Gesellschaft für Informatik (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of
• Seminars
• Proceedings
• Dissertations
• Thematics
current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English.

Information: http://www.gi-ev.de/service/publikationen/lni/

ISSN 1617-5468
ISBN 978-3-88579-259-8

In 2010, the annual International Conference on Innovative Internet Community Systems (I^2CS) 2010 takes place in Bangkok, its first Asian venue. The organizers continue combining contributions on foundations, technology, applications and socialization of virtual communities. Beside paper presentations a lot of space for discussion, brainstorming and other creative activities is given to the participants. The present LNI volume contains 21 carefully selected contributions of this conference. Since the 10^{th} I^2CS is also a mentionable event, this anniversary makes the volume a jubilee edition. Therefore, it contains a selection of excellent, still recent unpublished I^2CS papers from the previous years.

G. Eichler, P. Kropf, U. Lechner, P. Meesad, H. Unger (Eds.): I^2CS 2010

165

GI-Edition

Lecture Notes in Informatics

Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.)

10^{th} International Conference on Innovative Internet Community Systems (I^2CS) – Jubilee Edition 2010 –

June 3 - 5, 2010
Bangkok, Thailand

Proceedings

Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad,
Herwig Unger (Eds.)

10th International Conference on Innovative Internet Community Services (I^2CS)

Jubilee Edition 2010

June 3 - 5, 2010
Bangkok, Thailand

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-165

ISBN 978-3-88579-259-8
ISSN 1617-5468

Volume Editors
Dipl.-Ing. Gerald Eichler
Deutsche Telekom AG, Laboratories, Innovation Development
D-64295 Darmstadt, Germany; Email: gerald.eichler@telekom.de
Prof. Dr. Peter Kropf
Université de Neuchâtel, Institut d'informatique
CH-2009 Neuchâtel, Switzerland, Email: peter.kropf@unine.ch
Prof. Dr. Ulrike Lechner
Universität der Bundeswehr München
D-85577 Neubiberg, Germany; Email: ulrike.lechner@unibw.de
Asst. Prof. Dr. Phayung Meesad
King Mongkut's University of Technology North Bangkok
10800 Bangkok, Thailand, Email: pym@kmutnb.ac.th
Prof. Dr. Herwig Unger
Fernuniversität Hagen, Faculty for Mathematics and Informatics
D-58084 Hagen, Germany; Email: herwig.unger@fernuni-hagen.de

printed by Köllen Druck+Verlag GmbH, Bonn

Foreword to the Jubilee Edition

Over the years, the Innovative Internet Community Systems (I^2CS) moved from a national workshop, founded by Thomas Böhme, Ilmenau Technical University, Germany and Herwig Unger, University of Rostock, Germany, towards a small but remarkable international conference. Alternately, an international and a German location are selected by the steering committee for the annual three-day event in June. The name is program: scientists, researchers, service providers and vendors form an already great community for a longer term.

With the 9th I^2CS (URL http://i2cs.uni-jena.de/) at Friedrich Schiller University Jena in 2009, the conference has started a close corporation with the Gesellschaft für Informatik (GI) to publish the presented papers as Lecture Notes in Informatics (LNI) proceedings.

A strong peer review by three members of the program committee guarantees the high quality of presentations. These proceedings are structured in nine topics, covering the best papers from 2010, 2007 and 2006. Please refer to LNI P-148 for the last I^2CS proceedings.

The selection of I^2CS topics for 2010 again covers a wide range of aspects, bundled into the three areas: “foundations”, “technology”, “applications and socializations”.

Foundations – Theories, models, algorithms for communities

- Distributed algorithms and simulation models
- Game theory, graph theory and cost models
- Innovative communication protocols
- Self organization and self stabilization
- Security and privacy protection
- Interoperability and IT-governance

Technology – Distributed architectures and frameworks

- Service-oriented architectures for communities
- Peer-to-peer and grid architectures
- Distributed community middleware for Web x.0
- Software agents for community support
- Adaptive cooperative information systems
- Community management in ad-hoc environments
- Information retrieval and distributed ontologies

Applications and socialization – Communities on the move

- Experiences with Mobile Internet applications
- Context and location awareness
- Personalization of components and tools
- Personal social networks and user behavior

- Social and business aspects of user generated content
- Expert profiles, collaborative filtering and matching
- Domain specific languages for semantic design

Many thanks to all volunteers, especially to Sabina Serbu, Université de Neuchâtel and Werner Schubert, Fernuniversität Hagen for their support. Also to those former authors, who re-wrote their papers, according to the LNI style and gave their permission for publication of the contributions in this jubilee edition.

The 11th I²CS conference, carried out by the Innovation Development of Deutsche Telekom Laboratories, will take place in Berlin, Germany from June 15 till 17, 2011. Please check the permanent conference URL http://i2cs-conference.org/ regularly for more details.

With kind regards on behalf of the steering committee and the editors' board

Darmstadt, April 2010

Gerald Eichler, volume editor in chief
Deutsche Telekom AG, Laboratories, Berlin & Darmstadt, Germany

Preface to the 10th I²CS 2010, Bangkok, Thailand

It is a great pleasure for us to welcome all participants of the 10th International Conference on Innovative Internet Community Systems (I²CS 2010) in Bangkok. On behalf of the organizing and program committee, we hope that you will have some nice and unforgettable days in the city of the angels, which is the original name of Thailand's capitol, Bangkok.

The 10th I²CS 2010 is organized in conjunction with the 6th National Conference on Computing and Information Technology (NCCIT 2010). As all conferences, this joint conference wants to support scientists and practitioners to exchange knowledge, skills, and experiences from multiple disciplines in computing and information technology both in theory and applications. Since the multi-disciplinary cooperation is the key to master the challenges of globalization, the organizers of the NCCIT 2010 and the I²CS 2010 are proud that we could bring people from such different areas of science and technology together. This also provides a channel to disseminate researches conducted by faculty members, researchers, students, stakeholders, partners, and other practitioners.

Both, the NCCIT 2010 and the I²CS 2010 conference, received a record of submissions this year. On one hand, this is a confirmation for the well thought concepts of both conferences. On the other hand, it allows us to select only high quality papers for presentation and printing in the proceedings. We have to thank both: all of our approximately 40 contributors of scientific papers and all of our program committee members for their excellent review work. Following our tradition, we again selected 18 full and four short papers for the presentation at I²CS. Prof. Kyamakya (University of Klagenfurt, Austria) and Dipl.-Ing Gerald Eichler (Deutsche Telekom Laboratories, Germany) will give a great outlook to the expected innovation in their field of research in their invited talks.

The organizers of I²CS hereby still follow the initial concept to keep the conference small and make it possible that all participants get to know each other. We won't contribute to the tendency that conferences just serve the purpose to increase the authors' publication number, but want to offer an inspired and unique atmosphere, in which everybody will feel comfortable. It is the culture of I²CS to bring researchers together not only in order to share their latest results, but also to get to know and to learn about each other. Contributors will not only attend our scientific sessions, but will also spend as much time as possible together to discuss, share opinions, or find new ideas for new projects in brainstorming meetings. Last but not least, an excellent cultural program around our session will give the participants the possibility to get in contact with the Asian culture and life style. A Chaopraya river cruise with an excellent Thai style dinner is therefore as well included in the program as the visit of the Siam Nirumit Show at the Thai Cultural Center introducing music, culture and history of Thailand to the visitors.

We are happy that over the years a lot of colleagues join again and again our conferences, but in the same manner we appreciate and welcome all people who participate in I^{2}CS for the first time, and hope that this will not remain the only contact to our community. Beside a lot of contacts and meetings, the next year I^{2}CS conference in Berlin (Germany) maybe therefore just another fix point where we can all meet again.

Last but not least, the organizers would like to say thank you to Assoc. Prof. Dr. Monchai Tiantong, Dean of Faculty of Information Technology, KMUTNB, and the IT staff members whose work significantly contributed to the success of this year's event.

Bangkok, April 2010

Phayung Meesad, King Mongkut's University of Technology North Bangkok, Thailand
Herwig Unger, Fernuniversität Hagen/GI e.V., Germany

Retrospection to the 7th I²CS 2007, Munich, Germany

More and more different types of applications are using the Internet as a large distributed system. Mobile users and pervasive systems pose new technological and organizational challenges. Many research problems have emerged with these developments. Solutions to these problems require multi-disciplinary collaboration among researchers from different fields.

The 7th Innovative Internet Community Systems Workshop dealt with theoretical, methodological, and technological aspects of the Internet Systems as well as business, governmental, and other applications. I²CS 2007 brought together researchers from various areas related to novel Internet Community Systems. Participants had the opportunity to discuss the current state of the art and identify promising directions for research.

Two invited presentations by Bettina Hoser, Technical University of Karlsruhe, and by Armin Mickler, University of Texas at Austin, presented cutting edge research in the Social Network Analysis and Computational Epidemiology. 18 papers and talk proposals were submitted and a part of them were presented at the workshop. The Jubilee edition of the Innovative Internet Community Systems series includes 7 papers as a selection of the publications.

The workshop was hosted by the Universität der Bundeswehr München at the campus in Neubiberg, June 20 till 22, 2007. The workshop series was established in 2001 and previous I²CS workshops took place in Ilmenau, Kühlungsborn, Leipzig, Guadalajara, Paris and Neuchâtel.

Members of the organizing committee were U. Lechner, A. Dannecker, T. Böhme and H. Unger. The program committee 2007 consisted of 34 members from Germany, France, USA, Switzerland, Norway and Sweden: T. Böhme, A. Brandstädt Y. Breitbart, M. Brinkmeier, A. Bui, M. Bui, N. Deo, G. Dreo-Rodosek, K.-P. Fähnrich, P. Felber, C. Fetzer, M. Fiedler, H. Fouchal, L. Garcés-Erice, M. Garofalakis, C. Gkantsidis, V. Goebel, I. Gupta, T. Haupt , G. Heyer, M. Koch, P. Kropf, G-S. Kuo, P. Kuonen, N-C. Liebau, A. Mikler, N. Schabanel, M. Schoop, R. Steinmetz, D. Tavangarian, G. Teege, D. Tutsch, H. Unger, T. Ungerer.

We thank all organizing and program committee members for their work in reviewing and selecting papers.

Neubiberg, April 2010

Ulrike Lechner, Universität der Bundeswehr, Neubiberg, Germany
Achim Dannecker, Universität der Bundeswehr, Neubiberg, Germany

Retrospection to the 6th I²CS 2006, Neuchâtel, Switzerland

An increasing number of applications types have been using the Internet as a large distributed system in the past few years. Many challenging research problems have emerged with this recent development. Solutions to these problems require multi-disciplinary collaboration among researchers.

The International Workshop on Innovative Internet Community Systems (I²CS) 2006 was the sixth in a series of annual workshops encompassing three complementary aspects of Internet Community Systems: theoretical foundations and models, distributed architectures and applications, and information and knowledge management. This profile has again helped to bring together researchers from both academic and industry fields to discuss current progress and future developments in these areas, and was most remarkably supported by the talks of the two invited speakers, Vaidy Sunderam, Emory University, USA, and Jens Nicolaysen, JNC, Germany.

This volume of the Lecture Notes in Informatics series contains 19 unpublished papers accepted and presented during I²CS 2006, which was held at the University of Neuchâtel, Switzerland, from June 26 till 28, 2006. We were fortunate to receive almost 50 high-quality papers, of which 20 were selected in a peer review process for presentation. Each submission was evaluated by at least three reviewers, and comments were sent to the authors. The final papers found in this volume have been prepared after the workshop taking into account suggestions and critics made by the reviewers and issuing from discussions at the conference.

This workshop owes its success to all the members of the Program Committee who greatly helped to create an outstanding technical program. We would like to thank all of them, the additional reviewers who helped out with specific expertise, and the staff of the Computer Science Institute for the local organization. Special thanks go to the authors contributing to this 2010's LNI Jubilee Edition Proceedings of I²CS, and to Sabina Serbu for her professional help in compiling the papers for these proceedings.

Neuchâtel, April 2010

Peter Kropf, Université de Neuchâtel
Pascal Felber, Université de Neuchâtel

I²CS Program Committee 2010

Heinrich Arnold, Deutsche Telekom Laboratories, Germany
Gilbert Babin, HEC Montreal, Canada
Andreas Böhm, T-Systems, Germany
Thomas Böhme, TU Ilmenau, Germany
Gareth Clayton, KMUTNB, Thailand
Christian Erfurth, Friedrich Schiller University Jena, Germany
Gerald Eichler, Deutsche Telekom Laboratories, Germany
Hacène Fouchal, Univesité Reims Champagne-Ardenne, France
Wolfgang Halang, Fernuniversität Hagen / GI, Germany
Gerhard Heyer, University of Leipzig, Germany
Hagen Höpfner, International University Bruchsal / GI, Germany
Philippe Hunel, University of Antilles-Guyane, France
Thippaya Chintakovid, KMUTNB, Thailand
Janusz Kacprzyk, Polish Academy of Science, Poland
Vassilka Kirova, Alcatel-Lucent, USA
Peter Kropf, Université de Neuchâtel, Switzerland
Soradech Krutjohn, KMUTNB, Thailand
Kyandoghere Kyamakya, Alpe-Adrua University of Klagenfurt, Austria
Ulrike Lechner, Universität der Bundeswehr, München, Germany
Franz Lehner, University of Passau / GI, Germany
Armin Mikler, University of Northern Texas, USA
Anirach Mingkhwan, KMUTNB, Thailand
Supot Nitsuwat, KMUTNB, Thailand
Christian Prehofer, Nokia Research Center, Finland
Lior Rokach, Ben-Gurion University, Israel
Wilhelm Rossak, Friedrich Schiller University Jena, Germany
Harald Sack, HPI, University of Potsdam, Germany
Volkmar Schau, Friedrich Schiller University Jena, Germany
Holger Schilder, nexum AG / GI, Germany
A. Tajuddin Bin Samsudin, Fernuniversität Hagen, Germany
Kit Sang Tang, City University Hong Kong, Hong Kong
Herwig Unger, Fernuniversität Hagen, Germany
Martin Welsch, IBM, Germany
Leendert Wienhofen, SINTEF, Norway
Nawaporn Wisitpongphan, KMUTNB, Thailand

I²CS Steering Committee 2010

Thomas Böhme, TU Ilmenau, Germany
Gerald Eichler, Deutsche Telekom Laboratories, Germany
Phayung Meesad, KMUTNB, Thailand
Monchai Tientong, KMUTNB, Thailand
Herwig Unger, Fernuniversität Hagen, Germany

Table of Contents

Invited Talk Abstracts **16**

Kyandoghere Kyamakya **16**
Mobile Service Concepts and Car-drivers related Internet Community Systems for Supporting both a Real-time Road Safety Assessment and a Novel "Dynamic ridesharing"-based Urban Smart Mobility

Gerald Eichler **18**
Web 2.0 versus Enterprise 2.0 – how Communities Influence Today's Company Culture

Chapter 1: Dynamic Ad-hoc Networks **19**

Olivier Flauzac, Bachar Salim Haggar, Florent Nolot **20**
Self-stabilizing Tree and Cluster Management for Dynamic Networks

Mirko Caspar, Matthias Vodel, Wolfram Hardt **30**
System Level Test of Service-based Systems by Automated and Dynamic Load Partitioning and Distribution

Volkmar Schau, Kathrin Kirchner, Christian Erfurth, Gerald Eichler **41**
Ad-hoc Community Composition of Rescue Forces in Action Situations

Cholatip Yawut, Béatrice Paillassa, Riadh Dhaou **53**
Adaptation Process for Ad hoc Routing Protocol

Soufian Ben Amor, Marc Bui, Ivan Lavallée **63**
Optimizing Mobile Networks Connectivity and Routing Using Percolation Theory and Epidemic Algorithms

Harry Gros Désormeaux, Hacène Fouchal, Philippe Hunel **79**
Optimizing Distributed Test Generation

Adnan Noor Mian, Roberto Beraldi, Roberto Baldoni **91**
Identifying Open Problems in Random Walk based Service Discovery in Mobile Ad hoc Networks

Gerald Eichler, Christian Erfurth **103**
Mobile and Smart Devices in a Human Community – the Challenge of Context-aware Distributed Networking

Marcos F. Caetano, Mario Antonio Ribeiro Dantas **116**
An Approach of Semantic Cache for Mobile Devices to Enhance the Performance of Applications

Chapter 2: Decentralized Network Systems **129**

Christian Spielvogel, Peter Kropf **130**
Application Layer Scalable Video Coding for the iPhone

Lada-On Lertsuwanakul **140**
Fuzzy Logic Based Routing in Grid Overlay Network

Miguel Angel Rojas González **150**
Performance Evaluation of two Self-Adaptive Routing Algorithms in Mesh Networks

Oleksandr Kuzomin, Illya Klymov **158**
Functional Approach to Decentralized Search Engine for P2P-Network Communities

Chapter 3: User Behaviour and Profiling **163**

Andrea Jersabek **164**
Why Western Approaches fail in Asia: a Classroom Action Research on Developing Creative Processes and Knowledge Sharing Abilities

Amir Gershman, Amnon Meisels, Karl-Heinz Lüke, Lior Rokach, Alon Schclar, Arnon Sturm **170**
A Decision Tree Based Recommender System

Chanattha Thongsuk, Choochart Haruechaiyasak, Phayung Meesad **180**
Classifying Business Types from Twitter Posts Using Active Learning

Coskun Akinalp, Herwig Unger **190**
The Limbic Characteristic and El-Farol Games

Gerald Eichler, Christian Erfurth, Volkmar Schau **196**
Enhancing Communities by Social Interactions in Mobile Environments

Gregor Heinrich **207**
Actors–media–qualities: a Generic Model for Information Retrieval in Virtual Communities

Birgit Wenke **223**
Use of Algorithms for a User Specific Reduction of Amounts of Interesting Association Rules

Chapter 4: Content Management **237**

Wongkot Sriurai, Phayung Meesad, Choochart Haruechaiyasak **238**
Improving Web Page Classification by Integrating Neighboring Pages via a Topic Model

Maleerat Sodanil, Supot Nitsuwat, Choochart Haruechaiyasak **247**
Improving ASR for Continuous Thai Words Using ANN/HMM

Nivet Chirawichitchai, Parinya Sa-nguansat, Phayung Meesad..... 257
A Comparative Study on Feature Weight in Thai Document Categorization Framework

Florian Holz, Hans-Friedrich Witschel, Gregor Heinrich, Gerhard Heyer, Sven Teresniak 267
An Evaluation Framework for Semantic Search in P2P Networks

Chapter 5: Community Structure Building..... 277

Sunantha Sodsee, Maytiyanin Komkhao, Zhong Li, Wolfgang A. Halang, Phayung Meesad 278
On the Convergence of a Leader-Following Discrete-Time Consensus Protocol

Panchalee Sukjit, Herwig Unger..... 386
HexaGrowth: a new Grid Generation with a Local Algorithm

Daniel Berg, Herwig Unger 296
n-Dimensional Border Growth

Vincent Levorato, Marc Bui 306
Modeling the Complex Dynamics of Distributed Communities of the Web with Pretopology

Alessandro E. P. Villa, Javier Iglesias, Solange Ghernaouti-Helie 321
OpenAdap.net: a Community-Based Sharing System

Chapter 6: Security and Theoretic Approaches..... 329

Sheikh Ziauddin..... 330
An Improved Hwang-Lee-Tang Remote User Authentication Scheme

Dejvuth Suwimonteerabuth 340
Computing Minimum-Height Certificate Trees in SPKI/SDSI

Sirapat Boonkrong..... 350
Some Remarks on Andrew Secure RPC

Duc Kien Nguyen, Ivan Lavallee, Marc Bui..... 359
Generalizing of a High Performance Parallel Strassen Implementation on Distributed Memory MIMD Architectures

Roberto Gómez, Gabriel Ramírez..... 371
Using Digital Images to spread Executable Code on Internet

Marco Aurelio Turrubiartes Reynaga, Orlando Ezequiel Rincón Ferrera, Leopoldo Estrada Vargas, Deni Torres Román, David Muñoz Rodríguez, Marlenne Angulo Bernal, Luis Rizo Domínguez 384
Characterization and Generation of Synthetic Data Traces for IP Traffic Modeling

Chapter 7: Reliability and Availability **496**

Gert Pfeifer, Christof Fetzer, Martin Steuer **497**
Rearchitecting DNS

Fares Saad Khorchef, Ismail Berrada, Antoine Rollet, Richard Castanet **409**
Automated Robustness Testing for Reactive Systems: Application to Communicating Protocols

Harry Gros-Desormeaux, Hacène Fouchal, Philippe Hunel **422**
A Distributed Cache Management for Test Derivation

Chapter 8: Overlays and Ubiquitous Computing **436**

Raphael Chand, Luigi Liquori, Michel Cosnard **437**
Resource Discovery in the Arigatoni Model

Christophe Guéret, Nicolas Monmarché, Mohamed Slimane **450**
Self-Organizing Ant-based Information Gossiping Algorithm for P2P Networks

Hyosook Jung, Jinhyun Ahn, Seongbin Park **462**
A JXTA-based System for Adaptive and Collaborative Learning

Sergio Maffiolett, Simon Schubiger, Michèle Courant, Béat Hirsbrunner **472**
A Homogeneous Service Framework for Pervasive Computing Environments

Chapter 9: Information and Knowledge Management **492**

Fabien Mathieu, Laurent Viennot **493**
Local Aspects of the Global Ranking of Web Pages

Benoît Garbinato, Ian Rickebusch **507**
Impossibility Results on Fair Exchange

Achim Dannecker, Ulrike Lechner **519**
On the Demand for E-Services by Health Communities

Gilson Yukio Sato, Jean-Paul Barthès **532**
A Multi Agent System Application to Support Communities of Practice: Preliminary Analysis

Jacques Savoy **545**
Stemming Strategies for European Languages

Invited Talk

Mobile Service Concepts and Car-drivers related Internet Community Systems for Supporting both a Real-time Road Safety Assessment and a Novel "Dynamic ridesharing"-based Urban Smart Mobility

Speaker: Prof. Dr.-Ing. Kyandoghere Kyamakya

Affiliation: Alpen-Adria-University of Klagenfurt, Austria

E-mail: kyandoghere.kyamakya@uni-klu.ac.at

Abstract:

Road safety is a still unsolved issue in our modern society. Today, road accidents kill more people than wars. Besides, several major cities are merely asphyxiated by traffic congestion. The socio-economical and environmental costs of these two problems are terrifying. Our society needs safe roads and an efficient mobility in both urban and inter-urban road networks.

This talk presents a series of ideas demonstrating how far novel forms of social networking (Pervasive Social Computing) coupled with both appropriate intelligent systems and the mobile Internet do meet Intelligent Transportation. The pervasive social computing should involve vehicles, smart infrastructures and travellers. Thereby, a very cost-effective and adaptive data and information collection infrastructure does result from this synergetic convergence. A series of interesting mobile service concepts are enabled. A form of real-time road safety support involving amongst others context-aware recommender and assistance systems comes into realistic reach.

Furthermore, novel forms of mobility support should also profit from such a smart infrastructure. Just for illustration, the example of a novel urban mobility concept called "Mobility Ebay" or in short "Mo-Bay" is presented. Mo-Bay is based on the use of intelligent communication and information systems technologies to coordinate mobility needs, mobility demand, goals and actions in real-time through the efficient combination of multiple urban modes of transportation (individual cars, public transportation (bus and trams) and taxis). The concept is particularly human-centred in the sense that any developments are attuned to the needs of the users living in the urban areas and their surroundings.

Curriculum Vitae:

Prof. Dr.-Ing. Kyandoghere Kyamakya obtained his Master of Science degree (in French: "Ingenieur Civil") in Electrical Engineering at the University of Kinshasa (DR. Congo) in 1990. He then became teaching assistant at the same University for three

years. After that, he became a scholar of the DAAD (German Service for Academic Exchanges) for doctoral studies at the FernUniversität-Hagen in Germany, where he obtained his doctorate degree in Electrical Engineering in October 1999. He then spent three years of postdoc-research at the Hannover University in the field of "Mobility Management in Wireless Networks and Location Based Services". From October 2002 to October 2005, he occupied a junior-professorship position in the area of "Positioning and Location Based Services in Wireless Networks" at the same university. Finally, since October 2005, he is full professor for "Transportation Informatics" and head of the Department for Smart Systems Technologies at the Alpen-Adria-University of Klagenfurt in Austria.

The research group Transportation Informatics (TIG) headed by Prof. Kyamakya (for more details, see URL: http://www.uni-klu.ac.at/tewi/ict/sst/tig/index.html) is one of three research groups of the Institute for Smart Systems Technologies of the University of Klagenfurt. Concerning research, teaching as well as consultancy activities, TIG is mainly involved in modelling, simulation, data analysis, optimization and adaptive control for a set of transportation related complex systems, which are: intelligent traffic systems, intelligent urban mobility systems, supply chain networks, and robust machine vision systems for advanced driver assistance systems.

In the research addressing these systems, a series of both theoretical and practical instruments are either extensively exploited or are source of inspiration for innovative solutions and concepts, namely: nonlinear dynamics and synchronization, coupled oscillators as a modelling paradigm, context-awareness and reasoning under uncertainty, cellular neural networks, nonlinear image processing, digitally emulated analogue computing for real-time computational engineering, systems science and computational intelligence.

Invited Talk

Web 2.0 versus Enterprise 2.0 – how Communities Influence Today's Company Culture

Speaker: Dipl.-Ing. Gerald Eichler

Affiliation: Deutsche Telekom Laboratories, Innovation Development, Berlin & Darmstadt, Germany

E-mail: gerald.eichler@telekom.de

Abstract:

Web-based communities like the Fotocommunity or MySpace are ubiquitous in many private areas. Heavy PC applications are out, mobile apps are in. With Wikipedia, public knowledge is preserved, and moods are reflected situation-based by Twitter immediately. Content creation complements consumption. But social networks, as formed by facebook, Studi.VZ or Xing, are not limited to private contacts anymore – they become a huge source for data mining and replace news gathering e.g. by Twitter.

In 2007, Don Tapscott introduced the term "Wikinomics" to describe how community technologies conquer the business markets. Against the tradition, such technologies coming from the field of collaboration – also known as peer production – infiltrate companies from the base, introduced by the collaborators and not by the management. There are absolutely different methods to handle this, which will be compared. Are the Wiki workplace, the global factory and the co-operative mind the real keys for the business culture of tomorrow?

Curriculum Vitae:

Since 1993 Gerald Eichler has been working as an engineer for the research and development divisions of the Deutsche Telekom AG. At the Technologiezentrum Darmstadt, he started with service development in satellite communications, moved to the protocol and signalling section, sharing and leading projects with special focus of ATM and IP networks and service quality issues, while contributing to standardization and several European research and innovation initiatives.

In 2000, moving to T-Systems Enterprise Services GmbH, his focus changed towards the areas of knowledge management and e-learning. His current fields within Deutsche Telekom Laboratories meet innovation development for convergent web technology-based and context-aware community solutions. In parallel, he gives lectures in communication and network technologies.

Chapter 1: Dynamic Ad-hoc Networks

Contributions to 10th I2CS 2010, Bangkok, Thailand

Olivier Flauzac, Bachar Salim Haggar, Florent Nolot
Self-stabilizing Tree and Cluster Management for Dynamic Networks

Mirko Caspar, Matthias Vodel, Wolfram Hardt
System Level Test of Service-based Systems by Automated and Dynamic Load Partitioning and Distribution

Volkmar Schau, Kathrin Kirchner, Christian Erfurth, Gerald Eichler
Ad-hoc Community Composition of Rescue Forces in Action Situations

Cholatip Yawut, Béatrice Paillassa, Riadh Dhaou
Adaptation Process for Ad hoc Routing Protocol

Contributions to 7th I2CS 2007, Munich, Germany

Soufian Ben Amor, Marc Bui, Ivan Lavallée
Optimizing Mobile Networks Connectivity and Routing Using Percolation Theory and Epidemic Algorithms

Harry Gros Désormeaux, Hacène Fouchal, Philippe Hunel
Optimizing Distributed Test Generation

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Adnan Noor Mian, Roberto Beraldi, Roberto Baldoni
Identifying Open Problems in Random Walk based Service Discovery in Mobile Ad hoc Networks

Gerald Eichler, Christian Erfurth
Mobile and Smart Devices in a Human Community – the Challenge of Context-aware Distributed Networking

Marcos F. Caetano, Mario Antonio Ribeiro Dantas
An Approach of Semantic Cache for Mobile Devices to Enhance the Performance of Applications

Self-Stabilizing Tree and Cluster Management for Dynamic Networks

Olivier FLAUZAC, Bachar Salim HAGGAR and Florent NOLOT

CReSTIC/SYSCOM
University of Reims Champagne-Ardenne
UFR Sciences Exactes et Naturelles
Department of Mathematics, Mecanics and Computer Sciences
E-mail: {olivier.flauzac, bachar-salim.haggar, florent.nolot}@univ-reims.fr

Abstract: The lack of infrastructure and dynamic nature of mobile ad hoc networks demand new networking strategies to be implemented in order to provide efficient end-to-end communication. Some researches proposed to organize the network into groups called clusters and use different routing protocols for inter and intra cluster to propagate an information. But with these solutions, the network needs first to be organized into clusters and next, we need to construct each routing table. Other researchers proposed to build a spanning tree on the network to forward informations on a tree but many solutions need to know the global network topology. In this paper, we propose a self-stabilizing algorithm both to construct cluster and simultaneously build a spanning tree on the network. Without any global knowledge, we use only one type of periodically exchanged messages of size $Log(5n + 3)$ bits, and we construct clusters and the spanning tree on the network with a convergence time of at most $D+6$ rounds.

1 Introduction

Today, wireless networks are increasingly popular because of ease of deployment. These networks provide information access to users regardless of their location. However, mobile networks are divided into two main categories: cellular networks and ad hoc networks [BKP02]. While cellular networks are characterized by centralized devices, ad hoc networks are characterized by the absence of infrastructure. Thus, an ad hoc network is a collection of mobile entities inter-connected by a technology without wire, forming a temporary network without the assistance of any management and any fixed architecture. The concept of ad hoc mobile networks tries to extend the notions of mobility to all the components of the environment, contrary to the networks based on the cellular communication. Due to mobility of nodes, the network topology may change quickly and unpredictably over time. The network is decentralized, meaning network organization and message delivery must be executed by the nodes themselves, i.e., routing functionality will be incorporated into mobile nodes.
Mobile ad hoc network can be widely and quickly deployed, without any support from an existing infrastructure or any other kind of fixed stations. The main characteristics of

ad hoc systems: they are self-organizing, fully decentralized and highly dynamic. These characteristics prohibit usage of many applications of algorithms which work in a wired network. On the other hand they provide opportunities for a range of new and interesting applications: conferences, meetings, wireless communication between vehicles in road traffic, disaster relief, rescue missions, military applications, etc. Such scenarios typically lack a central administration or wired infrastructure and, hence, ad hoc systems are very useful for them. Under the limited resources such as network bandwidth, memory capacity, and battery power, the efficiency of routing schemes in ad hoc wireless networks becomes more important and challenging.
In this paper, we proposed a new self-stabilizing algorithm to create clusters on ad hoc network which simultaneously constructs a spanning tree of the network. In each cluster, a node can be clusterhead, gateway or ordinary node. A clusterhead manages data forwarding in its cluster. A gateway is charged to relay messages between clusters. An ordinary node has no particular function, it is neither a clusterhead nor a gateway. With this solution, we have a new solution, with few messages, to forward information over the network.

2 Related Works

We present in this part some existing works on clustering and spanning tree problem. Many solutions for clustering ad hoc networks are intended to identify a subset of nodes geographically closed in a network.
In the Lowest-ID Cluster Algorithm [EWB88], each node in the network must hold an unique identity. The node with the lowest identity over all its neighbors is elected cluster head and the cluster is formed by the cluster head and all its neighbors. In High-Connectivity Clustering [GPL99] and [YC03], cluster head election is based on degree of each node instead of node identity. A node is elected as a cluster head if it has the highest connected node.
The three previous cited algorithms are not self-stabilizing solutions. So they need another algorithm to maintain clusters. Least Clusterhead Change Algorithm (LCC) [Chi97] is designed to minimize cluster head changing. Cluster heads only change when they come neighbors, or when a node becomes disconnected from all cluster heads. This is an improvement (in stability) over existing algorithms which select the clusterhead every time the cluster membership changes.
A different approach of clustering is taken by Basagni in [Bas99]. He presents two clustering algorithms, Distributed Clustering Algorithm (DCA), for "quasi-static" network and Distributed and Mobility-Adaptive Clustering algorithm (DMAC) for mobile network. Each node reacts locally to any topological change in its neighborhood. Both DCA and DMAC assign to nodes different weights and assume that each node is aware of its respective weight. A node is chosen to be a clusterhead if its node-weight is higher than any of its neighbors node-weight. In the DMAC protocol, if two clusters leaders become neighbors, the one with the smaller weight must revoke its leader $Status$. In [JN06a], [JN06b] and [JN09] the authors propose a self-stabilizing version of DCA and DMAC. Moreover,

their solution is robust.
In [CR09] all the previous cited algorithm, Mobility Metric Based Algorithm (MOBIC) [BKL01], Weighted Clustering Algorithm (WCA) [CDT02], [CDT00a], [CDT00b], and Weight Based Clustering Algorithm (WBCA) [YZ07] are studied and compared.
The cluster construction algorithms are not a solution to propagate any information over the network. We need either routing algorithm or spanning tree. We concentrate now our study on existing spanning solution on cluster network. Some authors propose LMST algorithm (Local Minimum Spanning Tree) as in [LHS03]. Each node builds a graph of its neighborhood and broadcasts periodically a $hello$ message which contains its identity and its position. Each node needs to use a system to gather its position, applying Prim's algorithm [Pri57] independently to obtain its local minimum spanning tree. In [CSS04] the authors propose Directed LMST Broadcast Oriented Protocol, an algorithm based on LMST and using directional antennas. The nodes require the knowledge of neighbors position. In [MJ06], the author presents a self-stabilizing distributed algorithms to build a spanning tree. Although this algorithm is self-stabilizing, the number of exchanged messages during operations is important. In [EOD08], distributed algorithms to construct a spanning tree over a network with cluster. In first time, the authors use HEED (Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach) to build the clusters of the networks. In HEED the cluster formation is based on the residual energy of a node and its degree. After clustering, the authors modify the distributed spanning tree formation algorithm for general networks. After formation of spanning trees, each node will have a unique subroot cluster head node. This algorithm uses different kind of messages and are not self-stabilizing.
From this study of existing algorithms, to the best of our knowledge, we can notice there exists no self-stabilizing solution which organize a network in clusters and simultaneously, without a full knowledge of the topology and without a positioning system, builds a spanning tree on the network and on the clusters, with only one type of message.

3 Contributions

From all existing algorithms which build clusters of diameter two, the built clusters can be overlapping, i.e., a node can be in two clusters simultaneously. The deterministic algorithm $MaxCwST$ proposed in this paper builds both clusters of diameter at most equal to two and simultaneously, a spanning tree on the network and on the built clusters. Moreover, it does need neither initialization phase, nor network discovery nor cluster maintain phase. To obtain this result, each node periodically exchanges only one type of message of size $Log(5n + 3)$ bits, when n denotes the number of nodes in the network. The convergence time of our algorithm is at most equal to $D + 6$ rounds, with D the diameter of the graph.

4 Preliminaries

We consider the network as an undirected graph $G = (V, E)$ in which V is the set of nodes and E the set of edges. The size of the network is denoted by $|V| = n$ and we say there exists a link between two nodes u and v if there is an edge $\{u, v\} \in E$. In this case we say that u and v are neighbors and the set of neighbors of a node $u \in V$ will be denoted in this paper by $Neigh_u$. The link to Node v is denoted by $link_v$. We also assume that every node u in the network has an unique identifier which will be u. We define $d(u, v)$ the distance between two nodes u and v in G as the number of edges along a minimal path between the two nodes in G and D is the diameter of the graph.
Clustering means partitioning network nodes into groups called clusters. A *cluster* (illustrated in Figure 1) is a subgraph of G and we assume that the diameter of a cluster must be lower or equal to two and each node belongs to only one cluster and the intersection between any cluster is empty. A node uses Variable $Cl\text{-}id$ to store the identity of its cluster and we denote a cluster by Variable Cl.
Each node exchanges only one type of messages : $hello$ message. This message contains some variables and we use m to denote a message. $m.x$ denotes the variable x contained in Message m. For other variable x, used by Node u, to avoid conflict reading we use notation x_u.
The algorithms presented in this paper are self-stabilizing. The *self-stabilizing* concept was introduced for the first time by E. Dijkstra in [Dij74] as a system, regardless its initial state, which is guaranteed to converge to a legitimate state in a finite number of steps. For the clustering problem, to define the legitimate state, we use the following definition.

Definition 4.1 (Cluster well formed) *A cluster is said well formed when it verifies the four properties :*

1. *it contains only one cluster head*
2. *the cluster head is the node with the largest identity among all nodes in the cluster*
3. *the diameter of the cluster is at most equal to two*
4. *for every pair x, y of clusterhead, x is not a neighbor of y*

From this definition, we can define the legal state as a network in which all clusters are well formed and all nodes are in one cluster.

5 $MaxC$ Self-Stabilizing Clustering Algorithm

The choice of the cluster heads is based on the identity of each node. The cluster head is the node which has the highest identity among all its neighbors, in its cluster. But without lost of generality, we could also choose the node which has the lowest identity. Moreover, from our algorithm, each node eventually satisfies the three following properties : (i) every node

in the network must belong to only one cluster,(ii) all nodes which are not cluster head, are at a distance at most one of a cluster head, and (iii) each cluster has only one cluster head.

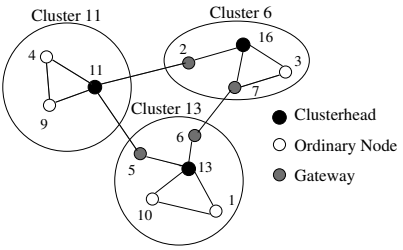


Figure 1: Cluster example

Each node executes the first enabled rule of $MaxC$ algorithm (Algorithm 1). It uses only two variables: $cl\text{-}id$ and $Status$. $cl\text{-}id$ stores the identify of the cluster in which it belongs. $Status$ stores the type of the node. Each node can be in one of the following types : cluster head (CH), gateway (GN) and ordinary node (ON). These three types can be described like this: a cluster head is a node which has the highest identity in its cluster like Nodes 11, 13, and 16 in Figure 1. Gateway node is a node which is adjacent to at least one node belonging to another cluster than him like Node 5 and 6 in Cluster 13 in Figure 1. Node 11 is in cluster 11 and it has two neighbors nodes, Nodes 2 and 5 which are, respectively, in Cluster 6 and 13. Finally, a node which has only neighbor in the same cluster is an ordinary node, like Nodes 4 and 3 for instance, in Figure 1.
We now present the modifications made on $MaxC$ algorithm in order to obtain $MaxCwST$ algorithm, the first algorithm which builds both clusters and simultaneously the spanning tree.

6 Spanning Tree Construction

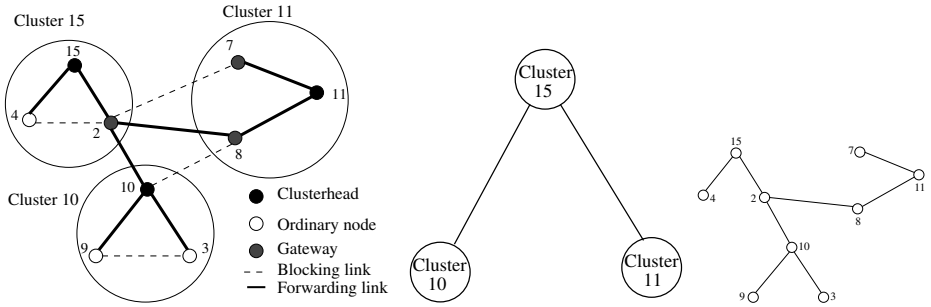
At the same time to clusters creation, we build a spanning tree of the graph and on the cluster, with very simple modification of $MaxC$ algorithm. We have just add two new informations in the $hello$ message: the identity of neighbor cluster and the identity of the gateway node which must be used to join the neighbor cluster.

6.1 MaxCwST algorithm principle

Our spanning tree algorithm is called $MaxCwST$. It is a modification of $MaxC$ algorithm in order to both construct clusters and simultaneously a spanning tree. But to avoid to write all $MaxC$ algorithm rules, we have just, in this paper, write the rules which constructs the spanning tree. It works according to following principle: each node of the graph

Algorithm 1 $MaxC$ Clustering Algorithm on a Node u

$cl\text{-}id$: Identity of the cluster of Node u.
$m.X$: The variable X in the message m
$Status \in \{CH, ON, GN\}$
On receiving $Hello(j, Status, cl\text{-}id)$
R1.a)
if $(Status = CH) \wedge (cl\text{-}id \neq id)$ **then**
  $cl\text{-}id \leftarrow id$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R1.b)
if $(Status \neq CH) \wedge (\ cl\text{-}id = id \vee (\forall\ m \in Hello, cl\text{-}id \neq m.j) \vee (\exists\ m \in Hello, cl\text{-}id = m.j \wedge m.status \neq CH))$ **then**
  $Status \leftarrow CH$;
  $cl\text{-}id \leftarrow id$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R2:
R2.a)
if $(Status \neq CH) \wedge (\forall\ m \in Hello, m.j < id)$ **then**
  $Status \leftarrow CH$;
  $cl\text{-}id \leftarrow id$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R2.b)
if $(Status \neq CH) \wedge (\exists\ m \in Hello, m.Status = CH \wedge m.cl\text{-}id > cl\text{-}id)$ **then**
  $Status \leftarrow GN$;
  $cl\text{-}id \leftarrow m.cl\text{-}id$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R2.c)
if $(Status \neq CH) \wedge (\exists\ m \in Hello, m.Status = CH \wedge m.cl\text{-}id < cl\text{-}id)$ **then**
  $Status \leftarrow GN$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R2.d)
if $(Status \neq CH) \wedge (\exists\ m \in Hello, (m.Status = GN \vee m.Status = ON) \wedge m.cl\text{-}id \neq cl\text{-}id)$ **then**
  $Status \leftarrow GN$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R2.e)
if $(Status \neq CH) \wedge (\exists\ m \in Hello, (m.Status = CH \wedge m.cl\text{-}id = cl\text{-}id)$ **then**
  $Status \leftarrow ON$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R3)
if $(Status = CH) \wedge (\exists\ m \in Hello, m.Status = CH \wedge m.j > id)$ **then**
  $Status \leftarrow ON$;
  $cl\text{-}id \leftarrow m.cl\text{-}id$;
  $Send\ Hello(id, Status, cl\text{-}id)$;
end if
R4)
$Send\ Hello(id, Status, cl\text{-}id)$;



(a) Cluster and Spanning tree obtained from $MaxCwST$ (b) Spanning tree view on the clusters (c) Spanning tree view on the network

Figure 2: Example of constructing spanning tree of network by $MaxCwST$

chooses only one neighbor node as father node in the spanning tree. Within each cluster, the cluster head will be the father of each nodes of its cluster. for instance, in Figure 2(a), Node 4 and 2 of Cluster 15 have chosen Node 15, their cluster head, as their father in this cluster. Nodes 7 and 8 have chosen Node 11 and Nodes 9 and 3 have chosen Node 10. All links between gateway or ordinary nodes in the same cluster will be in a blocking state. In this state, link will forward only $hello$ messages. Now, between each cluster, we need to make the spanning tree. So, only one gateway between two clusters need to "activate" its link. So, the cluster head will choose which gateway can be use to join a neighbor cluster. For instance, in Figure 2(a), Node 7 and 8 can be used to join Cluster 15. So, to avoid to make a loop, and to construct a spanning tree on the network and on the cluster, we have to activate only one link between Cluster 11 and 15. In the $hello$ message, we add the identity of the neighbor cluster. So, Node 8 will receive an $hello$ message from Node 2 which contains the cluster identity 15. This information will be forward to the cluster head, in the $hello$ message send by Node 8. The same exchange is made between Node 2 and 7 and next, between Node 7 and 11. When the cluster head 11 has received the two $hello$ messages from Node 7 and 8. It can choose only one gateway. This choice is based on the identity of this gateway. The gateway node with the highest identity will be chosen. In our example, cluster head of cluster 11 will choose gateway 8. We need also avoid another case. When a gateway has a link to more than one cluster, like Node 8 in Figure 2(a). In this case, the gateway always chooses the hello message from the highest identity cluster. So, Node 8 chooses Cluster 15 as father. From our algorithm, we obtain a spanning tree on all clusters (Figure 2(b)) and on the network (Figure 2(c)).

