_1, S_2, and S_3 apply different probing strategies. Each vantage point in S_1 is equally assigned an IP list, which includes all the 245,878 IP addresses that the ISP cloud can assign to its hosts. The list is organized in an increasing order. Each vantage point in S_1 probes the ISP cloud as follows (Fig. 3). First, it selects a random position of the list, and starts to retrieve upward IP addresses in the list for probing until the end of the list. After that, it tries backward IP addresses from the point immediately before the originally position until the beginning of the list. In each probe, the vantage point runs three traceroute [20] instances to capture routers along the paths from itself to a given IP address. Because we plan to start all vantage points at the same time, this strategy can avoid overloading the same routers in a short interval.

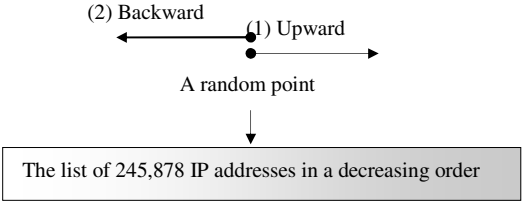


Fig. 3 The probing strategy of every vantage point in S_1.

The vantage points of S_2 and S_3 randomly select IP addresses from the IP list for probing. In addition, an IP address won't be selected if it has already been probed. The reason for doing in this way is many vantage points of S_2 and S_3 are located far outside the ISP cloud, so it is possible that probing adjacent IP addresses in the list may result in "old" paths that share many routers and links.

3.3 Data Collection

All vantage points start to collect topology information in 2:00-2:30 AM BST, September 11, 2005. The data collection process is divided into three steps.

First, each vantage point probes the IP addresses that selects from the IP list. In each probe, it sends out a series of ICMP "Echo-Request" packets with the time-to-live (TTL) fields set to different integers, and then it extracts the source IP addresses of intermediate routers from the ICMP "Time-Exceeded" and ICMP "Echo-Reply" packets returned by these routers [21]. In particular, routers that respond to the neighboring ICMP "Echo-Request" packets sent toward the same destination are assumed to be connected by links. In this way, the vantage points collect information of nodes and links. In S_1, the fastest vantage point takes 46 hours to probe the whole IP list, while the slowest one consumes 53 hours. The vantage points in S_2 and S_3 stop probing when those in S_1 all finish.

Second, after probing, S_1, S_2, and S_3 begin to resolve IP alias. In fact, the nodes that vantage points collect in the above step are the IP addresses of network interface cards (NICs) of routers, and many routers have more than one NIC. Therefore, we have to resolve IP alias, i.e. determining which IP address belongs to the same router. To do so, the vantage points in S_1, S_2, and S_3 implement Ally functions arrange the collected IP addresses in the unit of router. Specifically, vantage points in S_1 try to search for IP alias in all the IP addresses they have collected, while S_2 and S_3 try Ally on those that are of the given IP list (245,878 IP addresses) rather than on all the collected IP addresses. In addition, the public traceroute servers in S_3 do not provide functions for alias resolution, so we later use the result of other vantage points to resolve alias in the datasets collected by these servers. The fastest vantage point takes nine hours to finish alias resolution, while the slowest one takes 17 hours.

Finally, vantage points transmit the topology information (datasets) that collect to a central host, which is a computer in the ISP cloud. But the dataset stored in each vantage point is so large that even the smallest one is about 50 MB, so transmitting these datasets at close time to a host would cause significant network congestion. Specifically, each vantage point begins to transmit its dataset after a random delay ranging from ten minutes to ten hours. The probing and alias resolution steps take 70 hours together; the whole data collection process takes 82 hours.

3.4 Topology Validation

After the datasets from S_1, S_2, and S_3 are stored in the central host, we turn to integrate and validate them. We first implement an extra alias resolution to update the

datasets from S_1, S_2, and non-traceroute-server hosts in S_3 as follows. If there are k datasets containing two IP addresses a_1, a_2, and more than datasets regard a_1 and a_2 as IP aliases, all the k datasets are updated to set a_1 and a_2 IP aliases, otherwise they are updated to reflect the situation that a_1 and a_2 belong to two routers. Finally, the datasets from traceroute servers in S_3 are also updated using all the known IP aliases.

After the above effort at alias resolution, we develop a topology by merging all the datasets from S_1 with the algorithms in [22]. This topology is named the almost complete topology T_C. Particularly, T_C includes 40,166 IP addresses that belong to 25,733 routers, and these routers are connected by 36,029 links.

Now topology T_C is validated in five steps. First, we conduct self-verification on T_C. Second, we use the datasets from S_2 and S_3 to check the completeness of T_C. Third, we compare T_C with the maps released by Skitter [8]. Fourth, we check whether or not T_C contains routers outside the ISP cloud. Finally, we employ the ISPs that we map to help with validation.

We verify T_C using a feature of traceroute. When a vantage point uses traceroute to probe a destination, it sends a series of ICMP packets p_1, p_2, ..., p_n to trigger ICMP replies from all intermediate routers r_1, r_2, ..., r_m in the path from the vantage point to the destination (suppose that $n > m$). By checking the original IP headers encapsulated in the ICMP replies, we are able to know an ICMP packet is returned by router at which hop. If r_j responds but r_i doesn't ($j > i \geq 1$), a router at hop i is loss. In this way, we find T_C loses 2.3% routers and 3.3% links. Note that if r_j isn't the destination and all routers (or hosts) behind r_j do not generate ICMP responses, we won't be aware of the path behind r_j.

Sets S_2 and S_3 observe no more IP addresses (or routers) than S_1, but 22 more links. By checking the geography location of small ISPs inside the ISP cloud and their IP address space, we find that all these 22 links are located at the border of the ISP cloud. Since most of the vantage points in S_1 are located inside the ISP cloud, it is easy to assume that S_1 may lose some edge links (and it does). But T_C can not lose a considerable number of edge links because all 18 vantage points in S_2 and S_3 can only observe about 0.05% additional links. Nevertheless, these 22 links are added to T_C.

The latest datasets released by Skitter do not introduce new routers or links. Skitter detects only 9,093 IP addresses and 15,022 links of the ISP cloud. About 97.9% IP addresses are included by T_C, but the left 2.1% IP addresses are unreachable (even during the validation). We suspect that since Skitter and we start probing at different time (12 days apart), Skitter happens to observe some routers and links that do not exist during our experiment. This indicates that T_C is not an instantaneous topology of the underlying network; instead, T_C is a snapshot of the ISP cloud over a time interval τ. If τ is too long, parts of the snapshot tend to be out-of-date. We argue that since the total time our vantage points take to probe and resolve IP alias is 70 hours, and since end-to-end route won't change over time scales of days to weeks [15], T_C is an accurate snapshot of the ISP cloud.

In addition, T_C includes six routers and 14 links that are outside the valid IP address space of the ISP cloud. Further investigation reveals that the USTC campus network has temporarily routed a small portion of its traffic through a local commercial network when one of its gateways breaks down. But these temporary routes are no longer available after the gateway is repaired. Since these six routers and 14 links are not part of the ISP cloud, they are cut out of T_C. Furthermore, we ask the network operators of seven other campuses if they encounter similar situations during our experiment, and they all report no.

We consult ten ISPs whose networks cover almost half of the ISP cloud, and they confirm that TC misses very few routers or links. The ISPs do not report specific ratios because their networks are so large that they do not have a complete map covering every corner except the backbones. But they claim that they are not aware of about 2% links, which connect to their backbone routers in T_C. These links indeed exist since we can still detect them after the experiment. We suppose this situation is caused by local network operators who arbitrarily deploy fibers between backbone routers without reporting to their administrators immediately. We ask a question about how many routers in the ISP cloud are configured so as not to generate any ICMP packet. All ISPs answer that most of their routers can generate ICMP packets, and they also use ping or traceroute like toolkits for troubleshooting. So we are more confident with the result of the above self-verification on T_C.

Finally, we believe that T_C captures most of the routers and their links of the ISP cloud.

4 Data Analysis

After collecting and validating the topology information, we now evaluate the sampling bias by comparing T_C with the aggregate topology built on a random set of vantage points in S_1. The comparison was focused on topology coverage, metrics, and node degree distribution.

All possible topologies that are built on the information of the 49 vantage points in S_1 are arranged in 49 groups, G_1, G_2, ..., G_{49}. The first group G_1 consists of 49 topologies observed independently by the 49 vantage points. The second group G_2 includes totally =1,176 topologies that are constructed by merging the topology information of every two different vantage points. Similarly, group G_x consists of topologies built on the information of every x different vantage points. Finally, G_{49} has only one topology which combines the views of all the 49 vantage points. Note that the topology in G_{49} is slightly different from T_C, which has 22 additional links observed by S_2 and S_3, and discards six routers and 14 links that are outside the ISP cloud (see topology validation, Section 3).

Fig. 4 shows the maximum and the minimum router coverage of the topologies in select groups G_1, G_5, ..., G_{49}. For example, a topology in G_1 can include at most 69% or at least 52% routers of T_C. We find that even though vantage points in S_1 are assigned the whole list of potential IP addresses of the ISP cloud, many

vantage points still fail to detect a large portion of routers. We check the topologies in G_1 and find many routers in several ISPs are unreachable to the vantage points in some other ISPs due to AS-level policies, as also found in [23]. In addition, a router with the target IP address may respond to our probing packets through NICs that are assigned other IP addresses [12]. Naturally, as the number of vantage points increases, the number of routers they observe increases quickly as well. The "max" column is much higher than the "min" one when the index of group is less than 21. As the index continues to grow, the "min" column catches up gradually.

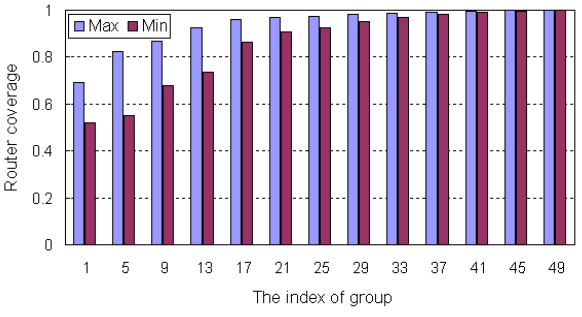


Fig. 4 The maximum and the minimum router coverage of topologies in G_x (x = 1, 5, ..., 49).

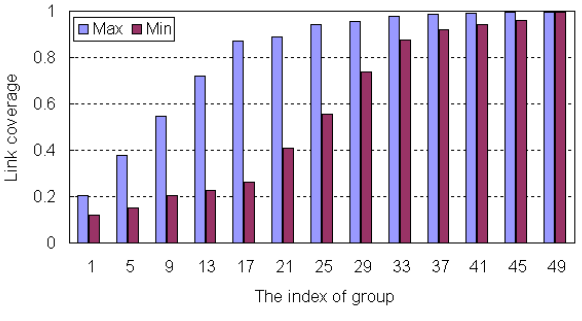


Fig. 5 The maximum and the minimum link coverage of topologies in G_x (x = 1, 5, ..., 49).