6.2 The proof

To prove the $MaxCwST$ algorithm, we use the following property:

Property 6.1 (Characteristic of a tree) *Let S be a subgraph of a graph G and n the size of G. S has exactly n-1 edges iff S is a tree of n nodes.*

Property 6.2 *Let i be a cluster. Each node which is not a cluster head, chooses only one link to another node in Cluster i. This link will be the link to the cluster head of cluster i.*

Proof. Only Rules $R3.a$ and $R3.b$ can be executed in this case. From Rule $R3.a$, the link to the cluster head will be in forwarding mode and from Rule $R3.b$, all other link to nodes in the same cluster ($m.cl\text{-}id$= $cl\text{-}id_i$) will be in blocking state. □

Algorithm 2 $MaxCwST$ on Node i

$MaxCwST$: Clustering and Spanning Tree Algorithm
id_i: Identity of node i
$m.cl\text{-}id$: Identity of the cluster in message m
$cl\text{-}id_i$: Identity of the cluster of Node i
$cl\text{-}id_{adj}$: An array that contains identity of gateway $cl\text{-}id_{adj}.id$ which can be used to access to neighbor cluster $cl\text{-}id_{adj}.cl\text{-}id$
$Forwarding$: In this state, the link will transmit data packets
$Blocking$: In this state, the link will transmit only $hello$ message
$Port \in \{Forwarding, Blocking\}$
$Port(cl\text{-}id_{adj})$: Identity of the link which can be used to join neighbor cluster $cl\text{-}id_{adj}$
$Port(Max(cl\text{-}id_{adj}))$: Identity of the link which can be used to join a neighbor cluster which have the highest identity
$Port(j)$: The link which connects a node i to a node j
NPC: contains $cl\text{-}id$ and identity of gateway chosen
$NPC.id$: identity of gateway contained in the variable NPC
R1)
if $(Status_i = CH) \wedge (cl\text{-}id_{adj}.id > 1)$ **then**
  $NPC \leftarrow Max(cl\text{-}id_{adj}.id)$
end if
On receiving $Hello(j, Status, cl\text{-}id, cl\text{-}id_{adj}, NPC)$
R2.a)
if $(Status_i = CH) \wedge (\forall\ m \in Hello, cl\text{-}id_i$ ¿ $cl\text{-}id_{adj}.cl\text{-}id)$ **then**
  $Port(cl_{adj}) \leftarrow Forwarding$;
end if
R2.b)
if $(Status_i = CH) \wedge (\exists\ m \in Hello, cl\text{-}id_{adj}.cl\text{-}id$ ¿ $cl\text{-}id_i)$ **then**
  $Port(Max(cl\text{-}id_{adj})) \leftarrow Forwarding$;
  $Port(\neg Max(cl\text{-}id_{adj})) \leftarrow Blocking$;
end if
R3.a)
if $(Status_i = ON \vee Status_i = NP) \wedge ((\exists\ m \in Hello, m.Status = CH) \wedge (m.cl\text{-}id$= $cl\text{-}id_i))$ **then**
  $Port_j \leftarrow Forwarding$;
end if
R3.b)
if $(Status_i = NP) \wedge ((\exists\ m \in Hello, m.Status \neq CH) \wedge (m.cl\text{-}id$= $cl\text{-}id_i)$ **then**
  $Port_j \leftarrow Blocking$;
end if
R4.a)
if $(Status_i = NP) \wedge ((\exists\ m \in Hello, m.Status = CH) \wedge (m.cl\text{-}id$= $cl\text{-}id_i) \wedge (id_i \neq NPC.id))$ **then**
  $Port(cl\text{-}id_{adj}) \leftarrow Blocking$;
end if
R4.b)
if $(Status_i = NP) \wedge ((\exists\ m \in Hello, m.Status = CH) \wedge (m.cl\text{-}id$= $cl\text{-}id_i) \wedge (id_i = NPC.id))$ **then**
  $Port(cl\text{-}id_{adj}) \leftarrow Forwarding$;
end if

From this property, for each node in a cluster, only one link to another node in the same cluster will be in forwarding state. So, in each cluster, we have a spanning tree.

Property 6.3 *Each cluster chooses only on father cluster*

Proof. We need to examine two cases. Either the node which can communicate with another cluster is a cluster head, or it is a gateway. For the first case, like a cluster head

is also a gateway, from Rule $R2.b$, only one link to another cluster will be chosen. In a second case, from Rule $R1$, the cluster head of each cluster chooses the highest identity of neighbor cluster and from $R4.a$ and $R4.b$ only one link will be chosen. So a gateway node chooses also only one father. □

From the previous properties, each cluster head chooses only one gateway node which has the permission to activate its link to a father cluster and in each cluster, each node chooses only one link towards a father node. Only the node with the highest identity activates all its links. So, on a graph of n nodes, only $n - 1$ nodes have only one link in forwarding state. From Property 6.1, we have a tree.

7 Conclusion

In this paper, we have proposed the first deterministic and self-stabilizing algorithm for partitioning a network into multiple clusters which simultaneously constructs a spanning tree on the network and on the cluster. After at most $D + 6$ rounds, the spanning tree is created and the clusters are formed. Each node just needs to discover its neighborhood and their identity. No global knowledge is required to make the spanning tree. Moreover, we do not need maintain phase to maintain the cluster. Our solution is self-stabilizing and self-organized on an ad hoc network. The presented algorithm may be easily and efficiently applied in a broadcasting protocol for a distributed network. Unfortunately, our solution overload the cluster heads because they are responsible of the choice of right gateway to propagate informations. We need to find solution to this problem to achieve an improved version of this algorithm.

References

[Bas99] Stefano Basagni. Distributed Clustering for Ad Hoc Networks. In *ISPAN*, pages 310–315, 1999.

[BKL01] Prithwish Basu, Naved Khan, and Thomas D.C. Little. A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks. In *In International Workshop on Wireless Networks and Mobile Computing (WNMC2001*, pages 413–418, 2001.

[BKP02] Claudio Basile, Marc-Oliver Killijian, and David Powell. A survey of dependability issues in mobile wireless networks. Technical Report 02637, LAAS, Toulouse, 2002.

[CDT00a] Mainak Chatterjee, Sajal K. Das, and Damla Turgut. An On-Demand Weighted Clustering Algorithm (WCA) for Ad hoc Networks. In *In Proceedings of IEEE GLOBECOM 2000*, pages 1697–1701. ACM Press, 2000.

[CDT00b] Mainak Chatterjee, Sajal K. Das, and Damla Turgut. A Weight Based Distributed Clustering Algorithm for Mobile ad hoc Networks. In *HiPC '00: Proceedings of the 7th International Conference on High Performance Computing*, pages 511–521, London, UK, 2000. Springer-Verlag.

[CDT02] Mainak Chatterjee, Sajal K. Das, and Damla Turgut. WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks. *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks*, 5:193–204, 2002.

[Chi97] Ching-Chuan Chiang. Routing In Clustered Multihop, Mobile Wireless Networks With Fading Channel, 1997.

[CR09] Suchismita Chinara and Santanu Kumar Rath. A Survey on One-Hop Clustering Algorithms in Mobile Ad Hoc Networks. *J. Netw. Syst. Manage.*, 17(1-2):183–207, 2009.

[CSS04] J. Cartigny, D. Simplot, and I. Stojmenovic. An Adaptive Localized Scheme for Energy-efficient Broadcasting in Ad hoc Networks with Directional Antennas. In I. Niemegeers and S. Heemstra de Groot, editors, *Proc. 9th IFIP Int. Conf. on Personal Wireless Communications (PWC 2004)*, volume 3260 of *Lecture Notes in Computer Science*, pages 399–413, Delft, The Netherlands, 2004. Springer-Verlag, Berlin. Best paper award.

[Dij74] Edsger W. Dijkstra. Self-stabilizing systems in spite of distributed control. *Commun. ACM*, 17(11):643–644, 1974.

[EOD08] Kayhan Erciyes, Deniz Ozsoyeller, and Orhan Dagdeviren. Distributed Algorithms to Form Cluster Based Spanning Trees in Wireless Sensor Networks. In *ICCS '08: Proceedings of the 8th international conference on Computational Science, Part I*, pages 519–528, Berlin, Heidelberg, 2008. Springer-Verlag.

[EWB88] A. Ephremides, J-E. Wieselthier, and D-J. Baker. A design concept for reliable mobile radio networks with frequency-hopping signaling. *NASA STI/Recon Technical Report N*, 89:17772–+, September 1988.

[GPL99] M. Gerla, G. P, and S-J. Lee. Wireless, mobile ad-hoc network routing. In *ACM FOCUS*, 1999.

[JN06a] Colete Johnen and Le Huy Nguyen. Self-stabilizing Weight-Based Clustering Algorithm for Ad Hoc Sensor Networks. In Sotiris E. Nikoletseas and José D. P. Rolim, editors, *Algorithmic Aspects of Wireless Sensor Networks, Second International Workshop, ALGOSENSORS 2006, Venice, Italy, July 15, 2006, Revised Selected Papers*, volume 4240 of *Lecture Notes in Computer Science*, pages 83–94, 2006.

[JN06b] Colette Johnen and Le Huy Nguyen. Robust Self-stabilizing Clustering Algorithm. In *OPODIS*, pages 410–424, 2006.

[JN09] Colette Johnen and Le Huy Nguyen. Robust self-stabilizing weight-based clustering algorithm. *Theor. Comput. Sci.*, 410(6-7):581–594, 2009.

[LHS03] Ning Li, Jennifer C. Hou, and Lui Sha. Design and Analysis of an MST-Based Topology Control Algorithm. In *INFOCOM*, 2003.

[MJ06] Ricardo Marcelin-Jimnez. Locally-Constructed Trees for Adhoc Routing. In *PWC*, pages 194–204, 2006.

[Pri57] R. C. Prim. Shortest connection networks and some generalizations. *Bell Systems Technical Journal*, pages 1389–1401, nov 1957.

[YC03] J.Y Yu and P.H.J Chong. 3hBAC (3-hop between adjacent clusterheads): a novel non-overlapping clustering algorithm for mobile ad hoc networks. 1:318–321, 2003.

[YZ07] Wei-Dong Yang and Guang-Zhao Zhang. A Weight-Based Clustering Algorithm for Mobile Ad Hoc Network. *Wireless and Mobile Communications, International Conference on*, 0:3, 2007.

System Level Test of Service-based Systems by Automated and Dynamic Load Partitioning and Distribution

Mirko Caspar, Matthias Vodel, Wolfram Hardt

Faculty of Computer Science, Chemnitz University of Technology, Germany
mica | vodel | hardt @cs.tu-chemnitz.de

Abstract: Load generation is a suitable concept for system level test of complex systems. We present a novel approach especially for service based systems. Test scenarios are defined as inputs and describe a quantification of the wanted load to a service. A test-automation component partitions and distributes these abstract load values to a set of clients. These clients execute service requests to generate the calculated load. Since the partition and distribution are done during the test runtime, it can handle a dynamic set of clients with fluctuating resources. The proposed framework will be able to test a wide range of systems. It is aimed to run system level tests under laboratory and field test conditions.

1 Introduction

The increasing complexity of systems causes a big challenge for *all* steps of the design process. Hence, not only the specification and implementation of a new system is difficult and complex but also the steps of testing. Many test paradigms aim not to use expert knowledge about the implementation of a module but to define specification related test cases. On the lower levels of the design process the specification of single modules can be very detailed. This allows the easy generation of suitable test cases. In contrast, the specification on the high level - especially on system level - is abstract and imprecise. The derivation of suitable test cases is difficult and needs experience. The execution of the test may be extensive.

It is some kind of best practice for laboratory system level tests to generate load for the system under test (SUT) by client devices. The test clients are part of dedicated test stations or allocated in a wide area and connected by a wired network. In most cases, test scripts for static or random scenarios are used. These scripts are not able to react dynamically to changes in the client infrastructure in a controllable or meaningful way.

Furthermore, field tests are used to check the SUT in a real world scenario. A restricted user group is allowed to use the system as it is intended to be used. The test cases are generated implicitly by the users, since they use the service of the SUT for daily work. The test engineers are completely dependent on the user behaviour and can hardly influence the generated test cases, respectively the load to the system.

In this paper we present a concept and a constitutive framework which supports the test engineer and solve the mentioned disadvantages of field / laboratory tests on system level. Our approach is optimised to test service providing systems, where the SUT offers any kind of well defined services for clients. The availability of the clients and their resources may change during runtime. For test support, we extended the well-known concept of load generation. Popular approaches base up on static scripts. In opposite, we aim to generate suitable test tasks for each test client automatically. The generation is done dynamically during the runtime and uses abstract test scenario functions as inputs. Another component maintains the set of clients and distributes the test tasks among them. The presented work is in state of proof of concept.

2 Classification and Related Work

A popular model for the description of the development and test process is the V-model [FG99]. It describes that development and testing are done within different levels of implementation/integration. The second important conclusion is that the specification of a test is done during the development in the according level. Hence, details about the implementation on lower levels cannot influence the test specification. On the other hand the executed test can only be used to check the system implementation of its own level.

Even the V-model has been proposed in the context of software engineering, the main ideas can be used for the realisation of all kinds of information systems. Our approach is placed on the level of system test and above. It is a framework to generate load on the SUT and to report about success or failures. The detailed analysis must be done by checking traces and performance keys of the SUT itself.

A lot of work has been done in the area of testing. We concentrate our inquest to the area of formal approaches for test case and load generation on system level.

IBM presents a linear programming test case generation for SoC [NSZ06]. The functional dependencies are modelled by a linear program to limit the set of test cases. Soft constraints are introduced to model random tests. The usage of system is restricted to generate binary vectors as test cases. The extension to complex service based systems is not useful.

Krishnamurthy et. al. present a system to test session based server architectures [KRM06]. They use scripts of user inputs to generate static load scenarios for the SUT. The load itself is generated locally on the server, so that no clients are used. The idea to use linear programming for the generation of 'good' test cases is interesting. All test cases are generated as preparation for the test. A dynamic adoption during test time is not provided.

ServMark is an approach related to ours. It is a framework for performance tests of grid systems and webservices. It is composed of DiPerf [DRR+04] and GrenchMark [IE06] and is set up on PC-based test clients that are connected by a wired network. Remote

procedure calls (RPC) are used by the test server to start tests on the clients. In contrast to our approach, the set of clients is fixed and cannot be changed during the test. A failure of a client causes the failure of the test. Additionally, ServMark is completely adapted to the test of grid systems.

3 Structure of Test System

As mentioned in the introduction, our approach is designed for testing service providing systems (SPS). An SPS is a system with a set of resources which are encapsulated by well defined interfaces. Requests of service consuming clients to these interfaces are processed and responded by the according service implementation. For our approach it is necessary that the load of each service is quantifiable. This means that there is a meaningful value $l_i \in N$ for the service i that describes the load for this service.

An easy example for an SPS is a computer running some server services, like a HTTP server and a network file system. Both services provide a well defined interface - the protocol. Clients with a routed network connection to this server can use the services. The loads l can be defined differently, depending on the situation. For example the load of the HTTP can be the number of open connections and the load of the NFS can be the current upload rates (to the clients).

Beside this simple example, more complex scenarios are possible. For instance, a whole cellular network can be seen as SPS. The entire infrastructure that is necessary to provide a mobile network is part of the SUT: antennas, hardware, software and the network infrastructure. The provided services may be voice calls, data calls and message services. The mobile phones represent the test clients.

This example illustrates the problems that have to be solved. Mobile phones are small devices with limited resources. They can lose the connection to the mobile network or run out of energy so that they cannot be used as test clients temporarily. In this section we will describe our approach for the test system and its components. It is illustrated in figure 1.

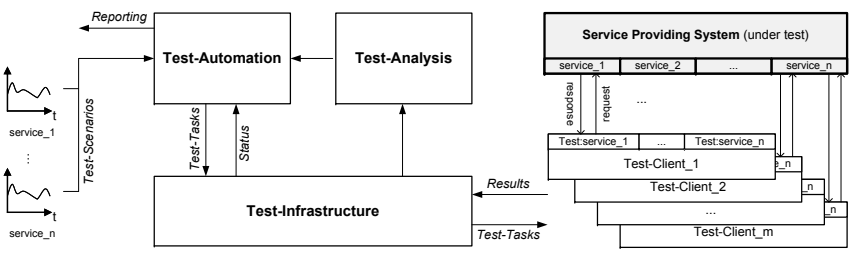


Figure 1: Overview over the whole test system structure.

The basic idea of our concept is to use the clients to generate a controllable number of requests to the SUT services. A central test server controls the clients and sends test tasks

to them. So the test of SPS can be influenced during runtime. In opposite to the approaches presented in section 2 we have to consider that the client architecture is not static. It may change during the run of the test. Furthermore, the available resources of the client devices may differ. Not every client is able to handle a special test task at any time. To use the clients remotely, they need to run a control unit which can receive test tasks and can start requests to the services of the SUT.

A test may consist of hundreds of test clients. A manual control of this set is not possible for a human test engineer. In consequence it is essential to provide the engineer a more abstract view and definition of the test. Hence, we define a test for a single service as a function of quantified load l depending on time. This describes how many requests for a service have to be executed by the clients at a defined time.

Obviously, this value of load has to be partitioned and distributed to test tasks for a set of test clients. The availability of each client and its available resources must be considered for the calculation. Hence, a simple and popular script based approach is not suitable anymore. An algorithm has to be implemented which is able to calculate the distribution based on the available client resources. Details about the model and the algorithm will be given in the next section.

Usually, some kind of communication network is necessary for the clients to communicate with the SUT. This can be used for the communication of the test server with the clients too. In special cases, the communication channel can also be part of the SUT. An example is the cellular network infrastructure that was mentioned above. The data links of this network can be used for communication between clients and test server even if they are a part of the services that has to be tested. This scenario leads to some special requirements that are not mentioned in detail here.

Since our concept aims for the load generation for black box systems, we do not have any information about the success or the performance of tests from the SUT itself. We introduce state and result messages which are sent from the clients to the test server. The test server manages the received messages and calculates the states of the test tasks and the test client. Additionally, all results are stored in a database too to allow an analysis of the test.

Furthermore, state changes of test tasks are reported to the test-automation system as well as information about the availability of the clients. This information is needed for the calculation of the next test tasks and the distribution to the clients.

A test-analysis component evaluates the results of the tests and calculates overall results that can be visualised for the test engineer.

4 Model and Algorithm

The test-automation component calculates the test tasks that are mapped to available clients. It is a generic part of the test system which is not adopted for the test of a special SUT. Hence, it needs two inputs as description of a concrete test: a model of the SUT and the conditions of the test itself.

The definition of the test is given by 2 sets. First, a set of functions $\{f_1(t), ..., f_n(t)\}$ where each one describes the wanted load l in time for a service. The function f for the service i is called test-scenario and defined by $f_i : T \rightarrow N$. Secondly, the test engineer has to define which client is able use which service at what point of time. It is defined as a set of functions $\{g_1(s,t), ..., g_n(s,t)\}$. The function g for the service i is defined as $g_i : S \times T \rightarrow \{0,1\}$, where S is the set of known clients. Obviously, the value 0 for a client i means that it shall not take part on the test. This set of functions can be defined manually by the test engineer or automatically generated, e.g. based on a random distribution.

The SUT itself cannot be described directly since it is a black box test and information about the SUT are rare. The only relevant information are the available services $1, ..., n$ that have to be tested. It is important for the test system to describe how the clients can use these services. Mainly, this is done by the implementation on the clients. It realises the necessary technologies and protocols to use the services.

The test-automation has to calculate the quantification of service load that has to be started by each client. Therefore, we have to model the resources of a client and optionally a simulated user behaviour. This can be easily described by a set of 2-dimensional cost functions $\{c_{s,1}(l_1, t), ..., c_{s,n}(l_n, t)\}$ for each client s. Each function is depending on time and the load quantification l_i (introduced in section 3). They have to be defined for each service and each client: $c_{s,i} : \mathbb{N} \times T \rightarrow \mathbb{R}$, where T is the time. Hence, $|S| \cdot n$ cost functions have to be defined. They can also be equal for classes of client devices.

Furthermore, it is necessary to define the maximum load that can be generated by a client for a service. A vector $\{a_{s,1}, ..., a_{s,n}\}$ defines the absolute values of client s for all services $1, ..., n$.

With the definition of this model and the test functions f and g the test-automation is able to generate test tasks for each available client. In detail, we are looking for a vector $\{l_{1,i}, ..., l_{n,i}\}$ of natural numbers, that defines the generated amount of load for each client $1, ..., n$ and the service i.

Necessarily, the runtime of the used algorithm will cause a discrete time system of the whole test process. For this reason the calculated load vectors must be considered as time dependent. To express this, the elements of the mentioned solution vector l will be tagged with the time index: $\{{}^t l_{1,i}, ..., {}^t l_{n,i}\}$. Since clients can get temporarily unavailable during the test (lose resources) this has to be modelled too. In addition to the input vector g the $(|S| \cdot n)$ matrix $G' = (g'_{s,i})$ is used to provide this information to the algorithm. The value

$g'_{s,i} = 1$ represents that the client s is currently available for the test of service i. The content of the matrix is maintained by status messages from the test-infrastructure.

The costs of the solution shall be optimised concerning the given cost functions $c_{s,i}$. The following equations and inequalities describe the common optimisation problem that has to be solved.

$$({}^t l'_{s,i}) \in N \quad : \quad \sum_{s \in S} \sum_{i=0}^{n} \left(c_{s,i} \left(({}^t l_{s,i} +^t l'_{s,i}), t \right) \cdot g'_{s,i} \cdot g_i(s,t) \right) = min! \tag{1}$$

$$\forall s \in S, i \in [1,n] \quad : \quad {}^t l_{s,i} +^t l'_{s,i} \leq a_{s,i} \tag{2}$$

$$\forall i \in [1,n] \quad : \quad \left| \sum_{s \in S} \left(({}^t l_{s,i} +^t l'_{s,i}) \cdot g'_{s,i} \cdot g_i(s,t) \right) - f_i(t) \right| \leq \varepsilon_i \tag{3}$$

Opposite to the mentioned absolute load value ${}^t l_{s,i}$ for each client we are now looking for a relative value ${}^t l'_{s,i}$. It describes the alteration of client load for the service i. This separation is necessary due to possible technical restrictions of the clients. For example, the alteration may be limited to adding more requests since the implementation does not allow cancelling requests.

The equations (2) and (3) describe the constraints that have to be satisfied by a correct solution of the equation system. On the one hand it has to be checked that the calculated value is not greater than the maximal allowed load value for this client. On the other hand the summation of all calculated values of a service i has to be equal to the wanted service load $f_i(t)$ (considering an uncertainty of ε).

We have chosen 3 different kinds of algorithms with different grades of optimisation accuracy for further analysis.

The first algorithmic approach is a linear program. Since we do not restrict the functions $c_{s,i}$, we cannot guarantee a convex solution space for the equation (1). A new target function has to be deviated to get a linear program. Hereby, the costs can only be integrated as a linear combination with the values ${}^t l'$, as given now with equations (4) and (5).

$$({}^t l'_{s,i}) \in N \quad : \quad \sum_{s \in S} \sum_{i=0}^{n} \left(d_{s,i} \cdot^t l'_{s,i} \right) = min! \tag{4}$$

$$d_{s,i} \quad = \quad c_{s,i}({}^t l_{s,i}) \cdot g'_{s,i} \cdot g_i(s,t) \tag{5}$$

Since the co-domain of ${}^t l'$ is discrete, the given problem is an integer linear program (ILP) which is known as NP-hard . To solve the problem in polynomial time, we decided to use LP-relaxation [HO02] and to cast the results to integer values.

The ILP model has the serious restriction, that the costs can only be modelled by a linear factor whereas the daily testing practice may generate non linear and even discontinuous functions. We decided to develop a heuristic algorithm that allows to consider any kind of cost functions and to calculate satisfying results in polynomial time. We are using the fact that the whole system is necessarily time discrete. Changes in the input values of the equations are derived from status messages of the clients or from changes in the function value f_i. Furthermore we introduce the restriction, that every cost function $c_{s,i}$ for a service i does not depend on the load value of any other service j. Under these conditions, load changes can be handled for a single service.

Since the described test automation problem is similar to partitioning problems we map it to a hierarchical-clustering algorithm [GVN$^+$94]. After a load change $\triangle l_i$ has occurred, it is separated in pieces of equal size and mapped to every available node. A closeness-matrix is calculated where the value $a_{i,j}$ expresses the difference of costs when the new piece of load is transferred from client i to j. The operation that leads to maximum saving is executed and the closeness-matrix is updated. This iterates until no more optimisation can be reached. The complexity of this algorithm is $O(n^2)$ which results mainly from the generation of the matrix.

As a third type of algorithm we are going to use a random distribution. It does not consider the cost minimisation of equation (1). Independently, the constraints of equations (2) and (3) must be fulfilled. This approach will be used to analyse the test quality of randomised partitioning and distribution of test scenarios.

5 Discussion

In this section we want to expose the advantages and prospects as well as some limitations of our approach. We will present our simulation concept to proof and optimise the system.

5.1 Concept

The most important advance of our approach is the ability to handle a dynamic client structure and heterogeneous resources of these clients. It is realised by the dynamic recalculation of the necessary load values which can take account on the availability of clients or resources. Furthermore, this changing availability can also be simulated in laboratory tests by the accordant calculation of g (see section 4). If all secondary conditions, like user behaviour or resource availability, are deterministic, the test can be reproduced.

On the other hand there are some limitations due to the proposed model, the algorithms

and technological conditions. The most distinguished limitation is the time discretisation that is caused by the runtimes of algorithms and messages. This leads to time periods on each client, which are not controllable in detail. Hence, differences between the wanted load (given by the test scenarios f) and the time actual real load may occur.

Technological restriction on the clients may lead to a second problem of accuracy. We know services, where load can only be generated indirectly. E.g. network traffic may only be generated by downloading files. Hence, the real load value l that is generated by a client is a function of time and depending on the available resources of the service itself, the network and the client resources. If this value cannot be measured and reported from the client to the test system, the value has to be estimated.

5.2 Implementation and Simulation

Our system is able to manage several hundreds of test clients. It is difficult to get access to so many 'real' clients. So we decided to use a special adoption of a simulation environment to check our concept, evaluate parameters and show the performance.

The basis is SimANet [VSC+08], a modular and extensible simulation environment. It is used to simulate the behaviour of huge wireless networks with different wireless communication standards [VCH10]. SimANet has been optimised to run on parallel systems to increase the simulation performance. Movement models are available to change the positions of network nodes.

We implemented the mentioned cellular network example. The clients and the mobile network will be simulated in SimANet. The automation- and infrastructure-system are implemented as a server that is communicating with the simulation. Each simulated client runs a program to start service requests and to communicate with the test server by virtual sockets of the simulated network.

To realise this, some enhancements of SimANet were necessary. This contains the implementation of an access point mode to simulate the behaviour of a cellular network. Furthermore, a capability to allow clients the allocation of all necessary services or resources is implemented now. Hence, we have the possibility, to maintain and check the current load to each service easily.

The implementation of the test-infrastructure component is finished. It is responsible to maintain a list of available clients, send test tasks to the clients and receive result messages from them. To test this implementation, we simulated up to 100 clients that has to be controlled and maintained by this infrastructure component. Just by the possibility to simulate so many parallel clients, we were able to find some critical errors in our test-infrastructure component. Most of them were related to parallel access to data structures. Due to this success we are confident that the simulation approach is feasible.

The test tasks are generated by static scripts since the proposed algorithms are in an evaluation process. The LP model is implemented as a Matlab program whereas the clustering heuristic is a dedicated Java-program. First runs of the heuristic algorithm show a good runtime but weaknesses in the optimisation results.

5.3 Timing

As explained above, the timing behaviour of all the test process is important for the quality of the load generation process. For this reason it is one of the most important properties that have to be analysed and optimised by the implemented simulation. The significant delays are shown in figure 2 (A).

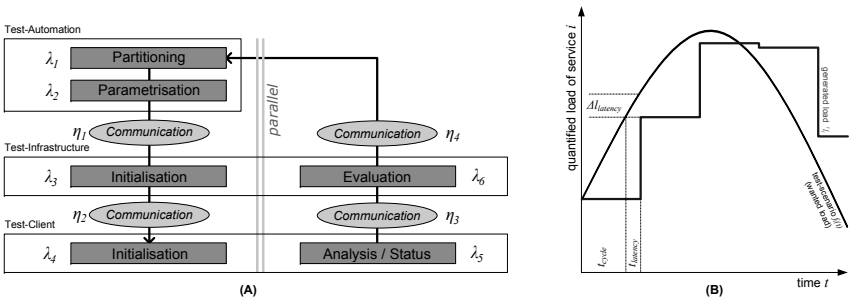


Figure 2: Timing behaviour of the test system.

Based on this timing model, two important values can be derived. The cycle time t_{cycle} is responsible for the update speed of calculation and hence for the accuracy of the load generation. The $t_{latency}$ is the delay that occurs after the calculation has been started until the clients execute the new calculated service requests. The cycle and latency of the system can be derived as follows:

$$t_{cycle} = max(\lambda_1, ..., \lambda_6, \eta_1, ...\eta_4) \tag{6}$$

$$t_{latency} = \sum_{i=1}^{4} \lambda_i + \sum_{i=1}^{2} \eta_i \tag{7}$$

Obviously, λ_1 is the runtime for the test-automation algorithm. Most likely, it will be the maximum value of all delays so that it is the basis for the interval time t_{cycle} of our time discrete system. It must be the aim to keep it as small as possible.

The other delays down to λ_4 are not eminent for the performance of the system. They are part of a kind of pipeline that has to be passed by each generated test tasks. The

summation of these delays is the time that passes until the generated test task is started as service request. Parallel to the transmission of a test tasks, status and report messages are sent to the automation system where they have to be evaluated. The delays are mainly depending on the communication and server system and not part of a critical path.

Figure 2 (B) illustrates the influences of t_{cycle} and $t_{latency}$ to the generated load. The cycle leads to periods of time, where the load is constant and cannot be adopted to the wanted load, given by f_i. The latency may lead to a difference between the wanted and the generated load at a point of time.

6 Conclusion

We presented an approach for the automated and dynamic load generation to support the system level test of service based systems. The primary objective deals to control a set of test clients from a central test server. This test server generates test tasks for all the clients based up on test scenarios. A test-infrastructure component transmits the tasks to the target client and manages status and result messages from the clients. The clients generate quantified amounts of service requests to the SUT following the instructions of the test tasks. In opposite to existing approaches, we are able to manage a heterogeneous and dynamic set of clients with fluctuating resources. Central component of the system is the test-automation. It has to partition and allocate the test scenarios to concrete test tasks.

Basic parts of the framework, like the test-infrastructure component, are finished. The work for the test-automation framework has been started whereas the LP model and the heuristic algorithm are already implemented. It is part of continuative work to improve the quality of the heuristic algorithm to find better solutions for the test-partitioning problem. With the framework and the presented simulation platform we are going to compare the performance of the LP, the heuristic and a random based algorithm.

When the system is stable with the given algorithms, we are going to analyse some prediction models to improve the mentioned inaccuracies caused by the latencies and the cycle.

Beside the area of testing, we noticed that there are similar algorithmic problems in the area of parallel computing. Load balancing aims to map tasks to the processors during runtime [SKH95] [MCS+09]. The result should be that no processor is idle when there are available tasks. Further research work will analyse if parts of our work can be used for this problem too.

References

[DRR+04] C. Dumitrescu, I. Raicu, M. Ripeanu, and I. Foster. DiPerF: an automated distributed performance testing framework. In *Grid Computing, 2004. Proceedings. Fifth IEEE/ACM International Workshop on*, pages 289–296, Nov. 2004.

[FG99] Mark Fewster and Dorothy Graham. *Software Test Automation: Effective use of test execution tools*. Addison-Wesley Professional, Harlow, 1999.

[GVN+94] Daniel D. Gajski, Frank Vahid, Sanjiv Narayan, and Jie Gong. *Specification and design of embedded systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994.

[HO02] Juraj Hromkovic and Waldyr M. Oliva. *Algorithmics for Hard Problems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2002.

[IE06] Alexandru Iosup and Dick Epema. GRENCHMARK: A Framework for Analyzing, Testing, and Comparing Grids. In *CCGRID '06: Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid*, pages 313–320, Washington, DC, USA, 2006. IEEE Computer Society.

[KRM06] D. Krishnamurthy, J.A. Rolia, and S. Majumdar. A Synthetic Workload Generation Technique for Stress Testing Session-Based Systems. *Software Engineering, IEEE Transactions on*, 32(11):868–882, Nov. 2006.

[MCS+09] A. Moreno, E. Cesar, J. Sorribes, T. Margalef, and E. Luque. Task distribution using factoring load balancing in Master–Worker applications. *Inf. Process. Lett.*, 109(16), 2009.

[NSZ06] Amir Nahir, Yossi Shiloach, and Avi Ziv. Using Linear Programming Techniques for Scheduling-Based Random Test-Case Generation. In Eyal Bin, Avi Ziv, and Shmuel Ur, editors, *Haifa Verification Conference*, volume 4383 of *Lecture Notes in Computer Science*, pages 16–33. Springer, 2006.

[SKH95] Behrooz A. Shirazi, Krishna M. Kavi, and Ali R. Hurson, editors. *Scheduling and Load Balancing in Parallel and Distributed Systems*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1995.

[VCH10] Matthias Vodel, Mirko Caspar, and Wolfram Hardt. Embedded Ambient Networking - A New, Lightweight Communication Concept (accepted). In *Proceedings of the 9th International Conference on Communications (ICC)*, Cape Town, South Africa, May 2010. IEEE Computer Society.

[VSC+08] Matthias Vodel, Matthias Sauppe, Mirko Caspar, and Wolfram Hardt. SimANet A Large Scalable, Distributed Simulation Framework for Ambient Networks. *Recent Advances in Information Technology and Security - Journal of Communications (EI Compendex)*, 3(7):11 – 19, Dezember 2008. ISSN: 1796-2021.

Ad-hoc Community Composition of Rescue Forces in Action Situations

Volkmar Schau[1], Kathrin Kirchner[1], Christian Erfurth[2] and Gerald Eichler[3]

[1]Friedrich Schiller University Jena, Germany
Volkmar.Schau@uni-jena.de, Kathrin.Kirchner@uni-jena.de

[2]University of Applied Sciences Jena, Germany
Christian.Erfurth@fh-jena.de

[3] Deutsche Telekom AG, Laboratories, Germany
Gerald.Eichler@telekom.de

Abstract: Secure energy and transport networks, Internet and telecommunications, are vital nerves of our highly networked society. Global mobility makes it difficult to combat and natural disasters and technological accidents can cause serious damage in a closer-knit world. In this paper we introduce the challenges in dynamic community composition of heterogeneous rescue forces (culture clash) for rescue and protection tasks and discuss our approach in the context of the "SpeedUp project" how to present a solution for ad-hoc communication, situation-aware representation, tracking and guiding in dynamic inhomogeneous communities.

1 Motivation

Mostly unexpected events like natural causes or major loss are challenges for rescue forces. Within shortest time rescue teams of different public authorities need to sum up the situation, recover injured persons and attend to them, safe the spot, secure evidence, and much more. Rescue forces account for a subset of such tasks and work task-driven according to their organizational structures. Thus typically police, fire service, emergency medical service, and other authorities have separated areas of operation and different objectives. However an interaction and cooperation between forces is important to ensure a fast response to a rescue mission.

According to their tasks every rescue force has a specialized information demand. While fire workers need to know critical places on site and their team leader monitor vital functions, police officers want to gather IDs of involved persons and register any kind of potential evidence for further investigations. Each rescue force can be seen as a community which needs its information to coordinate actions. Information sources and the flow of information is different and depends for instance on type of information, type of situation (context of rescue mission), and organizational structures. In advance official instructions try to define regulations for information needs, flows, and organizational buildups. However every situation has its own instance and specialty. In addition the practical im-

plementation of instructions is often slightly different (and often has a local interpretation too). Generally interfaces to other forces are not defined. In part this leads to repeated information gathering. Information sharing between communities would be preferable since one could see the different rescue forces as a community as a whole.

The first information on a critical event which requires rescue response is typically given via an emergency call. This is an incoming call at an emergency control center[1] providing the service to inform rescue forces depending on the type of event. The received information and potentially further relevant information (e. g. access ways) will be overhanded to the rescue forces. The communication is done using radio. The control center receives status information on rescue forces (e. g. arrived at spot). Quickly the first arrived team sums up the situation and decides whether additional support is necessary or not. The control center is a kind of back office which will care for further resources (e. g. additional man power, tools, special information, food) needed by teams on the spot. Partly needed information is acquired by rescue forces itself. Collected data on site are tried to be gathered at a (local) central point. This is typically done with paper and pencil and blueprints. In an emergency case of a larger scale information is lost, late, incomplete, wrong, or even duplicated and, therefore, may prevent efficient rescue missions.

2 Challenges of IT Support

One consequence which can be drawn from the introducing section is apparent: The field of rescue needs support. Modern IT technology may help if applied in a meaningful way. Investigations of the European Security Research and Innovation Forum (ESRIF) have defined European Research and Security needs in its final report[Eur09]. The working group on crisis management has outlined core challenges, e. g. strengthening response forces, situation awareness and decision making, cooperation, managing resources, and recovery logistics. Information and information flows, communication support, simulation, and training are some of the research and innovation needs. However improvements are partly hard to achieve. Complex aspects need to be taken into account for the implementation of a system and its establishment, e. g.:

- constraints of typical missions,
- clear goals of technology application,
- different organizational structures,
- technological base,
- readiness of use by rescue forces,
- handling, ergonomic aspects

For an identification of effective IT support an extensive analysis of the rescue organizations (static, cultural aspects) and typical processes in rescue including communication is essential. Rescue workers of different organizations cooperating in missions are in the

[1] In Germany we have control centers operated by police, by fire service, by medical service or combined versions.

focus of the collaborative project SpeedUp[2] [Spe10] which has started with such an analysis. SpeedUp seeks for an community focused IT solution by supporting organizational structures and needs as well as improving communication within and between organizations. Rescue workers will be supported in providing better information and, thereby, in coordinating missions more efficient. Figure 1 points out the interplay of organizations, information, and IT with main challenges.

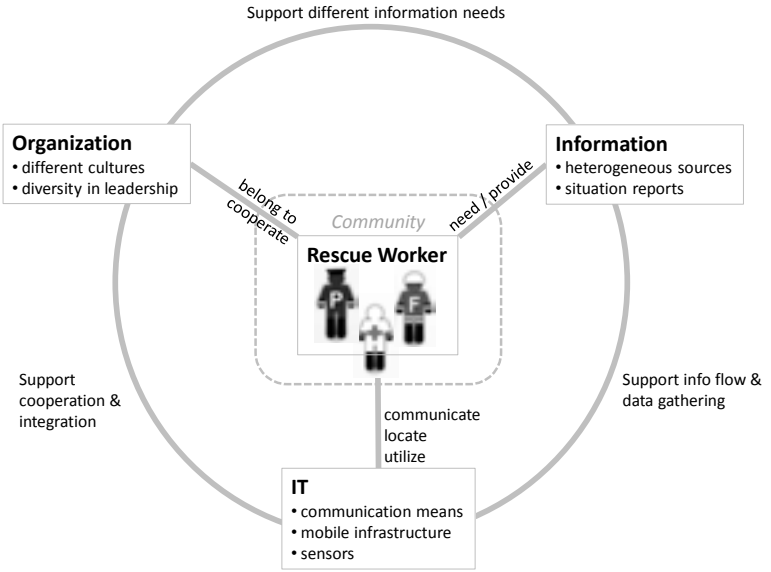


Figure 1: Interplay of organizations, information, IT and challenges with the focus on rescue workers

The main question which rises on the technological level is: How can we move toward a solution which supports various requirements resulting from this interesting application domain and its constraints? The paper outlines relevant fields of research and technology which may lead to an promising architecture if combined meaningful. In SpeedUp some of the ideas will be evaluated and integrated in an suitable framework.

3 The Information Building

Rescue forces have a strong need for any kind of information related to mission in progress. Similar to communities information will be communicate to co-workers. We would like to propose an information building which enables fast information flows, filtering of relevant

[2]SpeedUp is funded within the Federal Government's program "Research for Civil Security" [Fed10] (call "Rescue and protection of people") by the Federal Ministry of Education and Research (duration: 1 May 2009 - 30 April 2012)

information, and automated information transmissions. Figure 2 presents an overview of related technological fields to cover several informational needs of rescue forces.

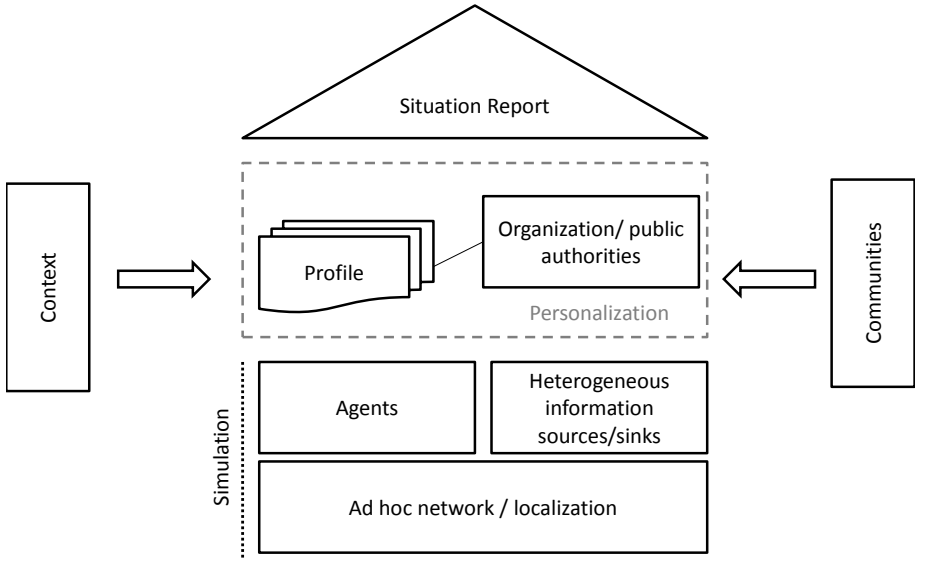


Figure 2: The information building: A technological landscape

As a prerequisite for an information flow a technical infrastructure has to be available or established. For the transmission of information devices need to be networked using ad hoc networks or if available public infrastructures like cellular networks. Localization of rescue forces and geotagged data is very helpful for further processing. A sophisticated transport of data and information as well as an seamless integration of data resources via standard interfaces is essential to cover a flexible usage of the whole system. The application of mobile software agents is promising therefore. This opens also the chance to establish a simulation system on basis of software agents for training purposes. The information flow will be accompanied by agents with the goal to deliver data to organizations and rescue workers with different roles. Which information is delivered and accessible for organization and rescue workers is guided by policies and profiles. Information needs are different for involved organization: they are task (context) and role dependent. The knowledge which is gathered during a mission should be shared by the community of rescue workers mostly independent of their organizations. This information building shows technical solution fields which will be examined in more detail toward a precise supporting solution.

3.1 Localization of Objects

Any rescue activity is usually map-driven. Therefore, it is essential to be informed about the current position of any interesting object. Objects are persons, materials and environments, where persons are rescue teams and victims (injured persons); materials are rescue vehicles and equipments; and environments describe to given local situation. Any of this objects can be associated with geo-coordinates within a certain area.

3.1.1 Network Support for Localization

There are typically four types of networks which can be primary or secondary used for localization of objects:

- the Global Positioning Satellite Network (GPS),
- the permanent mobile phone network (GSM/GPRS/UMTS),
- the special rescue forces digital network (TETRA)[3],
- permanent and ad-hoc meshed wireless networks (WLAN/Bluetooth).

All of these networks can be extended by additional base stations on demand with local resources, mounted on vehicles to support two goals: extend the communication capacity and increase the accuracy of localization. To incorporate different localization technologies the exact position (geo-coordinates) of any type of base station is required. Furthermore, special rescue communication devices, called multi-functional devices with multiple communication interfaces devices can act as mediator. For example, the exact position of such a movable device is found by GPS, while it acts as WLAN base station too.

3.1.2 Accuracy of Localization

GPS provides a localization accuracy of about 5 m, GSM using triangulation of about 50 m assuming multiple cells are receivable. Both holds for outdoor localization. GPS is not available in most indoor cases. However, active and passive repeaters might help slightly in rooms with non-metallic covered windows. Risky buildings with fixed and dense WLAN infrastructures could be modeled in advance for exact positioning. Ekahau systems reach accuracies down to 1 m there. Installed Bluetooth tags can enrich this as they operate in the same radio frequency band as WLAN. The following chapter will introduce, how ad-hoc mobile networks can overcome at least some issues.

Areas can be marked as dangerous by modeling polygons. As soon as an unauthorized rescue person is about entering such an area an alarm on the multi-functional device is generated or a special communication channel is established. The entire scenario should be given such a polygon as border to detect new incoming our leaving outgoing resources.

[3]The new federal German communication network of ”Behörden und Organisationen mit Sicherheitsaufgaben”.

3.1.3 Tagging

As the number of multi-functional devices is limited, permanent tags - applicable indoor and outdoor are introduced and attached to any interesting (semi) static object in the rescue scenario. There are two types: passive and active tags. Passive tags might be simple 2D-barcodes as self-adhesive labels, scanable by the camera of the multi-functional device. The tags are attached manually to objects by rescue workers, scanned at the moment of attachment and stored with its unique ID and geo-coordinates in the central database of the coordination team. Although QR-codes are currently the most spreaded barcode labels, ZigBee codes are recommended, as they have the charm to combine machine and man readable parts within one label. To omit the time consuming photo process, RFIDs in combination with visible tags are recommended. If static, later on such tags can be used as reference points. [ELAS09]

Active tags are either long range interactive RFIDs or Bluetooth tags. In contrast to the passive tags there is no need for active scan as they are recognized automatically by multi-functional devices from a certain distance. The interaction can be controlled by shaking or turning the device in a defined pattern.

3.2 Mobile Assistants

The combination of personal mobile devices and wireless ad-hoc networks allows the concept of mobile emergency ad-hoc information system, consisting of a highly dynamic, decentralized and self-organizing network of autonomous and mobile devices that interact as peers. Each mobile device represents one or more rescue units using information, feeding data or being a peer. According the rescue task such a mobile ad-hoc network must be self-organizing to be a benefit for rescue and protecting people. Thus, researchers and developers have to deal with a new set of problems peculiar of these systems, due to user and device mobility, variable bandwidth, transient loss of connectivity and no centralized structures. It is evident that, for these classes of systems, applications cannot be designed according to the conventional architectural paradigms.

Based on an emergency call rescue forces move out. The received information and potentially further relevant information like access ways will be overhanded by the emergency control center using radio. Arrived on site the first rescue team tries to get a situation report for starting rescue operations. In contrast to an individual case of an emergency the first team is exposed to a situation of radical change by making the critical decisions with far-reaching consequences. Proceeding as an individual case of an emergency rescue operations may fail to tactical procedures at the expense of human life. So the first rescue team plays a decisive role for the entire occurrence. In the first instance they have to supervise and arrange. Afterward they can save human life. Otherwise, the order is missed and there is no on-site management organization. This and the overall chaos will make future rescue operations difficult or impossible. [PMU01] Situation reports therefore have great importance. According them emergency forces initiate rescue activities immediately. Situation reports are continuously issued by on-site management. Started by the first rescue

team these reports are overhanded to the next levels of on-site management organization. Thus, we have a highly dynamic situation and management. Moreover, gathering all relevant situation data is done by radio or in a paper driven process. Disappearing of data is unavoidable. On-site command control assistants try to manage paper and information flooding. Heading this way first integration of mobile IT in Germany demonstrates significant advantages. So rescue units have identified that they could save radio effort, obtain clean data sets and keep calm. But mobile IT is a mixed blessing. Using mobile technology device mobility denotes wireless networks and no centralized structures. Furthermore, integrated rescue mobile technology requires easy to use and self-organizing structures. On site there is no time for (re)configuration or management of connectivity.

According common rescue operations let us assume we have command control assistants in software form supporting connectivity management, configuration and self-organizing IT structures. Software command control assistants mean a multiplication of command control assistants able to organize network, command and information structures. Erfurth et al. [EKR+08] present mobile assistants as one way to support networked worlds. In understanding of Erfurth et al. mobile assistants are similar to (mobile) agents. Therefore, a mobile agent is a special kind of software which can execute autonomously. [BR05] Once dispatched, it can hop from peer to peer performing data processing autonomously, while software can typically only execute when being called upon by other routines. Therefore, it seems to be that mobile agents are meant for supporting rescue forces in autonomous self-organizing networks. Self-organizing networks mean ad-hoc networks ready to use for rescue units at all times. Specifically for the rescue operations we need mobile ad-hoc networks (MANET). But rescue units pay no attention to organization and communication within the network. This part is done by mobile agents. So we call such a mobile agent mobile ad-hoc network 2MANET.

3.2.1 Mobile Agent Mobile Ad-hoc Networks

In general, mobile ad-hoc networks (MANET) are communication networks built up of a collection of mobile devices which can communicate with each other via wireless connections. [BSF08, SGF02] Peers can join or leave at any time. Routing is the task of directing data from a source peer to a given destination. Based on no fixed infrastructure all peers are equal and there is no centralized control or overview. So the routing task is quite hard in mobile ad-hoc networks. Peers serve as routers for each other, and data packets are forwarded from peer to peer in a multi-hop fashion. Due to the mobility of mobile peers and the lack of centralized structures, routing algorithms should be robust and adaptive working in decentralized and self-organizing way. [QW01, CDG05]

The new challenges include:

- the variety of communication channels;
- the communication bandwidth per channel is quite different;
- the communication bandwidth for wireless network is much lower;
- the environment is more unreliable, causing unreliable network connection and increasing the likelihood of input data to be in faulty; and

- fixed routing is impossible.

Many MANET routing algorithms have been proposed. In the literature, the classical distinction is between table-driven and demand-driven algorithms. Table-driven algorithms, such as DSDV, are purely proactive: all nodes try to maintain routes to all other nodes at all times. This means that they need to keep track of all topology changes, which can be difficult if there are a lot of nodes or if they are very mobile. Demand-driven algorithms, such as AODV, are purely reactive: nodes only gather routing information when a data session to a new destination starts, or when a route which is in use fails. Reactive algorithms are in general more scalable since they reduce routing overhead, but they can suffer from oscillations in performance because they are never prepared for disruptive events. In practice, many algorithms are hybrid (e.g. ZRP), using both proactive and reactive components.

The advantage of this kind of network is, that it does not require or even need any kind of infrastructure, like a base station in a cellular network. Therefore ad hoc networks are best suited for an environment, which is not able to provide any kind of infrastructure like for disaster recovery.

In traditional distributed networks, data are collected by source peers, and then transmitted to a higher-level processing peers which performs data fusion. During this process, large amount of data are moved around the network, as is the typical scenario in the client/server paradigm. [TZA03]

By transmitting the computation engine instead of data, in our understanding mobile agents as a special kind of software which can execute autonomously, the new formed Mobile Agent Mobile Ad-hoc Network (2MANET) offers the following important benefits:

- Network bandwidth requirement is reduced. Instead of passing large amount of raw data over the network through several round trips, only the agent with small size is sent. This is especially important for real-time applications and where the communication is through low-bandwidth wireless connections.
- Stability. Mobile agents can be sent when the network connection is alive and return results when the connection is re-established. Therefore, the performance of 2MANET is not much affected by the reliability of the network.

In our research we pursue a two way strategy. In the field we are dealing with mobile devices be part of the rescue operations. In contrast most of the time we only carry out experiments under lab conditions. Research under rescue conditions in action seems too great a risk for rescue forces in workaday life therefore we use exercises of precautionary measures in the field. Figure 3 presents a snap-shot of 2MANET rescue scenario in an exhibition hall. Each spot symbolizes one rescue staff member in action and the way covering his distance as light gray line. Other colors are indicated as follows: Red colored spot means the starting point for data. The destination is marked by green spot. Blue colored spots are whistle stops attended on the way.

The data way or the way of an agent depends on the environment availability (communication channels, communication bandwidth, peer reliability, services, etc.) per peer part

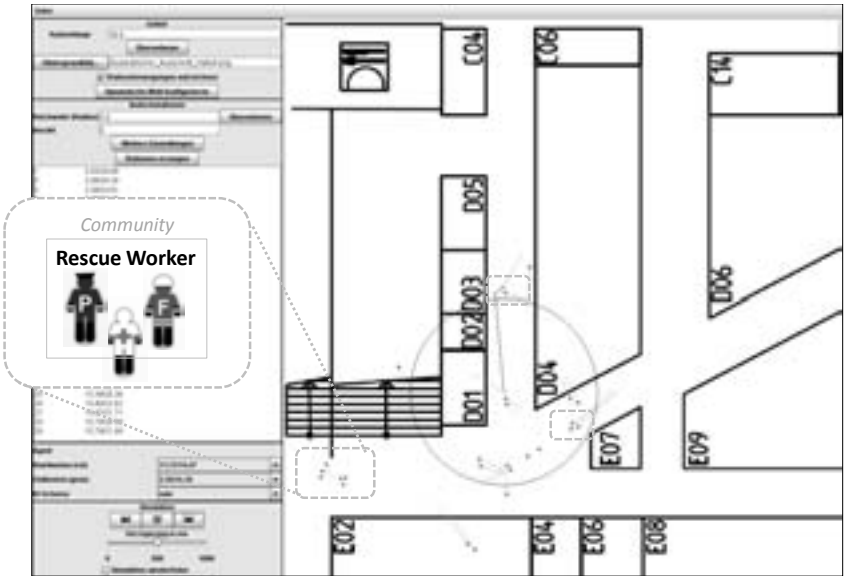


Figure 3: Mobile Agent Mobile Ad-hoc Network simulation

of that highly dynamic network occurence (in contrast a MANET is ad-hoc network in a fixed configuration). As preliminary research result we combine conventional methods of MANET hybrid routing policies, time-based memory of whistle stops, cloning strategies and decision making on peer site. This complex process is done within each agents supported by peer services. [DCG] It is the way to achieve emergency force demands of ad-hoc community composition. Depending on chronological situation on site data are information addressed to different emergency units in variable transmission. So figure 3 presents three heterogeneous communities (in the area of D03, E07 and E02) composed by firefighter, rescue service and police force. Based on the situation they are interested in differend information maybe regulated by law. Therefore, we form a metalevel as an ad-hoc community upon on site heterogenous forces. An ad-hoc community composition of rescue forces defines a situation driven need for information regulated by operation and/or by act of law.

3.3 Personalization

In our rescue force scenario, understanding the situation is a key priority for rescuers. Communication between actors is important for coordinating actions in place.

A lot of information is collected via radio communication or telephone and is written down on sheets of paper. Information can get lost, written down twice or inconsistently, and sometimes come to late to the people who urgently need this information. Besides, a huge

amount of information is collected, e.g. in the control, and rescuers face an information overflow and have to figure out relevant information.

Our research therefore focuses on new ways of offering information to the rescue force members. The rapidly increasing amount of data available needs accurate compilation depending on local workflows and the individual needs of a rescuer. Giving all information to all rescuers in an emergency situation will not be appropriate for huge amounts of data. The information should be offered according to the current place of the rescuer and near real-time. Furthermore, it is important to consider the rights of involved rescuers to receive certain information. E.g., policemen are allowed to get statistical information about the number of persons injured, but no details about concrete injuries of a certain person. Organizational aspects also play an important role in information allocation and filtering. Members within the same rescue force on different organizational levels, and members of different rescue forces have different information needs, goals and tasks to perform according to their location.

Therefore, legal, organizational and situation dependent rules have to be defined to make sure that the right information comes to the right people. A personalization of information is important. In contrast to customization, where users specify their preferences manually, automatic personalization means automatic adaptation according to user profiles. Personalization is a data-intensive process and is based on the characteristics of the user (user data, usage data) and the user's context in which he/she is located or a certain action is performed. In our context, several types of personalization can be applied: personalization of content, structure or modality (Fig. 4).

Figure 4: Different forms of personalization

Personalization of content means the automatic tailoring of information according to users' profiles, that include needs or level of expertise. Relevant information can be retrieved and the content can be selected according to the special location or context of activities that have to be performed at a given point of time. [MC00] If all information would be recorded electronically via a central database, it could be distributed with the help of a personalized approach. In our scenario, e.g. special alarm words can be used to assign information to a certain person (like information including the word injured people should first go to

emergency medical service units). Another possibility is a semi-automatic personalization approach, where people writing down there information electronically decide via click for which group of people this information should be relevant. People that receive this information can mark information that is unimportant for them, so the system can learn from user's needs and can better decide about relevant information for a specific user.

Personalizing the structure of an application refers to altering the location of content including text, images or links. This helps to identify important information, but can also facilitate a personalized navigation. [SC02] In rescue situations, relevant information have to be easy to identify. Therefore, information relevant to the exact place of the rescuer, and among these the most urgently things to be done should be displayed at first sight.

The personalization of modality enables changes from text to other types of media to present to the user. [ABD+04] These could be images, videos, maps or audio, if they are available in the system. The selection of the modality can be done according to the kind of content, or the user characteristics. For quick decision making in rescue situations, maps to visualize places of injured people is easier and quicker to understand as a long textual description.

4 Conclusion

In our paper,we introduced relevant fields for an ad-hoc community composition of rescue forces. For implementing an overall system, we have to consider the complex application domain. The understanding of processes and organizational cultures of different rescue forces defines the basis for an successful IT support. In this context, the SpeedUp project tries to integrate the users to evaluate relevant technologies and their possible implementation in rescue scenarios.

Due to different organizational structures of rescue forces and federalism in Germany,the requirements for an overall system are quite diverse. Therefore we need flexible technologies to support a lot of scenarios - starting from small action situations to major incidents. Our paper discusses a first approach to combine relevant technologies and research paradigms on different levels to outline an overall system.

In order to introduce such a system, rescue workers have to become familiar with the handling. Therefore, it is essential to simulate different rescue scenarios in a training phase. Our approach allows authentic training using the same system by simulating an infrastructure and scenarios with the help of mobile software agents. Scenarios can be displayed several times so that rescue workers can learn from different decision situations. With this focus the proposed system is promising to achieve acceptance of rescue forces.

References

[ABD+04] Stefan Arbanowski, Pieter Ballon, Klaus David, Olaf Droegehorn, Henk Eertink, Wolfgang Kellerer, Herma van Kranenburg, Kimmo Raatikainen, and Radu Popescu-Zeletin. I-centric Communications: Personalization, Ambient Awareness, and Adaptability for Future Mobile Services. *IEEE Communications Magazone*, pages 63–69, 2004.

[BR05] Peter Braun and Wilhelm R. Rossak. *Mobile Agents: Basic Concepts, Mobility Models and the "Tracy" Toolkit*. Morgan Kaufman, 2005.

[BSF08] Remi Badonnel, Radu State, and Olivier Festor. Management of ad-Hoc Networks. In Jan Bergstra and Mark Burgess, editors, *Handbook of Network and System Administration*, pages 331–360, 2008.

[CDG05] Gianni Di Caro, Frederick Ducatelle, and Luca Maria Gambardella. Swarm intelligence for routing in mobile ad hoc networks. In *In Proceedings of the 2005 IEEE Swarm Intelligence Symposium (SIS*, 2005.

[DCG] Frederick Ducatelle, Gianni Di Caro, and Luca Maria Gambardella. USING ANT AGENTS TO COMBINE REACTIVE AND PROACTIVE STRATEGIES FOR ROUTING IN MOBILE AD HOC NETWORKS.

[EKR+08] Christian Erfurth, Steffen Kern, Wilhelm Rossak, Peter Braun, and Antje Leßmann. MobiSoft: Networked Personal Assistants for Mobile Users in Everyday Life. In *Proceedings of 12th International Workshop, CIA 2008*, pages 141–161, Prague, 2008.

[ELAS09] Gerald Eichler, Karl-Heinz Lüke, Aykan Aydin, and Roland Schwaiger. Barcode Application Innovation for Smartphones. In *Proceedings of 39.Jahrestagung der Gesellschaft für Informatik 2009*, Lübeck, 2009.

[Eur09] European Security Research & Innovation Forum. ESRIF Final Report, December 2009.

[Fed10] Federal Ministry of Education and Research. Security Research - Research for Civil Security. `www.bmbf.de/en/6293.php`, 2010. [Online; accessed 10-April-2010].

[MC00] B. Mobasher and R. Cooley. Automatic Personalization based on Web Usage Mining. *Communications of the ACM*, 43:142–151, 2000.

[PMU01] Hanno Peter, Thomas Mitschke, and Theodor Uhr. *SEGmente 3 Notarzt und Rettungsassistent beim MANV*. Verlagsgesellschaft Stumpf+Kossendey m.b.H., 2001.

[QW01] Hairong Qi and Feiyi Wang. Optimal Itinerary Analysis for Mobile Agents in Ad Hoc Wireless Sensor Networks, 2001.

[SC02] Barry Smyth and Paul Cotter. Personalized Adaptive Navigation for Mobile Portals. In *Proceedings of ECAI/PAIS'02*, Lyon, 2002.

[SGF02] Rüdiger Schollmeier, Ingo Gruber, and Michael Finkenzeller. Routing in Mobile Ad Hoc and Peer-to-Peer Networks. A Comparison. In *In Int.Workshop on Peer-to-Peer Computing. In Networking 2002*, 2002.

[Spe10] SpeedUp Consortium. SpeedUp Homepage. `www.speedup-project.de`, 2010. [Online; accessed 10-April-2010].

[TZA03] Lang Tong, Qing Zhao, and Srihari Adireddy. Sensor Networks with Mobile Agents. In *in Proc. 2003 Military Communications Intl Symp*, pages 688–693, 2003.

Adaptation Process for Ad hoc Routing Protocol

[1]Cholatip Yawut, [2]Béatrice Paillassa, [2]Riadh Dhaou

[1]Faculty of Information Technology, King Mongkut's University of Technology North Bangkok, Bangkok – THAILAND

[1,2]Univeristy of Toulouse, IRIT laboratory – ENSEEIHT, 2 rue Camichel 31071 Toulouse – FRANCE

{cyawut, Beatrice.Paillassa, Riadh.Dhaou}@enseeiht.fr

Abstract: Because of several constraints in ad hoc networks, an adaptive ad hoc routing protocol is increasingly required. In this paper, we propose a synopsis of an adaptation process for an adaptive ad hoc routing protocol. Next, we put into practice the analysis of the process of adaptation to mobility by realizing an adaptive routing protocol: CSR (Cluster Source Routing) which is an extension of a widely used ad hoc routing protocol: DSR (Dynamic source Routing). Mobility and density metrics are considered to CSR<->DSR mode switching, it moves from a flat architecture working in DSR to a virtual hierarchical architecture. With this mode switching, CSR can enhance the scalability of the DSR routing protocol.

1 Introduction

In an ad hoc Network [MM04], a node communicates either through single-hop transmission if the destination is in its transmission range, or by relying through intermediate nodes using a routing protocol. If one of the nodes on the ad hoc network has an internet connection, it is possible to share it with the other nodes on the network, like in the case of a traditional local network. The standard ad hoc routing protocols normalized by IETF (Internet Engineering Task Force) are OLSR (Optimized Link State Routing) [CJ03], TBRPF (Topology Dissemination based on Reverse-Path Forwarding routing protocol) [OTL04], AODV (Ad hoc On-demand Distance Vector) [PBD03] and DSR (Dynamic source Routing) [JMH07].

However, users require an ad hoc routing protocol which performs more adaptive than a standard ad hoc routing protocol because of several basic characteristics in an ad hoc network, for example: a limited battery capacity, limited and varied bandwidth, dynamic topology and mobility. Researches in adaptive ad hoc routing protocols have focused to adapt protocol behaviors according to these constraints.

To easily understand the mechanism of an adaptive ad hoc routing protocol, we propose a synopsis of adaptation process. With this synopsis, we put into practice the analysis of the process of adaptation to mobility by realizing an adaptive routing protocol: CSR (Cluster Source Routing). CSR [JP07] developed by the IRIT laboratory is an example of such an adaptive ad hoc routing protocol. CSR is a cluster-based extension of the DSR protocol, a popular deployed ad hoc routing protocol. CSR improves the scalability of DSR in high-density and low-mobility networks. CSR using a mode adaptation is regarded in the aspect of Cluster Head and Server selection. As CSR protocol defines a mode switching (it moves from a flat architecture working in DSR to a hierarchical architecture).

The rest of the paper is organized as follows. Section 2 describes a protocol adaptation view about related works, adaptation in the protocol stack. Section 3 illustrates CSR fundamental and how it adapts its behavior according to mobility and density metrics.

2 Protocol Adaptation View

2.1 Related Works

An adaptive ad hoc routing protocol uses metrics to adapt its behaviors. In [YBD09], we define the definition of metric; the metric is a measure indicating the state of a node, its neighborhoods, or the entire network, for example, energy level, transmission power, mobility etc... The measure may be a combination of several parameters such as a number of nodes, number of links, energy state...

Many researches use a mobility metric to adapt the system. Adaptive Routing Protocol for Manets (ARPM) [Se06] begins with using the proactive behavior and dynamically eliminates routing tables and switch to reactive behavior whenever the mobility degree exceeds a certain threshold. Adapting to Route Demand and Mobility (ARM) protocol [AS02] uses the rate of neighbor change as mobility metric. Fast-OLSR [BMA02] considers the number of neighbor changes as mobility metric. A node reduces its Hello-Interval when this metric reaches a predefined threshold. In [YPD07], we use link duration metric to improve MPR selection process.

In other ways, some protocols use a density metric to adapt their behavior. Density adaptive routing protocol (DAR) [Li08] utilizes the local network density to determine the packet forwarding zone; in dense areas, it narrows the forwarding range to reduce the total number of participants in flooding; in sparse area, it enlarges the forwarding scope to enclose enough nodes for packet relaying. LAKER [LM03], a LAR-based protocol, utilizes population density distribution and other knowledge for route guiding and passing around the void area.

2.2 Adaptation in the Protocol Stack

Our work focuses on a metric strategy to improve an adaptive routing protocol. First, it

is necessary to understand how adaptation process works and what it consists of. In this step, we propose a synopsis of an adaptation process (Figure 1) by considering three sets of element: network environment, which is perceptible through metric, behavior to adapt, i.e. the algorithms to be applied to depend on metric values, and performance which we are trying to optimize. Therefore, the elements of the adaptation process consist of :

A) Metric

The metric gives values that can be used to adapt protocol behaviors. Several metrics could be used by the algorithm of adaptation, such as a routing protocol adapts its operation to the network density and mobility [JP07]. An environmental metric can be used to select a route or to establish a routing structure in an adaptive routing protocol.

B) Types and Policy of Adaptive Algorithms

Adaptive algorithms can be classified in 2 types: Auto and Cross-layer adaptations. The parameters taken into account in metric calculation may come from a single layer called as auto-adaptation, or several layers called as cross-layer or multiple-layers [SM05] adaptation. In this paper, auto-adaptation is evaluated because of its simplicity.

The adaptive algorithms can be also classified in 2 policies: Parameters and Mode adaptations.

Parameter adaptation: a protocol sets its parameters according to the state of its environment. Examples of such adaptation are:

- A transport protocol computes its window according to the emission rate parameter of congestion;
- A routing protocol chooses neighbors, links, or route depending on the stability or on the delay;
- A routing protocol adjusts its broadcast timer (for proactive routing) based on the network mobility: if the network is high mobility, the broadcast timer is short duration while it is important in low mobility case [QK06].

Mode adaptation: protocol changes its behavior depending on its environmental condition. For example:

- A transport protocol stops to increase its retransmission timer if an important level of mobility is detected [CPJ07];
- A routing protocol works in a mode (e.g. reactive, non architecture) and the number of mobile nodes increases, it switched to the other mode (e.g. proactive, with cluster architecture) [JP07].

C) Performances

Last element is the performance which adaptation seeks to optimize. We investigate the conventional performance characteristics of fixed network: throughput, delay, which adds energy.

A synopsis of an adaptation process is illustrated in Figure 1:

- Initially (Figure 1-a), a given node receives information (a set of parameters) coming from a single layer or multiple-layers. This information can be acquired from local source; such as node itself, or global source; such as neighbor nodes or network.
- Next (Figure 1-b), the parameters are taken into account in metric calculation.
- An adaptation strategy uses the values of metric calculation (Figure 1-c), for example the strategy may consider the M1 and M2 Metrics at the same time or only M1 metric or M2 metric beyond M1 metric or etc…
- The results of adaptation strategy are (Figure 1-d) to adapt a protocol behavior.
- After this adaptation, network parameters are changed by a node's mobility, giving new information to the given node to repeat the adaptation process (Figure 1-e).

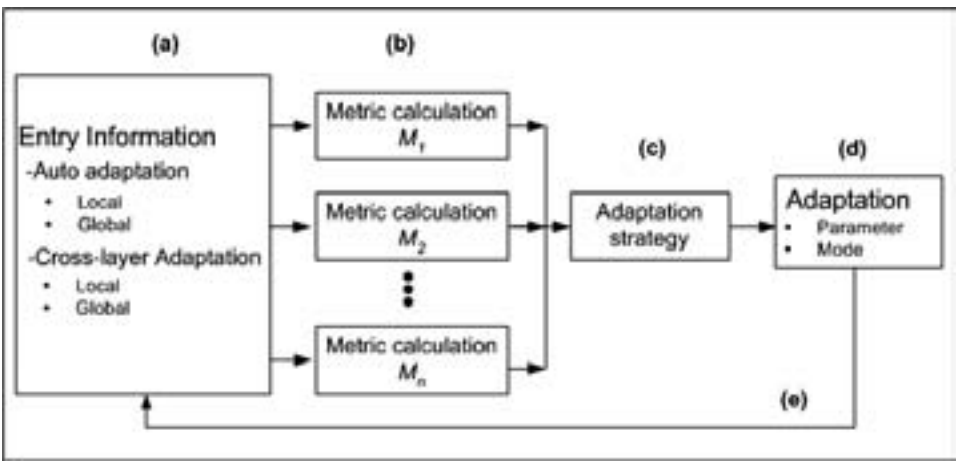


Figure 1: Synopsis of an adaptation process

3 Cluster Source Routing Protocol (CSR)

In this section, we put into practice the analysis of the process of adaptation to mobility by realizing an adaptive routing protocol: CSR (Cluster Source Routing) which is an extension of a widely used ad hoc routing protocol: DSR (Dynamic source Routing).

CSR is a source routing protocol and an architecture routing protocol. This protocol aims to increase the scalability of DSR with regard to network size and node mobility in an adaptive way. CSR is an auto-adaptation considering local information: a number of route errors as mobility metric and a number of neighbors as density metric. These metrics are obtained from node itself. CSR takes a strategy using a combination of these two metrics to change its mode and uses density metric to select a Cluster Head and Server.

Nodes can switch from DSR to CSR (DSR<->CSR) mode if the network stability and the local density are sufficient. DSR<->CSR mode is gainful on dense network configuration. The benefit of DSR-CSR especially grows with node density.

The CSR extension procedures are totally transparent and ensure full compatibility between native DSR and DSR<->CSR nodes. In fact, the DSR packet format is conserved. Native DSR and DSR<->CSR nodes can communicate since CSR integrates the DSR protocol. The CSR procedures are carried out through the DSR option mechanisms. Appropriate option codes are chosen to allow native DSR nodes to treat packets if necessary.

CSR [JP07] aims to increase the scalability of DSR with regard to network size and node mobility in an adaptive way. Nodes can switch from DSR to CSR (DSR<->CSR) mode if the network stability and the local density are sufficient. DSR<->CSR mode is gainful on dense network configuration. The benefit of DSR-CSR especially grows with node density.

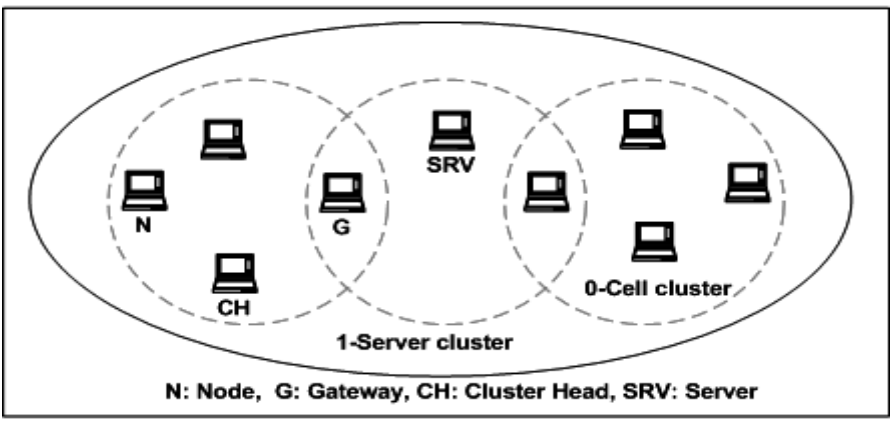


Figure 2: CSR model

The network is partitioned according to a 2-level hierarchical architecture (Figure 2). The lower level of cluster is the cell (0-Cell cluster). Each node within the cell is 1-hop away from the Cluster Head. Communication between 0-Cell clusters is completed through gateway nodes. The upper level of cluster (1-Server cluster) is formed by a set of cells. The associated cluster leader is named Server. Each node has four statuses:

- Undefined: the node has not yet obtained a valid status and is running the native DSR protocol.
- Node: a station which can use the CSR mode.
- Cluster Head: the cluster leader of the 0-Cell cluster.
- Server: the cluster leader of the 1-Server cluster.

Mobility and density metrics can be considered in an individual or combined way. In CSR, the mobility and the density metrics are used, by default, separately in order to perform the mode switching (Figure 3). The number of Route Errors is selected as the mobility metric and the number of neighbors in the route cache provides the density metric. Metrics are periodically computed. The values of the thresholds are as follows: M1 (low) = 2, M2 (high) = 4, D1 (low) = 2 and D2 (high) = 5.

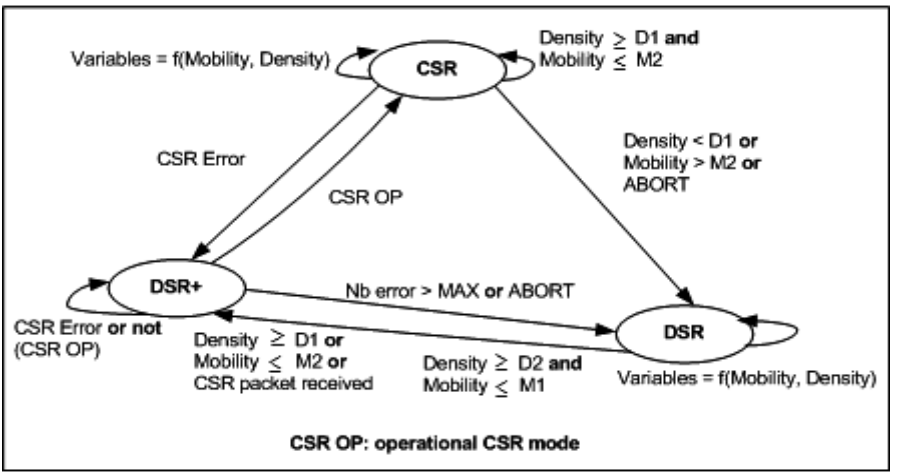


Figure 3: CSR states

A node switches to DSR mode if it experiences more CSR errors than the predefined MAX value. The MAX value is set to 3. CSR errors are caused by failures in setting up the architecture. On receiving an ABORT packet from the Server, a node also switches to DSR mode. Server sends an ABORT packet when it is about to give up its role.

Mobility and density metrics can be considered in an individual or combined way. In CSR default, the mobility and density metrics are separately used to perform the mode switching. The number of Route Errors is selected as the mobility metric and the number of neighbors in the route cache provides the density metric. Metrics are periodically computed. The computation timer value is set to 6s. The mode change is detailed below. Each node which runs DSR-CSR protocol could experience three states (Figure 5):

- DSR: The node uses DSR Route Discovery and DSR Route Maintenance. If network dynamics are favorable (high density and low mobility), the node enters DSR+ state. Two thresholds of mode switching are defined for both the Mobility metric (M1<M2) and the Density metric (D1 < D2):
    - Mobility > M2 or Density < D1: the node stays in DSR mode (high mobility and/or low density).
    - Mobility ≤ M2 and Density ≥ D1: the node changes from DSR to DSR+ mode if it receives a CSR packet (average mobility and density).
    - Mobility ≤ M1 and Density ≥ D2: the node switches from DSR to DSR+ mode (low mobility and high density).
- DSR+: The node uses DSR Route Discovery and Route Maintenance. However, clustering procedures are used to set up or to recover the CSR architecture. If clustering procedures succeed, the node enters CSR state. Else, it goes into DSR state. After its election, Server sets a timer and waits for Cluster Heads Registration to obtain its routing information. On timer expiration, Server is operational. When Server sends back a Registration Reply, it indicates to Cluster Head whether the CSR architecture is active or not. During DSR+ state, Server could stop CSR mode

by broadcasting an ABORT packet in the network (for example, few registered Cluster Heads indicating a low global density) and pass into DSR mode.

After its election, each Cluster Head registers to the Server. If the Registration procedure fails, Cluster Head will initiate a Server election. On MAX election failures, Cluster Head passes into DSR state. When Cluster Head receives a Registration Reply, it checks whether the CSR mode is operational or not. If so, it enters the CSR state and signals operational CSR mode to its cluster members through each Cell Maintenance packet. Else, it sets a timer and only goes into CSR state on its expiration. On receiving an ABORT, each Cluster Head switches to DSR state.

On receiving a Cell maintenance packet, the node checks whether the CSR is operational (use of CSR Route Discovery) or not (use of DSR Route Discovery). Each Node switches to DSR state if it receives an ABORT.

- CSR: Node uses CSR Route Discovery and Route Maintenance. CSR mode is operational and Cluster maintenance procedures are applied. Server broadcasts an ABORT message when it is about to give up its role because of network dynamics and switches to DSR state. If Server receives a packet from a higher criterion Server, it becomes Cluster Head and enters the DSR+ state. If Server is unreachable, the Cluster Head locally broadcasts a Cell Maintenance packet indicating to its cluster members that CSR architecture is not operational. Then, it applies the Registration procedure and switches to DSR+ mode. On receiving an ABORT, Cluster Heads and Nodes switch to DSR state.

3.1 CSR Procedures

The required procedures to operate the CSR extension are divided into two categories: Routing procedures serve to the discovery and maintenance of routes and Clustering procedures serve to the establishment and maintenance of virtual architecture.

3.1.1 Routing Procedures

Instead of diffusing a route request in the entire network, a route request in CSR mode is directly managed to the Server (transparently to the nodes, a Cluster Head manages this request). When a node has to send a packet, its Route is firstly searched for a route reaching to the destination. If such a route is not discovered, a Route Discovery (Figure 4) is launched using the non-propagating Route Request of DSR (Time To Live=1) by locally broadcasting a Route Request in its cell. If such a route exists in the cache of the Cluster Head of the cell, it is replied to the source node. If no route is known, the Route Request is transmitted to the Server (the path is known based on the periodic Topology Discovery of the Server). The Server verifies whether the destination survives. If so, the route between the source and the destination is created using its topology knowledge and a Route Reply is sends it back to the source node. Otherwise, Server asks all the Cluster Heads to find the required destination. Each Cluster Head investigates the destination in its cell using a non-propagating Route Request. A positive reply is send back to the Server, if a Cluster Head localizes the destination,

The Server modernizes its topology information and replies to the source node. If it cannot indicate the destination, it sends back a Route Error packet (informing the unreachable destination) to the source. On receiving this Route Error from the Server, the node reinitiates a DSR Route Discovery.

In case of successive failures (maintenance clustering phase) route request is broadcasts through the network (DSR or DSR+ mode). CSR routing is completely transparent to the nodes: a DSR node without this extension can operate in the network.

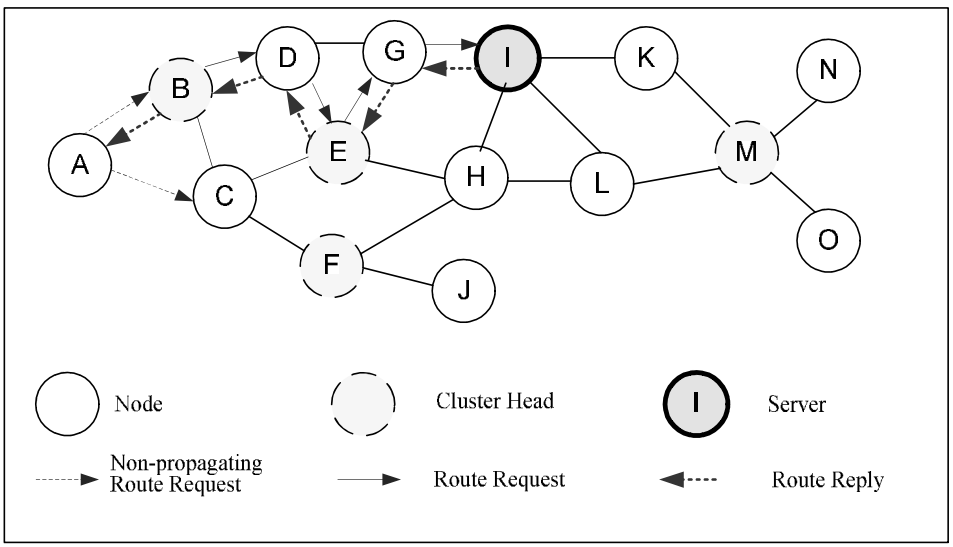


Figure 4: CSR Route Discovery

3.1.2 Clustering Procedures

In CSR, each node must obtain a status (Node or Cluster Head) to proceed with the CSR mode set up. Each Node is managed by a Cluster Head which forwards its Route Requests to the Server.

The set up of cells is based on the highest-connectivity degree algorithm; we can say that it is based on a density (D) metric. When a node enters CSR routing mode, it initiates the GetStatus procedure. Nodes which do not belong to a cluster are called uncovered nodes. To obtain a status, each uncovered node locally broadcasts a Route Request which contains its election criterion and indicates its undefined status (a Status packet). A specific option code is used to prevent neighboring native DSR nodes from processing the packet [JMH07]. Once the Status packet is broadcasted, the node waits for a GetStatus period. If a packet from a Cluster Head is received before GetStatus expires, the node initializes its status to Node. Else, on receiving a Status packet, the node checks its routing mode: CSR mode*:* it compares the packet election criterion with its own, DSR mode: it checks its adaptation criterion. If its criterion is suitable enough to switch to CSR mode, the node starts the GetStatus procedure and native DSR: it just discards the packet (unknown option code).

If the node has the local highest criterion (the lowest ID is preferred in case of tie), it sets up its status to Cluster Head and broadcasts a Cell Maintenance packet indicating its status. Thus, its neighbors take the Node status and stop their GetStatus procedure. If a node does not have the local highest criterion and does not hear any Cluster Head, it becomes itself Cluster Head at the end of the procedure.

Periodically, each Cluster Head locally broadcasts a Cell Maintenance packet to maintain its cell. If Node has not heard any Cluster Head during a Status period, it applies again the GetStatus procedure. The selected Cluster Head revocation algorithm is LCC (Least Cluster Change) in order to control the number of Cluster Heads [Ch97]: when two Cluster Heads are within transmission range, the lower-criterion one gives up its role and becomes Node. Thus, Cluster Heads are at least 2-hop away.

3.2 Server Selection Algorithm

The Server is elected among Cluster Heads and selected on the election criterion. At the beginning of the procedure, Cluster Heads initialize their candidate criterion variable with their election criterion value and the candidate address variable with their own address. Each Cluster Head which initiates an election broadcasts an Election packet in its 3-hop neighborhood. This Election packet is a DSR Route Request which contains the election criterion of the Cluster Head.

4 Conclusion

In this paper, the synopsis of an adaptation process is proposed to easily comprehend how an adaptive ad hoc routing protocol works. With this synopsis, we put into practice the analysis of the adaptation process to mobility by realizing in an adaptive ad hoc routing protocol: CSR. Cluster Source Routing adapts the DSR routing protocol to various conditions of mobility and density in ad hoc networks. The strategy of CSR <-> DSR mode switching is a combination of the number of Route Errors and the number of neighbor nodes as mobility and density metric respectively. These metrics are considered because it is simple; readily available the studied protocol, but efficient; signaling minimizing and no extra modification.

Improving the scalability of the DSR routing protocol can be achieved by realizing on a 2-level hierarchical scheme (0-Cell and 1-Server clusters) in CSR. Route Requests are transmitted to the 1-Server leader, considered as an upper level of Route Cache, to prevent network flooding. Then, data are transferred based on native DSR. Clustering procedures are specified to set up and maintain the CSR architecture. Each station separately adapts its routing mode (DSR or CSR) based on the mobility and density metrics. Computation methods of adaptation criteria are illustrated to authorize the change between modes and to adapt the routing variables.

Bibliography

[AS02] Anh, S.: Shankar, A.U.: Adapting to route-demand and mobility in ad hoc network routing. In Computer Networks: The International Journal of Computer and Telecommunications Networking, Volume 38, Issue 6, ISSN:1389-1286, April 2002; p. 745 – 764.

[BMA02] Benzaid, M.: Minet, P.: Alagha, K.: Integrating Fast Mobility in the OLSR Routing Protocol: In Fourth IEEE Conference in Mobile and Wireless Communications Networks, 2002.

[Ch97] Chiang, C. et al.: Routing in Clustered Multihop, Mobile Wireless Network with Fading Channel. In Proc. of IEEE Singapore International Conference On Networks (SICON'97), Singapore, 1997; p. 197-211.

[CJ03] Clausen, T.: Jacquet, P.: Optimized link state routing protocol (olsr), RFC3626, October 2003.

[CPJ07] Charoenpanyasak, S.: Paillassa, B.: Jaddi, F.: Experimental study on tcp enhancement interest in ad hoc networks: International Conference on Wireless and Mobile Communications (ICWMC 2007), Guadeloupe, 04-09, IEEE, Mars 2007.

[JMH07] Johnson, D.: Maltz, D.: Hu, Y.: The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks for IPv4, RFC 4728, February 2007.

[JP07] Jaddi, F.: Paillassa, B.: An Adaptive Hierarchical Extension of the DSR: the Cluster Source Routing. In: Journal of Universal Computer Science, John Wiley and Sons, Special Issue of the Journal of Universal Computer Science, Vol. 13 N. 1, 2007; p. 32-55.

[Li08] Li, Z. et al.: A Density Adaptive Routing Protocol for Large-Scale Ad Hoc Networks: In Wireless Communications and Networking Conference, 2008 (WCNC 2008); p. 2597–2602.

[LM03] Li, J.: Mohapatra, P.: Laker: Location aided knowledge extraction routing for mobile ad hoc networks: In Wireless Communications and Networking, (WCNC), Vol. 2, 2003; p. 1180–1184.

[MM04] Murthy C. S. R.; Manoj, B.: Ad Hoc Wireless Networks : Architectures and Protocols (Prentice Hall Communications Engineering and Emerging Technologies Series): Prentice Hall PTR, Special edition, June 3, 2004.

[OTL04] Ogier, R.: Templin, F.: Lewis, M.: Topology Dissemination based on Reverse-Path Forwarding routing protocol (TBRPF): Network Working Group, RFC3684. February 2004.

[PBD03] Perkins, C.: Belding-Royer, E.: Das, S.: Ad hoc on-demand distance vector (aodv) routing, RFC 3561, July 2003.

[QK06] Qin L.: Kunz, T.: Mobility metrics to enable adaprtive roiting in manet: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, 2006 (WiMob'2006), 2006.

[Se06] Seba, H.L.: 2006 ARPM: An Adaptive Routing Protocol for MANETs. In ACS/IEEE International Conference on Pervasive Services, Volume 26-29 June 2006; p. 295 – 298.

[SM05] Srivastava, V.: Motani, M.:_Cross-layer design: a survey and the road ahead: IEEE Communications Magazine, vol. 43, no. 12, December 2005; p. 112-119.

[YBD09] Yawut, C. : Paillassa, B.: Dhaou, R.: Adaptation à la mobilité dans les réseau ad hoc : PhD thesis, INPT-ENSEEIHT, Sep 2009.

[YPD07] Yawut, C.: Paillassa, B.: Dhaou, R.: Mobility versus Density metric for OLSR enhancement: Asian Internet Engineering Conference (AINTEC 2007), Phuket, Thailand, Vol. 4866, Springer, LNCS, November 2007.

Optimizing Mobile Networks Connectivity and Routing Using Percolation Theory and Epidemic Algorithms

Soufian Ben Amor, Marc Bui and Ivan Lavallée

Laboratoire d'Informatique et des Systèmes Complexes (LaISC)
41 rue G. Lussac, F-75005, Paris, France.
Email : {sofiane.benamor, marc.bui, ivan.lavallee}@ephe.sorbonne.fr

Abstract: Mobile Ad-hoc NETworks (MANETs) are complex systems presenting a phase transition phenomenon : an abrupt change in the behavior of the network around a critical value of a certain key parameter. The reliability and connectivity of MANETs depend on their size and on the efficiency of the routing protocol. In this paper we propose a new approach in MANETs modeling, combining percolation theory and epidemic algorithms. Using percolation theory we show the existence of a connectivity threshold (in a square lattice) needed to guarantee the communications in the network, in particular when the number of direct links are limited. Epidemic algorithms are used to provide a good propagation of information in the network wile minimizing the resources cost (energy, number of messages...).

1 Introduction

A complex system is a network composed of mutually interacting elements, where the global behaviour of the system can not be deduced from the sum of its components and their properties. One of the most important particularities of complex systems concerns the global *phase transition* phenomenon, which occurs around a critical value of a key parameter leading to the appearance of a new property in the system. From the phase transition point of view MANETs can be considered as a complex system.

A mobile ad-hoc network (MANET) is a network composed of a set of nodes communicating over paths composed of one or a sequence of wireless links. A wireless link is established when tow nodes are within a certain distance corresponding to the transmission radius (figure 1). Nodes mobility implies unpredictable wireless links formation and removal explaining the dynamic topology of the network. Since we are dealing with a propagation of information over a random structure, percolation theory (see section IV) offers an adapted theoretical framework to study the behavior of such a system.

The mobility allowed by MANETs and the facility of their deployment (they do not rely on a preexisting infrastructure to communicate) permitted important and various applications such as mobile detection systems and military communications. But, nowadays, the popularity of MANETs is due to the widespread availability of wireless devices such as cell phones, PDAs and WiFi/Bluetooth enabled laptops. [SCS03]

Recent studies [GK98], [KWB01] and [KWBP02] showed the existence of phase transition phenomena in MANETs. In [KWB01] tow aspects are presented :

- there is a critical value of the energy (*i.e.* transmission radius) spent by each node to guarantee the connectivity of the network with high probability.
- It is possible to conceive an efficient routing protocol using probabilistic epidemic algorithms. These algorithms are able to diffuse information on the whole network with high probability when the probability of retransmission of a message at each node is higher than a critical value p_c.

The new tendency in the study of phase transition phenomena observed in MANETs, and more generally in large-scale networks, concerns the applicability of percolation theory [KWB01], [SCS03] and the usefulness of the probabilistic epidemic algorithms [EGKM04], [KMG01]. From our point of view percolation and diffusion theory may be complementary in the MANET context. In fact, it is possible to solve, using both theories, tow dependent aspects : a good diffusion of information in the network (needed for routing, broadcast and communication) and its connectivity (needed to reach each node). The use of epidemic algorithms instead of the classic flooding approach permits a limitation in the redundant messages and the waste of limited resources (energy, bandwidth,...). We can also avoid the problem known as *Broadcast storm problem* corresponding to a situation where too much redundant messages are generated at the same time impeding communications.[SCS03]

The rest of the paper is organized as follows : in section II, the basic phase transition phenomena observed in MANETs are presented. The section III concerns random graph theory, traditionally used to model the behavior of complex networks, and presents the mathematical formulation of the phase transition property. Percolation theory and diffusion theory are subject of section IV. We present, in section V, our mixed modeling approach using both percolation theory and diffusion theory to guarantee the connectivity and the broadcast in the network while reducing the resources cost. We conclude, in section VI, with a discussion of the results and future work.

2 Phase transition in MANET

There are two main phase transition phenomena in MANETs with uniform fixed radius : the first one concerns the conductivity in the network and the second one is related to the efficiency of epidemic algorithms in broadcasting information in the whole system.

2.1 The network conductivity

The communications in a network need not only a good routing protocol, but also the existence of an open path (sequence of wireless links) between each pair of nodes. The network conductivity expresses its ability to propagate a message between two nodes.

MANET's topology is organized according to the relative distance between the nodes. Tow nodes establish a direct link, if they are within a certain distance corresponding to the transmission radius. The dynamic topology of MANETs, due to the mobility of the nodes, requires a new algorithmic approach to ensure a good conductivity of the network and to guarantee the functioning of communication protocols.

Recent works [GK98] then [KWB01] and [KWBP02] have shown the existence of a critical level of transmission power provided by each node to ensure with high probability the connectivity of the network. Percolation theory is a good theoretical framework to study this phase transition phenomenon because we can determine, using simulations the value of the threshold of connectivity.

2.2 Message broadcasting

The mobility of the nodes poses also an other problem concerning the routing protocol[1] used in the network [BBBS03]. In fact, because of the changing topology of the network, broadcasting is a very important communication primitive for routing. The classical broadcasting method uses flooding : an algorithm for distributing messages to every part of a connected network[2]. But this approach is not optimal and generates a high number of redundant messages, wasting limited resources such as bandwidth and energy. There is an other approach [EGKM04], [SCS03] consisting in forwarding messages with probability p. In these diffusion models there is a phase transition around a critical value p_c [KWBP02]. For $p < p_c$ the probability that the message reaches each node of the network is very low and for $p > p_c$ this probability is very high. The value of p_c depends on the topology of the graph modeling the network.

3 Random graphs

Random Graph Theory was introduced in 1959 by Paul Erdös and Alfred Rényi. A random graph G, is a graph generated using a stochastic process called *random graph model*. In a random graph model, there is generally a key parameter permitting to vary the average density of the graph. Here are the basic random graph models :

- Fixed edge number model (the original Erdös-Rényi model): $G = G(n, e)$, given a number of edges e and a number of vertices n, choose G uniformly at random among all possible graphs (n, e). This model is not adapted to the MANET context because of the variability of the number of links (*i.e.* edges).
- The Bernoulli model (known as the *binomial model*): $G = G(n, p)$, given a number

[1] Routing is a mean of discovering paths in computer networks along which information can be sent. Routing directs forwarding, the passing of logically addressed packets from their source toward their ultimate destination through intermediary nodes, called routers.

[2] The name derives from the concept of inundation by a flood.