Fig. 5 plots the maximum and the minimum link coverage of topologies in G_1, G_5, ..., G_{49}. In contrast to router coverage, all topologies in G_1, G_5, and $G9$ observe a very small portion of links. Moreover, when the index of group x is less than 21, different combinations of x vantage points obtain diverse link coverage.

In fact, what a vantage point observes is a tree-like graph (not necessarily a tree). Particularly, the topologies in G_1, G_5, and G_9 seem like a bundle of trees spread over T_C. Many links are still undetected though the composite of a few trees indeed covers a large portion of T_C. Furthermore, if a set of vantage points

are logically close nodes in T_C, it observes many common links and its link coverage would be comparatively low. In contrast, loosely connected vantage points often reach high link coverage because their views do not share many links. In the rest of this paper, the topology that achieves the highest link coverage in group G_x is termed G_x–Max, while the topology with the lowest link coverage in G_x is named G_x–Min.

5 Discussion

What is the cause of sampling bias? The sampling bias is mainly determined by topology coverage. The percentage of target network that the inferred topology covers can strongly bias the observations of the target network. Furthermore, the number of vantage points and their locations will significantly affect the topology coverage. Finally, an inferred topology is just a snapshot of real network. So the time period during which a network is measured should be as short as possible, otherwise the network would undergo considerable changes and the inferred topology is prone to inaccuracy or incompleteness. Note that this paper focuses largely on link coverage, but the link coverage is enough to tell the importance of topology coverage in sampling bias.

What does the sampled information tell us about the real network? Here, the sampled information refers to the inferred topology, which is sampled because it is usually impossible to obtain a complete and instantaneous picture of target network. Therefore, we have to characterize the network using sampled information, and we suspect that the sampled information can tell us any possible information about the target network. Though the study of an inferred topology would also lead to the same conclusions on a few properties as one does with a real topology, it may not be very safe to assume other properties of the inferred topology match those of the real one.

How to capture an accurate topology with as few measurements as possible? Despite the challenges of mapping networks, it is possible to capture the accurate topology of a target network with a small number of vantage points. The prerequisite is that these vantage points should be placed in suitable locations of target network in order to achieve high all-point-distance. To do so, we need to make careful trade-offs between topology coverage and measurement time. First, to maximize the topology coverage, the all-point distance of a fixed number of vantage points in all available positions must be computed so as to find suitable locations. In addition, an appropriate probing strategy is necessary.

What is an accurate topology, anyway? The answer to this question is metric-specific, meaning that it depends on which metric is under estimation. For example, in our analysis, G13–Max is accurate if only distortion is taken into account, but it is not accurate as resilience is involved. In addition, the answer also varies with the required exact level of metrics. For example, if our purpose is to check whether or not the node degree distribution is a power law rather than calculate the parameters of distribution precisely, most of the inferred topologies seem accurate.

Therefore, to obtain accurate estimation of metrics from a comprehensive perspective, an accurate topology should be the topology that achieves high topology coverage. But how much coverage can be regarded as "high" coverage depends on the required exact level of the metrics that we are interested in.

6 Conclusions

Understanding the sampling bias is very important because it enables us to link an inferred topology to the real network reasonably. This paper systematically evaluates the sampling bias of network topology inference. Our basic idea is to compare inferred topologies with an almost complete topology of a specific and large-scale network from various perspectives. To do so, we identify an ISP cloud, spread vantage points over the ISP cloud and the world, collect topology information by probing a fixed list of IP addresses, merge the views of all vantage points to produce the almost complete topology, which consists of 25,733 routers and 36,029 links, and validate this topology.

We find that sampling bias, if undetected, could significantly undermine the conclusions draw on the inferred topologies. Moreover, an inferred topology that shares the same properties of target network may still be thought inaccurate if other properties are involved. Finally, sampling bias is associated with topology coverage (especially link coverage) that the inferred topology can achieve. To weaken the effect of sampling bias, researchers should carefully select the geography location of vantage points so as to achieve high all-point-distance, focusing on specific metrics, and predict the scope of target network before measurement starts.

Acknowledgment

We gratefully acknowledge the financial support of the Project 211 supported coordinately by the State Planning Commission, Ministry of Education and Ministry of Finance, China.

References

[1] Meyer, D.: Routeviews, http://www.routeviews.org/
[2] Spring, N., Mahajan, R., Wetherall, D., Anderson, T.: Measuring ISP topologies with Rocketfuel. IEEE/ACM Trans. Networking 12(1), 2–16 (2004)
[3] Broido, A., Claffy, K.: Internet topology: connectivity of IP graphs. In: Proc. SPIE ITCom WWW conference, August 2001, pp. 172–187 (2001)
[4] Burch, H., Cheswick, B.: Mapping the Internet. IEEE Computer 32(4), 97–98 (1999)
[5] Govindan, R., Tangmunarunkit, H.: Heuristics for Internet map discovery. In: Proc. IEEE INFOCOM, pp. 1371–1480 (2000)

[6] Breitbart, Y., Garofalakis, M., Jai, B., Martin, C., Rastogi, R., Silberschatz, A.: Topology discovery in heterogeneous IP networks: the NetInventory system. IEEE/ACM Trans. Networking 12(3), 401–414 (2004)
[7] Kernen, T.: traceroute organization, http://www.traceroute.org/
[8] Cooperative Association for Internet Data Analysis (CAIDA): http://www.caida.org/
[9] Faloutsos, C., Faloutsos, P., Faloutsos, M.: On power-law relationships of the Internet topology. In: Proc. ACM SIGCOMM, September 1999, pp. 251–262 (1999)
[10] Chen, Q., Chang, H., Govindan, R., Jamin, S., Shenker, S., Willinger, W.: The origin of power laws in Internet topologies revisited. In: Proc. IEEE INFOCOM (2002)
[11] Willinger, W., Govindan, R., Jamin, S., Paxson, V., Shenker, S.: Scaling phenomena in the Internet: critically examining criticality. Proc. National Academy of Sciences 99(suppl.1), 2573–2580 (2002)
[12] Paxson, V.: Measurements and analysis of end-to-end Internet dynamics. Ph.D. dissertation, Univ. California, Berkeley (1997)
[13] Lakhina, A., Byers, J.W., Crovella, M., Xie, P.: Sampling biases in IP topology measurement. In: Proc. IEEE INFOCOM, pp. 332–341 (2003)
[14] Abilene Network., http://www.internet2.edu/abilene/
[15] Zhang, Y., Duffield, N., Paxson, V., Shenker, S.: On the constancy of Internet path properties. In: Proc. ACM SIGCOMM conference on Internet measurement, pp. 197–211 (2001)
[16] Floyd, S., Paxson, V.: Difficulties in simulating the Internet. IEEE/ACM Trans. Networking 9, 392–403 (2001)
[17] China Education and Research Network (CERNet),, http://www.edu.cn/HomePage/english/cernet/index.shtml
[18] Resilient Overlay Networks (RON), http://nms.lcs.mit.edu/ron/
[19] PlanetLab, http://www.planet-lab.org/
[20] Prtraceroute, http://www.isi.edu/ra/RAToolSet/prtraceroute.html
[21] Postel, J.: Internet control message protocol. IETF, RFC 792 (1981)
[22] Zhou, H., Wang, Y.: RichMap: combining the techniques of bandwidth estimation and topology discovery. Journal of Internet Engineering 1(2), 102–113 (2007)
[23] Tangmunarunkit, H., Govindan, R., Shenker, S., Estin, D.: The impact of policy on Internet paths. In: Proc. IEEE INFOCOM (2001)

Computer Network Reverse Engineering

Hui Zhou, Wencai Du, Shaochun Xu, and Qinling Xin

Abstract. Software reverse engineering has undergone many milestones and stepped from research to industry quickly in recent ten years. By analogy, we have found that it is also possible to apply reverse engineering to computer networks. The goal of network reverse engineering is to annotate a map of the networks with properties such as node distribution, connectivity, and bandwidth usage. It is necessary, but also challenging, to employ reverse engineering to computer networks. To do this, we first comparatively analyze the reverse engineering of both software and network from five basic perspectives: source, data analysis, presentation, validation, and prediction. And then, RichMap system has been developed to automatically infer the topology and link available bandwidth of a network. The experiment result indicates that, after applying the object snapshot concept of software, RichMap can smoothly capture and present complete router-level snapshots and significantly decrease the network load that it generates.

1 Introduction

Software reverse engineering is, in practice, one of the most important endeavors in software engineering. This stems from the fact that software systems are complex and often poorly specified and documented. As a result, software practitioners need to spend a substantial amount of time understanding the source code from a structural and behavioral perspective, before carrying out any maintenance task. In this context, most reverse engineering processes follow the same pattern: a program is analyzed through static or dynamic analysis and the collected low-level program information is transformed into a higher level, more abstract presentation. The presentation helps engineers understand the rationale of the code and thus facilitate future refactoring.

Given the dynamic nature of the Internet, keeping track of network information manually is a daunting (if not impossible) task. Network operators generally can't

Hui Zhou · Wencai Du
Hainan University, Renmin Ave. No. 58, 570228, Haikou, China
e-mail: wencai@hainu.edu.cn

Shaochun Xu
Algoma University, Sault Ste, Marie, Ontario, P6A2G4, Canada

Qinling Xin
Central China University of Technology, Wuhan, China

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 227–239.
springerlink.com

draw a complete map of their networks since many internal parts can undergo different scales of changes but will not report these changes immediately. Therefore, they need reverse engineering systems to detect underutilized and congested links, plan network capacity upgrades, and deploy security infrastructure. In addition, many users also need to verify whether they get the network service stated in their service-level agreements with the Internet service providers (ISPs).

It has become obviously necessary to employ reverse engineering to computer networks. However, network reverse engineering is a challenging task. The key reason is that the design of the Internet can't provide explicit support for end nodes to obtain information about the network internals. A network typically consists of many small networks; such networks are under different administrative control, so there is no single place from which one can obtain a complete picture of the specified target network. Furthermore, the Internet is so heterogeneous that an approach found to be useful in a certain networks may not be effective elsewhere [1].

This paper makes two contributions. First, we analyze the differences between software reverse engineering and network reverse engineering from five basic perspectives: source, data analysis, presentation, validation, and prediction. In addition, we build RichMap system, which need to be installed on a single client host, to characterize and monitor its surrounding computer networks. The experiment result proves that, after adopting the snapshots concept from software domain, RichMap is able to present a series of router-level views of a large-scale network. And it can also effectively illustrate the changes of topology, congested links, and delay without injecting noticeable probing packets into target network.

This paper is organized as follows. Section 2 summarizes the related work on network reverse engineering domain. Section 3 analyzes the reverse engineering techniques of both software and networking, and then Section 4 presents RichMap, which draws a series of streaming snapshots about designated networks. Section 5 discusses our findings, and finally Section 6 concludes the paper.

2 Related Works

The field of software reverse engineering and its closely related fields, such as program comprehension or software analysis, have undergone many successes over the past 20 years. In addition, software reverse engineering environment has been equipped with various intelligent tools: extractors, analyzers, and repositories [2]. During the same time, along another thread, network community has introduced quite a few measurement systems to gathering and presenting the information of network properties [3]. The theories, protocols, techniques, tools, overlay framework, and the released data archives have initially make up the main body of network reverse engineering.