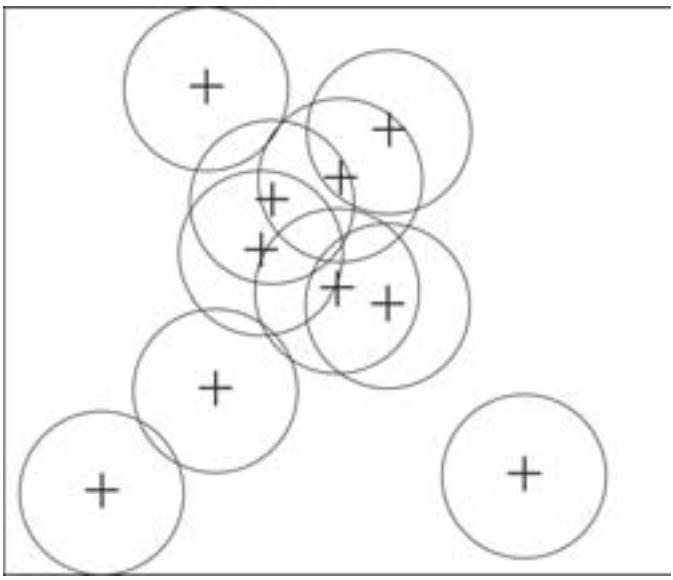


Figure 1: MANET's structure.

of vertices n , and a probability p, generate the graph G such that, for each pair of vertices, there is an edge connecting them with probability p. This model is the most studied one, but it is not really adapted for MANETs modeling because it does not take into account the relative distance between the nodes. In fact, in MANETs even if the configuration of the network changes randomly, its topology is determined by the transmission radius constraint. Contrarily, in the Bernoulli model, each couple of nodes is susceptible to have an edge even if the distance separating them is important.

- The random geometric graph : $G = G(n, r)$, given a number of vertices n randomly placed, according to some probability distribution in the Euclidian plane, generate a graph such that there is an edge between each X_i and X_j if and only if :

$$|X_i - X_j| < r$$

where $i \neq j$ and $i, j \in \{1, 2, ..., (n-1), n\}$ and r a fixed parameter. This model is similar to the MANETs link construction rules, and for this reason it is appropriate to describe this kind of networks.

- Dynamic model : $G = G(n, t)$, given a number of vertices n, the graph is constructed by adding uniformly at random an edge at each time-step. This model is useful to describe the evolution and functioning of static networks, but it does not allow a realistic modeling of MANET.

Formally, $G(n,p)$ is a probability space over graphs. Given any graph theoretic property A there will be a probability that $G(n,p)$ satisfies A, which we write $P[G(n,p) \models A]$. When A is monotone $P[G(n,p) \models A]$ is a monotone function of p. For example, let A be the event "G is triangle free". Let X be the number of triangles contained in $G(n,p)$. Linearity of expectation gives

$$E[X] = \binom{n}{3} p^3$$

This suggests the parametrization $p = \frac{c}{n}$. Then

$$\lim_{n\to\infty} E[X] = \lim_{n\to\infty} \binom{n}{3} p^3 = \frac{c^3}{6}$$

The distribution of X is asymptotically Poisson and

$$\lim_{n\to\infty} P[G(n,p) \models A] = \lim_{n\to\infty} P[X = 0] = e^{\frac{-c^3}{6}}$$

We can see that

$$\lim_{c\to 0} e^{\frac{-c^3}{6}} = 1$$

and

$$\lim_{c\to\infty} e^{\frac{-c^3}{6}} = 0$$

The first triangles always appear at $p = \Theta(\frac{1}{n})$. This means that the probability that the graph $G(n,p)$ contains a triangle approaches 1 as n approaches infinity. It was a central observation of Erdös and Renyi that many natural graph theoretic properties become true in a very narrow range of p. They made the following key definition:
$r(n)$ is called a *threshold function* for a graph theoretic property A if

- when $p \ll r(n), \lim_{n\to\infty} P[G(n,p) \models A] = 0$
- when $p \gg r(n), \lim_{n\to\infty} P[G(n,p) \models A] = 1$

In the case of our example $\frac{1}{n}$ is a threshold function for $A =$"G is triangle free".

The transition observed around the threshold function is similar to the characteristic *zero-one law* in the first order properties in random graphs. First order properties are those that can be described using a language based on the usual logic Boolean operators ($\wedge, \vee, \neg$), the equality ($=$) and adjacency ($\sim$) relations and the universal quantifiers[3] ($\exists, \forall$). For example, the property "there is a triangle" can be written ($\exists_u \exists_v \exists_w s.t.(u) \wedge (v \sim w) \wedge (u \sim w)$).

Theorem 3.1 *For all first order graph property A:*

$$\lim_{n\to\infty} P[G(n,p) \models A] = 0 \text{ or } 1.$$

[3]Quantification only over vertices

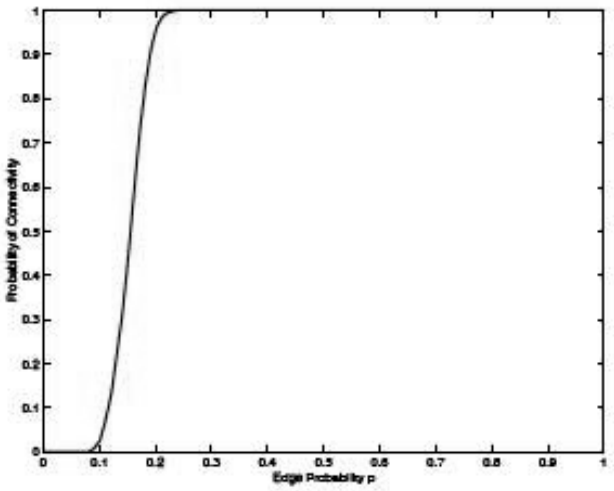


Figure 2: Phase transition in the probability of connectivity in the Bernoulli model (with n=15). Source: [KWB01].

The transition is obtained when p reaches a critical value p_c called *transition threshold*. Although the property "the graph is connected" is not a first order property, it exhibits a zero-one transition in the Bernoulli model (figure 2). In [KWB01] is given the following conjecture:

> *properties which satisfy a zero-one law for Bernoulli Random Graphs also satisfy a zero-one law for the Fixed radius model.*

The existence of a phase transition phenomenon in the fixed radius geometric random graphs offers an opportunity to study the behavior of MANET's connectivity and message delivery, respectively, as a percolation and diffusion processes.

4 Percolation and epidemic diffusion

As complex systems presents generally a phase transition phenomenon according to a certain property and its global behavior can not be deduced from the behavior of its components, it is important to study them using the appropriate theoretical tools. Among theories using a *holistic* (or *systemic*) approach to explain the passage from the individual to the collective, from the micro to the macro, Percolation Theory is the most adapted in the MANET's context : it studies the deterministic propagation of a fluid (or an information) on a random medium (or structure).

4.1 Definition

The percolation model was first introduced by Simon Broadbent et John M. Hammersley in 1957, using the example of a porous stone immersed in a bucket of water. This fundamental question was asked: What is the probability that the center of the stone is wetted? Equivalently, what is the probability that an infinite size percolation cluster of pores exists [Gr99]. Of course this depends on the porosity of the stone (*i.e.* the density of pores)

4.2 The general description of a percolation model

The physical problem is mathematically modeled as a network of elements (or vertices) where the connections (or edges) between each two neighbors may be open (allowing the liquid to pass through) with probability p, or closed with probability $(1-p)$. For a given p, what is the probability that an open path exists from the top to the bottom? Mostly we are interested in the behavior for large n. As is quite typical, it is actually easier to examine infinite networks than just large ones. In this case the corresponding question is : does there exist an infinite open cluster ? That is, is there a path of connected points of infinite length "through" the network. In this case we may use Kolmogorov's zero-one law to see that, for any given p, the probability that an infinite cluster exists is either *zero* or *one*. Since this probability is increasing, there must be a critical probability p_c such that (figure 3) :

$$P(p) \begin{cases} = 0 & si \quad p < p_c \\ = 1 & if \quad p > p_c \end{cases}$$

Where P is the percolation probability which indicates the probability of appearance of the giant cluster in the system.

A model where we open and close vertices rather than edges, is called *site percolation* (figure 4 a) while the model described above is more properly called *bond percolation* (figure 4 b). The model where the uncertainty concerns both sites and bonds is called *mixed percolation* (figure 4 c).

Even if this mathematical model was initially used to describe critical phenomena in statistical physics, its polyvalence and efficiency to characterize non-linear phenomena, led the scientific community to apply this theory to model the behavior of biological systems, social networks and economical systems. Recently, in [SCS03] the usefulness of Percolation Theory to ensure the broadcast on MANETs is exposed, and [EGKM04] recommend the use of probabilistic epidemic algorithms to ensure the efficiency of routing on large networks. From our point of view, it is important to take into account both approaches in the MANET's context. It is important to be interested in epidemic diffusion because it exhibits a phase transition phenomenon on one hand, and allows to tackle the problem from a complementary point of view to that of percolation on the other hand. The latter, is generally presented as being the dual of diffusion theory where the stochastic mechanism concerns the propagation process.

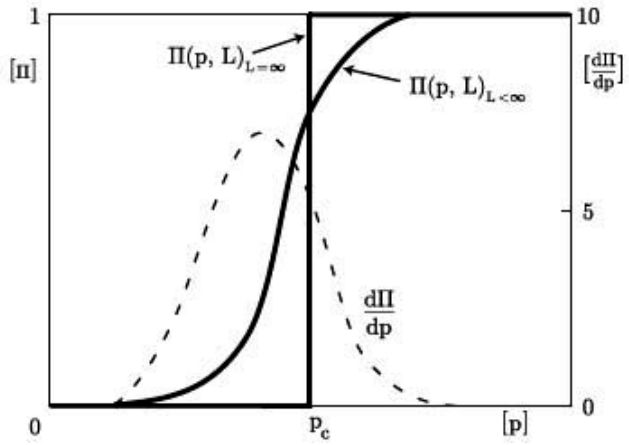


Figure 3: Phase transition around p_c. (Source : Stauffer and Aharony (1992), In [Pa01] p.50)

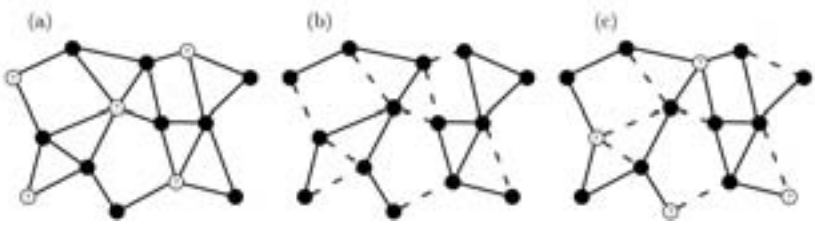


Figure 4: Basic percolation models

4.3 Epidemic diffusion

Epidemic diffusion is a particular case of diffusion theory. It was initiated by Francis Galton in the XIX^{th} century in order to study mathematically the chances of survival of the names of noble families. His model known as *Galton-Watson process* or *Branching process* [AN72], considers a population of males x_r belonging to the generation r where each individual gives birth, independently from the others, to k individuals with a probability p_k that will participate in the generation $r + 1$. Starting with generation 1 with only one individual the probability of extinction is

$$p_{ext} = \sum_{k0} p_{ext}^{k} p_k$$

The probability of extinction depends on the average number of descendants $f = \sum_{k0} k p_k$ and thus on the weights of probability p_k. By varying the parameter f a phase transition appears around a critical value f_c : the probability of extinction is very weak ($P \rightarrow 0$) for $f > f_c$ and very high ($P \rightarrow 1$) for $f f_c$.

Based on the epidemic diffusion paradigm, different algorithms known as epidemic algorithms were developed to guarantee a good propagation of information in particular in the context of distributed applications but also to improve routing protocols in large networks. These algorithms are known as *pro-active algorithms* in the sense that they disseminate information through the network to avoid a potential failure of certain links or certain nodes [EGKM04]. However, the majority of the algorithms used currently are of *réactifs* type : they react to a failure by sending a second time the lost information (or message). Epidemic algorithms are based on the following properties :

- each node of the network is potentially implied in the dissemination process.
- each node retransmits the received message in a probabilistic way to a subset of nodes of the network.
- each algorithm is characterized by a set of key parameters which differ according to the probabilistic diffusion rules

The general epidemic diffusion model is characterized by a stochastic retransmission process of the messages received by each node. The nodes have a capacity of reception b, a number f indicating the maximum number of randomly selected nodes to retransmit the message to them and a parameter t indicating the number of repetition of the same procedure by the same node. The differences between the models of epidemic diffusion lie in the values of these three parameters b, f and t.

5 A mixed model to optimize MANETs connectivity and routing

The communications in MANETs require a general diffusion of certain messages which requires, itself, a total connectivitity of the network. There are thus two dependent prob-

lems, but of different nature, to solve : the connexity (of topological nature) and diffusion (of algorithmic nature). In order to solve at the same time the two aspects of the problem, we propose a mixed model made up of two complementary sub-models. A percolation based modeling of MANETs can guarantee the connectivity of the network with very strong probability while minimizing the resources necessary (energy of transmission). Then, on the random structure thus created, is applied a stochastic epidemic diffusion algorithm allowing to ensure a total diffusion of a message through the network, while minimizing the total number of messages to be retransmitted to disseminate information. For the structural representation of MANETs we used a correspondence between a random geometrical graph and site percolation. Because models are mathematical representations for which the relevance must be checked, we specify in this document the basic assumptions and the different components of the model.

5.1 Hypotheses of the model

We consider a mobile network composed of nodes with a fixed and uniform transmission radius. We suppose that the speed of nodes is lower than that of the transmission of messages. It is also supposed that the transmission radius of nodes is sufficiently small (compared to the size of the network) in order to allow a modeling of node's mobility by discrete steps. Lastly, because our model is two-dimensional, we suppose that two nodes cannot occupy the same co-ordinates (*i.e.* site) at the same time in the Euclidean plan and that each node cannot be connected directly to more than eight neighbors (Moore neighborhood).

5.2 MANETs connectivity using percolation theory

Figures 5 and 6 represent an instantaneous configuration of a MANET with a fixed and uniform transmission radius and the corresponding connectivity. In the beginning we are interested in guaranteeing the connexity of the graph modelling the network. Indeed, we do not consider the conductivity of the network (*i.e* its capacity to propagate a message between two nodes), we rather seek a threshold guaranteeing the connexity of the graph allowing to reach each node of the network. Therefore, we cannot use the traditional model of percolation considering this theory is closer to the concept of conductivity than of connectivity (or connexity), because it only indicates the existence of an infinite cluster allowing to join the edges of the system, but does not guarantee the existence of a single cluster on the network (there can be small finite size clusters when percolation occurs). We thus propose a site percolation model which we called a *percolation-connectivity* model. At the theoretical level, even if we have a theorem due to P. Gupta and P.R. Kumar [GK98] concerning the connectivity of the mobile networks with fixed transmission radius this formula is not adapted when we have a restriction concerning the number of direct neighbors (for example in the case of entities communicating using optical communications

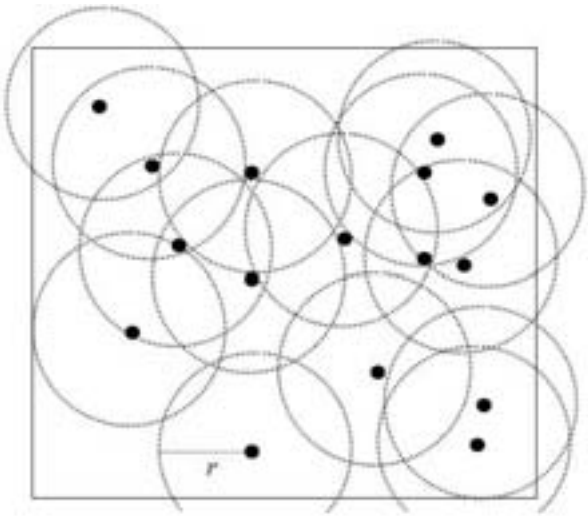


Figure 5: A random configuration of a MANET with a fixed transmission radius.

in a square lattice). In fact, the determined critical value $r_c = \sqrt{\frac{\log n}{n}}$ guaranteeing the connectivity of the network implies that the mean number of direct connections per node around the threshold is $\Theta(\log n)$.

By establishing a correspondence between a geometrical random graph and a site percolation model, we can expect to observe a phase transition concerning the connectivity of the simulated mobile network. This correspondence is built according to following elements :

- To model the evolution of a MANET configuration over time, we use successive generations of a random geometrical graph in a square lattice defined in Z^2 (the occupied sites will represent the nodes of the network). let N_1 be a random variable representing the number of nodes. For a given realization the N_1 nodes are distributed randomly (according to a uniform distribution) in a square surface $c \times c$. Two points x and y are directly connected if and only if $d(x, y) < r$ where r is the transmission radius.

- From the percolation point of view, we are interested in determining the critical density necessary to ensure the connexity of the network with a high probability. We consider only the case of site percolation meaning that when nodes are in the neighborhood of each other, the established communication link is functional with a probability $p_l = 1$. We consider a square network in which if a site is occupied, the transmission radius covers its closer neighbors. We choose here a Moore neighborhood. Thus, for a geometrical random graph $G(N_1, r)$ the transmission radius covers all the sites in the neighborhood of the active site (figure 7). Supposing a large network, the configurations of the n sites (total number of sites of the square

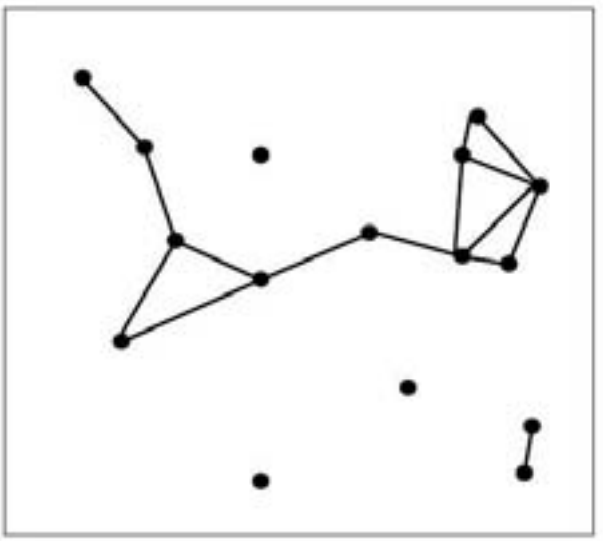


Figure 6: The resulting connectivity of the configuration shown in figure 5.

lattice) are given by the state vector e where $e_i = 1$ if the site is occupied by a mobile node and $e_i = 0$ if not. The number of occupied sites by nodes N_1 is a random variable $N_1(p, n)$ and the concentration corresponding to the total rate of active sites is also a random variable $c(p, n) = N_1(p, n)/n = \sum_i e_i/n$. Then :

$$\lim_{n\to\infty} c(p, n) = \bar{e} = p$$

where p is the probability of activity of a site. The probability of activity of a site thus merges with the density of active sites in the network.

Conjecture :

In the percolation-connectivity model that we propose, there exists a threshold of the density of active sites at which the graph is connected with a very strong probability.

$$\exists \quad p_c \quad | \quad \forall (S_i, S_j), i \neq j \begin{cases} P[S_i \sim S_j] & = 0 \quad if \quad pp_c \\ P[S_i \sim S_j] & = 1 \quad if \quad pp_c \end{cases}$$

where $\sim$ expresses the existence of a relation between two sites of the network. Two sites are related to each other if and only if, they belong to the same cluster.

We can then determine the threshold of connectivity by estimating the critical density using simulation. A preliminary simulations carried out using NetLogo confirm the existence of a threshold of connectivity, related to the density of the network. This value ranges between 0,58 and 0, 61 (figure 8 and 9).

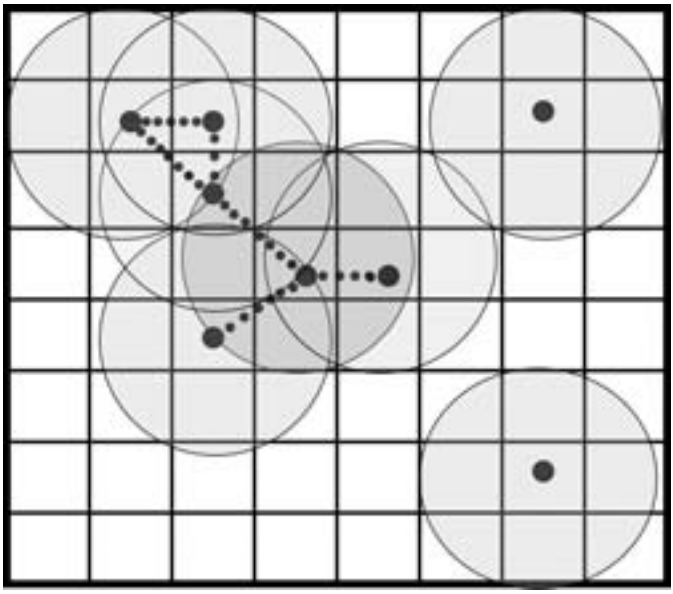


Figure 7: Correspondence between a discrete geometric random graph and a site percolation model.

Figure 8: The network is not totally connected for $p = 0.58$.

Figure 9: The network is not totally connected for $p = 0.61$.

5.3 Epidemic diffusion over the network

We propose in the following an algorithm of epidemic diffusion in which we incorporate rules aiming at reducing the number of redundant messages on the network. Our model simulates not only the connectivity of the network according to the density of the nodes but it simulates simultaneously the epidemic diffusion on the network. This allows to see the evolution of the quality of the diffusion according to the density (i.e.conductivity) of the network. The objective is to determine a threshold which allows to optimize at the same time conductivity and diffusivity (*i.e.* capacity to disseminate information) on the network.

To study the diffusion in a random network produced by a model of percolation, P. - G De Gennes introduced the method of "the ant in a labyrinth". It is a method of analysis of diffusion on a structure given through a random walk, based on a relation established by Einstein between conductivity and diffusion in an identical network [Pa01]. The original model consists in parachuting an ant (in our situation, it is the initialization of a message) on an active site, then at each time step, the ant evolves/moves randomly through contiguous active sites. The clusters of active sites are interpreted like possible paths. For a critical rate (*i.e.* probability) of sites activity, the ant entirely crosses the network. This value is generally given with the method of Monte Carlo. There are several alternatives to this model and the difference lies in the rules of displacement. Thus, an ant can be, for example, equipped with a mnemonic intelligence allowing it to leave a trace on its passage. In our problems, we want a quick and global diffusion over the network. With this intention, we associate the stochastic diffusion to the epidemic model. We want to be at the same time above the threshold of percolation to guarantee the conductivity of the network

and at the needed threshold of diffusion in order to limit the costs in term of resources (a number of messages, band-width...).

Stochastic and epidemic diffusion algorithm

Let m be a message to disseminate over the network.

- $\forall i \in [0, n-1]$
- choose uniformly at random an active site s_0.
- initiate a message m at s_0
- **if** s_i receives the message for the first time
  then it forwards it to his neighbors (excepting the one from which he received the message) with a probability p.
- **if** s_i have already received the message
  then it stops forwarding the message(to avoid unnecessary messages)

The complexity of the algorithm in number of messages is $O(n)$. In fact, in the worst case the node where the message is initiated have just only one and new neighbor at each time step and all these nodes will not propagate the message. In that case the initial node will send n messages.

6 Conclusion

Our proposed approach is a first attempt to model and simulate complex systems, for which analytical solution is very difficult to establish, by coupling dependent parameters. At the level of possible applications, our modeling allows to study, for example, the problem known as *walkers problem* where a set of mobile agents (robots, mobile detection systems,...) are moving on a rectangular grid and where the communications between the agents are established using wireless links (radio waves, optical systems,...). More generally, this model is adapted to simulate certain systems where the concepts of neighborhood and distance are important and where the geometrical structure of the support of the network is rectangular. According to the results of the simulation carried out with this model, it is possible to improve the model, in particular concerning the dimension of the network (a three dimensional modeling of the network is more realistic) and the topology of the graph modeling the network in term of distribution of connections which may have an important influence on the value of the thresholds.

References

[AN72] K. B. Athreya and P. Ney, *Branching Processes*, Springer-Verlag, New York, 1972.

[BBBS03] M. Bui, A. Bui, T. Bernard et D. Sohier, *A new method to automatically compute processing times for random walks based distributed algorithms*, in ISPDC 03 Second IEEE Internationnal Symposium on Parallel and Distributed Computing, vol 2069, pp 31-36 IEEE Computer Society Press, octobre 2003.

[EJY05] R. B. Ellis, X. Jia, C. H. Yan, *On random points in the unit disk*, 2005.

[EGKM04] P. Eugster, R. Guerraoui, A. M. Kermarrec, and L. Massoulié, *From Epidemics to Distributed Computing*, IEEE Computer, 37(5):60-67, May 2004.

[Gr99] G. Grimmet, *Percolation* , Springer-Verlag, Berlin, 1999.

[GK98] P. Gupta and P. R. Kumar, *Critical Power for Asymptotic Connectivity in Wireless Networks*, in Stochastic Analysis, Control, Optimization and Applications, Eds. W.M. McEneaney et al., Birkhauser, Boston, p.547-566, 1998.

[KMG01] A. M. Kermarrec, L. Massoulié and A. J. Ganesh, *Probabilistic reliable dissemination in large-scale systems*. Technical report, Microsoft Research, June 2001.

[KWB01] B. Krishnamachari, S. Wicker, and R. Bejar, *Phase transition phenomena in wireless ad-hoc networks*, Proceedings of the Symposium on Ad-Hoc Wireless Networks, GlobeCom2001, San Antonio, Texas, November 2001.

[KWBP02] B. Krishnamachari, S. B. Wicker, R. Bejar, M. Pearlman, *Critical Density Thresholds in Distributed Wireless Networks*, Kluwer, December 2002.

[Pa01] S. Pajot, *Percolation et économie*, thèse de doctorat de l'Université de Nantes, 2001.

[SCS03] Y. Sasson, D. Cavin, A. Schiper, *Probabilistic Broadcast for Flooding in Wireless Mobile Ad hoc Networks*, Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2003).

[Sp87] J. Spencer, *Ten Lectures on the Probabilistic Method*, SIAM, 1987.

Optimizing Distributed Test Generation

Harry Gros-Désormeaux
ironb972@gmail.com
Hacène Fouchal
Hacene.Fouchal@univ-reims.fr
Philippe Hunel
Philippe.Hunel@univ-ag.fr

Abstract: This paper presents two optimizations of the master-slave paradigm which provides more robustness. The former uses a mobile master which randomly jumps from node to node in order to dispatch the bandwidth load during the application execution. The latter aims to fully decentralize the paradigm like peer-to-peer systems. We present and experiment our new schemes on an application dedicated to test generation used in protocol engineering. We show that they can help for unknown dynamic environments where no information is given about nodes configuration.

Keywords : load balancing, master-slave, distributed networks, peer-to-peer

1 Introduction

The Master-Slave paradigm is a well-known concept used in distributed computing for decades. The master distributes works to several slaves for computing. Whenever a slave ends its computation, it sends back results to the master or to some repository. Actually, most of distributed computing environments leverage this concept to distribute the load across the network. Performances are very interesting for some types of application from which parallelism can be expressed as multiple independent tasks.

Numerous scientific applications use the master-slave paradigm and lack optimization for new dynamic environments as the peer-to-peer ones, Internet, etc. Such networks constantly evolve and decrease the relevance for using the master-slave paradigm according to load balancing aspects as well as robustness ones. Indeed, it is hard to compute efficient chunk size for works which have to be given to the slaves. Too small chunks will produce bandwidth overhead and too large ones provide computing overhead. On other hand, the fixed location of the master naturally becomes a soft spot. To overcome some of these downsides, we suggest to move the master in a randomly way on the network to enhance the robustness of the paradigm. Finally, a full decentralization is proposed through a total replication of the master behavior on all the network nodes.

The structure of this paper is as follows : section 2 recalls some relevant works done in the field of load balancing for the master-slave paradigm. Section 3 describes the two schemes that we have studied to cope with dynamic and heterogeneous environments.

Some preliminary experiments are exposed in section 4 where performance results are given. Finally, section 5 concludes and details future works to complete this study.

2 Related Work

Load Balancing for Master-Slave distribution is an important field in where a lot of works have been done. Bellow, we describe some known studies in this area. Heymann et al. [HSLL04] present an adaptive distribution strategy which optimize two criterions : efficiency, that is the optimal number of slaves set up according to the application speedup, and the make-span, by monitoring execution time for each distributed work over iterations (time splits).

An asymptotic optimal scheduling algorithm is given in [BLR03] for indivisible tasks following a linear cost model. The optimal number of slaves is solved with a linear programming theory and is used in the computation of the optimal number of tasks. Another complementary study, Almeida et al. [AGM06] derive the scheduling problem in a task allocation problem [IK88].

[Hag97] designs a sophisticated heuristic relevant for homogeneous environment that uses mean and standard deviation with the execution time of a processor to compute the batch element size. Despite the high complexity, this algorithm gives better results than others.

In some similar way, Cesar and al. [CMSL06] adjust in a regular way a ratio which gives the size of non-allocated elements from a batch work using predictive formulas.

Yang and al [YSF03] suggest a scheduling algorithm for loosely coupled applications by estimating the average load of a processor from its history. This algorithm is more efficient than ones given in [Dai02] which leverage the NWS service[Wol98, WSH99] to estimate processors load.

We suggest two schemes that enhance traditional ways to cope with load balancing in a Master-Slave environment and allow this class of application to become more decentralized. The former is an adaptation of the original paradigm in which the master moves. The latter emphasizes the actual trend used in decentralization processes : a load balancing emergent property results from local nodes interactions.

3 Incentives and Model

3.1 The master-slave paradigm and its drawbacks

The Master-Slave algorithm is well known in distributed programming and is commonly described as follows : one process performs an administrative role by allocating independent subtasks (the whole is the entire work to be done). In general, results are gathered on the master but can be stored on other repository (see Figure 1). More formally, a process p_m divides work W in some independent works w_n with $n \in \mathbb{N}^*$ which are sent to k

process p_k with $k \in \mathbb{N}^*$.

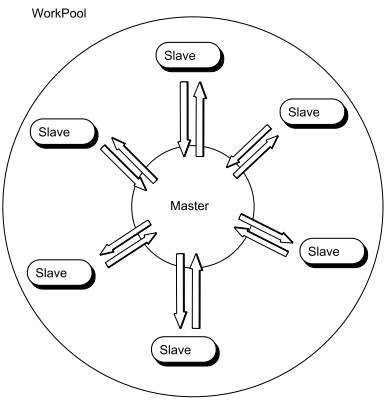


Abbildung 1: The Master-Slave Paradigm

As emphasized by Crutchfield and Mitchell [CM94], centralized applications exhibit three major drawbacks with respect to :

- the speed: a central coordinator can be a bottleneck to fast information processing,
- robustness: central failure destroys system structure,
- balanced resource allocation: the main controller draws a lot of resources.

These problems can arise on all Master-Slave schemes which limit the use of this paradigm on dynamic systems. For sake of reusability, we suggest some mechanisms to overcome these drawbacks.

3.2 The mobile Master-Slave paradigm

The mobile Master-Slave paradigm consists of permanent move of the master across the network. The paradigm can be formalized as above :

Definition 3.1 (Mobile Master)**.** The *mobile master* is an administrative process p_{i_k} hosted by some resource H_k which must achieve coordination and distributes the work $W = w_1, \ldots, w_j$ over a set of processes $p = p_1, p_2, \ldots, p_n$ respectively hosted by $H_i (1 \leq i \leq n)$.

By moving the coordinator during the application execution, the resource allocation is shared along the whole process over the nodes of the network. Furthermore, bandwidth allocation changes continously during the application execution and de facto, inherently shares in a balanced way the bandwidth between all computational resources.

Definition 3.2. We call trajectory $\mathcal{T}$ of the mobile master, the ordered set of hosts which possessed the administrative process.

New masters are elected randomly. The current master randomly selects the new master after a fixed period of time (time frame). Each computational resources keeps in a list of master locations. To ensure a minimal fault tolerance, soft state can maintain the system consistency : if for some reasons, the current master is not reachable, the previous one is reached in order to re-elect a new one. To enable such functionality, our computational resources store the master trajectory during our application execution.

Although our paradigm does not use the full peer-to-peer paradigm, it turns out to make the system more balanced. In fact, all system agents share the same capabilities. An example of such a system can be seen in Figure 2.

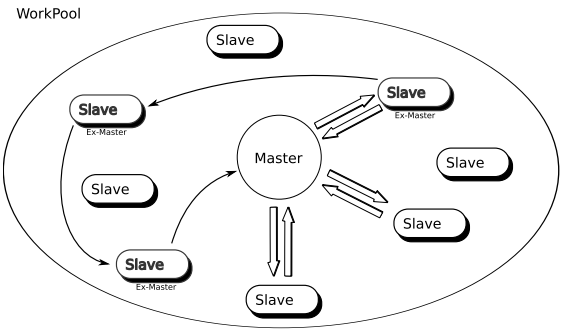


Abbildung 2: The Mobile Master Paradigm

Unfortunately, in such case, our scheme is not efficient. Moving too much data can struggle the overall performance of the system. For this reason all data relevant to the master computations are replicated on each slave. Thereby, just work index can be used to determine the new data which has to be processed by a slaver. So, very small information need to be moved whenever a new master is elected. Then results have to be stored on any repository on the the master.

Proposition 3.1. *If the whole work $W = w_1, \ldots, w_j$ is already on each host then the work index $i \in [1..j]$ is sufficient to characterize the task to be handled by a slave.*

Definition 3.3. We call *trace* of the mobile master the set of works $w_i (1 \leq i \leq j)$ which have already been distributed and processed.

Corollary 3.1. *If propositon 3.1 holds, then the trace of the mobile master can only be its work indexes.*

The model presented above is rather simple. More complex mechanisms can be implemented to leverage the load balancing techniques described previously in our related work section (see section 2). These schemes often keep information computed from scratch about nodes in the network to achieve their goal. In this case, the masters can create their

own subnetwork by binding each other and update[1] themselves on a regular way. All information is then centralized in this *Master Network*. So, such a scheme virtually allows almost all the load balancing scheme described in section 2 to be used with our mobile master paradigm without too much transformations.

Proposition 3.2. *The more the masters we have, the more the application gains in robustness.*

Beweis. More formally, let p be the probability for a node in the network to be flawed. N is the cardinal of the network and thus, the probability for a node to be the master is $\frac{n}{N}$ where n is the total number of different masters used during the application. A master is unreachable with probability $p\frac{n}{N}$. Finally, the probability that a node fails to find the master is $(p\frac{n}{N})^k$ with $1 \leq k \leq n$, the actual number of ex-masters. We notice that this probability lowers whenever the master moves inducing a more robust application to each move. □

3.3 Full Decentralization

Our previous scheme uses mobility to overcome problems which arise in centralized applications whereas our new one emphasizes ubiquity. To fully decentralize the Master-Slave paradigm, we propose to replicate the master process on each nodes in the network.

Definition 3.4 (Decentralized Masters)**.** Decentralized masters are computational resources H_k which bear an administrative process p_k and which have their own batch of works W_k.

Full peer-to-peer applications are the ones for which each peer is a decentralized master. So, load balancing becomes a more challenging task because each slave becomes its own master and because the data has to be initially at some node, it has to be spread over the network by load balancing mechanism. Bahi and al.[BCV05] have proved that synchronous distributed load balancing can be used on dynamic networks. We propose to use the GAE algorithm to distribute works of the batch pool between all network nodes.

3.3.1 The GAE algorithm

The GAE algorithm derives from GDE algorithms used in [HLM$^+$90]. These algorithms find an edge coloration of the overlaying communication graph $G(V, E)$, where V is the vertices set and E, the edge set, and uses edge color as dimension to balance the load. The graph $G(V, E)$ is transformed in a k-color graph $G_k(V, E_k)$ where E_k is a set of 3-tuples $\{i, j; c\}$ with (i, j) an edge of E and c, the color of the edge (i, j) $(0 \leq c \leq k - 1)$.
At time t, each node balances along one dimension with its neighbor its load

$$w_i^{(t+1)} = \begin{cases} w_i^{(t)} + \lambda(w_j^{(t)} - w_i^{(t)}) & \text{if } \exists j \text{ such that } \{(i, j); c\} \in E_k \\ w_i^{(t)} & \text{otherwise} \end{cases} \quad (1)$$

[1] Note that this update can be full or partial according to our objectives

The parameter λ is empirically fixed according to the network topology.

In order to use this algorithm in our decentralized scheme, load has to be considered as the size of the batch and nodes swap a part of this batch between each other to balance the load in the network. This algorithm does not take into account the fault tolerance aspect. However, this problem can be solved through batch elements replication.

We give here our interpretation of this algorithm that joins the current mainstream for massive distributed computations, that is *Emergence and Self-Organization* [Joh06, SGK06, WH04]. It is said that emergent properties are necessary to compute at massive scale when central coordinators (servers) cannot cope with the high load they encounter. These properties are obtained through local interactions between all agents which give at the macro-level the desired properties for the global system. For example, ant-based algorithms leverage this paradigm as well as many other complex systems that can be found in life. Wolfram[Wol02], through cellular automata, studies this principle and notably defines a class of these systems corresponding to complex ones (class IV) that can even simulate life. The GAE algorithm can be seen as local interactions from each node which balances its load locally. As a consequence, the distributed application turns out to be well balanced w.r.t the load.

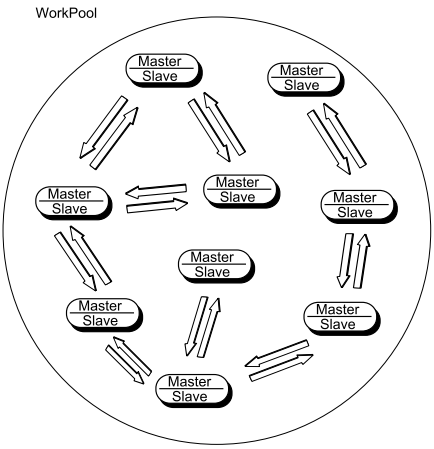


Abbildung 3: Full Decentralized Master-Slave Paradigm

4 Experimentation

We experiment our decentralization scheme on a distributed application used for test sequence generation [GDFH06]. This application computes UIO (Unique Input Output) sequences[SD88] used in conformance testing in a distributed manner. To test a system, its specification have to be described by a deterministic automaton. Computed from the automaton, unique sequences of transitions which characterize controllable states (states

on which the user can act) are drawn on the system implementation in order to detect incorrect behaviors. The distributed test generation application leverages the master-slave paradigm : the master distributes to computational resources: automaton and states for which unique test sequences have to be found when they exist.

Implementation is done through the ProActive API [BBC+06] which leverages the power of the active object model. Active objects can move freely accross the network and possess their own execution context. They can be accessed by any other process or object owned by the application. Furthermore, several mechanisms (pointer redirection, address update, etc.) ensure active object consistency whenever they move on the network.

4.1 ProActive

The ProActive framework provides the active object model used to implement our applications. An active object is a distributed object bound to its own thread and to a list of possibly remote methods calls to be processed. Indeed, each remote call to an active object is asynchronous, that is, the caller continues its execution flow until the end or some barrier mechanism. The request is put in a list of pending method calls in order to be processed later (by default, in a FIFO manner) or consumed according to a user-defined strategy. At the caller, each remote procedure result is bound to an abstraction, a *future*, which empowers asynchronicy in the application. Accessing futures only puts synchronization bareer the program if the remote procedure results are not yet provided by the active object. This synchronization mechanism is also called *wait-by-necessity*.

4.1.1 The Mobile Master

Our implementation of the mobile master for test generation use three active objects :

- the *master*, the process which distributes controllable states to the computational resources,
- the *slave*, the computational resource which computes unique sequence from the received controllable states,
- and the *storage*, which keeps the unique test sequences found at the slaves.

As described previously in section 3.2, the mobile master periodically moves on each slave using the ProActive mobility primitives. We fixed empirically each move to be done after 10 seconds. Moreover, each slave stores the master trajectory, that is, it keeps each master address during the test generation. If for some reasons, the master cannot be reached, slaves contact the previous slave which hosted the master in order to respawn the administrative process. We fixed a quorum of 70% of slaves to contact the ex-master before it can respawn the master. This let suppose that we assume a connected network without more than 30% of unreachable nodes.

4.1.2 The Decentralized peer-to-peer application

In this implementation of the distributed test generation, each peer inherits the master capabilities. This yields to an active object *peer* for each computational resource on the network. As seen previously, an active object *storage* is used to store unique sequences found by each slave. We used a third active object, the *time server*, to synchronize the load balancing by color. This object is also used to synchronize our greedy coloration algorithm : each node of the network runs algorithm 1 at its turn.

Algorithm 1: Distributed Edge Coloration

Data: Set of neighbors V of node i
foreach *neighbor* j **do**
    if *edge* (i, j) *is not colored* **then**
        Take the first available color c;
        `/* Check if c is not already used by the neighbor j,`
           `and update it if needed */`
        c = `UpdateColor(i,j,c);`
        Color (i, j) with d.

Function `UpdateColor` (*node i, neighbor j, color c*)

if c *is not available at node* j **then**
    Update c with the first available color following c;
Color (i, j) with c;
return c

The parameter λ used in equation 1 is empirically set to 0.5 and each node updates its load with its neighbor every five seconds.

4.2 Performance Evaluation

Experiments were conducted on 10 personal computers (INTEL P4 2.4 GHZ - 128 Mo RAM) with an OS-based Linux (FEDORA CORE 6) linked together with a switch 100 Mbps. Written in JAVA 1.5, our software uses JDK 1.5.0 as well as ProActive 3.2.

To compare our schemes, we generate several random automata whose specifications largely encompass real ones. Twenty 5000-states automata were used during these preliminary experiments for each scheme. Figure 4 shows performance comparisons between the two methods: the mobile and the decentralized one.

The mobile master outperforms the decentralized method on our samples. First, it is due to our testbed conditions. Indeed, our decentralized method is designed to cope with dynamic

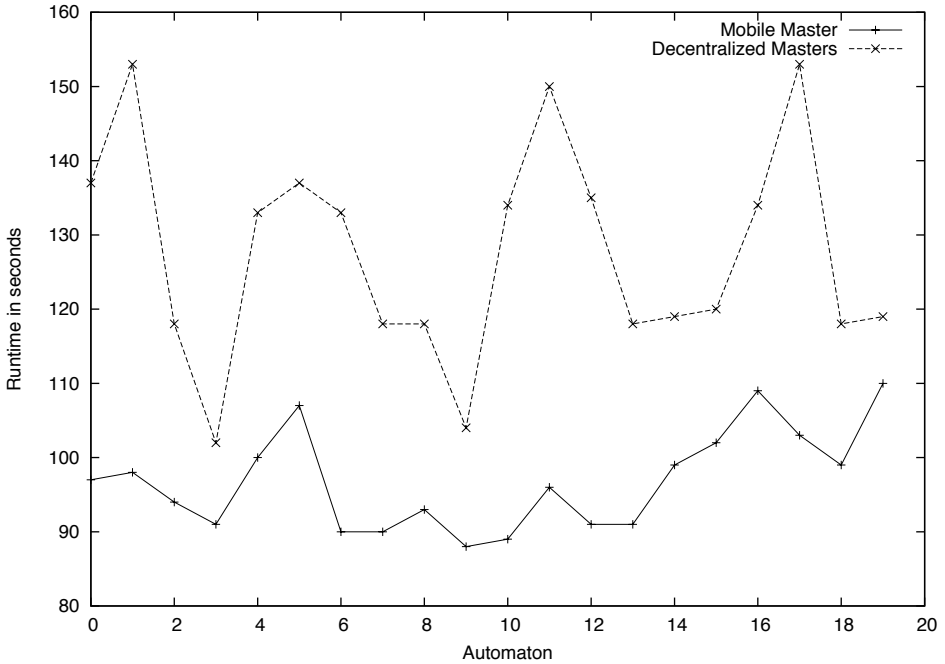


Abbildung 4: Mobile Master vs Decentralized Masters

large scale networks whereas our network environments is small, homogeneous and fast. Further, it suffers at starting from the edge coloration overhead which increases its runtime. On the other side, very low communication overhead is generated for the mobile master scheme since only work indexes (states number) are exchanged between the master and its slave. As a consequence, this scheme tends to be optimal more especially when distributed works are small. Last but not least, the distributed masters update their load every five seconds along a color and de facto, our application converges more slowly toward optimal resource allocation since works are centralized initially. Finally, some limitations hold such as our computer memory, which cannot get very large automata like millions states ones. In these conditions, the optimal number of computational resources can be maintained low and benefits the mobile master scheme. Figure 5 gives the speedup of the Mobile master application speedup (mean : 4.8) as well as the Decentralized masters one (mean : 3.7).

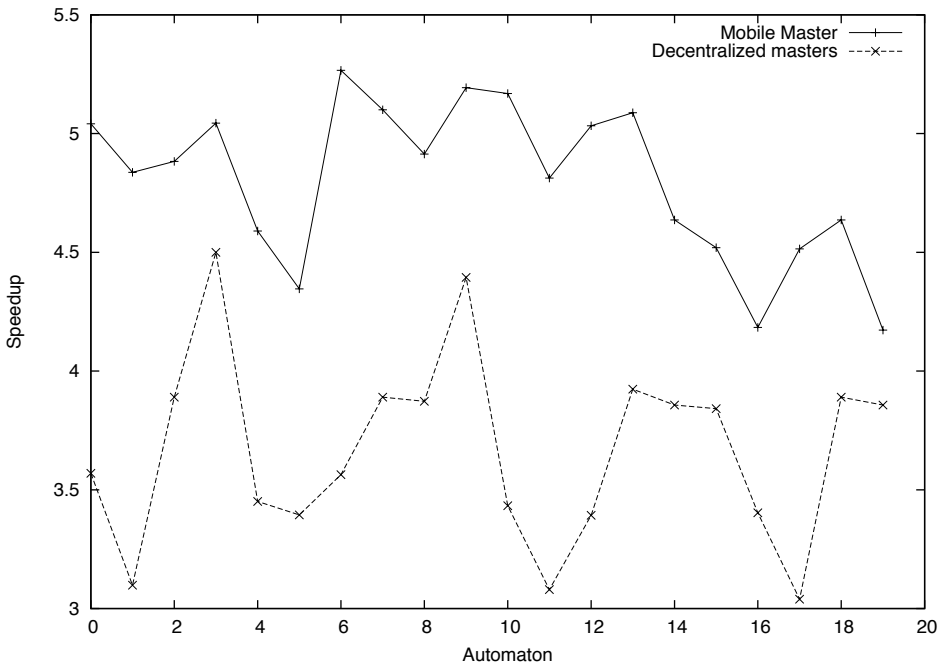


Abbildung 5: Speedup for Mobile Master and Decentralized masters

Although our decentralized masters scheme seems to be not efficient in our experiments, it is very likely it gains advantage in dynamic large scale networks. We simulate our two schemes on a parallel machine (quadri-XEON 3.6 ghz - 4Go RAM). We set the number of computational resources to 40. Unfortunately, communications exchange use the native interface memory of the system, so bandwidth is larger than real network ones. The mobile master scheme never succeeds to run properly. Investigations are on the way to know whether our implementation have memory leaks or crashes because a lot of stress are put on the system.

5 Conclusion

We suggested in this paper two schemes that enhance the master-slave paradigm. This paradigm tends to be inappropriate for large scale dynamic networks. Indeed, centralized applications are be proved to not be efficient for these kind of environments. Our first scheme uses mobility to overcome robustness downsides known for the master-slave paradigm in dynamic networks. The latter uses ubiquity and follows the actual mainstream which consists of solving problems through emergent properties on large scale networks. Our schemes were implemented on a distributed test generation application and then compared on a small homogeneous environment. This preliminary study reveals that the master-slave paradigm, enhanced with our schemes, can still be used in our actual networks without a lot of modifications and opens the way to automatic transformations for this old parallel method.

Our two schemes must be compared more thoroughly to validate our assumptions. They have to be deployed not only on a larger and more dynamic environments but also must be applied on others applications which leverage the master-slave paradigm. Some work is in progress to provide a generic platform which ease automatic transformations of applications leveraging the master-slave paradigm.

Literatur

[AGM06] Francisco Almeida, Daniel Gonzalez und Luz Marina Moreno. The master-slave paradigm on heterogeneous systems: A dynamic programming approach for the optimal mapping. *Journal of Systems Architecture*, 52(2):105–116, Februar 2006.

[BBC+06] Laurent Baduel, Françoise Baude, Denis Caromel, Arnaud Contes, Fabrice Huet, Matthieu Morel und Romain Quilici. *Grid Computing: Software Environments and Tools*, Kapitel Programming, Deploying, Composing, for the Grid. Springer-Verlag, January 2006.

[BCV05] Jacques Bahi, Raphael Couturier und Flavien Vernier. Synchronous Distributed Load Balancing On Dynamic Networks. *Journal of Parallel and Distributed Computing*, 65(11):1397–1405, November 2005.

[BLR03] O. Beaumont, A. Legrand und Y. Robert. Optimal algorithms for scheduling divisible workloads on heterogeneous systems, 2003.

[CM94] James P. Crutchfield und Melanie Mitchell. The Evolution Of Emergent Computation. Bericht 94-03-012, Santa Fe Institute, 1994.

[CMSL06] E. Cesar, A. Moreno, J. Sorribes und E. Luque. Modeling Master/Worker applications for automatic performance tuning. *Parallel Computing*, In Press, Corrected Proof:–, 2006.

[Dai02] H. Dail. A Modular Framework for Adaptive Scheduling in Grid Application Development Environments, 2002.

[GDFH06] H. Gros-Desormeaux, H. Fouchal und P. Hunel. Testing Timed Systems Using a Distributed Environment. In *Sixth IEEE International Symposium and School on Advance Distributed Systems*, 2006.

[Hag97] Torben Hagerup. Allocating Independent Tasks to Parallel Processors: An Experimental Study. *J. Parallel Distrib. Comput*, 47(2):185–197, 1997.

[HLM+90] Hosseini, Litow, Malkawi, McPherson und Vairavan. Analysis of a Graph Coloring Based Distributed Load Balancing Algorithm. *JPDC: Journal of Parallel and Distributed Computing*, 10, 1990.

[HSLL04] E. Heymann, M. A. Senar, E. Luque und M. Livny. Efficient resource management applied to master-worker applications. *Journal of Parallel and Distributed Computing*, 64(6):767–773, Juni 2004.

[IK88] T. Ibaraki und N. Katoh. *Resource Allocation Problems: Algorithmic Approaches*. MIT Press, Cambridge, MA, 1988.

[Joh06] Christopher W. Johnson. What are emergent properties and how do they affect the engineering of complex systems? *Reliability Engineering & System Safety*, In Press, Corrected Proof:–, 2006.

[SD88] K. Sabnani und A. Dahbura. A Protocol Test Generation Procedure. 15:285–297, 1988.

[SGK06] Giovanna Di Marzo Serugendo, Marie-Pierre Gleizes und Anthony Karageorgos. Self-Organisation and Emergence in Multi-Agent Systems: An Overview. In *Informatica, An International Journal of Computing and Informatics, ISSN 0350-5596*, Jgg. 30, Seiten 45–54, Ljublana, Slovenia, janvier 2006. The Slovene Society Informatika.

[WH04] Tom De Wolf und Tom Holvoet. Emergence Versus Self-Organisation: Different Concepts but Promising When Combined. In Sven Brueckner, Giovanna Di Marzo Serugendo, Anthony Karageorgos und Radhika Nagpal, Hrsg., *Engineering Self-Organising Systems*, Jgg. 3464 of *Lecture Notes in Computer Science*, Seiten 1–15. Springer, 2004.

[Wol98] Richard Wolski. Dynamically forecasting network performance using the Network Weather Service. *Cluster Computing*, 1(1):119–132, 1998.

[Wol02] Stephen Wolfram. *A New Kind of Science*. Wolfram Media Inc., 2002.

[WSH99] Rich Wolski, Neil T. Spring und Jim Hayes. The network weather service: a distributed resource performance forecasting service for metacomputing. *Future Generation Computer Systems*, 15(5–6):757–768, 1999.

[YSF03] Lingyun Yang, Jennifer M. Schopf und Ian Foster. Conservative Scheduling: Using Predicted Variance to Improve Scheduling Decisions in Dynamic Environments. In *SC'2003 Conference CD*, Phoenix, AZ, November 2003. IEEE/ACM SIGARCH. ANL.

Identifying Open Problems in Random Walk based Service Discovery in Mobile Ad hoc Networks

Adnan Noor Mian, Roberto Beraldi, Roberto Baldoni

Dipartimento di Informatica e Sistemistica
Universitá di Roma *La Sapienza*,
Via Salaria 113, Roma, Italy.

adnan@dis.uniroma1.it
beraldi@dis.uniroma1.it
baldoni@dis.uniroma1.it

Abstract:[1] Service discovery in mobile ad hoc networks (MANETs) is a challenging issue. The nodes in a MANETs offer spontaneous and variable connectivity. Also the proximity of a given service as well as the kind and the number of services vary unpredictably with time. Traditional directory based architectural solutions can hardly cope with such a dynamic environment while a directory-less approach has to resort to network-wide searches. Some solutions integrate a Service Discovery Protocol (SDP) with the routing protocol. These can improve performance but still there is need for network wide searches which is a source of inefficiency. There has been lot of work on the problem of service discovery by leveraging on the random walk based search in wired peer-to-peer networks. These works present interesting results that can be useful for MANETs and can be good candidates for SDP, as these methods require fewer resources as compared to SDPs using some sort of flooding. In this paper we have tried to identify some of the open problems in service discovery in MANETs that use random walk.

1 Introduction

A service is defined as a software that can perform specific function/functions on the behalf of users and applications over the network [AKS05]. Gibbins and Hall [GH01] define service discovery as the process of discovering location of software entities/agents that can provide access to network resources such as devices, data and services.

The number of storage and computing devices are increasing. Some of the devices provide lot of resources/services but are fixed and some are small, mobile and scarce in resources. The mobile devices can also provide services that may be of interest to other mobile devices. So there is always a need for having a mechanism that can discover and utilize services provided by other devices. The service discovery techniques are used for this

[1]This work has been partially supported by the following projects. RESIST, an EU Network of Excellence in dependability and IS-MANET, an Italian project on building mobile ad-hoc infrastructures for hostile environments.

purpose. Most of the existing service discovery protocols not only include algorithms for finding the location of a particular service but also algorithms for service announcement, service selection and dealing with mobility. Service Description is also an important aspect of a service discovery protocol. Proper descriptions facilitate the searching of a service. Usually these aspects are an intrinsic part of a service discovery protocol. As far as our present work is concerned we shall restrict to that aspect of service discovery that is concerned with finding the location of a particular service in mobile ad hoc networks.

Most of the protocols use flooding as a basic mechanism for searching a particular service. In some protocols advertisements are flooded to all of the nodes in the networks and in some query is flooded. Most of the protocols use hybrid of both methods. This wastes a lot of bandwidth of the network, making it very difficult for using such protocols for MANETs in which the bandwidth is already a scarce resource. The use of random walk as a mean of searching for an item is a well-established technique in wired unstructured Peer-to-Peer (P2P) networks [Ra02]. These random walk based method however use very little network resources and do not disturb whole of the network during the service discovery process. Such methods are very attractive for MANETs. The topology of a mobile ad hoc network is however structurally different from wired P2P network. The suitability of random walks, thus have to be studied carefully. In the following sections we shall identify some open problems regarding the service discovery in MANETs using random walk method. These open problems are identified as we go through this paper. The rest of the paper is organized as follows. In section 2 we shall describe some of the important work done in service discovery in wired networks and wireless ad hoc networks. In section 3 we shall explain the motivation for using random walk for service discovery. In section 4 we shall introduce hint as a bias and then based on hint we shall present a generic random walk based mechanism for service discovery. Our understanding is that most of the service discovery protocols that use random walk in mobile ad hoc networks use or will use this basic algorithm. The algorithm presented use hint for biasing the random walk. In section 5 we give an example that calculates hint and then uses random walk for service discovery. In section 6 we discuss some of the issues that are open for further investigation and finally in section 7 we give conclusion.

2 Existing Work in Service Discovery

There has been quite a good amount of work in service discovery regarding the wired networks but the problem has not been addressed very successfully in MANETS.

2.1 Service Discovery in Wired Networks

In wired networks four types of architectures have emerged.

2.1.1 Directory-based Architecture

In this architecture some nodes with better computation and memory resources are selected as Directory Agents (DAs) that keep a repository of all the service information in the network in a directory. These DAs advertise themselves to other nodes. Service provider nodes register with these DAs. Clients contact these DAs to get the location of service providers. Examples include Jini [Su99] by Sun Microsystems, Universal Description, Discovery and Integration (UDDI) [Oa95] by OASIS Consortium and Salutation [Sa99] by IBM. This approach is suitable for infrastructure-based networks or when changing topology is not an issue (as in 1-hop wireless networks) but not suitable for MANETS where the topology of the system keeps on changing due to the mobility of nodes.

2.1.2 Directory-less Architecture

In this architecture there is no service coordinator. Clients contact service provider directly by flooding the service query. This results in a high overhead produced due to flooding. The flooding of the query message consumes lot of bandwidth, computational and battery resources, which are already scarce in MANETS thus making this architecture unsuitable for MANETS. Examples of this architecture include Service Location Protocol (SLP) [Pe99] by IETF and Universal Plug and Play (UPnP) [Mi99] by Microsoft Corporation.

2.1.3 Hybrid Architecture

This architecture is hybrid of directory-based and directory-less architectures. In this architecture servers may either register their services with DAs (if they are available) or wait for the client service query. Client may send a service query to DAs (if they are available) or directly to service providers using flooding. This architecture is again not suitable for MANETs for the reasons method in the previous two architectures.

2.1.4 Integrating Service Discovery with Route Discovery

Service discovery can be integrated with the route discovery function of on-demand or hybrid routing protocols, as both exploit network-wide broadcasting. Such a cross-layer design principle can improve the performance but there is still need to resort to network wide searches. For example the protocol by Polyzos and Ververidis [PV05].

2.2 Service Discovery in MANETS

Some of the important service discovery protocols proposed for MANETS are following. Kozat and Tassiulas proposed a distributed service discovery architecture for MANET, which relies on a virtual backbone for locating and registering available services within a dynamic network topology [TK04]. The architecture consists of two independent parts:

backbone management (BBM) phase and distributed service discovery (DSD) phase. A similar approach has been proposed recently by Sailhan and Issarny [BBL05] for implementing a scalable directory service for MANET. The architecture is based on homogenous and dynamic deployment of cooperating directories among the network arranged in a virtual network. Despite the goal of the architecture as achieving scalability, the paper presents performance results only for a few nodes (90 nodes). Another protocol is Konark [Ve03] that is designed specifically for the discovery and delivery of services in multi-hop ad hoc networks. It supports both push and pull modes, with a cache in all devices and uses multicast to advertise and discover services. Allia [Ra02] by Ratsimor, Chakraborty et. al. also follows a decentralized directory-less approach in which the nodes, which are geographically close form groups called alliances. Klein and Konig-Ries et. al. propose a protocol called Service Rings [KKO03b] that form an overlay structure by grouping of nodes that are physically close and offer similar services. Each service ring has a designated service access point (SAP) through which the nodes within the ring can be accessed as it has all the information about the services offered within the ring. These SAPs are also connected with SAPs of other service rings thus forming a hierarchical structure. The directory information is kept in chosen edges that are dynamically selected. Another protocol that forms overlay network is Lanes [KKO03a] by Klein, Konig-Ries et. al. It is inspired by Content Addressable Network (CAN) protocol, which is used for service discovery in wired peer-to-peer networks. Some nodes are grouped together to form an overlay network forming lanes of nodes. Each group is called a lane. Nodes in the same lane have the same directory replicated in each node cache. There are different lanes in a network that are loosely coupled with each other. A field theoretical approach to service discovery has been proposed by Lender et al. [MLP05]. A service is modeled by a (positive) point charge, and service request packets are seen as (negative) test charges that are attracted by the service instances. They map the physical model to a mobile ad hoc network in a way where each network element calculates a potential value and routes service requests towards the neighbor with the highest potential and hence towards a service instance.

3 Motivation for Using Random Walk and Biased Random Walk

Given a graph and a starting point, we select a neighbor of it at random and move to this neighbor. We then select a neighbor of this point at random and move to it and so on. The random sequence of points selected this way is a random walk on the graph. The interested reader is referred to [Lo93] in which the theory of random walks on a graph has been explained in detail. In this approach the node, which is interested in a particular service sends a query message. This query message plays the role of the walker that randomly moves on the graph formed by MANET and eventually approaches the service. The advantage of this approach is that there is no flooding that wastes lot of scarce bandwidth resource in MANET. Also there is no broadcast storm problem that results in collision of packets. In random walk on the graph [Lo93] the neighbor is selected randomly, but a biased selection can also be made. There is an interesting result in this

regard. Let us consider a simple case of N+1 nodes arranged in one-dimensional form as shown in Figure 1.

Figure 1: Random Walk on N+1 points

Suppose a node i wants to search a service present in node s. For this purpose, to forward a query message there are always two options available; one is sending the query message towards the service provider node s and second is sending the query message away from it. Let the probability p of forwarding the query message to the neighbor node that is near to service provider node s be 0.5. Then the probability q of selecting a node away from the node s is also 0.5. Thus the selection of node for forwarding the query packet is completely unbiased. In this case the mean hitting time is N^2 where the hitting time is the expected number of steps or hops in a random walk before a node s on the rightmost side of Figure 1 is visited for the first time, starting from the leftmost node in Figure 1 [Lo93] [GG87]. Let us take another case in which if each node selects a neighbor node towards the service provider node with probability $p = 1$, that is, the selection is biased, then there are exactly N hops. Thus biasing the selection process from 0.5 to 1 reduces the number of hops from N^2 to N, a query message takes to reach the required service. Although this result is just for a particular case of having the nodes present in a linear fashion and the search-initiating node being the left most but even then this is quite interesting. From this we can anticipate that in a more general case biasing decreases the number of hops of a query message to reach a target. In a service discovery protocol if we can find some way to bias the selection of next neighbor towards the service provider, such that the probability is $0.5 < p < 1$, we can then considerably decrease the number of hops required for the discovery of service. The problem remains to find some metric that can be used to bias the next hop selection for service discovery packet. This metric will help in selection of nodes for forwarding the query message to nodes that can form a bridge to the service provider such that the query message reaches the service provider in a few number of hops. In the next section we shall describe a generic random walk based algorithm that uses such a metric.

4 A Generic Algorithm for Random Walk based Service Discovery

A Service Discovery Protocol (SDP) based on random walk can be biased. Let us call the biasing information as hint. Hint is calculated with respect to a service. It provides the proximity of a node i to a node s at time t. In other words it gives the information that the

node i has some time ago remained in contact with the service provider node s. Contacts can be of two types, direct or indirect. Direct contact means to be in the neighbor, that is, in the wireless range and indirect contact means nodes are not in wireless range of each other but can contact each other through some other node(s). No hint will be available in case node i has never been in the contact of node s. Hint can be calculated by any suitable method. One of the methods is to use Global Positioning System (GPS) information. Here we shall describe another method to calculate hint, which is given in section 5. This method makes use of node speeds.

Let us now describe a generic algorithm for service discovery that uses hint. This algorithm can be described to occur in two phases. In the first phase hints are calculated and distributed in the network. These hints are stored in the nodes. It is not necessary that every node may have enough information to calculate the hint. So in a mobile ad hoc network some nodes may have hints and some nodes may not have a hint. In the second phase, when a node wants to search for a particular service, it sends a query message to its neighbor nodes. A forwarding protocol that runs on each node forwards the query message according to a policy. The algorithm can be described as follows.

Hint creation and distribution phase

1. Node i, after every ΔT secs, when passing near a service provider exchange some specific information (that helps in calculating hint) and calculate hints with respect to the particular service.
2. The calculated hints are stored in the service table of the node i, which in addition to hints also store information about the service corresponding to the hint. For example the service table contains the description of the service, ID of the node having the service, etc. The previous values of hints are deleted from the service table after every ΔT secs.
3. A node while moving sends its stored hints to the newly encountered neighbors that have not been able to calculate hints due to lack of some specific information (example in next section).

Forwarding protocol

1. A node wishing to discover a service S generates a request message containing the description of S.
2. If there are some nodes that have hints regarding the requested service, then among these nodes, a node is selected probabilistically for the next hop.
3. If hints are not available then the selection of the next hop node is at random.

There are some points that need further clarification.

- In the hint creation and distribution phase the time interval ΔT secs is mentioned. This time interval needs not to be constant. It can vary with the mobility of nodes. For example ΔT can be small for node that are moving fast and it can be long for node moving slowly.

- There is a reference to some specific information. This specific information depends on the method of calculation of hint. This will be further explained in the next section by giving an example.

- In the forwarding protocol the node for next hop is selected probabilistically among nodes that have hints. In fact the hints do not always give the correct indication of proximity of the service provider. Small value of hint may or may not be close to the service provider as compared to another small value. This is because a node after calculating hint may go completely in a direction that make it away from the service provider as compared to other nodes. So we probabilistically select among the given choices and the probability of selection of a particular node depends on the confidence of the correctness of the value of hint.

5 An Example Implementing the Generic Algorithm

The most obvious method in determining the nearest node to the service provider is using the Euclidean distance. We can then have a Euclidean metric for biasing the selection process for the next node. But this metric requires GPS to determine the relative distances of the nodes from the service provider.

GPS is not always possible. Another interesting technique, based on finding the relative speed of different nodes with respect to the service provider can be used. Hint h_{is} [BBL05] is calculated and stored by node i for a service provider node s.

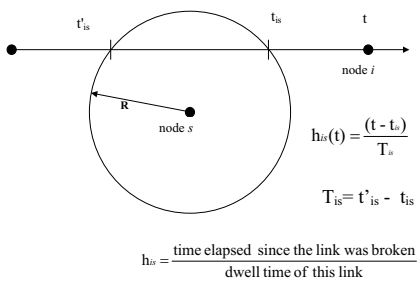


Figure 2: Calculation of hint using relative speeds

To calculate hint, node s sends an advertisement message after every ΔT secs containing the description of the service. This advertisement message is picked up by a mobile neighbor node i, which then updates its service table. The service table at node i, in addition

to other information also records the duration of the last wireless link established with s and time elapsed since the link with s was broken. The node i after every ΔT secs calculates hint and while moving sends the hint to the newly encountered neighbors. The nodes that have low values of hints are more probable to be near to the node s. For details see [BBL05].

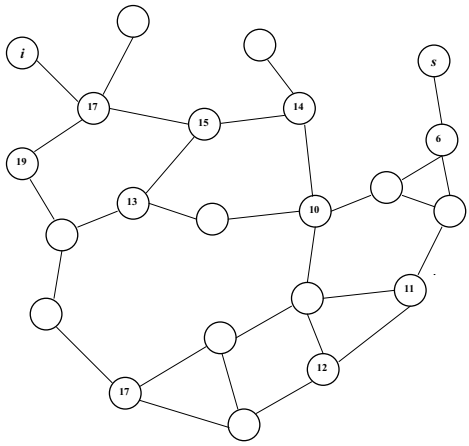


Figure 3: Mobile nodes with and without hints

Figure 3 shows the first phase of hint creation and distribution. The values shown in the node are the hints. Note that some nodes have hints and some nodes do not have hints.

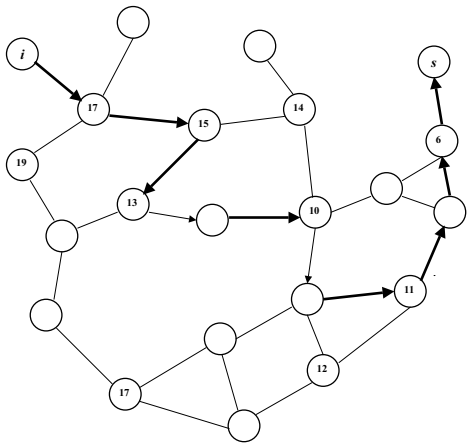


Figure 4: Biased random walk of query message

Figure 4 shows the forwarding protocol in action. The query messages starts from node i and following a random selection (shown as thin arrow) and probabilistic selection (shown as thick arrow) eventually reaches the node s.

6 Discussion

In the previous section we discussed an algorithm that is a "generic biased random walk based service discovery algorithm". We call it generic, as most of the random walk based algorithms would have the same form except that the hint may be calculated in a different way. We see that there are many issues that still remain to be solved.

6.1 Flooding is not better than Random Walk.

One of the main issues is to show that flooding is not better than random walk or under what conditions flooding is better is than random walk. It has been proved in [SGM04] that random walk is better than flooding under some specific conditions but these results are mainly for wired peer to peer networks. Such an effort is required for broadcast based wireless ad hoc networks.

6.2 Next hop selection

Selection of next hop node is based on a hint, if available. We gave an example of hint that is calculated using times. The problem still remains to find a better hint, which can guide the query message to the service provider in least number of hops or the shortest path available in network.

6.3 Termination criteria

In the presented generic algorithm no criteria has been defined for the termination of random walk search. The search can be terminated when the random walk first encounters the requested search or it may continue to search for other service providers till a specific TTL is reached. There could be other termination criteria. For example a search can be terminated if a node is visited a specific number of times, etc. One can investigate other termination criteria and their effect on the service discovery process.

6.4 Simultaneous random walks

Instead of one random walk, there can be many random walks simultaneously searching for a service. This can increase the probability of finding the service and decrease the searching time. An interesting investigation would be to investigate the number of simultaneous searches that can be made for efficient discovery of a service.

6.5 Loop formation

Since the next hop selection for a query message is probabilistic so there is no possibility of loop formations under the condition of mobility of nodes and probability of having incorrectness in the values of hints. But if the system becomes static and all of the nodes have hints that give correct information about the proximity of the service provider then there can be a possibility of loop formation and a query request can just keep on moving in a loop instead of moving in the whole network for the discovering the service. Conditions under which looping may occur and methods to avoid looping of the query message should be investigated.

6.6 Effect of mobility

An interesting investigation would be study the effect of mobility on service discovery using the biased random walk method presented here under different mobility scenarios.

6.7 Effect of different hints

The method of hint calculation also affects the discovery mechanism. Hints calculated in different way would affect the way random walk is progressed for service discovery. One can study properties of random walk under the effect of different hints.

6.8 Distributed selection

In the method presented the process of selecting a node is done in a centralized way. A node finds its neighbors and receive this information and then locally based on this information selects a particular node for the next hop. Another method can also be investigated in which a node forwards the query message to its all neighbors. All these neighbors may be running some protocol for selection and eventually one of the neighbors decides to forward the query message.

7 Conclusion

In this paper we tried to present some of the open problems in service discovery that use random walk in MANETs. We explained different concepts of random walk and service discovery and while explaining these, we identified some of the problem areas and problems that still need attention. We presented an algorithm that we called as a generic random walk based algorithm and then discussed different aspects of it and problems and issues

that need further investigation.

References

[AKS05] Hitha Alex, Mohan Kumar, Behrooz A. Shirazi. Service Discovery in Wireless and Mobile Networks. University of Texas at Arlington, 2005.

[BBL05] R. Baldoni, R. Beraldi, L. Querzoni. A hint-based probabilistic protocol for unicast communications in manets. Elsevier Ad Hoc Networs, 2005.

[GG87] Gefen and Goldhirsch. Biased random walk on networks. Physical review, Feb. 1987, 35(2).

[GH01] N. Gibbin, W. Hall.Scalability Issues for Query Routing Service Discovery. Proceedings of the Second Workshop on Infrastructure for Agents, MAS and Scalable, 209-271, May 2001.

[IS05] V. Issarny, F. Sailhan. Scalable service discovery for manet. In Proceeding of the 3rd IEEE Int'l Conf. On Pervasive Computing and Communications (PerCom2005), Kauai Island, Hi, USA, 8-12 March 2005. IEEE.

[KKO03a] M. Klein, B. Konig-Ries, P. Obreiter. Lanes - a light weight overlay for service discovery in mobile ad hoc networks. Technical Report 2003-6, University of Karlsruhe, May 2003.

[KKO03b] M. Klein, B. Konig-Ries, P. Obreiter. Service rings - a semantic overlay for service discovery in ad hoc networks. In DEXA Workshops, pages 180-185, 2003.

[Lo93] L. Lovasz. Random walks on graphs: a survey. In Combinatorics Paul Erdos in Eighty, Vol. 2, Budapest, 1993. J'anos Bolyai Mathematical Society.

[Mi99] Microsoft corporation. Universal Plug and Play (UPnP) forum. http://www.upnp.org, 1999.

[MLP05] M. May, V. Lenders and B. Platter. Service discovery in mobile ad hoc networks: A field theoretic approach. Elsevier Pervasive and Mobile Computing, 2005.

[Oa95] OASIS consortium. The universal description, discovery and integration (UDDI) Feb 2005. http://www.uddi.org

[Pe99] C. Perkins, J. Veizades, M. Day, E. Guttman. Service location protocol, Version 2. RFC 2608, Internet Engineering Task Force (IETF), June 1999.

[PV05] G. Polyzos, C. Ververidis. Routing layer support for service discovery in mobile ad hoc networks. In Proceedings of third IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'05), Washington, DC, USA, 2005. IEEE.

[Ra02] O. Ratsimor, D. Chakraborty, A. Joshi, T. Finin. Allia: Alliance-based service discovery for ad hoc environments. In ACM Workshop on Mobile Commerce WMC'02, September 2002.

[Sa99] Salutation architecture specification, ver 2.1. http://www.salutation.org, 1999.

[SGM04] A. Saberi, C. Gkantsidis, M. Mihail. Random walks in peer-to-peer networks. In the Proceedings of INFOCOM 2004, Hong Kong, March 7-11, 2004, IEEE.

[Su99] Sun Microsystems. Jini network Technology. http://www.jini.org, 1999.

[TK04] L. Tassiulas, U.C. Kozart. Service discovery in mobile ad hoc networks: an overall perspective on architectural choices and network layer support issues, 2004.

[Ve03] V. Verma, C. Lee, S. Helal, N. Desai. Konark, A service discovery and delivery protocol for ad hoc networks. In the proceedings of the third IEEE Conference on Wireless communication Network (WCNC), New Orleans, USA, 2003, IEEE.

Mobile and Smart Devices in a Human Community – the Challenge of Context-aware Distributed Networking

Gerald Eichler[1] and Christian Erfurth[2]

[1]T-Systems Enterprise Services GmbH, Technologiezentrum ENPS
Deutsche-Telekom-Allee 7
D-64295 Darmstadt, Germany
[2]Friedrich-Schiller-University Jena, Department of Computer Science
D-07740 Jena, Germany
gerald.eichler@t-systems.com; erfurth@cs.uni-jena.de

Abstract. This paper addresses the new and changing role of mobile devices and services in a dynamic human community, taking individual profiles and the current personal context into account. Binding and interaction are analysed in order to recognise future trends. Considering a community network as a distributed information source, filtering and matching of information tend to become a calculation intensive process. With mobile personalised assistants the way of processing information will be changed. Sample scenarios outlined in this paper show the basic function of such assistants. With the ongoing evolvement of mobile devices and first available frameworks like the Tracy Toolkit an implementation of mobile community supporting systems is possible. The process of contextualisation is defined. Finally, the paper attempts a view in requirements of community support as well as requirements for mobile devices.

1 Pervasive and Personal Communication Services

Buzzwords sell! While Europe focuses on the "social challenge" with the term *Ambient Intelligence* (AmI), America calls the same ideas of their "winner technology" *Pervasive Computing*, which is sometimes adapted to *Pervasive Communication*. Asia recollects their "emperor mentality" when creating the "Invisible Computer". Information technologies are more and more driven by social aspects. While individuals tend to be more independent from each other than ever, networks penetrate all areas of daily life. Moreover, information are taken from the internet considered to be right, just from the fact that they are there. This *network proof phenomena*: "I've seen it on the internet, it's true" leads to unexamined URL citation, even in the scientific community.

Driven by an increasing number of mobile devices using IP inter-communication, fixed network infrastructures are extended by ad-hoc networks, which are established on demand and change all the time. This is a mirror of human communities, where devices represent their users. Furthermore, a user has his/her own personalised assistant which represents him/her in the network, even if the user's device is offline. Fixed links are replaced by autonomous assistants, migrating on a big virtual network when required.

Chapter 2 describes the new role of mobile devices, while chapter 3 takes a look at how context awareness will drive the community aspect by means of mobile personalised agents. A first set of requirements for community mobility is presented, too. A compact vision of the future of contextualisation is provided in chapter 4.

2 Role of Mobile Devices

In the long term past incumbent telecommunication operators always propagated their systems. The most classic example might be the worldwide Plain Old Telephone *System* (POTS). The next era centred the services. Logically, the first complete digital communication network was called Integrated *Services* Digital Network (ISDN). With liberalisation of the telecommunication market the customer came to the fore. The Digital *Subscriber* Line (DSL) stays abreast of changes, being the starting point of all IP networks. A similar evolution holds for the devices' domain.

Looking at framework architectures, the same development leads the design. System middleware like the Common Object Request Broker Architecture (CORBA) was replaced by web services, resulting in the Service-Oriented Architecture (SOA). Today's business cases are based on subscriber bundles. We will call both effects the SSS or "triple S evolution" (S^3), which stands for *System* $\Rightarrow$ *Service* $\Rightarrow$ *Subscriber*, representing the experienced alteration. But what is next: the device, the peering or the Personal Area Networks (PAN) again?

Communication, computing and society grow together, which is reflected in the current European R&D programme [EC02]. Mobile devices bring much more dynamics into established networks, including IP migration. The following seven theses summarise our experience noticed over the resent years:

1. **Communities go mobile.** Smaller and powerful mobile devices allow increasing consumption of seamless services (e.g., mobile weblogs).
2. **Private meets business.** One mobile device with multiple functions is used for both. Private and business databases are merging (e.g., PDA).
3. **Users are ad-hoc involved.** Mobile communities imply short term memberships. The creation of content increases compared with its consumption.
4. **Personal assistants feed individual needs.** Due to a big information flood the selection of relevant content requires intelligent support (profiling, filtering)

5. **Personal data become a value.** Loss of physical device or a system crash become less important than a stolen network identity or lost configuration data.
6. **Security is taken for granted.** The belief in useful and secure configuration data is higher than the active adjustments for personal protection (defaults).
7. **Life cycles speed up.** Mobile devices are replaced in short terms and new services emerging more often, while reliability of both decreases.

2.1 Mobile Device Bindings

End user devices become smaller and portable, accompanying their owners everywhere. While in the past, communication and entertainment devices such as telephones, television sets or personal computers were shared by several persons living together (fig. 1a), universal mobile equipment (laptop, PDA or mobile phone) reflects a first step in social paradigm change towards personal devices and service consumption accessing different networks (fig. 1b). More recently, cheap service specific devices (MP3 player, game consoles, navigators) flood the market. The users' majority owns already a bundle of individual devices (fig. 1c). This enforces the trend of emerging community services to share individual experiences, using new channels of peer-to-peer-communication, while still forming a virtual service grid.

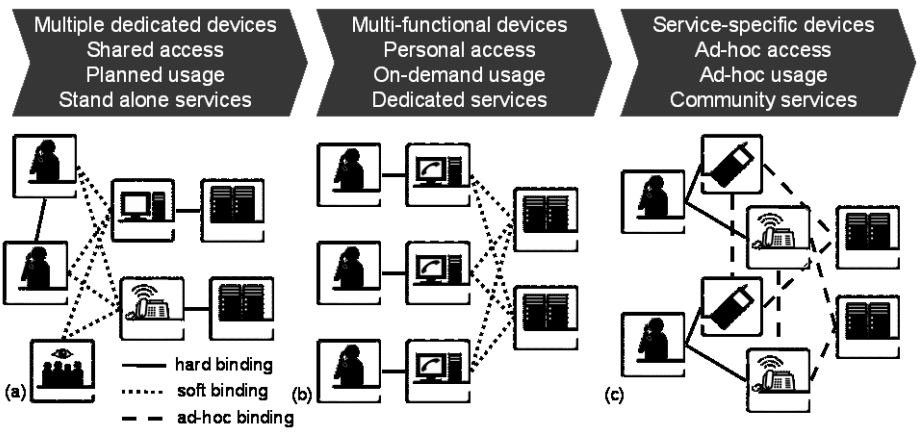


Figure 1: Changed binding of end user devices and services. (a) system binding, (b) user binding, (c) community service binding.

The initial service centered approach did change to a device centered approach, giving the user a new feeling of freedom. The big variety of possible network access technologies hinders easy-going ad-hoc peering, while quasi-standard service specific end user devices like Apple's iPod bring the idea forward.

2.2 User and Environmental Interaction

Today's mobile devices reflect dramatically the short development cycles. First priority is given to *technical features* no matter how easily they can be applied. Second attention is paid on fashionable *design trends* to sell new devices on the market. Unfortunately, *intuitive usability* ranks on very low priority for several reasons e.g., expensive and long development iteration including user group specific testing or difficult sales promotion. The life period of devices becomes even shorter and sales profit is taken from service bundle sales, not from the devices themselves.

An increasing number of features and interfaces leads to an increased power consumption and therefore decreased stand-by time. There are multiple power consuming network access opportunities e.g., GPRS/UMTS, Bluetooth, WLAN, IrDA on the one hand and many new *multi-modal user interfaces* e.g., touch screen, voice control or acceleration & orientation sensors on the other. Even the integrated camera becomes a new environmental interface using visual codes and maps for interaction [RZ05]. Nokia provides special device shells with an RFID reader. Bar code readers are available as SD cards. These are just staring points for new ways of *tag-based* Near Field Communication (NFC) which lead the users from a spamming driven *push* mentality to becoming active by invocation of simple *pull* actions.

2.3 Mobile Community Services

With mobility users become more active. Pure entertainment moves to edutainment [MO05]. Mobile diaries are written, content is created and will be shared within a next step. Social networking as known from the web office application openBC [BC06] will develop to a new level. But currently, the focus is still on the new mobile features.

A survey among 60 users of smart mobile devices, aged from 20 till 60, indicated that still information services are seen as most potential. Surprisingly, there were no significant variations in the results of different target groups divided by age, gender or profession. Within a scale of 1 (low relevance) to 5 (high relevance) travel & leisure information gets a 3.7 closely followed by consumption services, covering personalized advertisement, shopping and product information with a 3.5. Mobile gaming, which has definitely the closest community relation of the evaluated service offers, gets with 2.9 the lowest overall result, because of missing personal interest. In general, private usage of mobile services seems to be more relevant than basic business services, like electronic business card exchange.

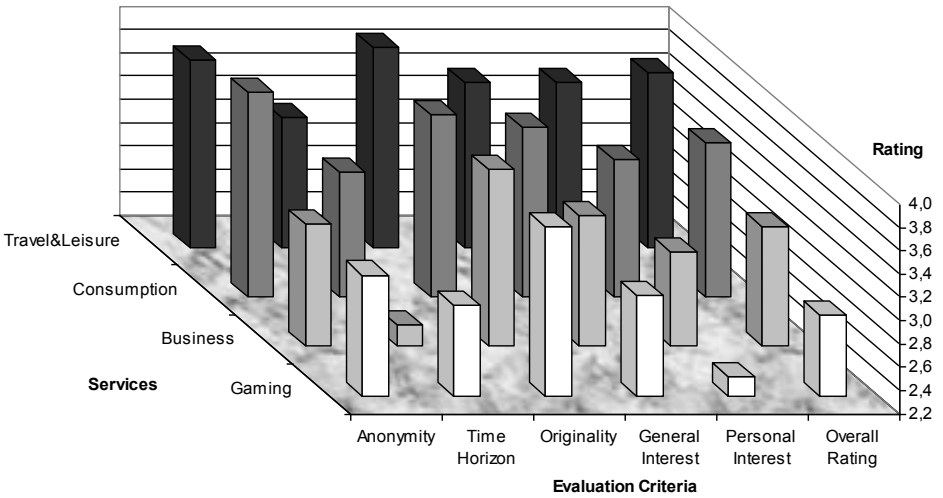


Figure 2: Survey on the acceptance of mobile service domains: travel & leisure, consumption, business and gaming among 60 users of mixed target groups (Nov. 2004 by T-Systems).

A more detailed impression about the survey results on special criteria, i.e. role of user's anonymity, expected time horizon, originality of the idea and general as well as personal interest is given in fig.2. The overall rating is calculated as a balanced average of these five criteria.

2.4 Sample Basic Services

Community applications rely on a set of extensible basic services. Except standard communication services e.g., mail/SMS, phone/video conferencing and application sharing, these are typically services, which help to establish meaningful ad-hoc relationships. Therefore, the current *location* [EB05] and personal *profile* [EW06] are the main sources – more general the *context*.

The *resource finder* assumes that a service provider has a database, where each object (point of interest) is associated with an offering profile. In that setting, the user is able to retrieve an ordered list of interesting resources, ranked by personal relevance, filtered against the user's profile based preferences.

The *contact builder* assumes that each user of the system has an associated pair of profiles (search and offer), describing relevant issues that are weighted positively or negatively. Similar to the resource finder, it is not only possible to find persons, who match a given search profile as opposed to one's own offer profile, but also to look for potentially interesting users by starting with a list of friends and finding out related persons filtered against one's own preferences.

The *annotation service* uses information tags e.g., points at which new content (text, voice message, photo) is instantly created and exchanged between mobile devices and resources. For instance, a discussion forum about a specific location may be directly associated with a hot spot in the vicinity, or a new product may have an information tag where it is possible to leave a rating for the manufacturer or community members.

When considering new ways of communication, such as weblogs or podcasts, this may alter the way of future human-to-human and group communication in an ad-hoc and asynchrony manner.

3 Context Awareness and Communities

The community idea can profit from and be extended by context aware systems. Personalisation as one instrument for context awareness is used to find partners in a community which have similar interests or helps to create value-added community features. The supporting effect of software (and devices) for a user will be improved: the user gets its personal assistant. Furthermore, the complexity of distributed software systems can be reduced if self-organising aspects are taken into account by realising community systems.

3.1 Classification of Context

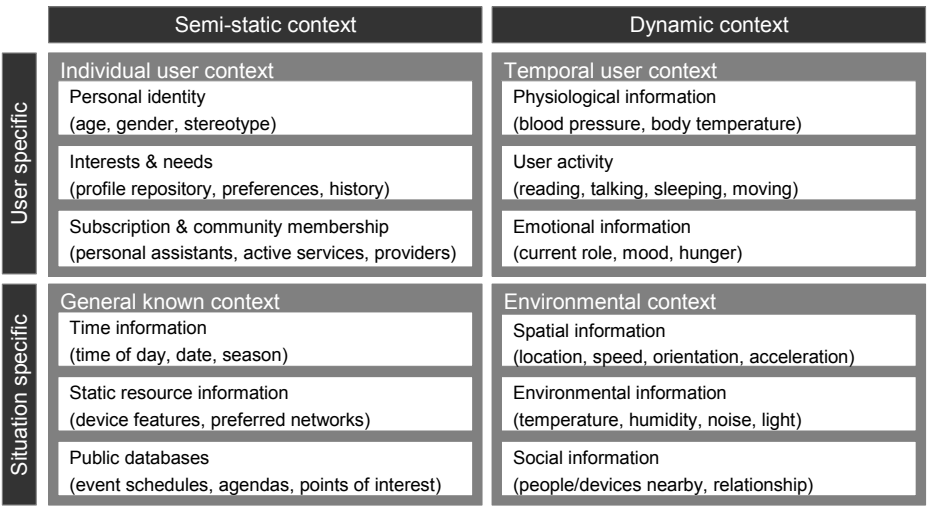


Figure 3: A context classification matrix.

An *event* is defined as a threshold driven change of one or more context elements. The meaning of an event depends always on the surrounding circumstances – the context.

So, one could just ignore an event if it is of no importance in the current situation. In another context the same event would be highly relevant. A challenge of developing context-aware software is to rate the relevance of an event in a given context. First of all, the context needs to be perceived by software. Therefore, the term context has to be defined more precisely.

In [DA99] an explanation can be found in which a context is defined as a set of information that can be used to characterise the current situation. There are also approaches to classify types of contexts [SAW94], [CK00] like computing context, time, user, physical, and history context. Following such approaches a user-centred classification of information is done as shown in fig. 3. The two dimensions show a classification of the context information: horizontally – user specific and situation specific information; vertically – separation of information about context in (semi-)static and dynamic parts.

3.2 Personalisation and Awareness

With *personalisation* the quantity of perceived context information is potentially high. Often only a small subset is of relevance. Through personalisation, information is filtered and individual information's relevance can be weighted according to users' interests and preferences. The process of filtering is tightly connected to matching. For an optimal matching, information are annotated with semantic information or classified in a domain-specific ontology. Languages like the Web Service Modelling Ontology (WSMO), the Resource Description Framework (RDF) or the Web Ontology Language (OWL) can be applied.

For communities personalised information is relevant to get involved. For instance, if one is seeking its surrounding area for people whose interests are common, everybody has to share this piece of information. This leads to the concept to divide personalised information into public and private data. It also makes sense to publish public data actively. The publishing process uses mobile software assistants which are able to migrate from a user's device to a set of reachable devices in its vicinity in a round trip manner. Users will be notified if matching interests are found by the mobile assistant. Therefore, such an assistant needs environmental information like location, orientation, acceleration etc. to be context-aware as well. Looking at fig. 3, information on subscription and community membership are relevant for personalisation of a mobile assistant (user specific, semi-static context).

3.3 Context Discovering Assistants

A user joins a service grid – a community network with services and resources – by just enabling the mobile device or by moving into a certain service grid area. The user wants to get involved by sharing data or services. After joining the network the following two steps will be passed:

1. Other (community) members, respectively their devices, get notified about this new participant and its contribution to the service grid.
2. The new user will receive information on the current situation within the community – its context.

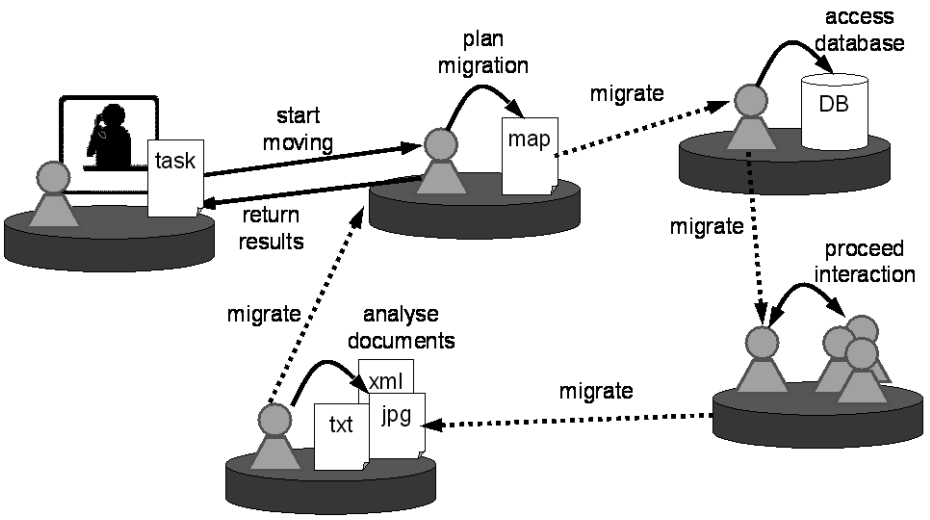


Figure 4: Information seeking personalised assistant.

Now the community has grown. But there are some open questions. In detail:

- Where are interesting services, shared data and other public resources in the community collected and stored?
- What information is passed to an entering user?
- Where does the filtering and matching process happen?

Especially, the last question is interesting in the case, where the user's device is limited within its resources (transmission, storage and computational power) like PDAs or smart phones. Depending on the size of the community the *quantity of information* might be quite high. In worst case a complete transmission of available resources in the community network (step 2 from above) would exceed the device's abilities or the operation duration to parse and filter data would frustrate the user.

A community network can be seen as a distributed information source. Various resources are distributed across the network. Whereby, the content of the information source is changing continuously. For a small community size, say a dozen ad-hoc connected users, it is not worth to create a central instance containing all information of this community. More efficient is the usage of *mobile agents* [BR05] or *assistants* which are seeking for information by *migration* as indicated in fig. 4. An adaptation of that simple scenario "joining a service grid" would be as following: The user's personalised assistant:

1. leaves his/her device and
2. visits all community members (migration),
3. informs them on the new user's contribution,
4. filters obtained information at its origin and
5. returns to its owner with the matching portion of relevant information.

Calculation costs are distributed and in case of a high filtering rate network traffic is reduced too. Updates could be propagated using mobile assistants. Further value-added features are imaginable: Mobile personalised assistants make appointments for their owners or negotiate the exchange of resources. In case of larger communities, a creation of a map with context information, showing the local region (vicinity) around members, will be a suitable solution [Er04], assuming information are fairly stable. Only rough information is stored at the map on far away areas. The maps are built up for orientation purposes: a mobile assistant will use them to gather information on places which are worth looking at by migration.