Basically, the reverse engineering of network mainly starts from measurement. Specifically, a router can be configured to passively record the information about its own performance, e.g. the number of packets received/sent by each of its network interface cards (NICs). A typical example is network traffic monitoring. Fig. 1 illustrates the bytes sent through the USENET bulletin board system, averaged over two-week intervals.

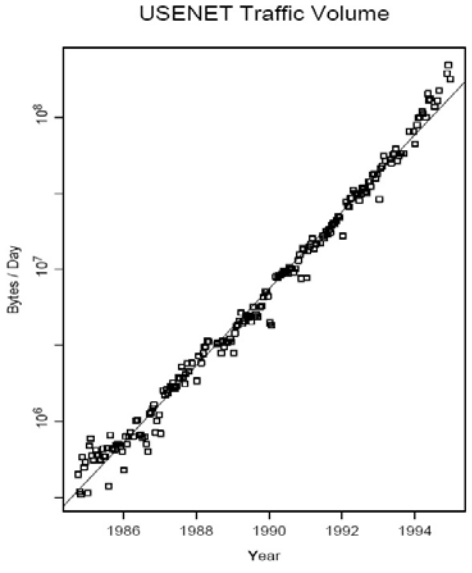


Fig. 1 USENET traffic monitoring information [4].

Furthermore, the measurement literature can further be classified according to different measurement targets: node, link, topology. Learning the role that a node plays is the first step to understand the network. Basically, each node has one of the following roles: client host; access router that aggregates the traffic from clients; and backbone router that transmits a large volume of traffic. The role problem has been frequently addressed, e.g. Rocketfuel [5] uses IP prefixes, DNS information, and topological ordering to identify role. In addition, many tools search for the bottleneck node with diverse heuristics [6].

In addition to role, the behavior of node has been a key reverse engineering target. For example, TCP features and supporting network services both can affect the composition of traffic of a node [7]. Practically, a node may hold multiple NICs, each with a different IP address. To provide a reasonable node-level, instead of IP-level, network analysis, we must decide which interface belongs to the same node [5].

Besides node, link is another important component. Generally, a link is the IP connection between two nodes that are only one IP-hop away from each other. Much research has been done to capture the usability, delay, and bandwidth capacity of a single link. Recently, the research community extends the study of link to end-to-end path, which can be regarded as a line of connected links. Measuring the properties of a path is very meaningful since it enables us to better understand how packets flow between nodes. Typically, tools use Internet control message protocol (ICMP) [8] timestamps to estimate the delay variation.

In addition to delay, the available bandwidth of path has attracted much attention since 1990s. Specifically, the available bandwidth is defined as the maximum rate that a path can provide to a packet flow, without reducing the rate of other flows in the path [9]. Measuring the instantaneous end-to-end available bandwidth is extremely difficult. We have examined 11 well-known available-bandwidth

measurement tools, and found that quite a few basic problems, e.g. system timing and end-host throughput, which can always lead to different scales of bias [1].

Finally, topology auto-discovery has strongly driven the study of active probing measurement. Network community has examined five categories of topologies: the graphs of connections between autonomous systems (ASs) [5], the point-of-presence (POP) topologies that interpret the structure of backbone using geography information, the IP-level topologies whose nodes are IP addresses and whose links are connections between the IP addresses, the router-level topologies that resolve IP aliases and group the IP addresses in the unit of router, and the connectivity of physical components.

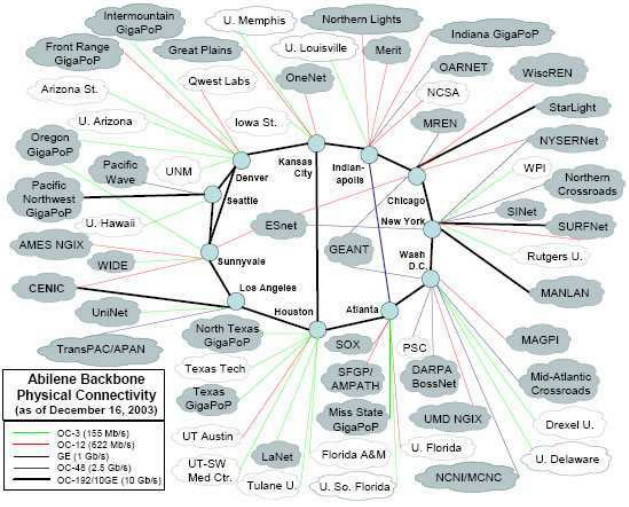


Fig. 2 The discovered topology of Abilene backbone [11].

For example, Breitbart et al. detected 3,888 nodes and 4,857 links in 2003 [3]. RocketFuel outputted a topology consisting of 228,263 nodes and 320,149 links in 2004 [5]. An ongoing project, Skitter, has been scanning the whole Internet for several years with tens of commercial network hosts, and it has released extensive graphs of Internet IP-level topologies [10]. As an example, Fig. 2 gives the result of a topology discovery work; the target network is Abilene backbone [11].

3 Comparative Analysis

We comparatively analyzed the reverse engineering of software and network from five basic perspectives: source, analysis, presentation, validation, and prediction.

3.1 Source

The source of software reverse engineering is code and code-related files such as log. Generally, software reverse engineering depends on performing some analysis of the source code in order to produce one or more models of the system under

analysis. Generally, source code is written by software engineers according to the well-designed specification of programming languages, e.g. ASM, Pascal, C/C++, and Java. A language often comes with a specification, to which compiler developer and software engineer must conform. Furthermore, the coding process is supported by various integrated development environments. As a result, no matter how well (or bad) the code is organized, software reverse engineering tools is built on a solid basis, i.e. the tools do understand the exact meaning of each line of code.

Unlike the source of software reverse engineering, the one of network reverse engineering mainly comes from measurement, and it is highly volatile. The volatility can be perceived in almost every parameter that we attempt to measure. For example, the round-trip time (RTT) of a pair of nodes is an important metric of network performance. Generally, RTT can be used as an indicator of end-to-end transmission quality. Here we attempt to measure the RTT of a short path, i.e. two directly connected computers C1 and C2. First, C1 sends an ICMP echo-request packet to C2. When C2 receives the packet, it immediately sends an ICMP echo-reply packet back to C1. In each active probe, the time from sending out an ICMP echo-request to receiving the corresponding echo-reply is regarded as a candidate of RTT. As shown in Fig. 3, the RTT is ever-changing with network traffic and time.

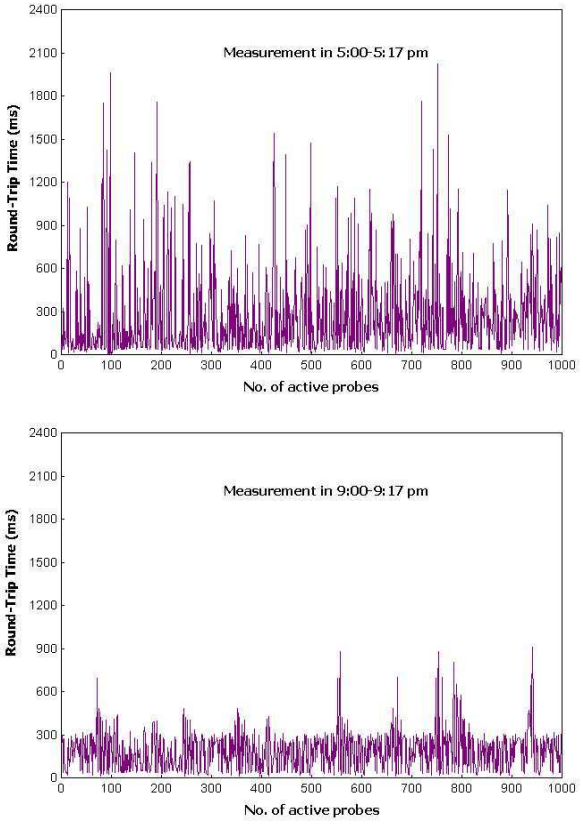


Fig. 3 Round-trip time of two directly connected computers.

3.2 Analysis

To analyze the source code, a software reverse engineering tool will first scan the source code. In most cases, reverse engineering tool assumes that the target source files won't undergo any change during the scan, which is done once and for all. In a very limited time interval, the source of software is safe to be regarded as static, while network is always a moving target. As a result, network tools must continuously collect the information about the designated network, in a never-ending style.

Moreover, as to network reverse engineering, analyzing the data source is challenging since it generally contains too much noises. But the analysis is valuable since it often provide insight into the network. For example, Faloutsos et al. discover some surprisingly simple power-laws of the network topologies [12]. These power-laws hold for three topologies between November 1997 and December 1998, despite a 45% growth of its size during that period. As shown in Fig. 4, log-log plot of the out-degree dv versus the rank rv in the sequence of decreasing out-degree.

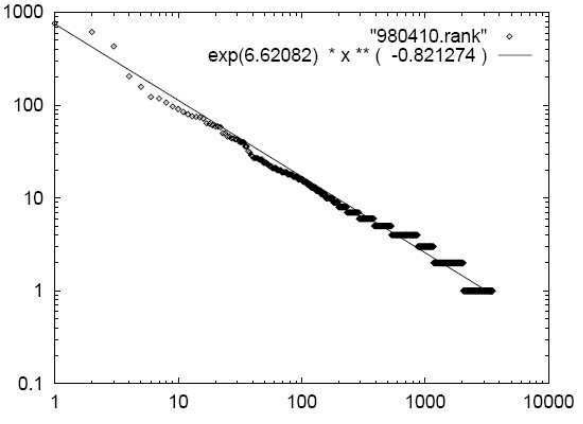


Fig. 4 The rank plots on dataset Intel-98 [12].

3.3 Presentation

Suppose that the presentation of software reverse engineering is a snapshot, the one of network reverse engineering can be regarded as a video. The parameters of target network can undergo changes as time passes, and thus lead to high dynamics. As shown in Fig. 5, the IP conversations of LAN captured by Sniffer Pro, which is a network packet sniffing tool installed in one node [13]. Since the target network is ever-changing, the presentation must trace the changes and output pictures that match.

Compared with software reverse engineering, the network reverse engineering tools can't support large-scale reuse since there isn't a universal accepted presentation standard. It is also hard to establish such a standard because each reverse engineering tool is built to study a specific question and work in a specific network environment.

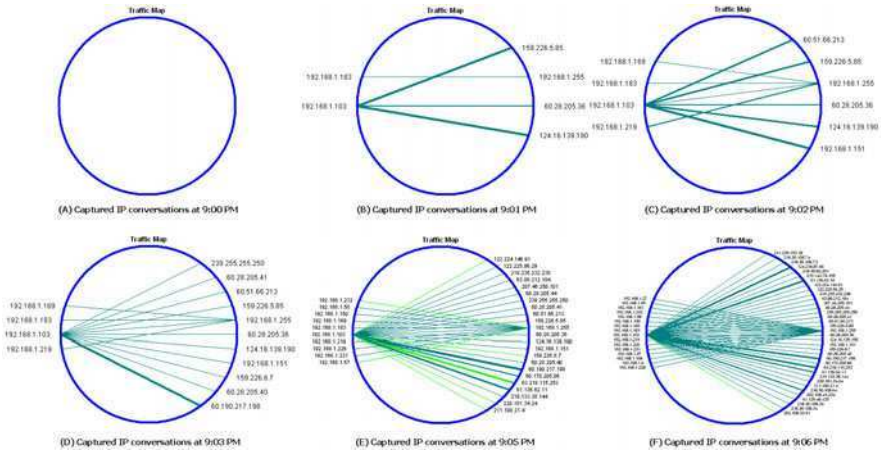


Fig. 5 IP conversations captured by Sniffer Pro in 9:00 – 9:06 PM.

3.4 Validation

To validate the available bandwidth of a path, researchers have introduced many inspiring techniques. It seems that comparing the estimation result with closely estimated bulk TCP throughput over the same path is a good idea [14]. However, available-bandwidth and bulk TCP throughput are indeed different. The former gives the total spare capacity in the path, independent of which transport protocol attempts to capture it. While the latter depends on TCP's congestion control. Fig. 6 typically shows the measurement result of the available bandwidth of an end-to-end path, which starts from Hainan University and ends at Chinese Academy of Sciences. In particular, Cprobe [15] and BNeck [6] are installed on hosts inside Hainan, Pathload [16] is installed in both end points, while TCP throughput is tested by maximized the parallel TCP connections of Iperf [17]. It is apparent that there isn't a curve that can exactly match the other.

As a result, we are not able to completely validate end-to-end available bandwidth. Furthermore, it is very hard to make sure the data we collect reflects the exact network status, even if we have success experience on a limited number of networks. The same problem is faced by almost all measurement techniques that rely on active probing. And this thus makes the network reverse engineering more challenging than its software counterpart.

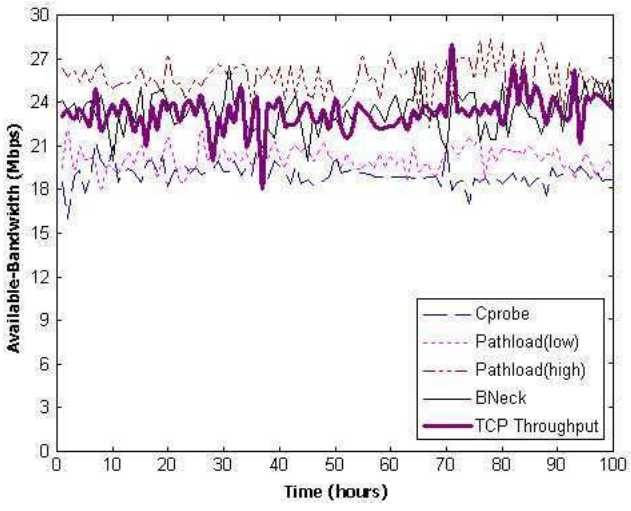


Fig. 6 Available bandwidth measured by different tools.

3.5 Prediction

Recently, there is a growing need of reverse engineering tools to support the prediction of changes in source. For example, through analyzing the history of the lines of code, managers can predict the code scale of a Java program in the next development iteration [2]. Surprisingly, though network contains much more noise than stationary software source code, many useful rules have been extracted, and used to predict the macro-behavior of networks.

Diurnal patterns of activity: It has been recognized for more than thirty years that network activity patterns follow daily patterns, with human-related activity beginning to rise around 8-9AM local time, peaking around 11AM, showing a lunch-related noontime dip, picking back up again around 1PM, peaking around 3-4PM, and then declining as the business day ends around 5PM. The pattern often shows renewed activity in the early evening hours, rising around say 8PM and peaking at 10-11PM, diminishing sharply after midnight. Originally, this second rise in activity was presumably due to the "late night hacker" effect, in which users took advantage of better response times during periods of otherwise light traffic load.

Self-Similarity: Longer-term correlations in the packet arrivals seen in aggregated Internet traffic are well described in terms of self-similar processes [18]. "Longer-term" here means, roughly, time scales from hundreds of milliseconds to tens of minutes. The traditional Poisson or Markovian modeling predicts that longer-term correlations should rapidly die out, and consequently that traffic observed on large time scales should appear quite smooth. Nevertheless, a wide body

of empirical data argues strongly that these correlations remain non-negligible over a large range of time scales. While on longer time scales, non-stationary effects such as diurnal traffic load patterns (see previous item) become significant. On shorter time scales, effects due to the network transport protocols, which impart a great deal of structure on the timing of consecutive packets, appear to dominate traffic correlations [19].

4 RichMap

To start network reverse engineering, and to accurately capture the running status of a network, we developed an experimental system: RichMap [21]. RichMap has three basic features. First, it is built on active probing technique, and is a single-node system instead of an overlay network system like Planet-Lab [22] that requires software to be installed on many nodes. Second, it automatically discovers the node-level topology of surrounding network, as well as link available bandwidth and delay variation. Finally, it utilizes the snapshot concept from software domain, and builds series of easy-to-understand network maps smoothly.

From boot time, RichMap starts a process to continuously measure the target network. When the RichMap is requested, it presents a map. If the request happens after the end of a measurement cycle and before the start of a new cycle, RichMap updates the repository with the information collected in the latest cycle. But, most of the time, the request occurs during the course of current cycle. At this time, RichMap displays the reverse engineering result of current cycle over the map of the last cycle, while the nodes and links of old map (judged by timestamp) are shadowed.

To evaluate RichMap, we installed it on a node that was in the same LAN of a backbone router in Tsinghua University, and configured RichMap to reverse engineer the network of teaching building No. 3. Fig. 7 gives the 54th and 60th hour snapshots of the outputted map. We observed that there were about ten high-speed links, connecting many local networks. About eight networks were built with high-performance equipments, while many others were not. It was also valuable to note that only the nodes with public IP addresses were drawn, a large number of nodes owned by individual department and accessed the Internet through network address translation technology were not included.

We also found that the available bandwidth of backbone links was steady, while the available bandwidth of non-backbone links tended to fluctuate. Link available bandwidth of the 54th-hour snapshot was generally higher than that of the 60th-hour one. The reason was that the 54th-hour snapshot was collected at night, while the 60th-hour one was in the morning.

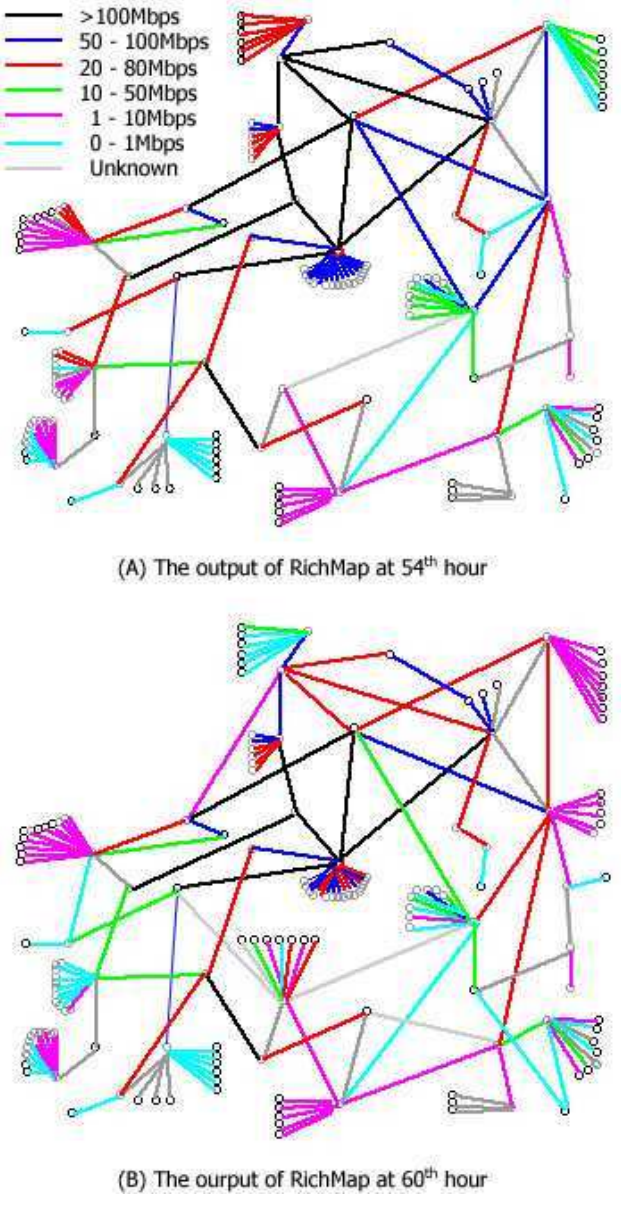


Fig. 7 Snapshots outputted by RichMap at 54th and 60th hour.

Besides the smooth presentation effect, adopting the snapshot idea could significantly decrease the network load. As shown in Fig. 8, when RichMap closed the snapshot option, it needed to actively probe the network one cycle by another. When the option was open, RichMap could pause a while in between two adjacent cycles. This was very useful especially when we choose to reverse engineering the

network at a specific time, and we found the number of nodes discovered by RichMap, no matter it turned on the option or not, were almost the same (Fig. 9).

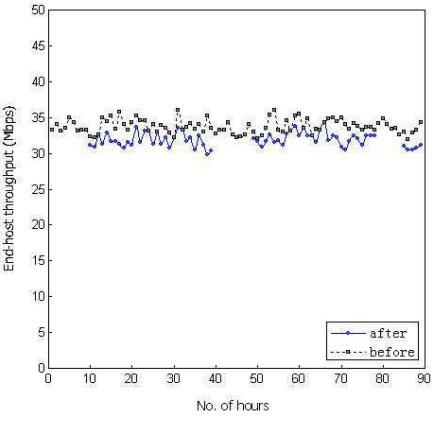


Fig. 8 End-host throughput.

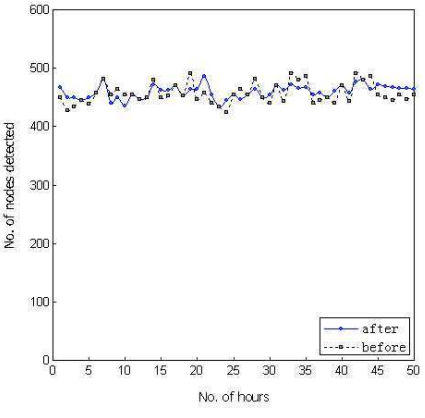


Fig. 9 The number of nodes detected by RichMap.

5 Conclusions

Reverse engineering is the process of studying the design of an object from its implementation. Reverse engineering has long rooted in software field, and now we found it useful to promote creative applications for the computer networks. A typical sample is the RichMap system; the snapshot concept enables it to present a series of steady maps of target network. With RichMap, we discuss the possibility and benefit of network reverse engineering, and argue that the reverse engineering is within the reach of both software and network communities.