As an example, a mobile assistant is used to gather information on wines at different online wine stores. The assistant is personalised regarding my preferences on taste, year of harvest, price, etc. In the vicinity of the assistant, at the local map, there are places which have information on wines respectively, which provide a service to gather information on wines. Services are classified at the map by means of ontologies. At the map there is also information on remote services. So the assistant is able to localise promising remote services e.g., beverage shops. Due to reduced information on remote services at the local map the assistant is not able to estimate whether the service could be helpful or not. The assistant has to migrate to these targets.

3.4 Community Scenario Campus.NET

At Friedrich-Schiller-University Jena the research project MobiSoft [MS06] explores the usage of personalised assistants on mobile devices. The publicly funded joint project between two local companies, the GODYO AG (http://www.godyo.de/) and the start-up company 'the agent factory GmbH' (http://www.the-agent-factory.de/), and the university is seeking for future application scenarios and realises necessary improvements of existing technologies for usage of mobile devices. The technical realisation of such a mobile personalised assistant is based upon mobile agent technology. The *Tracy Toolkit* [Tr06] is a result of the university's research activities which has reached product status by 'the agent factory'. This toolkit implemented in Java has a lightweight kernel with minimal basic functions to make it useable on small devices. Additional features are available via plug-ins, as pictured in fig. 5.

A sample scenario called Campus.NET – a community network at the campus – will be prototypically realised as one selected MobiSoft scenario to get a realistic feedback of the technology's abilities and user's acceptance. Various university-related content and services will be accessible with a personalised assistant from a mobile device e.g.,

lesson index and news, library services, information on departments. As an example my personalised assistant could notify me when my favourite meal is available in one of the student canteens. Of course, my assistant knows my courses and can observe them. If new course material is available, my assistant comes back to me with these documents in its luggage. Furthermore, my lecturer can push new information actively by sending his assistant to the students of the course.

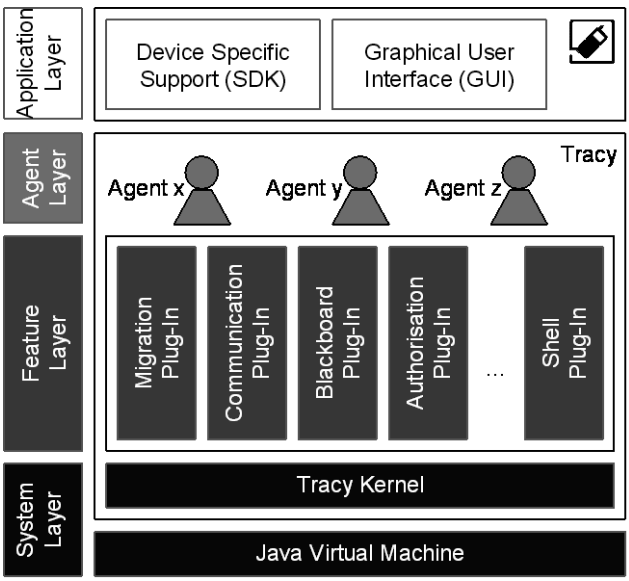


Figure 5: Tracy Toolkit - architectural overview of a typical device implementation.

Exchanges between mobile users are in focus: a bill-board with flat share offers, ride offers, games, learning groups, etc. is planned. But also direct interaction between community users will be implemented. A user can send out its assistant to find friends (contact builder function) or documents (resource finder function) within its surrounding area. NFC over Bluetooth will be possible to make appointments or hand over documents. In such community scenarios social aspects come to the fore, especially when someone could profit from modifying the system software. To counteract such tendencies an evidence based system as proposed by [Ob06] could be used to improve acceptance of the system.

In general mobile agents seem to be a promising concept to support communities. Especially when information or data is distributed over multiple sources, mobile agents can be used to analyse these sources for statistic purposes. For doctors such statistic agents could assist to confirm a diagnose or to investigate a disease: Agents could visit distributed information sources of different hospitals to access and analyse reports or X-ray photographs. It is not necessary to carry away the data itself and even store relations to persons. Extending this scenario, agents could be used to seek and involve experts for a detailed dispute.

3.5 Requirements for Community Mobility

Recent evolutions of mobile devices and device's connectivity allow communities to go mobile. With a tight delay new features are used to implement enhanced value added services. Community members are able to benefit from new, (hopefully) easier to use devices and services. Requirements for mobile communities can be derived in consideration of this evolution (table 1). Especially, the process of contextualisation requires new communication interfaces with the surrounding equipment.

Service Requirements	Hard and Software Requirements
▪ Context-awareness – presence and reachability services ▪ Contact filters for instant group join – short term membership – "social" verification and role support ▪ Appliance of – "contextualisation" – "personalisation" – "communitisation"	▪ Multiple tag receivers – visual codes, RFID, Bluetooth, … ▪ Movement and physical sensors – acceleration, orientation, temperature … ▪ Ad-hoc online/community access ▪ Programming support – communication JSRs ▪ "smart" devices with long uptime ▪ Mobile web performance (Web 2.0)

Table 1: Requirements for mobile community services and devices.

This list of requirements is for sure not a complete one. But from these requirements further work can be derived yet: development of context ontologies, of portable and exchangeable personal profiles, integration of location based services, utilization of new technology approaches (e.g. mobile assistants, self-organising middleware), etc. In addition for a wider acceptance, NFC and the support of various sensors for mobile devices have to be improved as well as the establishment of accepted standards for software development in this area.

4 Next Steps regarding Contextualisation

Many pieces, which will be part of new Context-Aware Services (CAS) are already in place. However, an entire framework for mobile, context aware, distributed services is still missing. [SFB06] proposes context modelling methods, while [DA06] focuses on a modular architecture for mobile personal services.

Multiple steps create a value added process chain to provide users with the right actions in the right place at the right time. Initial trigger events have many faces. Any changing contextual part could be the initiator. Personal assistants will help to unify this diversity. Community requirements are still an open issue. Fig. 6 illustrates the process of *contextualisation*.

The availability of freely accessible, rich up-to-date data bases (critical mass) seems to be an important success factor. This raises the question of ownership of context data.

A clear separation of privacy and non-privacy data is required by law. Certified gateways will act as a mediator focusing on *pseudonymisation* [BLR04].

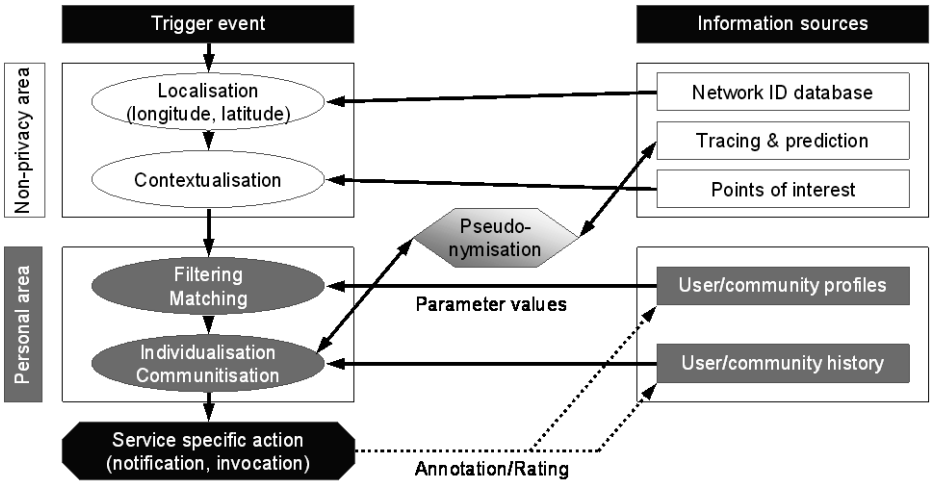


Figure 6: Event triggered recombination of information sources in public and private context using open and protected sources.

Furthermore, security and usability issues in combination with new mobile devices are a big challenge to develop reasonable business models. The mixed processing of public and private data requires several protection levels. So the S^3 evolution will experience a further development onto S^4, becoming an integrated *solution*.

Recent years have shown that device's abilities are getting more comprehensive, but especially, mobile platforms are still different from standard PCs. There are various OS-variants even for a single device. Due to this heterogeneity and restrictions, systems like the Tracy Toolkit need to be adapted and ported. Some features of the toolkit are not yet available for mobile devices. In some cases it is hard to access communication interfaces from Java-side e.g. for Bluetooth a special JSR is necessary. The development process of the Tracy Toolkit will focus on adaptation and porting of features as well as on new features.

References

[BC06] openBC. URL http://ww.openbc.com/ and http://de.wikipedia.org/wiki/OpenBC/

[BLR04] Böhm, A.; Leiber, T.; Reufenheuser, B.: Trust and Transparency in Location-Based Services - Making Users lose their Fear of Big Brother. Conference presentation at Mobile HCI 2004, Glasgow, UK, Sept. 13 –16, 2004

[BR05] Braun, P.; Rossak, W.: Mobile Agents - Basic Concepts, Mobility Models and the Tracy Toolkit. Morgan Kaufmann Publishers/dpunkt.verlag, San Francisco, USA, MK ISBN 1-55860-817-6, dpunkt ISBN 3-89864-298-4, 2005

[CK00] Chen, G.; Kotz, D.: A Survey of Context-Aware Mobile Computing Research. Technical Report TR2000-381, Dartmouth College, Department of Computer Science, 2000

[DA06] The DAIDALOS Project: Designing Advanced network Interfaces for the Delivery and Administration of Location independent, optimised personal Services. URL http://www.ist-daidalos.org/ , 2002-2008

[DA99] Dey, A. K.; Abowd, G. D.: The Context Toolkit: Aiding the Development of Context-Aware Applications. In: Human Factors in Computing Systems: HCI 99, pp. 434-441, Pittsburgh, 1999

[EB05] Eichler, G.; Böhm, A.: Einheitliche Einzellokalisierung – Voraussetzung für Track & Trace Mehrwertdienste. In: Praxis in der Kommunikationstechnik, 1/2006. pp. 9-14

[EC02] The European Commission: Sixth Framework Programme (FP6). 2002-2006. URL http://europa.eu.int/comm/research/fp6/

[Er04] Erfurth, C.: Proaktive autonome Navigation für mobile Agenten: ein Schritt in Richtung mobiler Agentensysteme der nächsten Generation. Disseration, Friedrich-Schiller-Univerity Jena, 2004

[EW06] Eichler, G.; Will, M.: Profiles and Context Awareness for Mobile Users – a Middleware Approach supporting Personal Security. In: Security in Pervasive Computing, Third International Conference, SPC 2006. LNCS No. 3934. pp. 134-148, York, UK, April 2006

[MO05] The European Commission, FP5: The MOBIlearn Project. IST-2001-37187. URL http://www.mobilearn.org/ , July 2002 – March 2005

[MS06] MobiSoft – Project Home. URL http://mobisoft.informatik.uni-jena.de/

[Ob06] Obreiter, P.: Kooperationsanreize für autonome Einheiten in selbst-organisierenden Informationssystemen. Dissertation, Friedrich-Schiller-University Jena, 2006

[RZ05] Rohs, M.; Zweifel, Ph.: A Conceptual Framework for Camera Phone-Based Interaction. PERVASIVE 2005. In: 3rd International Conference on Pervasive Computing, Pervasive 2005. LNCS No. 3468, Springer-Verlag, Munich, Germany, May 2005

[SAW94] Schilit, B.; Adams N. and Want, R.: Context-aware computing applications. In Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pages 85-90, Santa Cruz, California, December 1994. IEEE Computer Society Press.

[SFB06] Universität Stuttgart, SFB 626: Nexus. Environmental models for mobile context-related systems. URL http://www.nexus.uni-stuttgart.de/

[Tr06] Tracy2 – the mobile agentsystem. URL http://www.mobile-agents.org/

An Approach of Semantic Cache for Mobile Devices to Enhance the Performance of Applications

Marcos F. Caetano[1], Marco Antonio Ribeiro Dantas[2]

[1] Computer Science Department (CIC)
[1] University of Brasilia (UnB), 70910-900, Brasilia, Brazil
[2] Department of Informatics and Statistic (INE)
[2] Federal University of Santa Catarina (UFSC), 88040-900, Florianopolis, Brazil
[1] caetano@cic.unb.br
[2] mario@inf.ufsc.br

Abstract: Wireless mobile computing has become a differential aspect to a large number of distributed applications. The main research goal related to this subject is to provide to applications a similar level of services found in structured networks. An example of interesting research topic is to improve the data replication scheme that exists in a mobile device. The use of an elaborate method can represent an improved strategy to enhance the performance and availability of applications to wireless users. In this article we present a semantic cache model study, and its implementation, as a differentiate policy for the local management of data replication. The approach assumes that an answer for a query can be satisfied totally by a local semantic segment, considering the gathering action of partial segments or helping the connection to a server and then receiving the answer. We tested the implementation of the model in a real environment, assuming three classes of queries. The first class was characterized by those queries that could be promptly answered locally. The second type of queries was answered after a local gathering operation of semantic segments. In the last situation, queries were answered after a send-receive operation to a server node in a structured network. The empirical results indicate that the approach has improved successfully the performance of the applications, avoiding unnecessary connections to a server when an answer could be reached using one local cache segment, or gathering information from the local semantic segments to compound the answer. In the case of inexistence of any local semantic segment information to reply a query, the implementation works transparently to connect to a server and then answer the wireless user.

1 Introduction

The fast growing rate of new mobile computing technologies has brought many benefits for wireless users. On the other hand, this fact promotes several research challenges to provide software services in a similar level found in structured networks.

Investigation efforts such as [CM05, HMM05, DM04] target to prove a performance enhancement of applications based upon the location of devices, efficiency of communication strategies between client and server, or the treatment of replicas and its reconciliation.

In contrast to the former researches initiatives, our research work has the goal to enhance the performance of mobile applications using a model of cache policy. The implementation of this paradigm was based on semantic cache. The main objective of this approach is to prove the right answer to queries using as much as possible the information stored at the local device. In other words, we first search for a complete or partial solution inside the mobile device. Therefore, the solution avoids unnecessary wireless communications with a server node. In the case that an answer could not be found correctly a communication is established to a specific node that will provide the answer for the query.

This paper presents the model that we have adopted and its implementation, considering real mobile devices in some case study configurations. Experimental results of the approach indicate that the proposed paradigm implementation has successfully reached the objective for a number of cases studies.

The article is organized as follows. In section 2 we present the fundamental theory related to semantic caches in mobile devices. In addition, in this section we also show some related work. The experimental environment employed for our case studies is illustrated in section 3. Finally, in section 4 we draw some conclusions related to the present research and point out some future work.

2 Semantic Cache

Semantic cache can be broadly defined as a collection of semantic segments [Sd96]. Each segment has a result of a query and its description, which was previously executed [AJ96]. Utilising the existing description it is possible to execute operations and verify whether a new query can be answered by this segment, or not.

Semantic cache has been used in centralised systems [CN94, Nr91], client-server environment[Sd96, AJ96], OLAP systems [Pd98], heterogeneous systems [PJ97] and also in mobile computing environments [KHA99, QM99].

When a query Q is submitted to a semantic cache approach, the entity responsible for the management of the cache verifies if a semantic segment of the entity can answer the query. This procedure is repeated for all queries, the semantic mechanism indicates if a result can be reached or not. In the case of a partial answer exists, the initial query is divided in two parts. The answered portion is associated to a segment and later is processed. On the other hand, the part that was not answered is submitted again for the next semantic segment.

The procedure is repeated until an answer is found to the second part inside the mobile device, or there is no more semantic segments to process the query. In this case, the second part of the divided query is submitted to a server node to be processed. It is expected that in the end, all parts are gathered and initial query Q answered.

It is interesting to mention that the new data that came from the server node will form a new semantic segment that can be used locally for future answers.

2.1 Cache Structure

Each semantic segment that forms a cache, it is defined by a tuple from five subsets[Qr00]:

$S = <S_R, S_A, S_P, S_C>$

Where, the above parameters, related to the *SQL* language, have the meaning:

- S_R – source relation that translates how the segment is form, thus it contains names of the tables declared in the section *from <table>*;
- S_A – it represents a set of attributes that are specified in the *select <atrib1,atrib2,....>* section;
- S_P – a set of predicates that are specific for the *where <pred1,pred2,...>* section;
- S_C – it represents a set of answers from segment *S*.

S	S_R	S_A	S_P	S_C	S_{ts}
S_1	Employee	{Id,Name, Age}	Age > 30	2	T_1
S_2	Employee	{Id,Name, Phone}	Age < 20	5	T_2

Table 1: An example of a semantic cache structure

Table 1 illustrates a structure model example that implements a cache segment. Segments S_1 and S_2 where created from the following queries:

- Q_1 *select* Id, Name, Age *from* Employee *where Age* > 30;
- Q_2 *select* Id, Name, Phone *from* Employee *where Age* < 20;

S_C represents a pointer that indicates the number of the page that contains the result of the semantic segment. On the other hand, S_{ts} is the representation of the *time stamp* that keeps the information time when the segment was created.

2.1 Query Partition

When a query is submitted to the local cache, a number of procedures will be realised targeting to solve this query. All the segments that form the cache can have a portion that together can answer the question. Therefore, a semantic segment *S* can contribute entirely, partially or not contribute at all to solve one question *Q*. Important to observe that even for the case that a small fraction of the question can be answered using a semantic segment this fact represents less wireless communication. In addition, if we avoid unnecessary communication we are also helping to provide to the mobile device a battery save feature.

However, in the case of partial answer the procedure will necessarily execute the division of the *Q* question into two queries:

- *Probe* – part of the query *Q* that can be answer by the semantic segment;
- *Remainder* – the present semantic segment could not answer this portion of the query *Q*.

Figure 1: Examples of queries submitted to a semantic segment *S*.

Figure 2 has a graphical illustration of five possible cases that a query can be classified. Each case can be understood using the examples shown in figure 1. Query Q_1 submitted to the *S* segment represents *case 1*, Q_2 translates the *case 2* and so forth. In [Qr00, QM03] it is possible to find a more formal description of these concepts.

The five cases presented in figure 1 can be understood as:

- *Totally Contained:* this represents the first case, when a query Q_1 is submitted to segment *S*. It is possible to verify $Q_P \Rightarrow S_P$ and $Q_A \subseteq S_A$. In other words, query Q_1 selects the same attributes of *S* (*Age,Name*) and the predicate Age > 38 is a subset that entirely exists inside the predicate Age > 24 from segment *S*. As a result, the answer for the query Q_1 is completely inserted in *S* segment;

- *Horizontally Partitioned*: When a query Q_2 is submitted to the segment S, it is possible to verify that $Q_A \subseteq S_A$ and $Q_P \wedge S_P$ satisfies the question. This fact can be understood because Q_2 selects the same attributes set from S, and the predicate *Age > 17* finds inside the segment results where values are greater than 24. In this case, the predicates from the query Q_2 are divided into two parts:
  - $Q'_2 = Q_p \wedge S_p$: a Q_2 subset that requires existing values inside S, i.e. values greater than 24;
  - $Q''_2 = Q_P \wedge \neg S_P$: a Q_2 subset that requires values that do not exist inside S,i.e. value between 18 and 24.

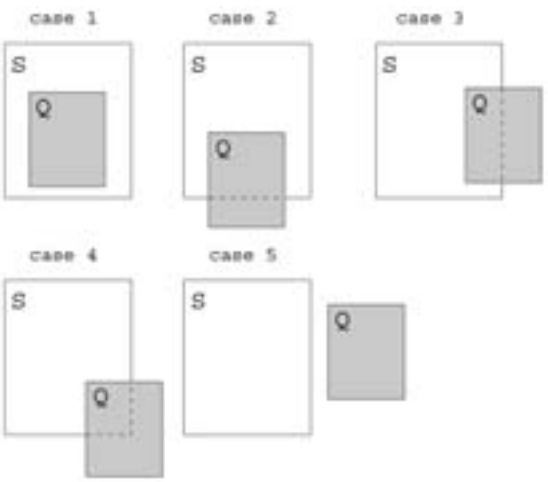


Figure 2: Possible partition semantic segment for query Q.

The Q'_2 query, called *probe query*, it will be entirely processed inside the segment *S.* In contrast, Q''_2, the query known as *remainder query*, will be redirected to other segment. In the case that no existing segment from the local cache could answer this query, it will sent to a server to be processed. Figure 3, presents the entire algorithm definition of the *query trimming*. In this section, we simplified the notation targeting to help the understanding of the idea.

- *Vertically Partitioned*: When query Q_3 is submitted to the segment S it is possible to verify that $Q_P \Rightarrow S_P$ e $Q_A \not\subseteq S_A$. Thus, the predicates from the query Q_3 are completely answered by the predicates from segment *S.* On the other hand, the attributes selected from Q_A (*Age, Phone*) do not entirely exist inside the segment attributes S (*Age, Name*). Therefore, the attributes

of the query Q_3 will be divided into two parts:

- $A_1 = (Q_A \cap S_A) \cup K_A$: compound by a subset of attributes that exists inside segment S (*Age*);

- $A_2 = (Q_A \cap \neg S_A) \cup K_A$; this part translates the attributes subset that does not exist inside the segment S (*Phone*).

The two subsets A_1 and A_2 will respectively form the *probe* and *remainder queries.* As we discussed in previous cases, the *probe query* will be executed locally. In contrast, the *remainder query* will be submitted to the other semantic segments. In the case that no local segment could match the query, it will be necessary a communication to a server to provide a answer. In the end, the subset will be gathered to solve the query Q_3. The concatenation of the subsets is done using the primary key from table *Employee.* Every query has the primary key of the manipulated table as an attribute. This attribute is defined as *included key segment* [QM03].

- *Hybrid Partitioned:* This approach can be considered the most complex, when it is compared to the previous cases. The reason for the complexity is the necessary utilisation of both *horizontal* and *vertical* partitions, respectively. When a query Q_4 is submitted to the segment S, it is possible to observe that:
  - $Q_p \wedge S_p$ can find a suitable answer. The predicate *Age > 17* finds inside the segment only values greater than 24. Therefore, values from the threshold 18 and 24 are not found*;*

  - $Q_A \not\subset S_A$ represents attributes selected from Q_A *(Age, Phone)* that are not completely inserted inside the segment *S (Age, Name);*

  - *Probe query*: a subset that represents values that exists in the segment S, i.e. $PQ = \pi\ (Q_A \cap S_A) \cap K_A \sigma\ Q_P \wedge S_P(Q_R)$. In other words, the attribute *Age* and the values of *Age > 24* are obtained in the segment.

- *Not Contained*: The query Q_5 can not be answered by segment S, i.e. $Q_P \wedge S_P$ is not satisfactory. Thus, the *probe query* is empty and the *remainder query* represents the query Q_5. This query will be processed by the next semantic segment until it can be solved. In the case that no answer is possible to be found from any segment, then the query is submitted to a server node.

The previous five described cases are organised in an algorithm structure presented in figure 3. The algorithm *Query Trimming* [QM03] receives as an entrance a semantic segment *S* and a query *Q*. As a result, the algorithm provides a structure that contains: a *type identifier* (that specifies which kind of partition was selected) and *sub-queries* (probe, remainder and amending query). An *amending query* is characterised by a query sent to a server node. In other words, this query represents the operation to find a segment that it is not present in memory of the mobile device.

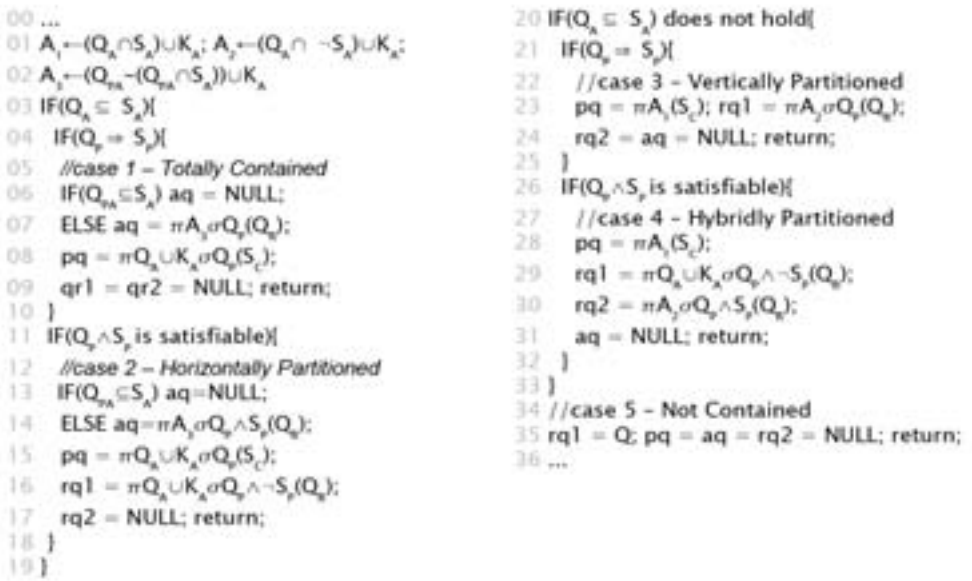


Figure 3: Algorithm Query Trimming.

In the following query structure we provide an example, where query *Q*' submits a search to segment *S*. A answer can not be found only using the local information available in the cache:

```
select Age, Name from Employee where Age > 30 and Salary > 300
```

It is possible to verify that the condition related to the employees, i.e. with Age > 30 and Salary > 300, can both be solve by the segment *S*, if the attributes exist in the local segment. However, if an attribute does not exist it will be impossible to solve the query. This example fits exactly in the case, because the local segment does not have the *Salary* attribute. Then, an *amending query* is created. This query submits to the server node a query to send to the mobile device the *Salary* attribute. When this attributes arrives at the local segment the query is processed and the result returned to the wireless user.

3 Configuration and Experimental Results

The environment considered in this article, it is a real configuration form by mobile devices that can have access to a structure local area network. We decided to utilise a real configuration, because we implemented the software portion of the server, the client and the communication module. Therefore, utilising this configuration it could provide a more realist assessment of some intrinsic problems found in wireless environments.

The design and development of the semantic cache model, presented in the theory in section 2, represents the main part of our contribution. As figure 4 shows, we image an environment that was formed by three elements: a database manager, a server and a client wireless node.
The database manager used in this research project was the Mckoi SQL [Md06]. We decided to use this database, because it is an open source package and provides all the functions necessary to the development of the proposed software environment.

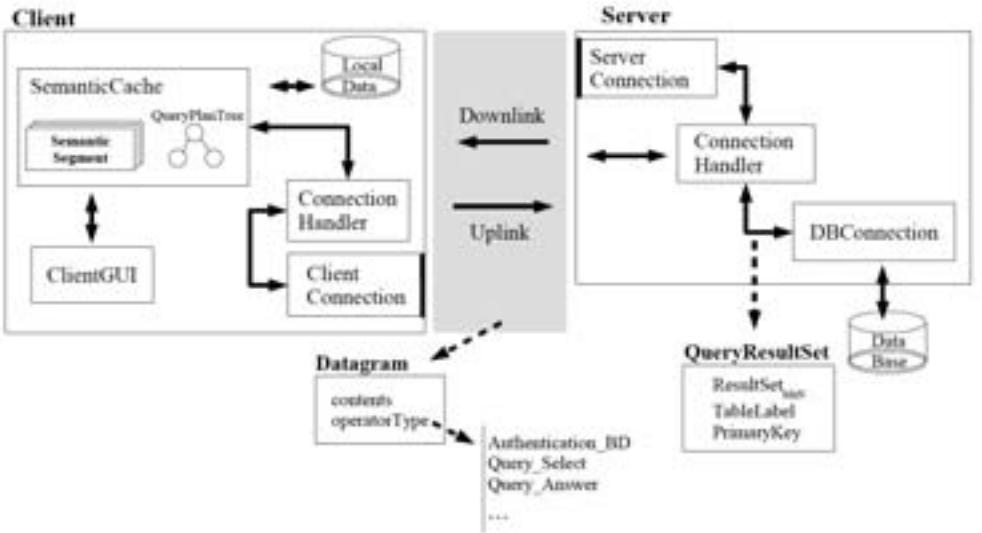


Figure 4: Proposed Environment.

The server node module was characterised for waiting and treating client connections and requests. This element was implemented having three software components:

- *Server Connection*: this element waits for the calls from wireless clients in a specific port and then redirects to the *Connection Handler* module to be treated;
- *Connection Handler*: this module is responsible for receiving calls from client nodes and establishes connections with the DBConnection. It was implemented adopting the multithread approach and has an application communication protocol developed to provide some functions such as:

authentication in the database, execution of queries and database re-initialisation.

- *DBConnection*: this element interfaces the communication between the server node and the database manager. The implementation was developed to provide to the server independence from the used database. Therefore, this module returns the results of all SQL queries as *labels*, indicating a line and a column and the primary key of the table. In other words, this software element was designed adopting a *QueryResultSet* structure.

On the client side, the enhanced semantic cache functionality was designed and implemented having the following components:

- *Client Connection***:** this element is responsible for establishing and closing a connection, sending and receiving requests in the client connections. These connections were characterised by *sockets* and this module provides two streams channels: an input and an output. Theses channels are managed and encapsulated through the functions:

```
public void send(Datagram msg) {...}
public Datagram receive() {...}
```

- *Connection Handler***:** this entity is responsible for treating the client connections. Thus, it provides a transparent interface, called as *Semantic Cache*, that encapsulates procedures to establish connection with a server and data structure;
- *Semantic Cache*: this module is the implementation of semantic cache policies that we discussed early. Therefore, we developed for this component some facilities such as: semantic segment, binary tree, QueryPlanTree, and PredicateSet. In addition, it also provides the entity *ClientGUI* that has a transparency feature related to the semantic cache policy. Thus, for an application that uses this interface, the processing procedure for local and remote queries are transparently. An example is:

```
public synchronized QueryResultSet
        executeQuerySelect( String queryConsult)
        {…}
```

- *ClientGUI:* this component of the proposed environment provides a user friendly interface to a user to have access to the facilities of the software package implemented;
- *Local Data:* the persistent local cache is store in this element. When, for example, the environment starts it has a number of segments that were generated in the last utilization.

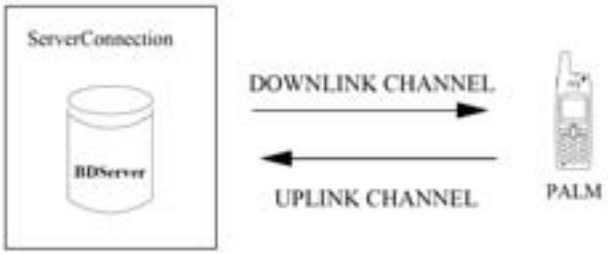


Figure 5: A general communication between client and server.

Figure 5 presents a general picture of the client and server communication. The *uplink* channel is used by clients to realize access and queries to a server node. This channel is also used to verify if enough energy exists and if the client is inside a wireless area from a server. The *downlink* channel is a conventional link where servers send their answers to clients.

Table 2 shows characteristics of the software and hardware components that were used to our empirical tests.

Component	Mobile Client	Server
Model	Palm Tungsten C	AMD Duron
Processador	400MHz	1,2GHz
Memory	64MB	256MB
OS	Palm OS 5.2.1	GNU-Linux-2.4
JVM	MIDP 2.0 - CLDC	Sun J2SE-1.4.2

Table 2: Environment Characteristics

3.1 Empirical Experiments

The experiments that we show in this section were realized to test all the existing components of the environment, shown in figure 4. In other words, we image a set of tests that could verify the interaction between the software elements and certify that the answers were expected results.

We started the mobile device with any information inside the local cache. After that we submitted the following queries:

1. Select Id, Name, Age from Employee where Age > 35;
2. Select Id, Name, Phone from Employee where Age < 40.

This first query was entirely submitted to the server and after that the information was added to a segment S_1. The second query, shown in figure 6, when submitted to the first segment S_1 generated a *hybrid partitioned*. The sub-queries were:

1. Select Id, Name from Employee where Age > 35;

2. Select Id, Phone from Employee where Age > 35 and Age < 40;
3. Select Id, Name, Phone from Employee where Age <= 35.

The first sub-query can be answer by the segment S_1, whereas the other two were sent to the server node. An example of the server behavior is illustrated in figure 7. This figure shows a server text interface informing the result sent the client. This message refers to the third sub-query.

The final answer of the second query Q is present in figure 8. This answer indicates that the designed and implementation reach successfully the objective of the wireless user. The research procedure was efficient, because it only collects the necessary data from the server node. In other words, avoiding unnecessary wireless connection to the server and saving resources once it does not considered the store of data that is already in a semantic cache segment.

On the other hand, figure 9 demonstrates the final content of the local semantic cache. After the *hybrid partitioned* operation two new semantic segments were created, S_1 *and* S_2. It is expected that for a future similar request, the mobile device could answer the query more efficiently, because the data now exists locally.

4 Conclusions and Future Work

The semantic cache is not a novel mechanism. However, this approach employed in small equipment, such as mobile devices, is an interesting research topic to enhance wireless applications. In this article we have presented a model and implementation of an environment to enhance the performance of mobile applications. We first described theoretically how the semantic segments work. After that we presented our differential contribution designing, developing and implementing a solution to receive queries from a local mobile device and then process efficiently these requests.

The proposed environment was compound by three software modules: a server, a client and a database interface manager. The server was implementing having entities such as a connection link element, a handler and a database connection. The client part representing the vital point of the contribution, it was characterised by having a connection server software package, a user friendly graphical interface and the semantic cache module. Finally, the database interface manager provides an open solution to the environment to work independently from a specific database or operating system. Empirical results indicate that we have successfully reached a contribution to improve the performance of local wireless applications.

As a future research work, we are planning to add locality dependence facility to the software package implemented and also consider *ad hoc* networks to form an environment to provide answers for mobile users.

Figure 6: The second query from the mobile device.

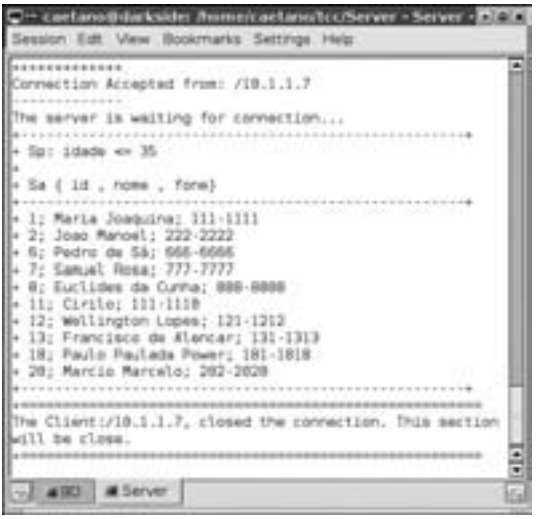


Figure 7: Server Text Interface.

Figure 8: Answer of the second query.

Figure 9: Semantic segment after the second query.

References

[AJ96] A. M. Keller and J. Basu. A predicate-based caching scheme for client-server database architectures. Parallel and Distributed Information Systems, 1994., Proceedings of the Third International Conference on, 5(2):35–47, September 1996.

[CM05] C.D.M. Berkenbrock and M. A.R. Dantas. Investigation of cache coherence strategies in a mobile Client/Server environment. *International Conference on Computational Science*, (3):987–990, 2005.

[CN94] C. M. Chen and N. Roussopoulos. The implementation and performance evaluation of the adms query optimizer: Integrating query result caching and matching. *In Proceedings of EDBT*, pages 323–336, march 1994.

[DM04] D. Pezzi and M.A R. Dantas. An experimental case study of replication and reconciliation in a wireless environment. Proceedings of The 18th International Symposium on High Performance, pages 179–182, 2004.

[HMM05] H. Monica, M. S. de Camargo, and M. A. R. Dantas. An architecture for location-dependent semantic cache management. ICEIS, (1):320–325, 2005.

[KHA99] K. C. K. Lee, H. V. Leong, and A. Si. Semantic query caching in a mobile environment. Mobile Computing and Communication Review, 3(2):28–36, 1999.

[Md06] Mckoi Database, http://www.mckoi.com/database/, available in April, 2006.

[Nr91] N. Roussopoulos. An incremental access method for viewcache: Concept, algorithms and cost analysis. ACM Transactions on Database Systems, 16(3):535–563, September 1991.

[Pd98] P. Deshpande, K. Ramasamy, A. Shukla, and J. F. Naughton. Caching multidimensional queries using chunks. *In Proceedings of ACM SIGMOD*, pages 259–270, june 1998.

[PJ97] P. Godfrey and J. Gryz. Semantic caching in heterogeneous databases. In Proceedings of DEXA Workshop, pages 414–419, August 1997.

[QM99] Q. Ren and M. Dunham. Using clustering for effective management of a semantic cache in mobile computing. In Proceedings of the International Workshop of MobiDE, pages 94–101, August 1999.

[QM03] Q. Ren, M. Dunham, and V. Kumar. Semantic caching and query processing. Knowledge and Data Engineering, IEEE Transactions on, 15(1):192–210, Jan.-Feb. 2003.

[Qr00] Q. Ren. Semantic caching in mobile computing. PhD thesis, Southern Methodist University, February 2000.

[Sd96] S. Dar, M. J. Franklin, B. T. J´onsson, D. Srivastava, and M. Tan. Semantic data caching and replacement. *Proceeding of the* 22*TH International Conference on Very Large Data Bases, Mumbai (Bombay), India*, pages 330–341, 1996.

Chapter 2: Decentralized Network Systems

Contributions to 10th I2CS 2010, Bangkok, Thailand

Christian Spielvogel, Peter Kropf
Application Layer Scalable Video Coding for the iPhone

Lada-On Lertsuwanakul
Fuzzy Logic Based Routing in Grid Overlay Network

Miguel Angel Rojas González
Performance Evaluation of two Self-Adaptive Routing Algorithms in Mesh Networks

Oleksandr Kuzomin, Illya Klymov
Functional Approach to Decentralized Search Engine for P2P-Network Communities

Application Layer Scalable Video Coding for the iPhone

Christian Spielvogel, Peter Kropf

University of Neuchâtel
Neuchâtel, Switzerland
{christian.spielvogel,peter.kropf}@unine.ch

Abstract: Videos streamed over the Internet can be received by mobile devices anywhere and anytime. The challenges for Internet streaming include avoiding data loss and playing back the content in the desired quality. A technique for achieving these goals is Scalable Video Coding (H.264-SVC) that adapts the content to network and device characteristics without transcoding. Since there is no player for rendering scalable video content on mobile phones using more than the base layer, we present our prototype implementation that is able to decode multiple layers of scalable content in the temporal, spatial or quality domain. Currently the prototype is available and has been tested on the iPhone 3G.

1 Introduction

Scalable Video Coding (H-264 SVC) [SMW07], the latest extension to the MPEG-4 Advanced Video Coding standard, enables content adaptation by removing parts of the encoded stream so that the sub-streams result in differently scaled variations. The availability of different variations is important when delivering the original quality is not possible, like in case of legacy devices not having enough resources or in lossy wireless network environments.
The variations can be produced in the temporal, spatial or fidelity domain. The temporal scalability is applied to reduce the frame rate, spatial scalability results in a reduced picture size and fidelity scalability decreases the signal-to-noise (SNR) ratio. A further but not very frequently used scaling method is the region-of-interest (ROI) scalability, where the sub-streams typically represent spatially contiguous regions of the original picture area.
While Scalable Video Coding features much flexibility to deliver best possible quality, to our best knowledge all video players for mobile devices support at most the decoding of the base layer, which is by definition H-264 compatible, but none of them is able to decode multiple layers. Being able to handle multiple temporal layers can be necessary to provide heterogeneous mobile devices with different frame rates while not limiting the decoded stream to the rate of the base layer. For example, devices with sufficient resources (like the iPhone) could get the full frame rate while devices with limited resources (older mobile devices) should receive a limited number of layers resulting in a lower frame rate (e.g. 12 frames per second). Another example stressing the necessity of being able to decode multiple layers in the spatial domain concerns the support of mobile devices with differ-

ent resolutions. For example a smart phone with a resolution of 160x240 pixels would receive only one layer an iPhone with a resolution of 320x480 pixels would get additional enhancement layers to render the video without using interpolation. Furthermore, we identify the necessity of being able to use multiple layers with respect to graceful degradation in the fidelity domain. Lets assume that the video is encoded in two layers, where the lossless transmission of the base layer requires 400 Kbit/s and the combined transmission of the base layer and enhancement layer requires 1000/s. Given that the network allows lossless transmission of both layers, only being able to decode the base layer on the mobile device - and discard the enhancement layer because of the inability to decode multiple layers - would result in a massive quality loss.
Section 2 gives and overview about Scalable Video Coding (SVC) in general, Section 3 presents current and future technologies for streaming media content to mobile devices, Section 4 describes our application layer approach for decoding multiple SVC layers on the iPhone, Section 5 presents the gain from using multiple Scalable Video Coding Layers instead of only a single one on a mobile device and Section 6 summarizes the presented work.

2 Background on Scalable Video Coding

Scalable Video Coding (SVC) produces streams consisting of multiple encoded sub-streams called layers. The first Layer is the base layer, all other layers are called enhancement layers and are numbered in ascending order. The architecture of a scalable video stream can be found in Figure 1.

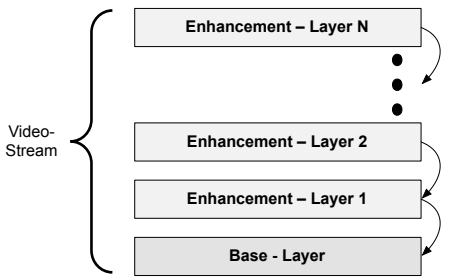


Figure 1: Scalable Video Coding Architecture

Decoding different subsets of the layers can result in N permutations. Decoding all layers together results in the best quality (see Figure2), decoding only the base layer results in the base quality (see Figure3).

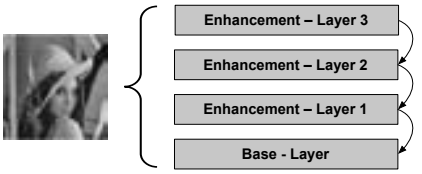


Figure 2: All layers are decoded

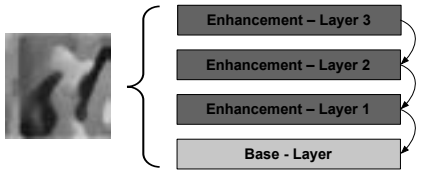


Figure 3: Only base layer is decoded

3 Related Work

At the moment mobile video services are largely based on H.263 and MPEG-4, but according to [SSW07] the availability of H.264/AVC capable terminals will increase over the next years, making the features of the new standard ubiquitous. Some examples for the emerging use of SVC in mobile video transmission can be found in [GLS06] [HHL08] [SGS+07]. A further indication for the upcoming availability of SVC, is the 3 GPP recommendation of using the H.264/AVC baseline profile for all mobile services, including packet-switched streaming services [Cur03], messaging services, and multimedia broadcast/multicast services [AK08]. Beside this, mobile broadcast services such as DVB-H [Wea07] and DAB [LFG+06] will also rely on the new standard.

4 Application Level Scalable Video Coding with multiple Layers

Contrary to the low level Scalable Video Coding implementation [SMW07] our approach has been implemented at the application layer and allows the transformation of MPEG-1,2,4, H-264 and QuickTime content into our proprietary scalable format. As illustrated in Figure 4, a video stream is received by our video proxy and transformed into multiple layers using the H-264 codec. In order to abstract the various video formats (MPEG-1,2,4, H-264 or QuickTime) we have implemented a codec-abstraction layer. The support of that many input video formats has become possible by using a combination of the X264 [Rao06] and the FFMPEG library [Tom06].
In the current implementation every stream which arrives at the proxy is transformed into the H-264 format producing one base layer and one enhancement layer in the temporal, spatial and fidelity domain. Once the transformation from the single stream to the layered format is accomplished, the content is forwarded to the mobile devices. On the mobile device the layers are processed by the scalability-abstraction layer. The scalability-abstraction handles the decoding of all H-264 layers into the RGB format as well as the

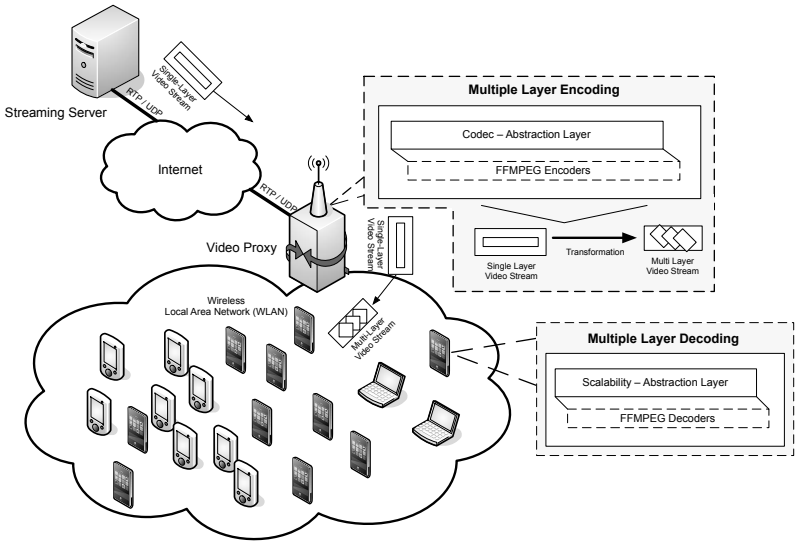


Figure 4: Scalable Video Coding at the Application Layer

merging and forwarding to the media player. For being able to decode the H-264 layers on the iPhone we have ported the X264 codec for the ARM architecture [Jag97].
The functionality of the codec-abstraction layer for encoding the layers in the temporal, spatial and fidelity domain as well as the decoding by the scalability-abstraction layer is explained in the following sections 4.1 to 4.3.

4.1 Temporal Scalability

The block diagram in Figure 5 shows our mechanism for producing one base and one enhancement layer in the temporal domain. The base and enhancement layers are created in 6 steps. The steps 1 and 6 are identical for both of them, whereas steps 2-5 are only required to produce the enhancement layer.

1. The raw video is temporally down-sampled, then transformed using the discrete cosine transform (DCT) and finally quantized (Q). Temporal down sampling is achieved by skipping a predefined number of frames. For example a down-sampling ratio of 2:1 is achieved by skipping every second frame from the input stream.

2. In order to produce the enhancement layer each frame is reconstructed by inverse quantization and the inverse discrete cosine transform (IDCT).

3. The input for the enhancement layer is temporally up-sampled to the original frame rate. Temporal up sampling is achieved by replicating frames. In the example of the

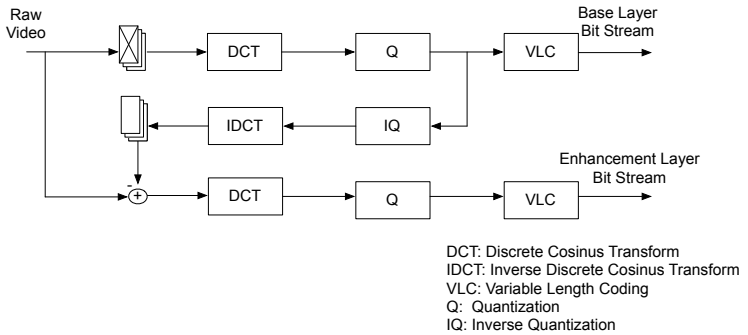


Figure 5: Block diagram Scalable Temporal Encoder

1:2 sampling rate, every frame is replicated.

4. The difference between the reconstructed (replicated) and the original frames is calculated. This difference is known as the residual, and stored in the enhancement layer.

5. The residual is transformed using the discrete cosine transformation (DCT) and quantization (Q).

6. The quantized coefficients of the (1) base- and (2) enhancement layers are encoded using variable length coding (VLC).

To decode the base and enhancement layers variable length coding (VLD), inverse quantization (IQ) and inverse discrete cosine transformation (IDCT) have to be applied (see Figure6). Afterwards, the final output is produced by up sampling the base layer and merging it with the enhancement layer.

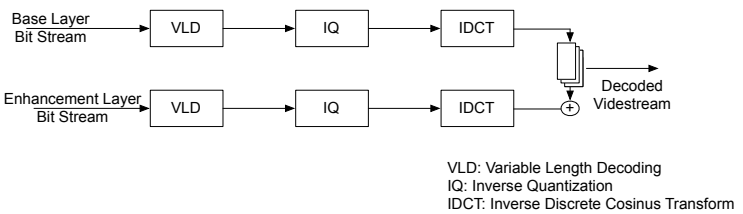


Figure 6: Block diagram Scalable Temporal Decoder

4.2 Spatial Scalability

A block diagram for encoding one base and one enhancement layer in the spatial domain can be found in Figure 7. Like for temporal scalability (see Section 4.1) the base layer and

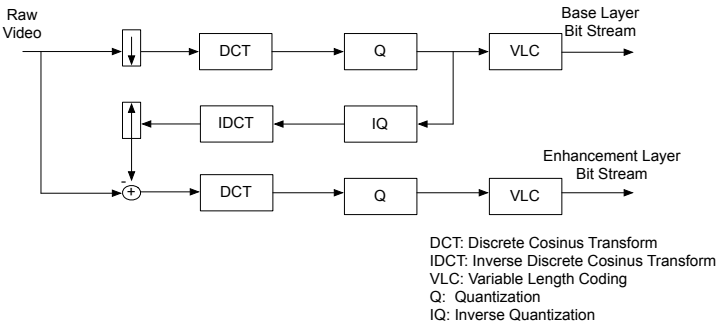


Figure 7: Block diagram Scalable Spatial Encoder

the enhancement layer are created in 6 steps, having in common steps 1 and 6 but applying steps 2 to 5 only to the enhancement layer.

1. The raw video is spatially down-sampled, transformed using discrete cosinus transform (DCT) and quantized to get the input for both layers.

2. The enhancement layer is produced by reconstructing each frame using inverse quantization (IQ) and the inverse discrete cosine transform (IDCT).

3. Each enhancement layer frame is spatially up-sampled to the original size using interpolation.

4. The up-sampled frames are XOR-ed with the corresponding original image, the resulting frame is known as the residual.

5. The residual is transformed using the discrete cosine transformation (DCT) and quantized (Q).

6. The coefficients from the base- and enhancement layer are encoded using variable length coding (VLC).

For decoding the two layers Variable Length Coding (VLC), Inverse Quantization (IQ) and Inverse Discrete Cosine Transform (IDCT) have to be applied (see Figure8).

After applying IDCT to both layers the frames are spatially up-sampled by XORing the base with the enhancement layer.

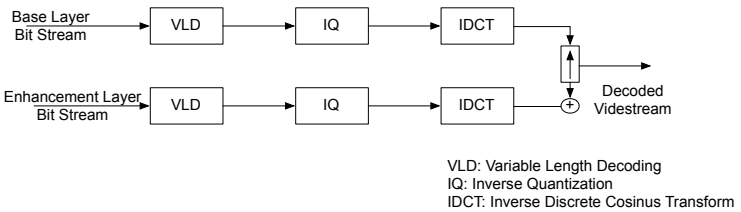


Figure 8: Block diagram Scalable Spatial Decoder

4.3 Fidelity Scalability

A block diagram for our application layer based production one base layer and one enhancement layer in the fidelity domain can be found in Figure 9.

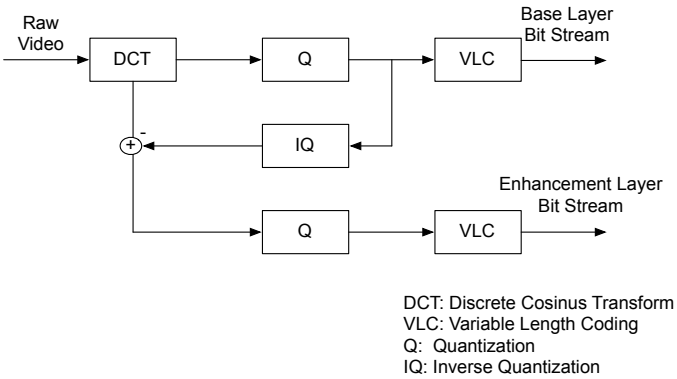


Figure 9: Block diagram Scalable Fidelity Encoder

Like for temporal and spatial scalability the base layer and the enhancement layer are created in 6 steps, having in common steps 1 and 6 but applying steps 2 to 5 only to the enhancement layer.

1. The luminance and chrominance values of the raw video are down-sampled, transformed using discrete cosinus transform (DCT) and quantized to get the input for both layers.
2. For producing the enhancement layer, each frame is reconstructed using inverse quantization (IQ) and the inverse discrete cosine transform (IDCT).
3. The luminance and chrominance components of each enhancement layer frame are up-sampled to the original quality.

4. Each up-sampled frame is XOR-ed with the corresponding original image. This difference is known as the residual.

5. The residual is transformed using the discrete cosine transformation (DCT) and quantized (Q).

6. The coefficients from the base- and enhancement layer are encoded using variable length coding (VLC).

For decoding the two layers Variable Length Coding (VLC), Inverse Quantization (IQ) and Inverse Discrete Cosine Transform (IDCT) have to be applied (see Figure10).

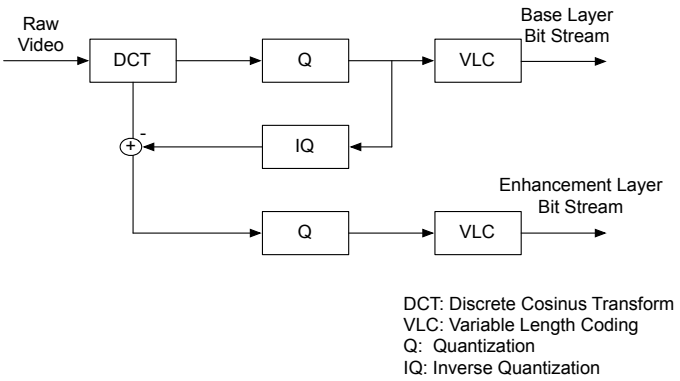


Figure 10: Block diagram Scalable Fidelity Decoder

After applying IDCT to both layers the frames are spatially up-sampled by XORing the base with the enhancement layer.

5 Evaluation

In our evaluation section we show the quality improvement obtained from our new approach where we are able to use the base layer in combination with the enhancement layer on the iPhone compared to only using the base layer. The proxy for producing the scalable stream is a standard linux workstation with a 2.13 GHz dual core processor. The decoding and rendering of the layers is performed on an iPhone 3G. The quality comparison is based on the peak signal to noise ratio (PSNR) [HCS01] calculation which is an automated standard method for evaluating video quality. The evaluation is done by emulating the quality impression of the human visual system (HVS) [JPH05] based on the following equation.

$$PSNR(n,k)_{dB} = 20log_{10}\left(\frac{2^k-1}{\sqrt{\frac{1}{N_{col}*N_{row}}\sum_{i=1}^{N_{col}}\sum_{j=1}^{N_{row}}(P_B(n,i,j)-P_{BE}(n,i,j))^2}}\right)$$

We have evaluated the quality of 18 randomly selected video sequences belonging to various categories ranging from sports to entertainment and news videos. For evaluating temporal scalability the base layer was encoded with 12 frames/second, and the enhancement layer with 13 frames/second. Spatial scalability was evaluated by producing a base layer with a resolution of 160x240 pixels and an enhancement layer increasing the resolution to 320x240 pixels. Depending on the different motion intensity and color distribution, the minimal, average and maximal gain from our implementation has been measured and is presented in Table 1.

	Quality Gain in the Temporal Domain	Quality Gain in the Spatial Domain
Min	23.6 %	6.8 %
Mean	41.58 %	23.34 %
Max	62.6 %	70.4 %

Table 1: Evaluation Results

Our comparison shows that the quality improvement for the tested videos, using the enhancement layer in the temporal domain, lies between 23.6 % and 62.6 % and the quality improvement from using the enhancement layer in the spatial domain is between 6.8 % and 70.4 %.

6 Conclusion

In this paper we have summarized the challenges of video streaming over wireless best effort networks. We have presented Scalable Video Coding (H.264-SVC) as the state of the art solution for handling requirements of heterogeneous mobile devices. Since there is no video player available for rendering scalable video content on mobile phones using multiple layers, we have presented our prototype implementation which is able to do so. Currently we are able to decode the base layer plus one enhancement layer in real-time, which allows to significantly improve the visual quality (see Section 5). A more sophisticated solution concerning the number of the decoded layers will be developed at a later stage of the project. Having more layers offers the possibility to serve a larger variety of devices and better possibilities of reacting to bandwidth fluctuations, resulting in higher end-user satisfaction. Increasing the number of enhancement layers will be realized by applying the existing algorithms recursively.

References

[AK08] Kari Aho and Janne Kurjenniemi. Multimedia Broadcast Multicast Service Performance and its Enhancements in WCDMA Networks. *Wirel. Pers. Commun.*, 46(2):115–142, 2008.

[Cur03] Igor D. D. Curcio. Multimedia streaming over mobile networks: European perspective. pages 77–104, 2003.

[GLS06] T. Wiegand G. Liebl, T. Schierl and T. Stockhammer. Advanced wireless multiuser video streaming using the scalable video coding extensions of H.264/MPEG4-AVC. *in Proc. IEEE Int. Conf. Multimedia Expo (ICME), Toronto, ON, Canada*, pages 625 – 628, 2006.

[HCS01] Lajos Hanzo, Peter Cherriman, and Jurgen Streit. Wireless Video Communications: Second to Third Generation and Beyond. 2001.

[HHL08] Mohamed Hefeeda, Cheng-Hsin Hsu, and Yi Liu. Testbed and experiments for mobile TV (DVB-H) networks. pages 995–996, 2008.

[Jag97] D. Jaggar. Arm Architecture And Systems. *IEEE Micro*, 1997.

[JPH05] Satu Jumisko-Pyykkö and Jukka Häkkinen. Evaluation of subjective video quality of mobile devices. pages 535–538, 2005.

[LFG+06] P. J. Lobo, M. A. Freire, M. J. Garrido, C. Sanz, F. Pescador, and D. Samper. The Prototyping Methodology of a Data Receiver for Digital Audio Broadcasting (DAB) Networks. pages 76–81, 2006.

[Rao06] R.S.V. Chandra D.J. Narayanan S. Rao, G.N. Prasad. Real-Time Software Implementation of H.264 Baseline Profile Video Encoder for Mobile and Handheld Devices. *IAcoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, 5:457 – 460, 2006.

[SGS+07] Thomas Stockhammer, Tiago Gasiba, Wissam Abdel Samad, Thomas Schierl, Hrvoje Jenkac, Thomas Wiegand, and Wen Xu. Nested harmonic broadcasting for scalable video over mobile datacast channels: Research Articles. *Wirel. Commun. Mob. Comput.*, 7(2):235–256, 2007.

[SMW07] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, September 2007.

[SSW07] T. Schierl, T. Stockhammer, and T. Wiegand. Mobile Video Transmission Using Scalable Video Coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(9):1204–1217, September 2007.

[Tom06] S. Tomar. *Converting video formats with FFmpeg*, volume 146. Linux Journal, 2006.

[Wea07] Ye Wang and et. al. Mobile video applications and standards. pages 1–6, 2007.

Fuzzy Logic Based Routing in Grid Overlay Network

Lada-On Lertsuwanakul

Faculty of Mathematics and Computer Science
FernUniversität in Hagen
Universitätsstraße 27 - PRG
58084 Hagen
lada-on.lertsuwanakul@fernuni-hagen.de

Abstract: With the aim to improve the quality of service of the modern distributed application, we propose a multi-criteria routing algorithm running in a mesh structured overlay network. Using a grid pattern can improve routing remarkably, since it provides alternative and partly disjunctive paths of equal length as well as the ability to measure distances between nodes in the overlay network. A Thermal Field approach is used to represent buffer stages on the nodes. The decision-making algorithms use fuzzy logic techniques to select the optimal path considering multiple constraints. The proposed algorithm is evaluated using P2PNetSim, a network simulation tool. The approach is compared to Shortest Path routing and probability functions using deterministic or adaptive approach. The result of routing with fuzzy logic shows superior routing performance than others both in delivery ratio and routing time.

1 Introduction

Quality of Service Routing (QoSR) is a key function for the transmission and distribution of digitized information across networks. It has two main objectives; finding routes that satisfy the QoSR constraints and making efficient resource utilization. Unfortunately, several factors can cause poor performance. So many problems still exist such as data loss because of overloaded incoming and outgoing message buffers, packet delay or expiry when residing in large queue or when using unsuitable routes. The complexity in QoS routing comes from multiple criteria, which often make the routing problem intractable. Typical criteria are node buffer capacities, residual link capacities, and number of hops on the path. Many routing algorithms [GRS05] have been developed in this research area. Expert systems, swarm intelligent systems, artificial neural networks, and fuzzy logic are applied for multi-constraint decision making. Many approaches focus on bandwidth and throughput optimizations.

We considered buffer utilization for improving robustness and to implement an efficient load balancing that is only based on local knowledge of the nodes involved. A fuzzy system was chosen because it provides a mathematical model for dealing with imprecision and uncertainty as given in common traffic situations in today's communication networks.

The overlay network approaches [Lu04] aim to support various features such as robust routing architectures, efficient searching and routing, load-balancing, distributed implementation of trust and authentication, and redundant storage, emerged in Peer-to-Peer (P2P) network. Structured P2P networks, such as CAN, Chord, and Pastry, provide Distributed Hash Tables (DHT) to identify relationships among nodes and files for searching and routing control. Whereas unstructured networks, ad-hoc systems, organize peers in a random graph and use flooding or random search on the graph to find the desired contents. In comparison, structured P2P networks can efficiently locate rare data since the key-based routing in scalable, but they incur significantly higher overhead for popular content.

The grid pattern with coordinate system provides many benefits for the routing process, because many paths with the same hop-count exist between two peers and they enable each peer to predict the shortest route without prior communication. Grid can be used to provide content-based coordinate systems generated from the distributed system's contents. The generated map can be changed dynamically according to overlaying application's requirements. The distance between peers is measured in Euclidean space. In [Be07] proposes routing in a mesh-like structure using the EPC code to establish an address space. Moreover, other applications such as Network Virtual Environments and Data mining application could be able to apply our adaptive routing approach on their content-grid structure.

The proposed algorithm takes the distance from the current node to the destination into account as well as the buffer usage level of each node's direct neighbors. The distance is measured by Euclidean space. To propagate the buffer levels in a node's neighborhood, a thermal-field-based approach is used. The locally executed decision-making process is based on fuzzy logic.

The rest of this paper is organized as follows: Section 2 provides related literature briefly and extensively description of classical routing methods on meshes, thermal field approaches, and fuzzy systems. Section 3 introduces our routing strategy considering multiple factors using fuzzy logic. Section 4 describes the simulation environment in P2PNetSim simulation tool, as well as the results and discussions. Finally, Section 5 concludes the paper and gives an outlook for future research.

2 Related Work

Many adaptive routing algorithms considering multi-constraint to improve QoSR have been introduced in before. Zhang and Zhu [ZZ05] introduced an algorithm considering number of hops and available buffer-capacities in general communication networks. FLAR [MTR07] and FACO [GDT09] describe routing algorithms applying ant colony systems and fuzzy logic to consider multiple constraints in Mobile Ad Hoc Network (MANET). FLAR considered route utilization and route delay but FACO considered buffer occupancy, remaining battery power and signal scalability. A fuzzy mixed metric approach, introduced in [US08], is used to make optimal routing decisions in packet switching network by considering one or multiple QoS factors.

The introduced routing approach is run on a mesh overlay network, in which the distance is measured by coordinate system. One of the well-known routing algorithms in grid-like networks was introduced by Jon Kleinberg [Kl00]. He introduced a decentralized algorithm in grid, where he added long-range links to forward messages from any source node to target within only $O(log^2 n)$ delivery time complexity. Meanwhile the probability proportional of long-range random links between nodes v and w is $d(v,w)^{-2}$. Some networks with coordinate systems are built in the lower layer of the network stack, e.g. [DGK09], but also approaches for building grid in the application layer exist, as in [BSU09, Be07]. In [BSU09], a grid structure is generated on top of a large-scale decentralized network. Their logical grid is built without centralized control and global instances; only local knowledge of each node is needed.

2.1 Routing in Grid-like Structure

Grid or mesh patterns have been used in many areas of communication networks. The advantage of mesh structures is their reliability and inherent redundancy of the connection. Many applications using grid structure are implemented in packet/circuit switching both in wireless networks and wired networks, vehicle problems, and software interaction [CL92, JVM95, LW04, RR91]. Some examples of routing algorithms in mesh-connected topologies are presented in [Me04]. A deterministic routing method in grid is called "XY" routing algorithm where packets are routed along X direction until reaching the X value of the target and then route the packet in Y direction to the target. This kind of routing can be refined, named the partial-adaptive routing algorithms; "West-First", "North-Last", and "Negative-First" approaches. These methods change packet routes dynamically by using a function that reacts immediately on network traffic, but in some specific conditions they use the deterministic ways.

All these classical routing methods have in common that they choose between multiple paths having the same hop count. The source node wants to send information to the target node. Then the best path depending on the algorithm is selected. The chance to find low QoS relies on the path selection function. If the selected route has many overloaded peers, then the delay time increases or the packet losses occur.

2.2 Thermal Field

Unger and Wulff [UW04] were introduced the Thermal Field approach which is used for searching nodes in P2P networks that keep the desired data; like a very frequently accessed data or recently update information. The temperature implies the intensity of the activities or changes of specific information in the node. Further, when a high temperature point occurs in the community, its heat spreads around. The spreading temperature decreases by distance between heat source and measurement point, also by distribution time. Finally, a point becomes colder if there is no heat fed in. The thermal approach can be applied to a P2P environment when the assumption is that members of the community cooperate with each others, and all peers contribute for community results. Whenever there is a message sent among members of the community, it means the heat is transported from source to neighbour.

However, there is a difference from nature that the virtual community is able to memorize temperatures from latest access of each neighbour. Consequently, when a message requests for special information, it can be transferred to the "hottest" neighbour that is kept in memory.

The Thermal Field approach is also used in routing in overlay network representing the level of buffer usage. In [LU09a], route decision is chosen either an adaptive or a directed route depending on a global predefined probability. This decision immediately reflects the buffer level of the considered node's neighbour at decision time. And in [LU09b], five adaptive probability functions of relative remaining distance were used to further improve the routing decision. It extended prior algorithm to avoid the global parameters. These approaches can gather remarkable advantages from the underlying grid-like structure, which offers a lot of alternative routing paths for a single message.

2.3 Fuzzy Logic

Fuzzy Logic introduced by Zadeh [Za65] allows a computer to model the same way that people do, not always precise. People think and reason using linguistic terms such as "hot" and "fast", rather than in precise numerical terms "90 degrees" and "200 km/hours" respectively. The fuzzy set theory models the interpretation of imprecise and incomplete sensory information as perceived by human brain. Thus, it represents and numerically manipulates such linguistic information in a natural way via membership functions and fuzzy rules. Some advantages of fuzzy logic are conceptually easy to understand, flexible, and tolerant of imprecise data. It can model nonlinear functions of complexity, and also can be built on top of the experience of experts.

A key feature of Fuzzy logic is to handle uncertainties and non-linearity, existing in physical systems, similarly to the reasoning conducted by human beings, which makes it very attractive for decision making systems. A fuzzy logic system comprises basically three elements: (i) Fuzzification, (ii) Knowledge base (rule and function), and (iii) Defuzzification. In Fig. 1 shows the generalized blog diagram of fuzzy system.

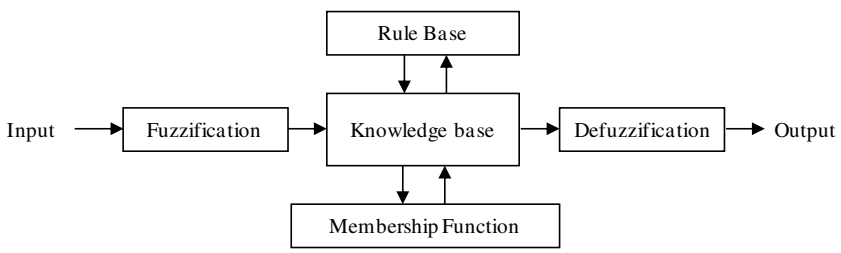


Fig. 1: A block diagram of a generalized fuzzy system

The function of the fuzzification is to determine the degree of membership to a crisp input in a fuzzy set. The fuzzy rule base is used to present the fuzzy relationship between input-output fuzzy variables. The output of the fuzzy rule base is determined based on the degree of membership specified by the fuzzifier. The defuzzification is used to convert outputs to the fuzzy rule base into crisp values.

In Section 2, we presented the classical routing in mesh, the thermal field approach and fundamental fuzzy logic technique. The next section, we will explain how the fuzzy logic works to find an optimal routes by considering relative distances in grid and buffer usage level.

3 Algorithms

In this section we describe our algorithm in details. The route decision process is constructed with the communication model transmitted temperature value by agents and use fuzzy logic to find the best path. Our approach considers decision position in terms of distance relationship and buffer usage status concurrently. The distance between nodes is measured by Euclidean distance from coordinate of grid. And the thermal field represented buffer usage level is used for communicating buffer information over the network. So that, every node has to keep its neighbours' temperatures and coordinate ID. The lower temperatures represent more available resources to handle new data. In the route decision process, the distance of original to target node, current to target peer, and neighbours to target location are measured.

3.1 Measuring the Temperature

The temperature θ represents the buffer usage of a peer that is the level of messages waiting to forward. At a current node c, the temperature θ_c is calculated at every simulation time. The value of θ_c is between 0 and 1: 0 denotes an empty buffer and 1 a full buffer.

$$\theta_c = \frac{\text{Messages in Buffer}}{\text{Buffer size}} \quad , \qquad 0 \le \theta_c \le 1$$

The latest buffer status is important to make a correct decision; hence, it is designed to attach the temperature value to all data packets sent through the community and in the corresponding acknowledgement packets. The packets and the acknowledgements work as a median of the temperature. They pass temperatures from one to another node until they reach their target or expire. Every current node c has a set of neighbours $N(c)$ where messages can be forwarded to and i is a number of neighbour, then $N_i \in N(c),\ 1 \le i \le 4$. There are three possibilities to update a neighbours' temperature, $\theta(N_i)$ on node c. Let β_i be the number of packets and μ_i be the number of acknowledgments which sent from neighbour N_i to current node.

1. If node c receives a packet or an acknowledgment from neighbour N_i , the old temperature is replaced with the new temperature.

$$\theta(N_i) = \theta_t \,, \qquad \text{if } \beta_i > 0 \text{ and } \mu_i > 0$$

2. If there is no message sent from neighbour N_p , the new temperature caused by the spread of source node then decreases exponentially, whereby t is the routing time.

$$\theta(N_i) = \theta(N_i) \cdot e^{-\lambda t}, \qquad \text{if } \beta_i = 0 \text{ and } \mu_i = 0$$

3. The new temperature is zero when no message arrives and no heat remains.

$$\theta(N_i) = 0, \qquad \text{if } \beta_i=0, \mu_i=0, \text{ and } \theta(N_i)=0$$

Next topic, we introduce how fuzzy logic system works when knows distance and temperature of its neighbours.

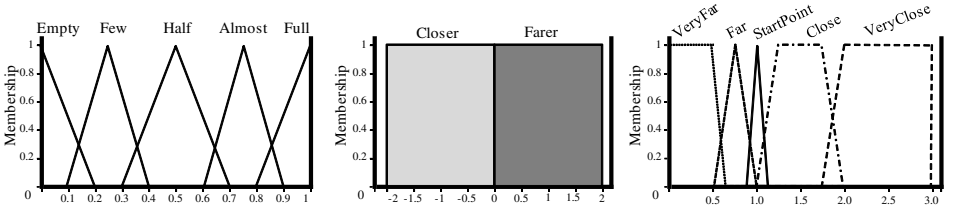
3.2 Fuzzy Logic

The inputs to the fuzzy controller to be designed for routing are: (i) buffer usage status, (ii) distance, and (iii) neighbour type. These three selection parameters make the route reflect the network status and the nodes' ability to reliability delivery network packet. The distance is defined current packet-holder position compares to source and target. It is calculated by equation (1), and neighbour type is in equation (2):

$$Distance = \frac{\sqrt{(x_d-x_s)^2+(y_d-y_s)^2}}{\sqrt{(x_d-x_c)^2+(y_d-y_c)^2}} \quad (1)$$

$$Neighbor\ Type = \sqrt{(x_{Ni}-x_t)^2+(y_{Ni}-y_t)^2} - \sqrt{(x_c-x_t)^2+(y_c-y_t)^2} \quad (2)$$

Where (x_c,y_c) is current peer, (x_s,y_s) is source node, (x_d,y_d) is destination, and (x_{Ni},y_{Ni}) is neighbours i of current node. The steps involved in calculation of neighbour preference rate are elaborated in Fuzzy Interference System (FIS). The three input variables to be fuzzified are the thermal value (buffer usage status), the relative distance, and the neighbour type. The terms "Empty", "Few", "Half", "Almost", and "Full" are used to describe the temperature field. "VeryClose", "Close", "StartPoint", "Far", and "VeryFar" are termed to explain the relation of distance and "Closer" and "Farer" are described neighbour types. In Fig. 2 shows the membership functions of input variable.



(a) Thermal (b) Neighbor (c) Distance

Fig. 2: Fuzzy Membership function of input variable

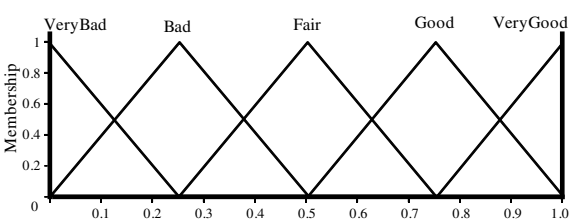


Fig. 3: Fuzzy Membership function of Neighbour Rate

Fig. 3 shows the membership functions of output, neighbour rate. We define nine terms for the values of neighbour rate from lowest to highest as “VeryBad”, “Bad”, “Fair”, “Good”, and “VeryGood”. The rules of the FIS are designed for an optimal path. Table 1 shows rule base for the FIS.

Neighbor Rate		Neighbor = Closer					Neighbor = Farer				
		Thermal					Thermal				
		Empty	Few	Half	Almost	Full	Empty	Few	Half	Almost	Full
Distance	VeryClose	VeryGood	VeryGood	Good	Fair	Bad	Good	Good	Fair	Bad	VeryBad
	Close	VeryGood	VeryGood	Good	Fair	Bad	Good	Good	Fair	Bad	VeryBad
	StartPoint	VeryGood	VeryGood	Good	Fair	Bad	Good	Good	Fair	Bad	VeryBad
	Far	VeryGood	Good	Fair	Bad	Bad	Fair	Fair	Bad	VeryBad	VeryBad
	VeryFar	VeryGood	Good	Fair	Bad	Bad	Fair	Bad	VeryBad	VeryBad	VeryBad

Table 1: Fuzzy Rule Base

There are 37 rules defined for this fuzzy system. The examples are showed in the following:

R1: IF thermal IS Empty AND neighbour IS Closer THEN neighbour_rate IS VeryGood;
R2: IF thermal IS Full AND neighbour IS Farer THEN neighbour_rate IS VeryBad;
…
R37: IF thermal IS Almost AND distance IS VeryFar AND neighbour IS Farer THEN neighbour_rate IS VeryBad;

The defuzzification is the process of conversion of fuzzy output set into a single number. The method “Center of Gravity” (COG) is chosen as show in equation (3)

$$Neighbor\ Rate = \frac{\sum_{i=1}^{n} X_i \cdot \mu(x_i)}{\sum_{i=1}^{N} \mu(x_i)} \quad (3)$$

Where x_i is the element and $\mu(x_i)$ is its membership function. COG method is the most widely defuzzification strategy, which is reminiscent of the calculation of the expected value of probability distributions.

The details proposed algorithm is explained. Next in Section 4, we will present some experimental results. The outcomes compare to shortest path method and adaptive probability functions to use thermal field.

4 Experimental consideration

We conducted experiments to evaluate the proposed protocol, and compare to the shortest path method and thermal approach by fixed functions algorithm. Since our approach run in decentralized network, each node knows only its neighbourhood peers. The route decision is made step by step when it hold message. The shortest path method finds the fastest way in terms of number of hop-count then the message is forwarded to the shortest neighbour to destination.

Other constraints aren't considered. On another factor, the thermal approach is used for considering buffer usage level [LU09b]. The functions for probability to select either low buffer route or shortest way are predefined. In this paper, we selected the two different formulas in comparison.

4.1 Environment Setting

The experiment was simulated using P2PnetSim, a network simulation environment. The tool is powerful and flexible in simulating, modeling and analyzing any kind of networks. Peers are configured collectively and individually using XML files for network setup, and Peer behaviors are implemented in the Java programming language. In our experiments, the network is organized into a grid structure with 1,024 nodes in two dimensions (32x32). The coordinates of a node within the grid form its node ID. The grid is overlaid on a virtual IPv4 network. Peers are connected in four directions to each other: left, right, up, and down. The buffer sizes and outgoing bandwidths are limited for most of the peers. Both buffer sizes and bandwidth values are assigned randomly follow the Power-Law distribution. There are two types of packets, data and acknowledgements. The acknowledgment is prioritized. Otherwise, the system handles the packets First-In-First-Out.

To generate traffic, the simulation defines different throughputs for nodes in terms of buffer sizes and outgoing bandwidths. In the trial, the 338 source nodes are uniformed randomly selected together with predefined target randomly in a specific area. Source nodes send a message to its target with probability 0.2. They generate a message in every simulation time until total messages are 50,000 packets. In order to evaluate algorithm performances messages success delivery ratio, message loss ratio, and message expired ratio are measured. Besides, a node (31, 28) is set to assess routing time. It generated a packet every simulation time until 500 messages sent to target (0, 0). The routing time that counts from launching the original node to reaching the target node. That time includes moving steps and waiting times in the traffic nodes.

4.2. Results and Discussion

The experiments report in this section compares to shortest path methods and thermal approach with two probability functions (4 & 5) as introduced in [LU09b]. Packet delivery ratio is important as it describes the successful rate that will be seen by the transport protocol, which in turn affects the routing quality that the network can support.

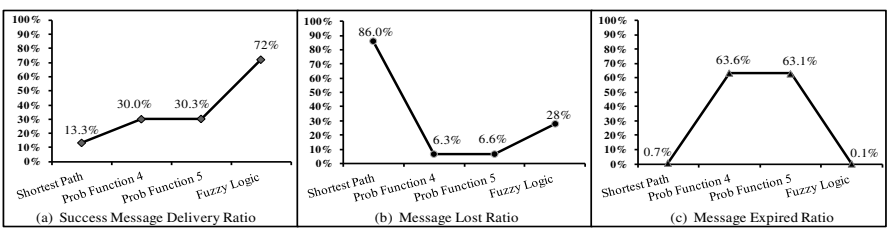


Fig. 4: Packet Delivery Ratio

Fig. 4(a) presents that packet success delivery ratio is the highest for routing with fuzzy logic, which is due to its ability to select a least congested routes thus having the lowest amount of loss as shows in Fig. 4(b). Moreover, a very few message expired ratio in Fig. 4(c) is also lowest which points that proposed algorithm is able to avoid very long indirect route and/or long queue in buffer peers.

Fig. 5 presents an example routing performance of evaluation node (31,28) which sent packets to its target peer (0,0). The average routing time is the average time to deliver a packet from launched at source to reached target, and it includes all possible delays such as waiting in buffer queue. And the average number of hop-count is the average number of peer that packets are passed during transmission. The minimum hop-count from node (31,28) to node (0,0) is 59 steps in grid. The shortest path method obviously shows every message transport to destination with a number of hop-count, 59 times. However, average routing time is quite high due to waiting time in long queue buffer nodes. And due to shortest path protocol aim to transmit in the fastest way, packet lost ratio as in Fig. 4(b) is highest. On the other hand, the thermal approach with predefined probability functions to select either available buffer path or shortest route shows value of average routing time and average number of hop-count in similar that means they are a good approach in terms of avoid long queue of busy nodes. But packet delivery ratios in Fig. 4 reveal that many long indirect routes are chosen to avoid congestion nodes made high percentage of expired packet. The best results are from multi-criteria using fuzzy logic. It balances avoidance of indirect route and escape from busy nodes.

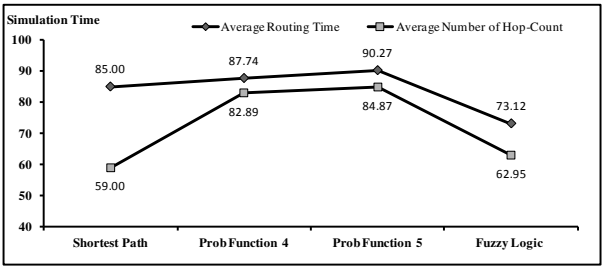


Fig. 5: The average routing time

5 Conclusion and Future Work

In this paper we introduced a multi-criteria routing algorithm using fuzzy logic system. Our method runs in grid-like structured overlay network then only local information is used to find an optimal path. The buffer usage status is applied by thermal approach. And the relative distance among peers are beneficial from coordinate in grid structure. Both criteria are taken into account to find the best path and balance resource utilization by fuzzy logic. The experiment results proof that the introduced algorithm enabled to find an appropriate path, and react to high buffer usage situations. In future work, more constraints, such as bandwidth will be considered for improving quality of service routing. In addition, the enhancement of routing algorithms will be studied by learning process.

References

[Be07] Berg, D.; Coltzau, H.; Sukjit, P.; Unger,H; Nicolaysen, J.: Passive RFID tag Processing using a P2P architecture: Malaysian Software Engineering Conference 2007; pp. 169-179.

[BSU09] Berg, D.; Sukjit, P.; Unger, H.: Grid generation in decentralized systems: In Proceeding of the International Workshop on Nonlinear Dynamics and Synchronization 2009 (INDS), Klagenfurt, Austria, 2009; pp. 95–99.

[CL92] Cristopher, J.G.; Lionel, M.N.: Adaptive Routing in Mesh-Connected Networks: Proceeding 12th International Conference on Distributed Computing Systems, 1992; pp.12-19.

[DGK09] Donnet, B.; Gueye, B.; Kaafar M.A.: A Survey on Network Coordinates Systems, Design, and Security: IEEE Communication Surveys & Tutorials, 2009.

[GDT09] Goswami, M.M.; Dharaskar, R.V.; Thakare, V.M.: Fuzzy Ant Colony Based Routing Protocol For Mobile Ad Hoc Network: International Conference on Computer Engineering and Technology, Vol. 2, 2009; pp. 438-444.

[JVM95] Jatin, H.U.; Varavithya, V.; Mohapatra, P.: Efficient and Balanced Adaptive Routing in Two-Dimensional Meshes: In International Symposium on High Performance Computer Architecture, 1995; pp. 112-121.

[Kl00] Kleinberg, J.: The small-world phenomenon: An algorithmic perspective: Proc. 32nd ACM Symposium on Theory of Computing, 2000.

[Lu04] Lua, E.K.; Crowcroft, J.; Pias, M.; Scharma, R.; Lim, S.: A Survey and Comparison of Peer-to-Peer Overlay Network Schemes: IEEE Communications Survey and Tutorial, March 2004.

[LU09a] Lertsuwanakul, L.; Unger, H.: A Thermal Field Approach in A Mesh Overlay Network: 5th National Conference on Computing and Information Technology, Thailand, 2009; pp. 610-615.

[LU09b] Lertsuwanakul,L.; Unger,H.: An Adaptive Policy Routing with Thermal Field Approach: 9th Int. Conf. on Innovative Internet Community Systems (I^2CS), Jena, Germany, 2009; pp. 169-179.

[LW04] Lee, A.; Ward, P.A.S.: A Study of Routing Algorithms in Wireless Mesh Networks: Australian Telecommunication Networks and Applications Conference, December 2004.

[Me04] Mello, A.V.; Ost, L.C.; Moraes F.G.; Calazans N.L.: Evaluation of Routing Algorithms on Mesh Based NoCs: Technical Report Series No.040, May 2004.

[MTR07] Mirabedini, S.J.; Teshnehlab, M.; Rahmani, A.M.: FLAR: An Adaptive Fuzzy Routing Algorithm for Communications Networks Using Mobile Ants: International Conference on Convergence Information Technology, 2007; pp. 1308-1315.

[RR91] Rajasekaran, S.; Raghavachari, M.: Optimal Randomized Algorithms for Multipacket and Wormhole Routing on the Mesh: Technical Report, University of Pensylvania, 1991.

[US08] Upadhayay, S.; Sharma, M.: Performance Evaluation of Fuzzy Routing Algorithms for a New Fuzzy Mixed Metric Approach: International Journal of Computer Science and Network Security, Vol. 8, No. 4, April 2008; pp. 21-28.

[UW04] Unger, H.; Wulff, M.: Search in Communities: An Approach Derived from the Physic Analogue of Thermal Fields: Proc. the ISSADS 2004, LNCS 3061, Guadalajara, Mexico, 2004.

[Za65] Zadeh, L.A.: Fuzzy sets: Information and Control, Vol. 8, 1965; pp. 338-353.

[ZZ05] Zhang, R.; Zhu, X.: Fuzzy Routing in QoS Networks: Proceeding of 2nd International Conference Fuzzy Systems and Knowledge Discovery, Part II, LNCS 3614, 2005; pp. 880-890.

Performance evaluation of two Self-Adaptive Routing Algorithms in Mesh Networks

Miguel Angel Rojas González

Faculty of Mathematics and Computer Science
FernUniversität in Hagen
Universitätsstraße 27, PRG
58084 Hagen, Germany
miguel.rojas@fernuni-hagen.de

Abstract: Following the seminal work of Unger et al. based on building mesh-like structures on top of a P2P network, we introduced an improved version of *Compass* - a routing and load balancing algorithm based on direction-scopes for message delivery in such kind of networks. In this paper we evaluate and compare the performance of this scope-based Compass against the self-balanced and self-adaptive routing algorithm named *ColorANT*. Moreover, we compare the Compass performance against two routing algorithms: Flooding and a constrained version of Hotpotato.

1 Introduction

Unger et al. in [BCS+07] introduced an adaptive routing algorithm for passive RFID tag information, monitored from any internet-connected tag reader. This algorithm is based on building mesh-like structures on top of a P2P-connected network of the manufacturer's servers. In such kinds of mesh network a node is identified by its X and Y coordinate and references up to eight neighbors (one for each orientation: down, up, right, left, as well as up-right, up-left, down-right and down-left). Due to the nature of the mesh structure (code-named *UTH* [1]) each node finds the closest neighbors through a fitness function. This function is based on the Euclidian distance. UTH mesh-based networks offer not only high stability and high speed of reaction by node-jittering in very large networks, but also more efficient and scalable than structured systems.

Several papers deal with online routing and related problems in geometric settings. Bose and Morin [BM99] classify routing algorithms based on their use of memory and/or randomization. In both cases, the message forwarding is based only on the coordinates of the node where the message is located. Bose and Morin study also a randomized and memoryless routing algorithm that works for any triangulation. Kalyanasundaram and Pruhs propose a 16-competitive algorithm to explore any plane graph. If the ratio between the

[1] UTH denotes the abbreviation for the "Udon Thani Airport Thailand", where the author invented the building mesh algorithm, while waiting.

length of any path found by algorithm A and the shortest path, is less than K, then A is called K-competitive. Cucka et al. [CNR96] have shown that greedy routing algorithm performs better than Compass routing algorithm ([BM99], [KSU99]) on random graphs, but does not do as well on Delaunay triangulations. Nevertheless, Compass is slightly more efficient in any case if the number of edges traversed is considered. In [BBC+00], Bose et at. show that, using Euclidean metric, there is no routing algorithm competitive for all triangulations. They also proved, that no routing algorithm exists that uses the number of traversed edges which are competitive for all Delaunay or greedy triangulations.

1.1 Motivation

In [RC07], we introduced an improved Compass routing algorithm. It delivers messages in the above mentioned mesh-like structure using on each hop the direction closest to the destination node. By using a range of possible directions that point to the destination, the so called *Scope*, Compass avoids paths under congestion conditions.

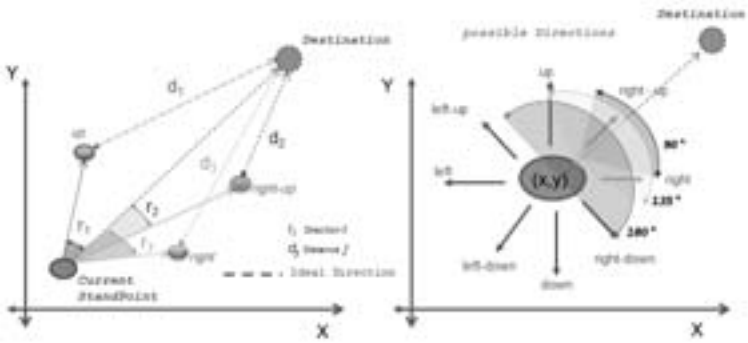


Figure 1: Compass mechanism performed on each peer. Left side shows the Euclidian distance and the angle between the closest neighbors with the ideal direction. Right side depicts the eight neighbors of a peer on the UTH overlay network. Furthermore it shows three different scopes: 90, 135 and 180 grades.

One of the open issues in routing algorithms is the *Hotspot* problem. A peer becomes Hotspot if it forwards messages with a higher frequency than other peers. Hence, a hotspot-peer induces quickly to congestion conditions. In this paper, we evaluate experimentally the performance of this improved Compass and prove its load balancing capability.

In [RUC07], a self adaptive and self-balanced routing algorithm was introduced for message delivery in large unstructured P2P networks. ColorANT adopts Ant Systems as introduced by Dorigo, Maniezzo and Colorini [FBP95]. It uses colored pheromones to look for alternative paths near to the optimal route in case of congestion conditions . As described

in [RC07] and [LU09], the colored routes marked by ColorANT can offer higher availability, coverage and up-to-dateness than the known best local routes approaches. Based on this, this paper aims also to evaluate the high availability and load balancing capability shown by ColorANT in very large networks.

In this paper we compare also the efficiency of two simple routing algorithms: Flooding and HotPotato in UTH networks. The HotPotato algorithm represents a general cacheless routing algorithm, which means the nodes of a network have no buffer to store temporal messages before they reach their destination. In HotPotato, each message that is routed is simply forwarded until it arrives its destination or it reaches its TTL (*time to live*) -then the message dies. Note that this procedure lacks any intelligence. On the other hand, there are many different implementations of the Hotpotato algorithm [FBP95], [TSG08], [HPL01], even buffered ones, when network limitations are considered. For comparison purposes, we modified the implementation of the HotPotato to be used in UTH overlay networks, keeping a list of each node load and its neighbors. It then equally distributes the incoming messages between half of the neighbors located close to the ideal direction of the destination on the mesh. By Flooding algorithm, each node replicates and forwards incoming messages to its neighbors until the message reaches its destination. Nevertheless, the forwarding process is limited by a TTL to avoid network saturation. Due to simplicity, flooding is being used even in many strategies for message delivery in unstructured P2P networks.

2 Performance Evaluation

2.1 Simulation Environment

Simulation Environment P2PNetSim version 2.5 was used for the above mentioned algorithms' performance evaluation. P2PNetSim is a distributable tool for simulating IP based network infrastructures and in particular for simulating large P2P Systems [Col06]. The analysis of simulation results can be classified into two basic groups: small networks (with 256, 512 and 1024 nodes) and large networks (dealing with 2048, 4096 and 8192 nodes). Each simulation scenario was executed 16 times on a cluster with 32 nodes AMD 64 bits Quad-Core 2.6 GHz, 4 GB RAM and 500 TB of storage. The platform used was JDK 1.6 u 18 under Linux Suse 12. The figures in this paper show the average of those 16 experiments according to each scenario.

2.2 Performance Measurements

In this section, we present several important performance measurements whose results will be compared by simulations in Section 3.

Number of steps: determines how many peers were visited starting from the sender until reaching the destination. The evaluation of this factor uses the average value of all delivered messages.

Network usage: This is a measurement for the distribution of used paths on the network. A marginal value means that many potential paths remain unused. This increases of course the probability of congestion over most used routes. The higher the network usage, the higher load balancing by message delivery. The higher value for network usage is 100%.

Amount of overloaded peers: describes how many peers in a given simulation step cannot accept incoming messages anymore, because their queue is full. The evaluation of this factor is calculated during the whole simulation.

Delay: This factor specifies how long a message stays in the queue before it is processed.

Latency: defines the elapsed time from a message is sent until it's delivered. This factor includes the delay experimented on each peer before the message reach his destination.

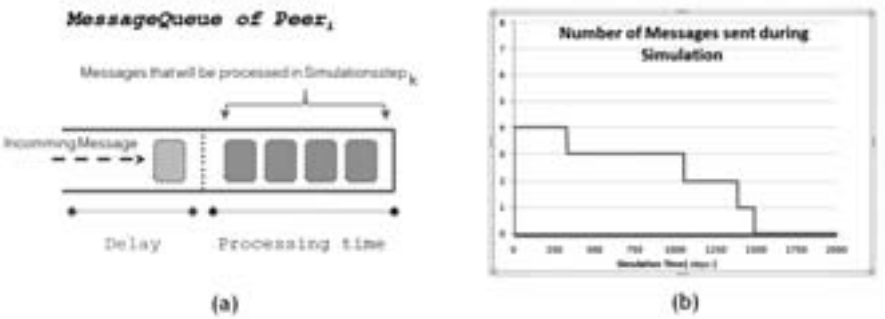


Figure 2: (a) Representation of delay and processing time of message on a peer. (b) Power Law function on number of messages to be routed by each simulation.