Acknowledgment

We gratefully acknowledge the financial support of the Project 211 supported coordinately by the State Planning Commission, Ministry of Education and Ministry of Finance, China.

References

[1] Zhou, H., Wang, Y., Wang, X., Huai, X.: Difficulties in Estimating Available-bandwidth. In: Proceedings of IEEE International Conference on Communications, pp. 704–709 (2006)
[2] Kienle, H.: Building Reverse Engineering Tools with Components. Ph.D. Thesis, Department of Computer Science, University of Victoria, Canada; 325 p (2006)
[3] Breitbart, Y., Garofalakis, M., Jai, B., Martin, C., Rastogi, R., Silberschatz, A.: Topology Discovery in Heterogeneous IP Networks: the NetInventory System. IEEE/ACM Trans. Networking 12(3), 401–414 (2004)
[4] Thompson, K., Miller, G., Wilder, R.: Wide-area Internet Traffic Patterns and Characteristics. IEEE Network, 10–23 (1997)
[5] Spring, N., Mahajan, R., Wetherall, D., Anderson, T.: Measuring ISP Topologies with Rocketfuel. IEEE/ACM Trans. Networking 12(1), 2–16 (2004)
[6] Zhou, H., Wang, Q., Wang, Y.: Measuring Internet Bottlenecks: Location, Capacity, and Available Bandwidth. In: Proceedings of International Conference on Computer Network and Mobile Computing, pp. 1052–1062 (2005)
[7] Padhye, J., Floyd, S.: Identifying the TCP Behavior of Web Servers. In: Proceedings of. ACM SIGCOMM (2001)
[8] Postel, J.: Internet Control Message Protocol. IETF RFC 792 (September 1981)
[9] Dovrolis, C., Ramanathan, P., Moore, D.: Packet Dispersion Techniques and a Capacity Estimation Methodology. IEEE/ACM Trans. Networking 12, 963–977 (2004)
[10] Cooperative Association for Internet Data Analysis (CAIDA), http://www.caida.org/
[11] Abilene Network, http://www.internet2.edu/abilene
[12] Faloutsos, M., Faloutsos, P., Faloutsos, C.: On Power-law Relationships of the Internet Topology. In: Proceedings of ACM SIGCOMM, Cambridge, USA (1999)
[13] Sniffer Pro, http://www.netscout.com/
[14] He, Q., Dovrolis, C., Ammar, M.: On the Predictability of Large Transfer TCP Throughput. Computer Networks 51(14), 3959–3977 (2007)
[15] Carter, R., Crovella, M.: Measuring Bottleneck Link Speed in Packet-switched Networks. Performance Evaluation 27(28), 297–318 (1996)
[16] Jain, M., Dovrolis, C.: End-to-end Available Bandwidth: Measurement Methodology, Dynamics, and Relation with TCP Throughput. IEEE/ACM Trans. Networking 11(4), 537–549 (2003)
[17] Tirumala, A., Qin, F., Dugan, J., Ferguson, J., Gibbs, K.: Iperf - The TCP/UDP Bandwidth Measurement Tool, http://dast.nlanr.net/Projects/Iperf/

[18] Zhang, Y., Duffield, N., Paxson, V., Shenker, S.: On the Constancy of Internet Path Properties. In: Proceedings of ACM SIGCOMM conference on Internet measurement, pp. 197–211 (2001)
[19] Paxson, V.: End-to-end Internet Packet Dynamics. In: Proceedings of ACM SIGCOMM (1997)
[20] Yuvrai, A., et al.: Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage. In: Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation (2009)
[21] Zhou, H., Wang, Y.: RichMap: Combining the Techniques of Bandwidth Estimation and Topology Discovery. Journal of Internet Engineering 1(2), 102–113 (2008)
[22] Turner, J., et al.: Supercharging Planetlab: a High Performance, Multi-application, Overlay Network Platform. ACM SIGCOMM Computer Communication Review 37(4), 85–96 (2007)
[23] Gkantsidis, C., Karagiannis, T., Vojnovi, M.: Planet Scale Software Updates. ACM SIGCOMM Computer Communication Review 36(4), 423–434 (2006)

CUDA-Based Genetic Algorithm on Traveling Salesman Problem

Su Chen, Spencer Davis, Hai Jiang, and Andy Novobilski

Abstract. Genetic algorithm is a widely used tool for generating searching solutions in NP-hard problems. The genetic algorithm on a particular problem should be specifically designed for parallelization and its performance gain might vary according to the parallelism hidden within the algorithm. NVIDIA GPUs that support the CUDA programming paradigm provide many processing units and a shared address space to ease the parallelization process. A heuristic genetic algorithm on the traveling salesman problem is specially designed to run on CPU. Then a corresponding CUDA program is developed for performance comparison. The experimental results indicate that a sequential genetic algorithm with intensive interactions can be accelerated by being translated into CUDA code for GPU execution.

1 Introduction

Genetic algorithm (GA) and other stochastic searching algorithms are usually designed to solve NP-hard problems [3]. The traveling salesman problem (TSP) is a famous NP-hard problem [5][10]. It aims to get the shortest wrap-around tour path for a group of cities. Since NP-hard problems cannot be solved in acceptable time, people aim to find acceptable solutions in acceptable time instead. To achieve this, various heuristic algorithms, such as the genetic algorithm, ant algorithm, tabu search, neural network, etc., are designed.

Genetic algorithm was inspired by the evolvement of chromosomes in the real world, which includes crossover, mutation, and natural selection. Viewing chromosomes as solutions to a TSP problem, crossover and mutation are

Su Chen · Spencer Davis · Hai Jiang · Andy Novobilski
Department of Computer Science
Arkansas State University, Jonesboro, AR, 72467, USA
e-mail:{su.chen,spencer.davis}@smail.astate.edu,
{hjiang,anovobilski}@astate.edu

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 241–252.
springerlink.com

changing phases for chromosomes, while natural selection is a sifting phase that will wash out the worst solutions so that better ones will stay.

To simulate this process in a computer program, programmers have to design sequences of numbers to represent chromosomes and perform certain operation on them. Different Problem will have different types of chromosome designs. For example, select participants from a group will have a design of $a_1a_2a_3...a_n$ as its chromosome, where a_i are either 0 or 1, where 0 means unselected and 1 means selected.

As the problem size increases, it takes a very long time to reach an optimum solution, or even a less-optimum but satisfying solution. In order to shorten the convergence time, artificial intelligence is usually introduced to make algorithms efficient. For the traveling salesman problem, 2-opt is a specifically designed mutation operator which takes longer time than ordinary operators, but guarantees fast and steady convergence. However, even with this efficient operator, computing time is still quite long when problem size is large. Recently, NVIDIA's CUDA programming paradigm enables GPU as a new computing platform [1][2]. Many-core GPUs can explore parallelism inside Genetic Algorithms for execution speedup and provide a cost effective method of implementing SIMD type solutions.

This paper intends to develop a heuristic genetic algorithm on TSP and then parallelize it with CUDA on GPUs for performance gains. The rest of the paper is organized as follows: Section 2 discuss the deployment of genetic algorithm on TSP problem. Section 3 addresses the issues of genetic algorithm implementation on GPUs with CUDA. Section 4 provides performance analyses on both CPU and GPU. Section 5 gives the related work. Finally, our conclusions and future work are described.

2 Genetic Algorithm on TSP

GA's input usually includes a waypoint number and a distance table. The Output of GA should be an optimized chromosome chain that represents the order of cities that the traveling salesman should follow. The general process of GA is given in Fig. 1.

The Initialization phase generates a group of chromosomes as shown in Fig. 2. The group size can influence quality of the final result and running time. Therefore, it needs to be properly chosen. Generally, when the group size increases, results are potentially better whereas the running time increases. Factors for both good results and a reasonable running time should be considered.

Crossover phase is an important part in GA to simulate the action where two chromosome individuals exchange partial sections of their bodies. This process helps increase diversity as well as exchange better genes within population. Crossover on real chromosomes is illustrated in Fig. 3. Unfortunately, in TSP, there are no two same numbers in one chain. Therefore, it is

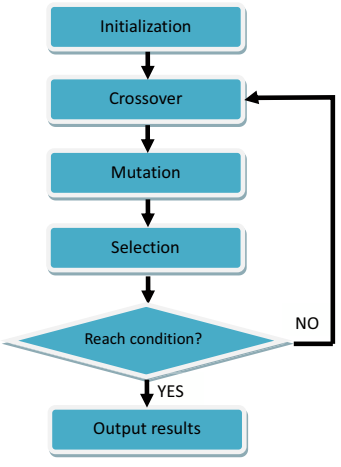


Fig. 1 Flow chart of the general process in Genetic Algorithm (GA)

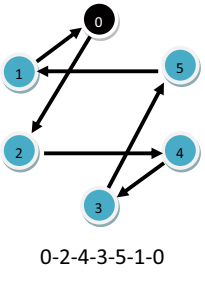


Fig. 2 Initial chromosome sequence generated from Genetic Algorithm (GA)

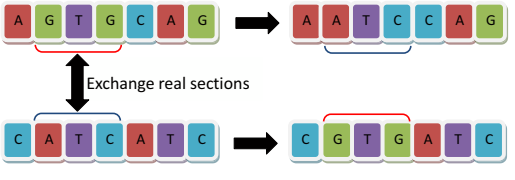


Fig. 3 A crossover example with actual exchange in the real world

impossible to do crossover directly, as chromosomes do in the real world. However, there are alternative ways to simulate this process. The strategy used by this paper is based on sequence orders not values, as shown in Fig. 4 where the crossover of the selected portion of chromosomes is reasonably done.

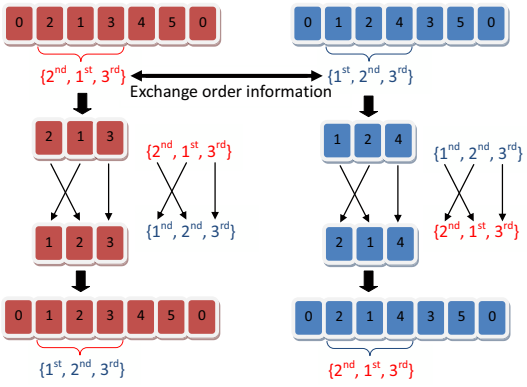


Fig. 4 The crossover in the GA on Traveling Salesman Problem (TSP)

Since good chromosomes are forced to stay in population and pass down their heritage information by crossover, after generations, chromosomes will assimilate each other. The mutation phase is designed to make unpredictable changes on chromosomes in order to maintain the variety of the population. Mutation operators can be arbitrarily designed but the effects taken by them will be hard to tell. Some mutation operators will slow down the convergence process, while others will accelerate it. In this paper, we select 2-opt as the mutation operator, which can make the algorithm converge much faster than ordinary GA. The 2-opt mutation operator is specifically designed to solve TSP and guarantees both diversity and steady evolvement [5][10]. However, this operator takes $O(n)$ time, and has larger time cost than that of simple operators. Details in 2-opt is given in Fig. 5.

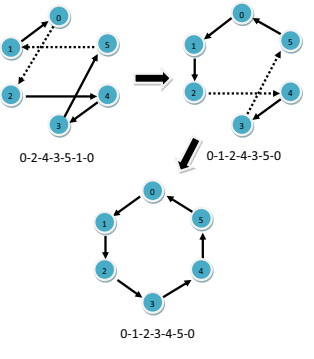


Fig. 5 One possible mutation example in Genetic Algorithm

Selection phase is usually placed after crossover and mutation. In this paper, simplest selection method is adopted and only better solutions are accepted. Each new chromosome will be compared with the older one and the better of the chromosomes stay in the population.

A termination condition should be set to stop the evolution process. In this paper, when the best result has stopped evolving for some generations, algorithm will stop and output the result. Though better solutions are expected when execution time becomes longer, after solutions are convergent, the probability for GA to update the best result becomes extremely small.

3 Genetic Algorithm Implementation with CUDA

3.1 CUDA Platform and GPU Architecture

CUDA (Compute Unified Device Architecture), developed by NVIDIA, is a parallel programming paradigm [1][2]. While graphics cards were originally designed only to process image and video flows, CUDA provides a platform to solve any general purposed problem on GPU. Rather than fetching image pixels concurrently, now threads in the GPU can run common tasks in parallel; however, as in other parallel programming platforms, task dependency problems should be considered by programmers themselves.

Besides hundreds of threads, Fermi (The latest GPU Architecture in 2010) provides shared memory that can be accessed by threads within the same block extremely fast. Shared memory can be thought of as a cache that can be directly manipulated by users. When the input of the problem is small and all its intermediate results can be loaded into shared memory, Fermi will do excellent job. On the other hand, if the input size is relatively large, the utilization of shared memory should be carefully considered.

Limitations of CUDA cannot be ignored. Recursion and pointers for functions are still not supported and debugging is a tedious job. The bus latency between the CPU and GPU exhibits as a bottleneck. All of these limitations should be avoided or considered during programming, and CUDA's architecture should be taken advantage of in their code.

3.2 Opportunities for GA with CUDA

Usually, in order to guarantee the diversity of species, GA maintains a group, or population, consisting of a good number of chromosomes. This can be thought of as the desire of the problem solver to create more directions in order to search a bigger area. It can be understood as that if more ants are dispatched to different directions, chance to find food becomes greater. In GA, these ants communicate with each other frequently and change their

searching directions based on the information they get. Though it contains many interactions and dependencies, it is possible to be parallelized for performance gains.

The most reasonable way to parallel this process is to map activities of chromosome individuals to separated threads. Since all chromosomes will do the same job, they roughly finish at the same time. This property prevents cores from being idle. Otherwise, synchronization will drag down performance severely.

CUDA platform and Fermi architecture provide good tools to parallelize the algorithm. First, GPU supplies hundreds of cores for executing threads in parallel. Second, threads can talk to each other easily and fast because they share address space in several levels such as shared memory and global memory levels. Third, as an extension of C, CUDA eases the programming task.

3.3 Random Number Generation in CUDA

Since GA is a stochastic searching algorithm, a random numbers generation strategy is required. Unfortunately, CUDA does not provide one yet. However, a pseudo random number generator can be easily simulated in different ways. In this paper, bit shifting, multiplication and module operators are used to generate random numbers.

CUDA programs on GPUs may slow down when threads compete for random seeds. To solve this problem, random seeds are generated in the CPU and assigned to each GPU thread. Equipped with a simple random number generator function, threads in the GPU can generate random numbers simultaneously without blocking or false sharing.

3.4 Data Management for GA

In CUDA architecture, threads can be arranged in blocks and grids to fit applications. In the latest Fermi architecture, cache and shared memory coexist to enable GPU cores to behave as CPU. This provides a greater chance to get better performance. Shared memory is as fast as cache and can be directly manipulated by users. However, shared memory space can only be accessed by threads within one block.

If shared memory is big enough for everything, programmers do not have to spend too much time on data manipulation. However, with limited shared memory size, only few frequently used variables and arrays have priorities to reside in it. For the GA on TSP problem, the distance table and chromosome group occupy the majority of the space and both are too large for shared memory. Since distance between two cities is Euclidean distance in this paper, the distance table can be discarded and coordinate arrays are used instead. This change will definitely harm CPU's performance because of duplicated

calculations. However in GPU, such computation redundance is encouraged since there always are many idle cores due to memory access latency. This design is proved to be valid by experimental results.

3.5 Parallelization of GA

Because each thread has an independent seed for random number generation, different threads can initialize chromosomes simultaneously. Since mutation of one chromosome has nothing to do with other chromosomes in this paper, there are no task dependencies between any two threads in these phases.

For the crossover part however, threads tend to find a peer to exchange information. Since threads are working on this part together, it is not possible for them to work on original chromosomes directly. Copies have to be made before crossover phase starts. However, these copies still cannot be changed directly because it is possible that two chromosomes choose the same target to communicate with. Since this operation not only reads, but also changes data, working directly on the copies is still not allowed. Therefore, each thread should make another temporary copy for the target chromosome to work on.

After crossover phase, the copy for previous group will be used as the group of the last generation. After mutation phase, selection phase needs to compare current chromosome and previous version and decide which one is better, and therefore stays. This process can be directly parallelized since no communication and task dependency exist among threads.

Updating the best chromosome needs to search for the minimum one in the adaptive value array for the new chromosome group. This can be implemented in complexity of $O(logn)$ using n processors instead of $O(n)$ done sequentially. The existence of shared memory and cache can reduce this time to an insignificant level, even doing it sequentially in one thread.

3.6 Synchronization in CUDA

Based on the task dependency analysis, necessary synchronization points have been detected and inserted as in Fig. 6. Synchronization needs to be addressed at four positions:

1. The copy phase should wait till best value is updated.
2. The crossover phase should wait till copy phase finishes.
3. Updating the best chromosome should wait until the selection phase finishes.
4. On the CPU side, the programmer should place a CUDA synchronization call to wait till all threads are idle, and then output the result. If this is not done, the result will be wrong since CPU does not know what is going inside the GPU.

Also, from crossover phase to selection phase, synchronization is not necessary due to the introduction of chromosome copy and algorithm design.

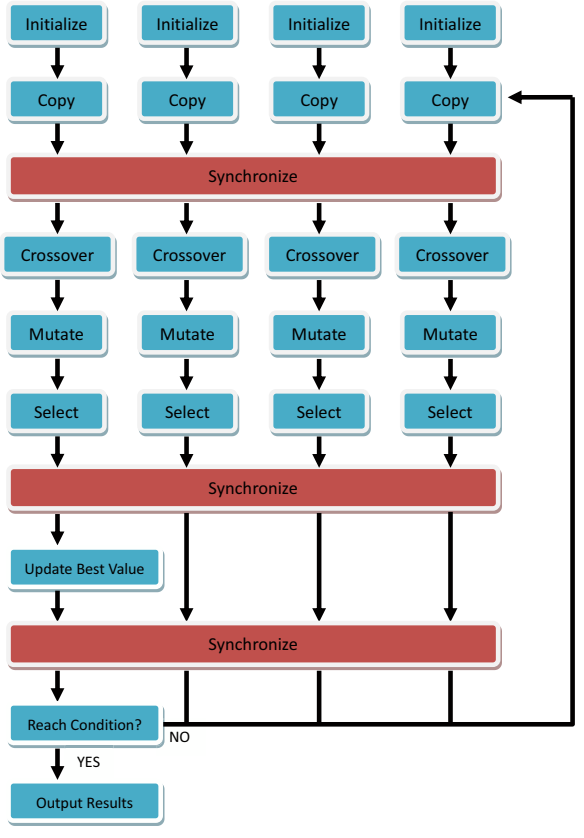


Fig. 6 Task dependency and synchronization points in CUDA programs

4 Experimental Results and Discussion

Both sequential and parallel programs were tested on a machine with two Intel Xeon E5504 Quad-Core CPUs (2.00GHz, 4MB cache) and two NVIDIA Tesla 20-Series C2050 GPUs.

Tests have been carefully made to determine how many chromosomes should be generated as a group and how large the termination generations should be. It turns out that we can get better solutions by setting 200 as the chromosome number and 1000 as the termination generation. Values that are larger than these two numbers do not provide further significant improvment to our solutions but increase the running time in a linear speed.

In general, the test data can be classified into two types: randomized and clustered, as shown in Figs. 7 and 8, respectively. Both of them occur in real life. However, it is easier for people to tell if clustered cities are well routed than random data through their intuitive observations. Even in programs, the work load of clustered data is smaller than randomized data. When a

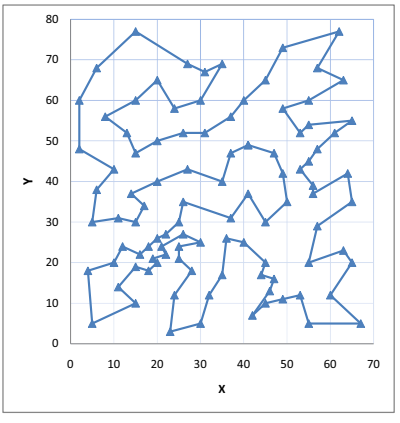


Fig. 7 An example of randomized test data for Genetic Algorithm

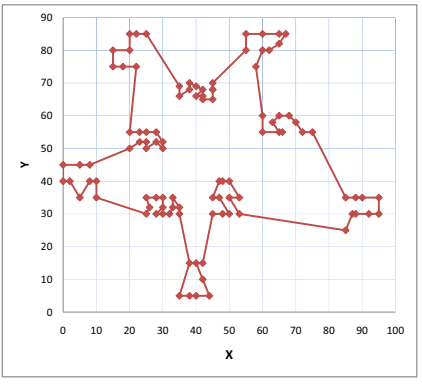


Fig. 8 An example of clustered test data for Genetic Algorithm

good solution is found, it is harder to find a better one for clustered data than for randomized data since only few tiny specific changes can update present the best solution for a cluster, while many more possible changes exist for randomized data. This inherent property associated with these two types of data makes their potential work load different in this paper. When dealing with clustered data, the algorithm will find it hard to update best value after it approaches some sort of line. Hence, the program ends early. On the other hand, the best value tends to update more times for randomized data, which causes a longer average running time. Comparison results are illustrated in Fig. 9. Both GPU and CPU give positive results for the above hypothesis, that is, algorithm on clustered data terminates earlier than that on randomized data. Another fact is that, for both types of data, GPU beats CPU.

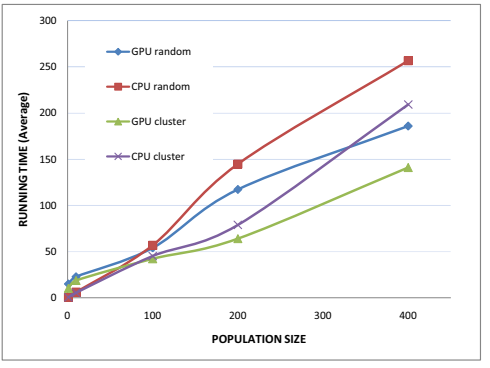


Fig. 9 Performance comparison with randomized and clustered data for GA

Questions may be raised about why GA in this paper only gets such insignificant speed up on GPU. As mentioned before, for the synchronization purpose, we only generate one block to run this program. Under CUDA architecture, threads in one block can only be served by one Streaming Multiprocessor (SM). However, in C2050, each GPU has 16 SMs and shared memory and cache are evenly assigned to each SM, which means we only used about 1/16 computing resource on one GPU, and the performance is still better than using CPU (single processor). In future work, we will try to expand the problem scale and keep the whole GPU or clusters busy, and the speed-up will increase significantly.

5 Related Work

Computer simulation of evolution started in 1950s with the work of Nils Aall Barricelli [3][4]. Since 1957, Alex Fraser has published a series of papers on simulation of artificial selection of organisms [7][8]. Based on this work, computer simulation of evolution became more popular in 1960s and 1970s. All essential elements of modern genetic algorithms were included in the book by Fraser and Burnell (1970) [9]. Goldberg (1989) first used genetic algorithm to solve the traveling salesman problem [10]. As a method for solving traveling salesman problems, 2-opt was raised by G. A. Croes (1958) in 1950s [5].

Muhlenbein (1989) brought up the concept of PGA (parallel genetic algorithm) [11], which aimed to implement GA on computer clusters. Ismail (2004) implemented PGA using MPI library [12]. In 2008, NVIDIA released latest CUDA SDK2.0 version, which bestowed CUDA much wider range of applications. Stefano et al. (2009) presented a paper about implementing

a simple GA with CUDA architecture, where sequential code for same algorithm was taken for comparison [6]. Another paper from Petr Pospicha and Jiri (2009) presented a new PGA and implemented it on CUDA [13]. However, performance comparison in Stefano's work was not based on the same algorithm. In 2010, NVIDIA developed latest version of its GPU architecture, which is called Fermi, for Tesla M2050 and M2070 [2] and corresponding programming guide under these architectures was released [1].

6 Conclusions and Future Work

Compared to Stefano's work in 2009 [6], this paper presents a more complex but parallelizable Genetic Algorithm (not specifically designed for certain GPU architecture) to solve TSP problem. Corresponding sequential C code for the same algorithm is carefully written for the performance comparison. Experimental results show the CUDA program with new Fermi architecture achieves some performance gains, although not so significant. However, considering the massive random memory accesses brought in by this much more complex algorithm and its relatively shorter execution time, this insignificant acceleration indicates that the current GPU architecture may have great potentials in speeding up the existing simulations of group evolution. More advanced performance tuning techniques such as asynchronous communication and zero copy will be applied for further performance gains in the future.

References

1. Nvidia cuda c programming guide 3.1 (2009)
2. Nvidia fermi tuning guide (2009)
3. Barricelli, N.A.: Esempi numerici di processi di evoluzione. Methodos, 45–68 (1954)
4. Barricelli, N.A.: Symbiogenetic evolution processes realized by artificial methods. Methodos 9, 143–182 (1957)
5. Croes, G.A.: A method for solving travling salesman problems. Operations Res. 6(1), 791–812 (1958)
6. Debattisti, S.: Implementation of a simple genetic algorithm within the cuda architecture. In: The Genetic and Evolutionary Computation Conference (2009)
7. Fraser, A.: Simulation of genetic systems by automatic digital computers. Australian Journal of Biological Science 10, 484–499 (1957)
8. Fraser, A., Burnell, D.: Computer models in genetics. Computers and Security 13, 69–78 (1970)
9. Fraser, A., Burnell, D.: Computer Models in Genetics. McGraw-Hill, New York (1970)

10. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Lkuwer Academic Publishers (1989)
11. Muhlenbein, H.: Parallel genetic algorithm, population dynamic and combinational optimization. In: Proc. 3rd, International Conference on Genetic Algorithms (1989)
12. Ismail, M.A.: Parallel genetic algorithms (PGAs): master slave paradigm approach using MPI. E-Tech (2004)
13. Pospichal, P., Jaros, J.: Gpu-based acceleration of the genetic algorithm. In: Genetic and Evolutionary Computation Conference (2009)

Design and Implementation of Sensor Framework for U-Healthcare Services

Haeng-Kon Kim

Abstract. Ubiquitous sensor network (USN) is one of the important key technologies for future ubiquitous life. USN nodes will be distributed at any place in the future such as street, in-building, campus, and so on. These USN nodes will play various roles like sensing, gathering, transmitting and receiving information about the surround. So, most of these are implemented as wireless communication system with simple hardware architecture. ZigBee protocol is one of the representative USN systems. So, many manufacturers are developing ZigBee hardware platform and their software protocol. To more efficiently implement and deploy USN, we need to know ZigBee protocols and their characteristics. In this paper, we design and Implement a sensor framework systems related to medical and surveillance that are significantly considered for enhancing human life. These are employed under USN environment to construct multiple health care services in which medical sensors are inter-connected to provide efficient management of them. For this configuration, Zigbee based wireless bio-sensors are established for portable measurement in which PSoC technique is utilized for compact implementation. As well, such Zigbee based embedded sensor equipment is devised for UPnP based sensor framework.

Keywords: USN, U-healthcare, Zigbee, UPnp, CBD.

1 Introduction

USN utilizes wire-line sensor networks and/or wireless sensor networks (WSNs). WSNs are wire networks consisting of interconnected and spatially distributed autonomous devices using sensors to cooperatively monitor

Haeng-Kon Kim
Department of Computer Engineering, Catholic University of Daegu, Korea
e-mail: hangkon@cu.ac.kr

R. Lee (Ed.): Computer and Information Science 2011, SCI 364, pp. 253–261.
springerlink.com

physical or environmental conditions (e.g., temperature, sound, vibration, pressure, motion or pollutants) at different locations. WSNs were generally implemented as isolated networks. Simple design of applications and services based on isolated sensor networks is made by capture and transmission of collected sensed data to designated application systems. Such isolated simple applications and services have been evolving over the years with network advancement, network and service integration, data processing schemes enhanced by business logics and data mining rules, context awareness schemes, development of hardware and software technologies, etc. These technical developments enable the ability to build an intelligent information infrastructure of sensor networks connected to the existing network infrastructure. This information infrastructure has been called ubiquitous sensor network (USN) opening wide possibilities for applications and services based on sensor networks to various customers such as human consumers, public organizations, enterprises and government. USN applications and services are created via the integration of sensor network applications and services into the network infrastructure. They are applied to everyday life in an invisible way as everything is virtually linked by pervasive networking between USN end-users (including machines and humans) and sensor networks, relayed through intermediate networking entities such as application servers, middleware entities, access network entities, and USN gateways. USN applications and services can be used in many civilian application areas such as industrial automation,

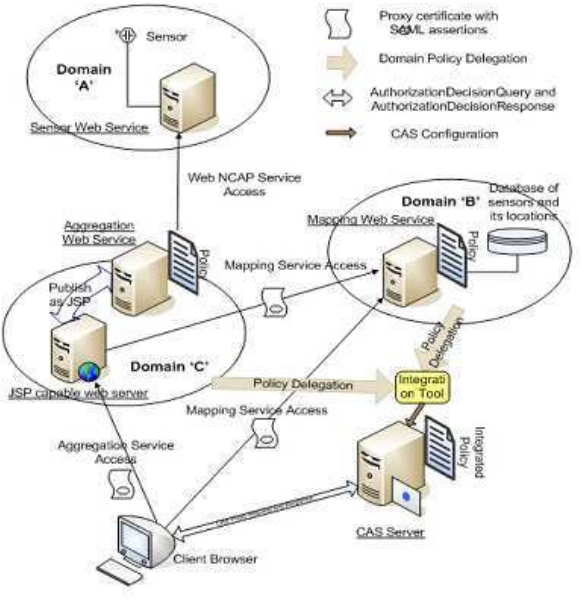


Fig. 1 USN Network

home automation, agricultural monitoring, healthcare, environment, pollution and disaster surveillance, homeland security or military field. Many industries invest cost and time for develop ubiquitous computing technology in IT fields. Ubiquitous computing is meant that there are multiple computers are embedded inside human and nature environment and they are inter-connected to be computed for alternative environment as in figure 1.

2 Related Works

2.1 Wireless Sensor Networks

Before looking at how wireless sensor networks can be used to assist firefighters in the performance of their duties, it is first necessary to know something about wireless sensor networks in terms of how they work; their capabilities and limitations. A Wireless Sensor Network (WSN) is a network comprised of numerous small independent sensor nodes or motes. They merge a broad range of information technology; hardware, software, networking, and programming methodologies. Wireless Sensor Networks can be applied to a range of applications [1] monitoring of space which includes environmental and habitat monitoring, indoor climate control, surveillance etc.; monitoring things for example structural monitoring, condition-based equipment maintenance etc.; and monitoring the interactions of things with each other and the surrounding space e.g., emergency response, disaster management, healthcare etc. The majority of these applications may be split into two classifications: data collection and event detection. Each mote in a wireless sensor network is a self-contained unit comprised of a power supply (generally batteries), a communication device (radio transceivers), a sensor or sensors, analog-to-digital converters (ADCs), a microprocessor, and data storage [2,3]. The motes self organize themselves, into wireless networks as in figure 2 and data from the motes is relayed to neighboring motes until it reaches the desired destination for processing. Each mote has very limited resources in terms of processing speed, storage capacity and communication bandwidth. In addition, their lifetime is determined by their ability to conserve power. These limitations are a significant factor and must be addressed when designing and implementing a wireless sensor network for a specific application.

2.2 UPnP

UPnP which is extensive from Plug-and-Play (PnP) based standard internet protocol popularly includes intelligent electronics, wireless machines and all personal computers to connect Peer-to-Peer in network points of view. Moreover, home network or SOHO and public regions are connected through the internet which provides flexible usage by employing TCP/IP network technology. This can be extensively established to provide PnP functions in

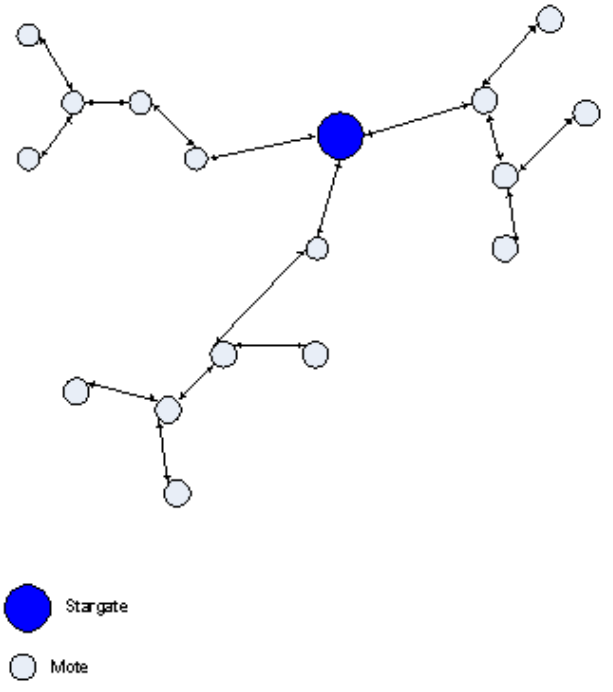


Fig. 2 Example of a Flat Network

printers, internet gateways, and home electronics. The devices transfer their ability to active networks through the UPnP services. That is, it uses Universal Control Point to control home applications after detecting and searching related devices. Sensor framework includes sensor searching, registration and deletion, monitoring control functions. This paper uses sensor framework implemented with equal framework and components to able to delete and add in Plug-in structures [3].

3 The Proposed Network Topology

3.1 Concepts

Generally, UPnP sensor based framework is a kind of software modules, which is unloaded with UPnP to present UPnP devices for interconnecting. This sensor framework can control the present UPnP devices with the protocol and unloaded devices to translate each protocol. Namely, it is likely to be an emulator of UPnP devices although non-UPnP in reality. We propose the system architectures to be design and implement as shown in figure 3.

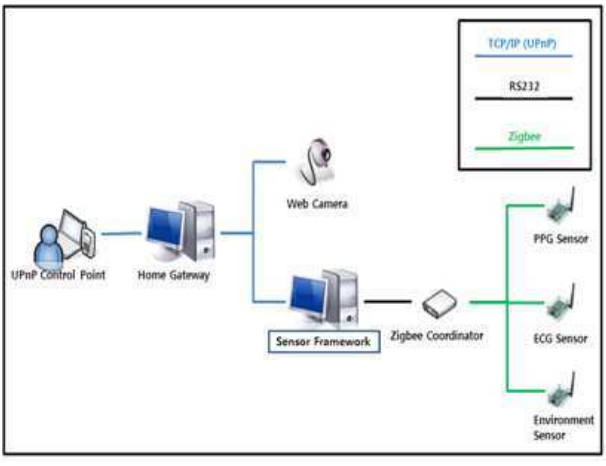


Fig. 3 Structure of our systems

The UPnP sensor framework is a device to connect bio-sensor and environmental sensor modules to Zigbee network which cannot UPnP stack. This is able to recognize several sensor modules through UPnP middle-ware, which includes bio and environment modules. Such framework is available to activate different UPnP devices and user control points based on DHCP servers. The UPnP must be constructed with the TCP/IP based UPnP stack to provide connectivity according to utility, flexibility, and standard through UPnP middle-ware. However, the sensor module is not connected with non-IP devices. Thus, we construct the UPnP sensor device modules inside the framework to recognize a virtual UPnP device. Figure 4 illustrates a software

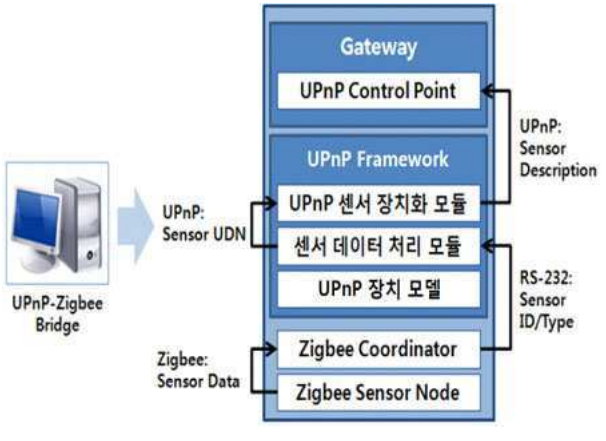


Fig. 4 Structure of Software Module

module of the UPnP framework proposed in this paper. The UPnP framework device module: The UPnP standard device model to connect the framework proposed in this paper to the UPnP device. Sensor data processing module: Data to be transferred to the framework is acquired and give its status continuously to the bio-data signal to user application. UPnP sensor equipment module: Based on sensor equipment information to be transferred from its data processing module, the loaded UPnP sensor is practically connected. The derived UPnP sensor device is linked with realistic sensor devices and UPnP control point from the virtual framework.

3.2 Design and Implementation of the UPnP Sensor Network

Realization of the UPnP based sensor network to equip Zigbee based senor systems is as follow in figure 5. The framework involves to connect the ports and activate modules of the whole software to acquire data from the sensor systems by the command BridgeStart(). The sensor devices continuously send 66 byte data sets including user ID, sensor type, and bio-data. The framework adds sensor devices listed in Device-ArrayList after acquiring data through the function GetsensorData(). The function SetsensorDevice() equips the sensor devices identified to connect based on Device-ArrayList. The function SetDataXML transforms the FLEX web application into the XML data type in order for the chart presentation. Through this procedure, the sensor equipment is connected with the UPnP control point for sensor device management and control of the user applications which can be identified from the transferred bio-data.

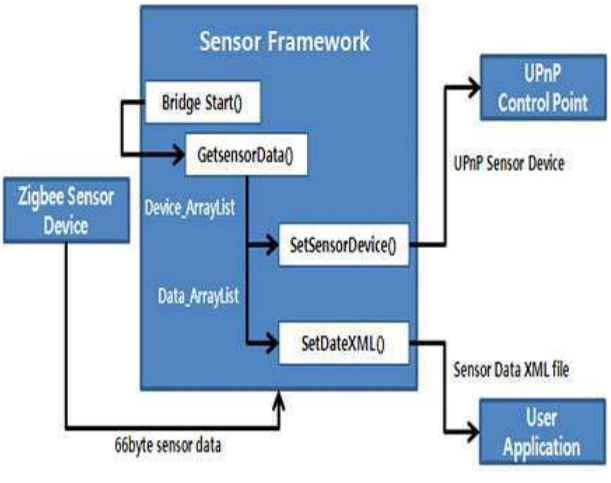


Fig. 5 Software module in our Frameworks

Fig. 6 testing environment for the utilized sensor modules

The UPnP device generally supports the Plug and Play connection to the hardware with its libraries which is possible to be activated under the window based PC or UPnP middle-ware. Figure 6 shows a testing environment for the utilized sensor modules and the web camera inside the home gateway network topology.

Figure 7 illustrates the UPnP control point program for identifying and control UPnP network connection. This program is installed inside the home gateway and equips the UPnP to show the web camera. As well, to control

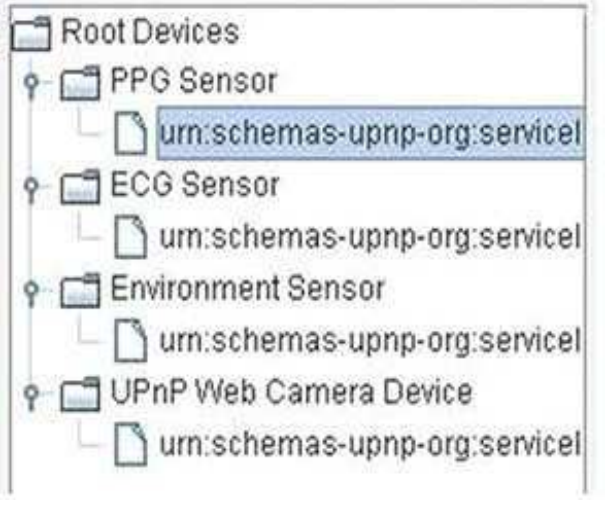


Fig. 7 UPnP Control Point Program

the sensor module, the command SetPower can control the power of the sensor systems via the control point action panel. The sensor command is constructed for power on/off action which is available from a Sleep mode of PSoC technique.

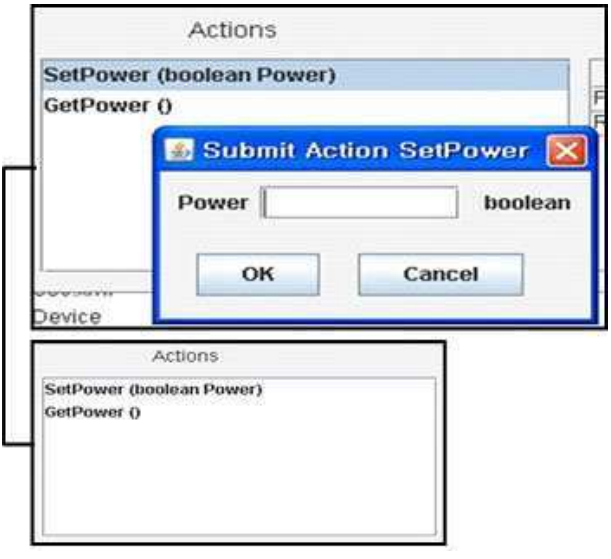


Fig. 8 Power control mode

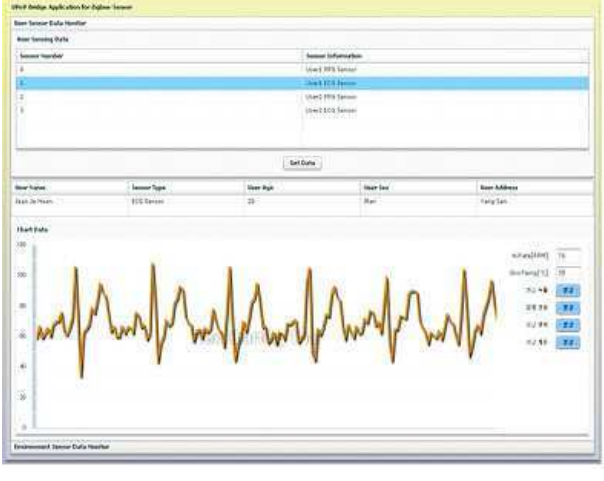


Fig. 9 UI Application

Fig. 8 shows an interfacing display to monitor data from the sensor modules implemented from the FLEX data service 2.0. An established user interface identifies what kind of sensor module is connected including selection and status of users and environment. Figure 9 show the UI applications for the frameworks.

4 Conclusions

This paper presents the logical UPnP single network construction which is no limit to connect different application systems provided possibly from standard connectivity and management under embedded USN environments. Main advantages of the proposed system include provision of the standardized connectivity under Zigbee based wireless communication network and effectiveness of device management and control through the UPnP control point program. These proposed topologies are able to extend and change multiple different application systems each other. In future work, we expand this investigation for more rapid and higher service provision in inter-connection of the single network configuration.

References

1. Schwiebert, L., Gupta, S., Weinmann, J.: Research Challenges in Wireless Networks of Biomedical Sensors. In: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking
2. Linnyer Beatrys Ruiz, J.M.S.N., Loureiro, A.A.F.: MANNA: A Management Architecture for Wireless Sensor Networks. IEEE Communications Magazine 41(2), 116–125 (2003)
3. Song, H., Kim, D., Lee, K., Sung, J.: UPnP-Based Sensor Network Management Architecture Real-time and Embedded Systems. Lab Information and Communications University (2007)
4. Eidsvik, A.K., Karlsen, R., Blair, G., Grace, P.: Interfacing remote transaction services using UPnPq. Journal of Computer and System Sciences 74, 158–169 (2008)

Author Index