3 Simulation Model and Results

As depicted in figure 2 the number of messages generated follows a Power Law function. Each simulation runs 2000 steps and during the last 500 steps, no new messages are generated. This time period is used for processing even the messages that are on the way to their destination. Due to comparison constraints and duration of simulation time, the number of peers generating messages was 4 and 8. The average link lenght of any pair sorce-destination peers was 2L, where L is the square root value of N (network size). The selection of these peers was random. Moreover the TTL for messages was set to 3L.

3.1 Evaluation of average hops by message delivery and network usage

As can be appreciated in figure 3, Compass and ColorANT clearly need less steps for message delivery than HotPotato and Flooding in small networks. Note that HotPotato requires the highest values (or maximum number) of hops. The reason behind is that HotPotato does not use a smart strategy for message forwarding. In large networks, Flooding shows not only the lower averages of hops but also exhibits the worst percentages of network usage. Additionally, Compass maintains around 100% of network usage. ColorANT presents similarl high network usage-, around 80% on networks with 256 peers. This value decreases softly when the networks grow. The network usage in ColorANT reaches approximately 75% with 8192 peers. This is an enhancement against HotPotato from 5% up to 75%. In large networks, Compass takes advantage of around 100% of network usage.

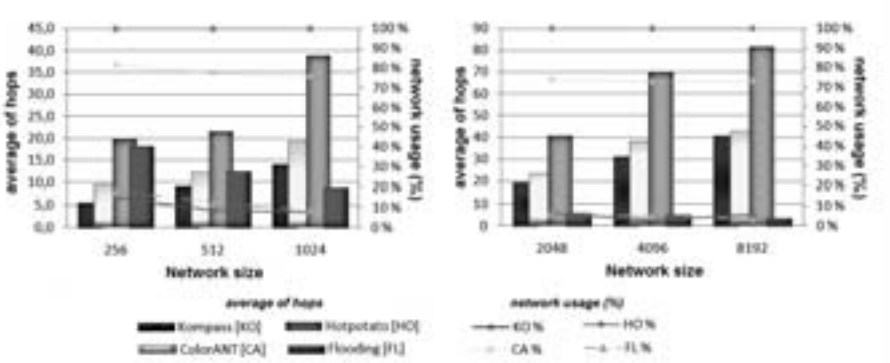


Figure 3: Evaluation of hops needed by message delivery and the network usage. The figure shows the evaluation of Compass in small networks (left side) as well as in large networks (right side)

3.2 Evaluation of the message queue length and number of overloaded peers

Figure 4 shows the simulation results regarding the average length of the message queue of all peers and the percentage of overloaded peers in small and large networks (left and right side respectively). Flooding reveals high values of queue length due to the replication of messages by forwarding. On the other side, HotPotato shows the shortest queues since this mechanism forwards the messages immediately and does not make any copies of the original message. Regarding the number of overloaded peers, ColorANT, HotPotato and Flooding have a similar behavior. This overloading factor lies between 80% and 90%. Compass emphasizes however with values between 30% and 35%. Note additionally that Compass behaves similarly in both small and large networks.

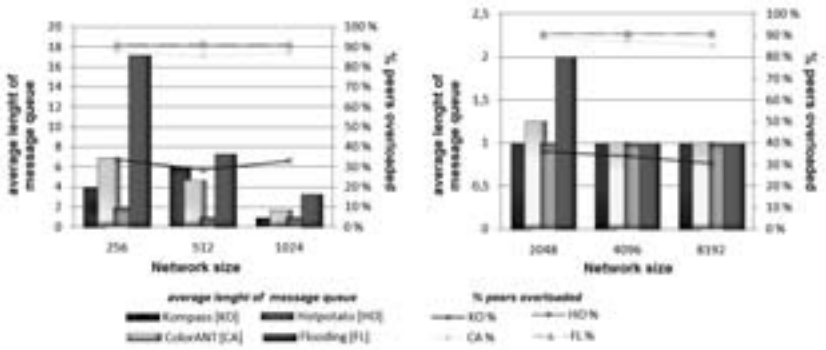


Figure 4: Evaluation of the message queue length and number of overloaded peers

3.3 Evaluation and Comparison of delay and number of hops by message delivering

Figure 5 represents the distribution of average delay and the number of hops in delivering messages of different sizes. In Flooding, a high value is observed for the delay which decreases as the network grows. The number of hops in Flooding is small although it increases lightly with the network size. Note that this value is maintained under 50 hops. This behavior is reflected by Compass and ColorANT as well. Nevertheless, in HotPotato, the number of hops increases strongly. The average delay for all algorithms, except Flooding, remains under 8 simulation steps for messages of 1KB. Compass exhibits a delay close to zero, which is remarkable. A comparable behavior is observed with messages of size of 2KB, 3KB and 4KB. The number of hops is maintained although the delay increases proportionally with the message size, as expected.

4 Conclusions and Future Work

In this paper, we evaluated the performance of two self-adaptive routing algorithms in UTH Overlay Networks. As it was demonstrated, Compass and ColorANT deliver messages with low delay values. In the same way, these algorithms are highly comparable with Flooding as these mechanisms maintain few hops for message delivery. The average delay in most cases remains marginal. Currently, we are evaluating the use of colored ants by the scope-based Compass algorithm. We believe the ants will increase the Compass performance during triangulation situations. In near future, we will evaluate the performance of this routing algorithm cooperation in larger networks that reflect the small-world principle.

Acknowledgements. The authors would like to thank to the University of Dortmund and the FernUniversitaet in Hagen for the facilities to run the simulations on their clusters.

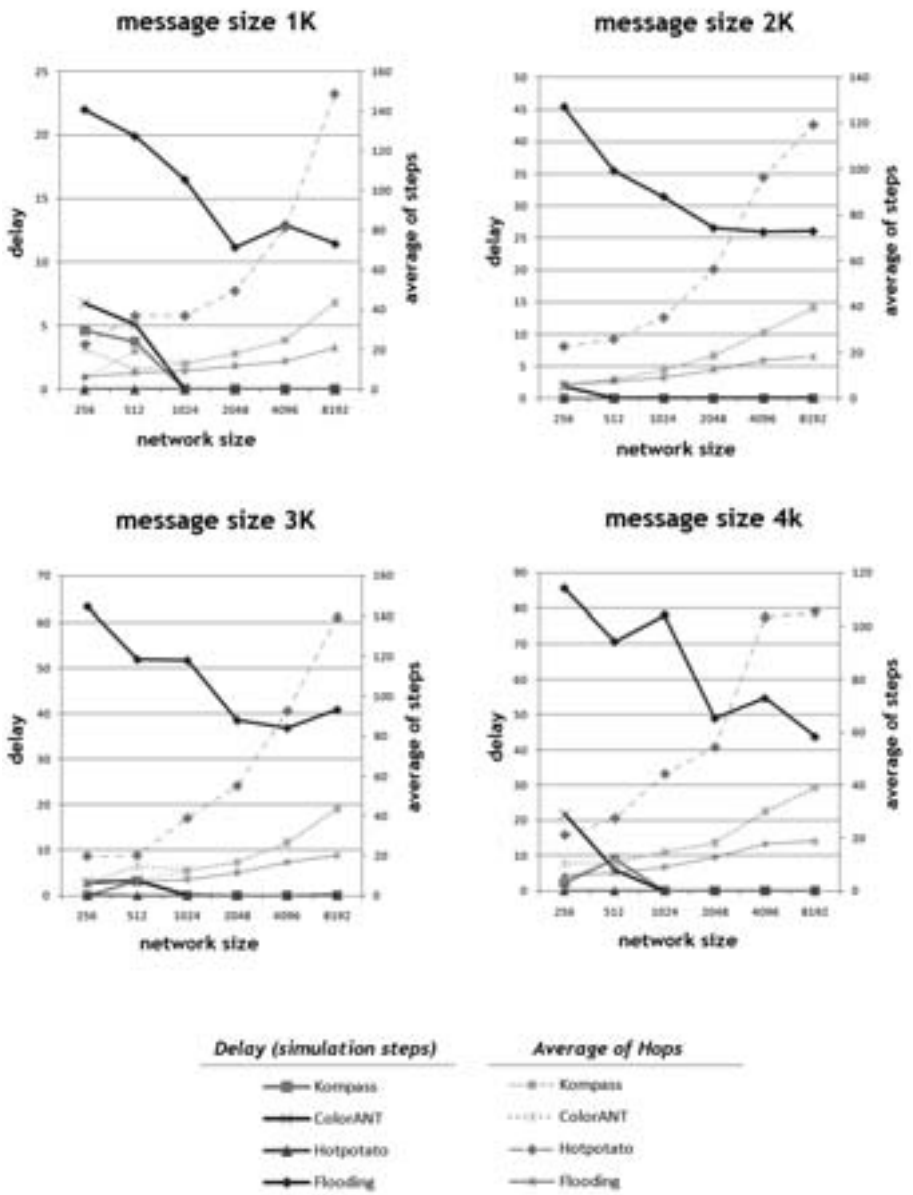


Figure 5: Evaluation and comparison of delay (in simulation steps) and number of hops by message delivery

References

[BBC+00] P. Bose, A. Brodnik, S. Carlsson, E. Demaine, R. Fleischer, A. Lopez, P. Morin, and J. Munro. Online Routing in Convex Subdivisions. In *International Journal of Pattern Recognition and Artificial Intelligence*, pages 429–446, Springer LNCS, 2000.

[BCS+07] D. Berg, H. Coltzau, P. Sukjit, H. Unger, and J. GM. Nicolaysen. Passive RFID tag Processing using a P2P architecture. In *Malaysian Sw. Engineering Conference*, 2007.

[BM99] P. Bose and P. Morin. Online Routing in Triangulations. In *Tenth International Simposium on Algorithms and Computation ISAAC 99*, pages 113–122, 1999.

[CNR96] P. Cucka, N.S. Netanyahu, and A. Rosenfeld. Learning in Navigation: Goal finding in graphs. In *International Journal of Pattern Recognition and Artificial Intelligence*, pages 429–446, 1996.

[Col06] H. Coltzau. Specification And Implementation Of A Parallel P2P Network Simulation Environment. Diploma thesis, Universität Rostock, 2006.

[FBP95] F. Forghieri, A. Bononi, and Paul R. Prucnal. Analysis and Comparison of Hot-Potato and Single-Buffer Deflection Routing in Very High Bit Rate Optical Mesh Networks. In *IEEE Transactions on Communications*, 1995.

[HPL01] R. Honkanen, M. Penttonen, and V. Leppänen. Hot-Potato Routing Algorithms for Sparse Optical Torus. In *International Conference on Parallel Processing Workshops (ICPPW'01)*, 2001.

[KSU99] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *11th Canadian Conference on Computational Geometry CCCG 99*, 1999.

[LU09] L. Lertsuwanakul and H. Unger. An Improved Greedy Routing Algorithm for Grid using Pheromone-Based Landmarks. In *World Academy of Science, Engineering and Technology*, Indonesia, 2009.

[RC07] M. Rojas and H. Coltzau. 'Compass', A Routing and load balancing algorithm in UTH Overlay Networks for largescale P2P Networks. In *IADIS WWW/Internet 2008 Conference*, Germany, 2007.

[RUC07] M. Rojas, H. Unger, and H. Coltzau. Self-Balanced and Self-Adaptive Routes in Unstructured P2P Networks. In *Second International Conference on Systems (ICONS'07)*, Martinique, France, 2007.

[TSG08] R. Teixeira, A. Shaikh, and T. Griffin. Impact of Hot-Potato Routing Changes in IP Networks. In *IEEE Press Piscataway*, NJ, USA Volume 16, Issue 6, 2008.

Functional Approach to Decentralized Search Engine for P2P-network Communities

Oleksandr Kuzomin, Illya Klymov

Informatics chair
Kharkiv National University of Radio Electronics
UKRAINE, Kharkiv, Lenina av. 14, office 437
kuzy@kture.kharkov.ua
illya.klymov@gmail.com

Abstract: In this paper we propose a simple powerful method to increase search performance in large p2p-network communities, especially, but not limited to Internet. Our approach is based on introducing common functional language methods "map" and "reduce" in distributed network environment. The method exploits different connectivity levels of p2p nodes, dividing them into two groups – master nodes, performing preliminary search processing and worker nodes, performing actual search. We show that approach could be effectively used in Grid network, built over common P2P-network utilizing benefits of routing in Grid environment. An experimental implementation of the method confirms the theoretical analysis of the method.

1 Introduction

An extreme amount of data available in an Internet requires an efficient tool in order to determine where we locate data matching our request. This process is not limited to web-search (solved by Google, Yahoo, etc.). In big P2P networks an efficient way for finding required chunk allows to significantly decrease overall bandwidth consumed by the network.

2 Problems of Using Common Search Engine Approaches in P2P-networks

A key idea of search engine functioning is an ability to retrieve links to new documents (let's speak about HTML pages for example) in process of indexing existing data. This is not applicable to P2P-networks since documents (files represented into network) have no interconnections between each other. On the other hand each node in P2P provides an interface to enumerate all documents, available on this node. In general the following problems renders common indexing algorithm almost unusable:

- Small peer TTL (time to live) – in normal indexing process we put recently discovered documents tot he end of queue, but when we end processing queue the peer, provided these documents, could already be offline. On other hand we could easy stuck on indexing one big, yet slow node, which is unacceptable for making quick searches
- Some peers could not be accessed directly (they have no white IP). This nodes still participate in P2P-network, they carry important network data, but we can't use „PULL" technique to get data from them - data must be directly pushed by that nodes
- P2P query consumes significant bandwith on the wide-area Internet; the total bandwith consumed by all queries (which should be performed in parallel if we want to provide an acceptable response time) must fit comfortably within Internet's capacity.

Another specific of P2P-search is that each document in network could be described in less than 1KB of data (including filename, metadata, node info) since we are not assuming performing full-text search on P2P-network. In other words we should pay significant attention to quick indexing each node, assuming that time of indexing document is not notable comparing to time for making connection and retrieving index of available node documents.

3 Search Engines in P2P-environments

3.1 Functional Approach to Search in P2P-networks

Any approach used for search in P2P-networks must be designed to meet the following goals: ability to operate in dynamic environment, performance and scalability and reliability. In order to start with functional approach in P2P-network let's define primary terms and operations over network.

We divide all nodes in two big groups – worker nodes and master nodes [MR09]. Worker nodes provide an index file of all its data on request: $\text{Index}(W_i) = L$, where W_i is a worker node and L – is an index in form of list, generated for quick search in it (consider it for example to be a binary search tree optimized for quick search or just a hashtable – selection of correct model greatly depends on specific of indexed data). Each worker node has at most one master node, to which it is connected.

Master nodes of 1 level store list of worker nodes, and store integrated index, which includes all data, provided by worker nodes, connected to this node (details of how worker nodes provide data to master ones are described in section 3.3):

$$\text{Index}(M_{i1}) = L \cup \text{Index}(W_1 \cdots W_k), W_1 \cdots W_k \in \text{Workers}(M_{i1})$$

Master node of n-th level is functioning as master node for all master nodes of (n-1)-th level and as worker node for at most one (n+1)-th level master node. Also each master node of n-th level contains a cached copy of all nodes of the same level, connected to (n+1)-th level master node, to which current node is connected.

Search operation could be implemented in two stages. First one is called "map". On this step each master node receives the query, spreads it to all its workers (except the source one) and transfer it to an upstream node. We guarantee that each node of level n has a node of level (n+1) connected if node of such level exists into network. The connection from worker nodes shown as dotted line because this nodes are not directly involved into search, but the data they provide is used for

Next step, called "reduce operation" is receiving results from all nodes and processing them into single instance. In order to generate a result for any node consider we have a function *Search* which returns results from an index. In such way results of node W_n could be described as follows:

$$\text{Search}(W_n) = \begin{cases} \text{Search}(\text{Index}(W_n)), & n = 1 \\ \text{Search}(\text{Index}(W_n)) \cup \text{Search}\,(\text{Workers}(W_n)), & n > 1 \end{cases}$$

Each node performs reduce operation before transferring query back in order to eliminate possible duplicates from response.

In such scheme the main weak point is first master node, which performs overall search. If this node goes offline during search requests it leads to all search results lost, wasting a huge amount of computing time and network bandwidth. In order to reduce losses in that case each master node provides a list of B backup nodes (where amount of B may vary and should increase with network growth). This backup nodes is an ordered list which is known both to workers and upstream nodes and describe which node should receive data in case current node goes offline unexpectedly. The main idea of this approach is shown on Figure 1:

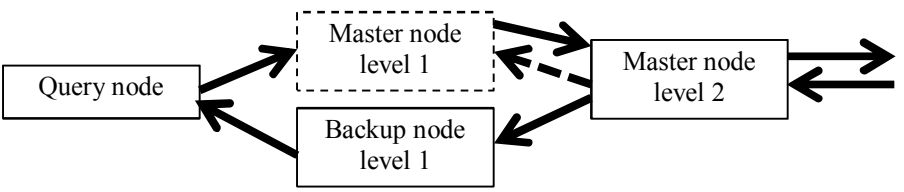


Figure 1: Fault-tolerance P2P search in action

On this figure it is shown how search reply from master node level 2 tries to reach master node level 1, but data transfer fails, for example due to timeout, the data is sent to the first backup node, which was provided in request, sent to master node level 2. This backup node notifies a data, which was not requested directly by that node, and analyzing the body of search response reveals that this data should be transferred to query node. In that manner we ensure that no results of the search would be lost during search process.

The main problem of such approach is that returning result of node is delayed until all downstream node formed it's response or timeout for waiting response is reached. This could be solved by selecting nodes, which are not heavily loaded in P2P-exchange as master nodes and decreasing timeout to low values, to make search quickly react on changing network structure [ZLX04]

3.2 Concept of P2P Search Engine using Proposed Approach

Now we are one step away from building a prototype of P2P search engine using proposed approach. First of all, it is time to decide how we can choose master nodes in the real grid network. Since master nodes would receive all requests, they should be not heavily loaded in real P2P-data exchange. This is a situation where the term "node temperature" becomes useful. We will measure node temperature as:

$$\theta_s = \frac{\text{Messages in buffer}}{\text{Buffer size}}, 0 \leq \theta_s \leq 1 \text{ [LU09]}$$

If node temperature is below some threshold θ_t this node could be considered as 1-level master node. All nodes with temperature lower than θ_t^2 will become master nodes of level 2 and so on.

Each master node of level *n* contains data about:

- all master nodes of n-1 level
- indexes received fron all master nodes os n-1 level (not necessary directly), with a TTL (time to live) mark attached to each
- master node of n+1 level to which it is connected.

In other words we are building a tree like system on top of existing grid mesh. Each node is functioning according to following algorithm:

1. Each X seconds requests if master node (node of higher level to which this one is connected) is alive. If not – all nodes which were using same master perform a voting process – node with least temperature becames a new master node (forced promotion)
2. If current temperature of node is lower than θ_t^k, where k>n (assume n is current level of node) for time more than Y seconds – node is promoting self to n+1 level. A new master is obtained from previous one, and all nodes, which were connected to previous master are notified and could change master

3. Each Z seconds a current index of all files is pushed to the upstanding master node. Master node merges all received indexes in one, includes own file index and provides this as new master node index
4. If current node has no master node privileges (worker node) it searches for any master node using blind search, receives list of all master nodes and using ant algorithm determines which node is most suitable to connect. After that it publishes own list to found master node.

Varying (X,Y,Z,S) we can vary intensity of updates in order to find an optimal solution between speed of reaction of overall search network to changes and an amount of service traffic, generated by network. Values of this parameters greatly depends on bandwidth between nodes, node lifetimes and how often node data is changed.

4 Conclusion and Future Work

In this paper, a method of building search engine in P2P-environment using map-reduce approach is introduced. This approach is provides a strong fault-tolerant backend for performing searches in a dynamic, decentralized system, avoiding common flooding approach widely spreaded in P2P networks. Moreover, by using this algorithm it is possible to significantly decrease a speed of most common searches in this environment, using cached map-results before performing reduce step.

In the future work, the algorithms must be studied more deeply to determine exactly the importance of different parameters of map-reduce search in real network environment. Additional research should be performed on determining sufficient and needed amount of master nodes for effective request processing depending on network size. An additional speed up could be obtained using effectively distributed master nodes all over the grid network (currently they are grouped around query node, which is not always effective). Another promising method is a possibility of integrating this approach with some parts of common blind search, granting benefits of quick reacting on new node appearance to proposed method.

References

[LU09] Lertsuwanakul L; Unger H.: A Thermal Field Approach in a Mesh Overlay Network: Proc. Of 5th National Conference on Computing and Information Technology, Bangkok, Thailand, May 22-23 2009

[MR09] Magharei N., Rejaie R.: PRIME: Peer-to-Peer Receiver-driven Mesh-based Streaming: Proc. Of IEEE/ACM Transaction on Networking (TON) Volume 17, Issue 4 (august 2009) ISSN 1063-6692

[SWU04] Sakaryan M.W.G.; Unger H.: Search methods in P2P networks: a Survey": Proc. Of Innovative Internet Community Systems (I2CS 2004), June 2003, Guadalajara, Mexico

[ZLX04] Z.Zhuang, Y. Lio, L. Xiao: Dynamic layer management in super-peer architectures.: Proc. Of the 2004 internation conference on parallel processing (ICPP 04), Montreal, Quebec, Canada, August 2004

Chapter 3: User Behaviour and Profiling

Contributions to 10th I2CS 2010, Bangkok, Thailand

Andrea Jersabek
Why Western Approaches fail in Asia: a Classroom Action Research on Developing Creative Processes and Knowledge Sharing Abilities

Amir Gershman, Amnon Meisels, Karl-Heinz Lüke, Lior Rokach, Alon Schclar, Arnon Sturm
A Decision Tree Based Recommender System

Chanattha Thongsuk, Choochart Haruechaiyasak, Phayung Meesad
Classifying Business Types from Twitter Posts Using Active Learning

Coskun Akinalp, Herwig Unger
The Limbic Characteristic and El-Farol Games

Contributions to 7th I2CS 2007, Munich, Germany

Gerald Eichler, Christian Erfurth, Volkmar Schau
Enhancing Communities by Social Interactions in Mobile Environments

Gregor Heinrich
Actors–media–qualities: a Generic Model for Information Retrieval in Virtual Communities

Birgit Wenke
Use of Algorithms for a User Specific Reduction of Amounts of Interesting Association Rules

Why Western Approaches Fail in Asia: A Classroom Action Research on Developing Creative Processes and Knowledge Sharing Abilities

Andrea Jersabek

Department of Information Technology
King Mongkut's University of Technology North Bangkok
1518 Pibulsongkram Road
Bangkok 10800
ajersabek@kmutnb.ac.th

Abstract: Critical as well as creative thinking, knowledge sharing abilities and intercultural competences are considered key competences in today's business world.
Basing on the results of a study at the King Mongkut's Institute of Technology North Bangkok (Thailand) the present contribution intends to compare the study behavior of Asian students in comparison with western ones.

1 Introduction

In today's globalized scenario, the major part of engineering students at a higher level will at some stage be exposed to an intercultural setting. This requires intercultural skills such as empathy for foreign cultures, knowledge sharing abilities, teamwork competencies, etc. Zorn (2005) emphasizes the significance of Virtual Communities for education, arguing that "VCs in international settings seem to offer great potentials for democratic exchange, collaborative learning, intercultural communication, free information flow, etc. if they florish". However, the research by Sarker on knowledge transfer in distributed U.S. – Thai teams showed, that "contrary to prior beliefs, this study showed that members of more individualistic cultures transferred/shared more knowledge", meaning that the Thai group members were less inclined to share. Sarker here bases this distinction between individualistic and collectivistic cultures according to Hofstede (1991). Burn and Thongprasert (2005), who conducted research on success factors of Virtual Education Delivery (VED), comes to similar conclusions, agreeing that certain Thai values, such as high power distance, collectivism and high uncertainty avoidance present barriers to knowledge sharing and collaborative learning.

2 Current Problems of the Thai Education System

At the annual conference of the Office for National Standards and Quality Assessment (Onesqa), held on November, 2007, in Bangkok, Thailand, several issues of the current problems of the Thai education system were addressed Among others, it was lamented that Thai graduates lacked adequate English skills, or other foreign language skills, leaving them unconfident dealing with foreigners. Furthermore, they need the ability to think critically and creatively in order to be able to 'think outside the box', "the ability to think differently with an open mind while respecting another's opinion. (Bangkok Post, 2007). Another problem which needs to be addressed is the students' knowledge-sharing abilities. In her study of Thai students in Thai and Australian universities, Thongprasert (2005) found that Thai students proved weak in their knowledge-sharing abilities.

One factor that is frequently addressed is that of learning styles with respect to Asian learners. According to Penporn and Pagram (2006) Asian students are reported to follow a strict teacher centered learning mode. Rote learning and 'chalk-and-talk' dominate the classroom atmosphere. Thai students "have never been taught to learn by themselves" (ibid.). What is more, unlike in many other Asian countries, English is considered and viewed as a "foreign language" (ibid.), not as a second language.

Asians are also much closer linked to their families and seem very dependent until well into their adult life. Generally, they also depend much more on authority figures throughout higher education. Moreover, Thais are distinctly social learners. They usually gather in groups after the lectures and then discuss among their peers what is unclear.

These statements are controversial, considering that the Thai education system was in fact modeled after the British and the American system. It should be remembered that rote learning and memorization were also the norm in Europe and America before educational reforms were introduced. Therefore it can be argued that the Thai educational system holds on to 'old' teaching methods. This should be considered in the following chapter.

3 Methodology

The sample comprised of students who took part in the course "Advanced Topics in Management Information Systems" in semester 2, 2009. The course focused on Knowledge Production, Knowledge Representation and Knowledge Management. It is part of the international Master's Program at the Faculty of Information Technology. The Learning Platform MOODLE (2009) was used to support student interaction. Evaluative methods included observation, surveys and interviews.

MOODLE was used as an LMS to support student-student and student-teacher activities; and the online conferencing software SABA Centra was introduced. Furthermore, the use of mind-maps and moderation techniques were introduced and practiced in workshops. The research was carried out within the course Advanced Topics in Management Information Systems during semester 1 2009. Lesson materials included supporting literature complementing classroom presentations and classroom discussions. Topics not only focused on technical issues, but included stimulating texts on technology and culture, creativity, conflict, philosophy, etc. Furthermore, a student-centered approach introducing mind-maps and brainstorming techniques was practiced.
MOODLE has various types of surveys predefined, one of them is ATTLS (Attitudes to Thinking and Learning Survey). It measures to which extent "a person is a 'connected knower' (CK) or a 'separate knower' (SK), whereby "people with higher CK scores tend to find learning more enjoyable, and are often more cooperative, congenial and more willing to build on ideas by others, while those with higher SK scores tend to take a more critical and argumentative stance to learning" (De Vega, 2009). For the below ATTLS questionnaire, the study in Thailand gave the following results (in []-brackets, mean value and deviation for the question are given):

1. In evaluating what someone says, I focus on the quality of their argument, not on the person who's presenting it. [4.2/1.3]
2. I like playing devil's advocate. [2.5/1.1]
3. I like to understand where other people are 'coming from', what experiences have led them to feel the way they do. [3.9/1.2]
4. The most important part of my education has been learning to understand people who are very different to me. [3.5/1.3]
5. I feel that the best way for me to achieve my own identity is to interact with a variety of other people. [3.9/1.2]
6. I enjoy hearing the opinions of people who come from backgrounds different to mine - it helps me to understand how the same things can be seen in such different ways. [4.4/1.1]
7. I find that I can strengthen my own position through arguing with someone who disagrees with me. [3.0/1.1]
8. I am always want to know why people say and believe the things they do. [3.5/1.4]
9. I often find myself arguing with the authors of books that I read, trying to logically figure out why they're wrong. [2.8/0.9]
10. It's important for me to remain as objective as possible when I analyze something. [4.4/0.8]
11. I try to think with people instead of against them. [3.6/1.1]
12. I have certain criteria I use in evaluating arguments. [3.1/1.2]
13. I'm prefer to understand someone else's opinion than to try to evaluate it. [3.5/1.4]
14. I try to point out weaknesses in other people's thinking to help them clarify their arguments. [2.7/1.0]
15. I tend to put myself in other people's shoes when discussing controversial issues, to see why they think the way they do. [3.3/0.9]
16. One could call my way of analysing things 'putting them on trial' because I am careful to consider all the evidence. [3.8/1.0]

17. I value the use of logic and reason over the incorporation of my own concerns when solving problems. [3.9/1.1]
18. I can obtain insight into opinions that differ from mine through empathy. [3.3/1.1]
19. When I encounter people whose opinions seem alien to me, I make a deliberate effort to 'extend' myself into that person, to try to see how they could have those opinions. [2.9/0.9]
20. I spend time figuring out what's 'wrong' with things. For example, I'll look for something in a literary interpretation that isn't argued well enough. [3.2/1.0]

The results were compared with a study carried out at the Autonomous University of Baja California (UABC) in Mexico during 2006 and 2008, also in the area of engineering. Although this study solely concentrated on e-learning, it is of relevance in comparing the outcome to Asian – in our case Thai – students. The comparison shows significant variations at only two questions concerning connected learning (13, 15) and one point in separate learning (14).

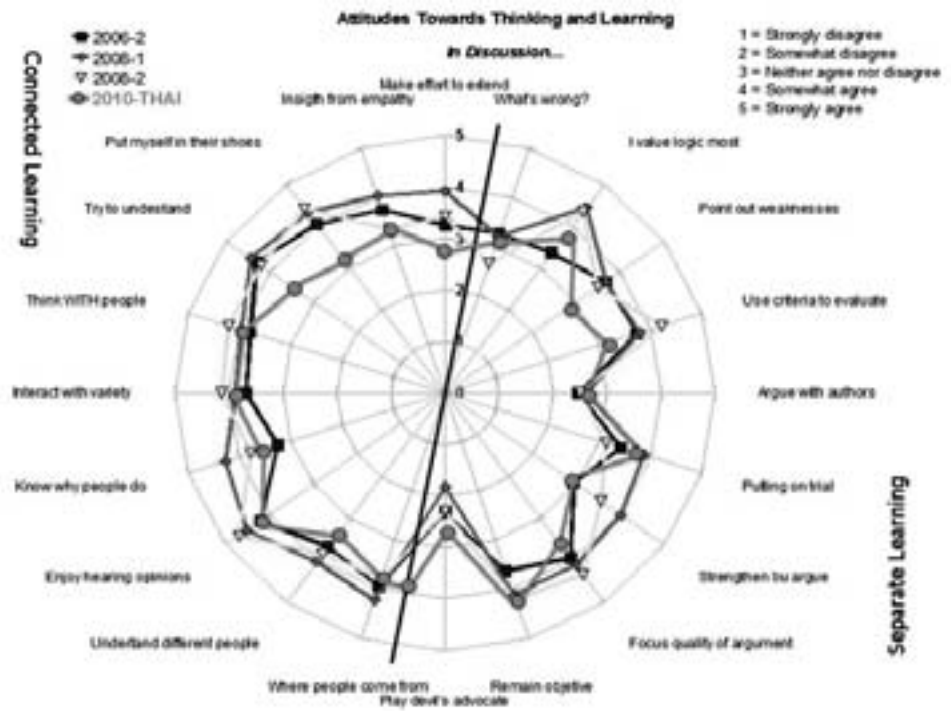


Figure 1: ATTLS results of Mexican and Thai students

4 Evaluation of Results

What Fig. 1 shows is in line with earlier studies mentioned above, namely, that Thai students can relatively quickly adapt to new learning styles, provided the learning environment is open to it. In the case of this research, the population was part of an international course, which means, the students had already been exposed to an international setting and most had already studies with western lecturers. What is more, the majority was working students or had worked prior to joining the international Master program, which suggests they are more used to working independently. This scenario is not the case in mainstream Thai university courses. It should be noted here that the comparison with a Mexican scenario is interesting in that Mexico, like Thailand, is also a newly industrializing country.

Overall, the interviewed students welcomed the student-centered approach, instigating discussions and using creative techniques like brainstorming and mind-maps. However, one student commented that Thai students prefer a passive leaning style, not an 'aggressive' one (like Western style). To further invite comments and opinions, students suggested addressing individuals directly, not to 'broadcast' questions to the entire class, i.e. no-one in particular.

As to the use of the LMS MOODLE, students generally liked the software, but found it difficult to use it as a medium for discussion. The students commented on the literature posted on the LMS, however, did not comment on other people's postings. Openly disagreeing they considered impolite. This is in line with research and findings introduced above, suggesting that it is the traditional value system that directs the learning and teaching styles. The students are eager and willing to embrace critical thinking and creative techniques, however, find it difficult do so within their cultural boundaries.

5 Recommendation and Future Work

The outcome of this study suggests improvements that can be made in the future as follows: Just as in team management, initial workshops that introduce students to more student-centered learning styles ought to be conducted. In these workshops team charters for learning teams could help redefine the role of the students and the teacher, and codes of conduct on how to deliver criticism constructively and in a way acceptable for the group can help students express their opinions freely and ask questions without compromising themselves or the teacher.

References

[Zi05] Zorn, I.: Do Culture and Technology Interact? Overcoming Technological Barriers to Intercultural Communication in Virtual Communities. ACM SIGGROUP Bulletin, vol. 25, 2, 2005, p. 8-13

[Hg03] Hofstede, G.: Culture's Consequences. SAGE, California, 2001.

[TB03] Thongprasert, N; Burn, J.: Identifying Strategies for Effective Virtual Education Delivery in Thailand. PACIS 2003 Proceedings. Paper 23, 2003 http://aisel.aisnet.org/pacis2003/23, February 3, 2010

[Tn09] Thongprasert, N.: Cross-Cultural Perspectives of Knowledge Sharing for Different Virtual Classroom Environments: A Case Study of Thai Students in Thai and Australian Universities. NIDA Development Journal, vol. 49, no. 4, 2009

[BP07] Bangkok Post, Learning Post 2007, http://bangkokpost.net/educationsite2007/cvnv2007.htm, November 22, 2007

[Mo10] www.moodle.org, November 8, 2008

[PP06] Pagram, P; Pagram, J.: Issues in E-Learning: A Thai Case Study. EJISDC, vol. 26, no. 6, 2006; p. 1-8

[AL09] Armijo, C.V., McAnally-Salas L., Lavigne, G.: Attitudes and Perceptions of Students in a Systems Engineering E-Learning Course. Acta Didactica Napocensia,. vol. 2, no. 2, 2009 http://dppd.ubbcluj.ro/adn/article_2_2_12.pdf, March 16, 2010

A Decision Tree Based Recommender System

Amir Gershman, Amnon Meisels
Department of Computer Science,
Ben-Gurion University of the Negev
Beer-Sheva, 84105, Israel
amirger, am@cs.bgu.ac.il

Karl-Heinz Lüke
Deutsche Telekom AG, Laboratories,
Innovation Development
Ernst-Reuter-Platz 7, D-10587 Berlin, Germany
Karl-Heinz.Lueke@telekom.de

Lior Rokach, Alon Schclar, Arnon Sturm
Department of Information Systems Engineering,
Ben-Gurion University of the Negev
Beer-Sheva, 84105, Israel
liorrk, schclar, sturm@bgu.ac.il

Abstract: A new method for decision-tree-based recommender systems is proposed. The proposed method includes two new major innovations. First, the decision tree produces *lists of recommended items* at its leaf nodes, instead of single items. This leads to reduced amount of search, when using the tree to compile a recommendation list for a user and consequently enables a scaling of the recommendation system. The second major contribution of the paper is the splitting method for constructing the decision tree. Splitting is based on a new criterion - the least probable intersection size. The new criterion computes the probability for getting the intersection for each potential split in a random split and selects the split that generates the least probable size of intersection. The proposed decision tree based recommendation system was evaluated on a large sample of the MovieLens dataset and is shown to outperform the quality of recommendations produced by the well known information gain splitting criterion.

1 Introduction

Recommender Systems (RS) propose useful and interesting items to users in order to increase both seller's profit and buyer's satisfaction. They contribute to the commercial success of many on-line ventures such as Amazon.com or NetFlix [Net] and are a very active research area. Examples of recommended items include movies, web pages, books, news items and more. Often a RS attempts to predict the rating a user will give to items

based on her past ratings and the ratings of other (similar) users.

Decision Trees have been previously used as a model-based approach for recommender systems. The use of decision trees for building recommendation models offers several benefits, such as: efficiency and interpretability [ZI02] and flexibility in handling a variety of input data types (ratings, demographic, contextual, etc.).

The decision tree forms a predictive model which maps the input to a predicted value based on the input's attributes. Each interior node in the tree corresponds to an attribute and each arc from a parent to a child node represents a possible value or a set of values of that attribute. The construction of the tree begins with a root node and the input set. An attribute is assigned to the root and arcs and sub-nodes for each set of values are created. The input set is then split by the values so that each child node receives only the part of the input set which matches the attribute value as specified by the arc to the child node. The process then repeats itself recursively for each child until splitting is no longer feasible. Either a single classification (predicted value) can be applied to each element in the divided set, or some other threshold is reached.

A major weakness in using decision trees as a prediction model in RS is the need to build a huge number of trees (either for each item or for each user). Moreover, the model can only compute the expected rating of a single item at a time. To provide recommendations to the user, we must traverse the tree(s) from root to leaf once for each item in order to compute its predicted rating. Only after computing the predicted rating of all items can the RS provide the recommendations (highest predicted rating items). Thus decision trees in RS do not scale well with respect to the number of items.

We propose a modification to the decision tree model, to make it of practical use for larger scale RS. Instead of predicting the rating of an item, the decision tree would return a weighted list of recommended items. Thus with just a single traverse of the tree, recommendations can be constructed and provided to the user. This variation of decision tree based RS is described in section 2.

The second contribution of this paper is in the introduction of a new heuristic criteria for building the decision tree. Instead of picking the split attribute to be the attribute which produces the largest information gain ratio, the proposed heuristic looks at the number of shared items between the divided sets. The split attribute which had the lowest probability of producing its number of shared items, when compared to a random split, is picked as the split attribute. This heuristic is described in further detail in section 3.

We evaluate our new heuristic and compare it to the information gain heuristic used by the popular C.45 algorithm ([Qui93]) in section 4.

2 RS-Adapted Decision Tree

In recommender systems the input set for building the decision tree is composed of *Ratings*. *Ratings* can be described as a relation $< ItemID, UserID, Rating >$ (in which $< ItemID, UserID >$ is assumed to be a primary key). The attributes can describe the

users, such as the user's age, gender, occupation. Attributes can also describe the items, for example the weight, price, dimensions. *Rating* is the target attribute which the decision tree classifies by. Based on the training set, the system attempts to predict the *Rating* of items the user does not have a *Rating* for, and recommends to the user the items with the highest predicted *Rating*.

The construction of a decision tree is performed by a recursive process. The process starts at the root node with an input set (training set). At each node an item attribute is picked as the split attribute. For each possible value (or set of values) child-nodes are created and the parent's set is split between child-nodes so that each child-node receives as input-set all items that have the appropriate value(s) that correspond to this child-node. Picking the split-attribute is done heuristically since we cannot know which split will produce the best tree (the tree that produces the best results for future input), for example the popular C4.5 algorithm ([Qui93]) uses a heuristic that picks the split that produces the largest information gain out of all possible splits. One of the attributes is pre-defined as the target attribute. The recursive process continues until all the items in the node's set share the same target attribute value or the number of items reaches a certain threshold. Each leaf node is assigned a label (classifying its set of items), this label is the shared target attribute value or the most common value in case there is more than one such value.

Decision trees can be used for different recommender systems approaches:

- Collaborative Filtering - Breese et al. [BHK98] used decision trees for building a collaborative filtering system. Each instance in the training set refers to a single customer. The training set attributes refer to the feedback provided by the customer for each item in the system. In this case a dedicated decision tree is built for each item. For this purpose the feedback provided for the targeted item (for instance like/dislike) is considered to be the decision that is needed to be predicted, while the feedback provided for all other items is used as the input attributes (decision nodes). Figure 1 (left) illustrates an example of such a tree, for movies.

- Content-Based Approach - Li and Yamda [LY04] and Bouza et al. [BRBG08] propose to use content features to build a decision tree. A separate decision tree is built for each user and is used as a user profile. The features of each of the items are used to build a model that explains the user's preferences. The information gain of every feature is used as the splitting criteria. Figure 1 (right) illustrates Bob's profile. It should be noted that although this approach is interesting from a theoretical perspective, the precision that was reported for this system is worse than that of recommending the average rating.

- Hybrid Approach - A hybrid decision tree can also be constructed. Only a single tree is constructed in this approach. The tree is similar to the collaborative approach, in that it takes user's attributes as attributes to split by (such as her liking/disliking of a certain movie) but the attributes it uses are general attributes that represent the user's preference for the general case, based on the content of the items. The attributes are constructed based on the user's past ratings and the content of the items. For example, a user which rated negatively all movies of genre comedy is assigned a low value in a "degree of liking comedy movies" attribute. Similarly to the collaborative approach,

the tree constructed is applicable to all users. However, it is now also applicable to all items since the new attributes represent the user's preferences for all items and not just a single given item. Figure 2 illustrates such a hybrid tree.

Consider a general case with a data set containing *n* users, *m* items, and an average decision tree of height h. The collaborative filtering based RS requires *m* trees to be constructed, one for each item. When a user likes to receive a recommendation on what movie to watch, the system traverses all trees, from root to leaf, until it finds an item the user would like to view. The time complexity in this case is therefore $O(h \cdot m)$. This might be too slow for a large system that needs to provide fast, on-demand, recommendations to users. The content based approach requires *n* trees to be constructed, one for each user. When a user likes to receive a recommendation, the system needs to traverse the user's tree from root to leaf once for each item, until it finds an item the user would like to view. The time complexity in this case is therefore $O(h \cdot m)$. Similarly, in the hybrid approach, the tree needs to be traversed once for each item, and the time complexity is also $O(h \cdot m)$.

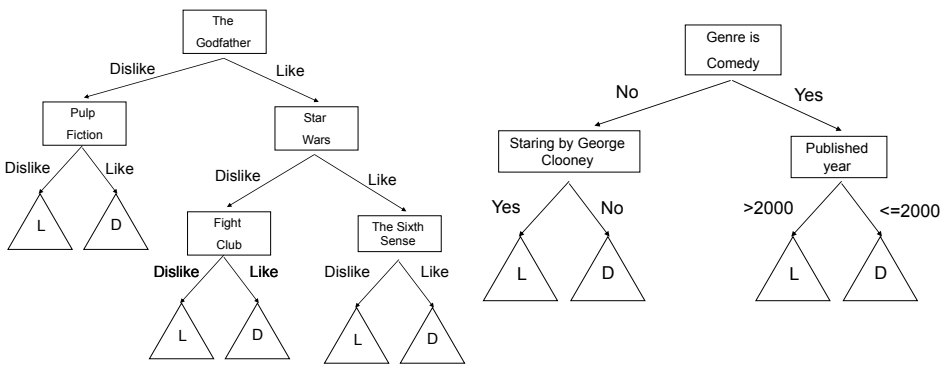


Figure 1: Left: A CF decision tree for whether users like the movie "The Usual Suspects" based on their preferences to other movies such as The Godfather, Pulp Fiction etc. A leaf labeled with "L" or "D" correspondingly indicates that the user likes/dislikes the movie "The Usual Suspects". Right: A CB decision tree for Bob.

In systems which require fast computation of recommendations and with many possible items to recommend, all the above decision tree based RS would be impractical. Therefore we propose a modification of the decision tree to better fit RS, and provide recommendations faster to users.

Our proposed algorithm is similar to the ID3 algorithm [Qui86] and uses the hybrid approach. Because we use the hybrid approach, only a single tree is needed, and the attributes to split by are only attributes that describe users. These attributes can be computed based on the user's past ratings and the content of the items, as shown in the example in figure 2 but they can also include user profile attributes such as age, gender etc. The major variation from the ID3 algorithm is in the leaf nodes of the tree. Instead of creating leaf nodes with a label that predicts the target attribute value (such as rating), we propose to construct a recommendation list out of the leaf's input set and save this list at the leaf node as its

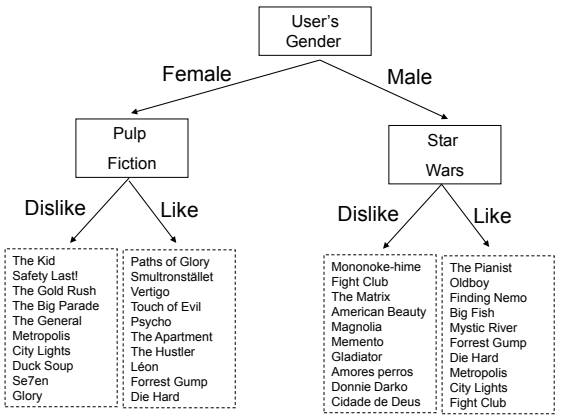


Figure 2: An example CF-CB hybrid decision tree

label. When a user wishes to receive recommendations, the tree is traversed based on the user's attributes until the leaf node is reached. The node contains a pre-computed recommendation list and this list is returned to the user. Thus the time complexity is reduced to only $O(h)$.

Building the recommendation list at the leaf node can be done in various ways. A simple solution selected here is to compute the weighted average of the ratings in all tuples at the leaf's *Ratings* set and to recommend the items with the highest weighted average first. Consider the rightmost leaf in the example tree in figure 2. For any tuple $< i, u, r >$ in the *Ratings* set (i,u,r denote an item, a user and a rating, respectively) at this leaf node we know that *u* has a degree of liking the genre Comedy more than 0.5 and a degree of liking movies with the actor George Clooney more than 0.7. All the items rated by users such as *u* appear in this *Ratings* set along with the ratings each user submitted. This leaf therefore contains the ratings of users similar to *u*. We assume that if we now pick the items which were rated the highest by the users similar to u, these would form a good recommendation for this user. Therefore, we order all items based on a weighted average (since items can appear more than once, when more than one user rated them) and set this list as the recommendation list of this leaf node. The algorithm is presented in detail in Algorithm 1.

3 Least Probable Intersections

Decision trees seek to provide good results for new unseen cases, based on the model constructed with the training set. To accomplish this, the construction algorithm strives for a small tree (in number of nodes) that performs well on the training set. It is believed that a small tree would generalize better, avoids over fitting, and forms a simpler representation for humans to understand [Qui86]. The C4.5 [Qui93] is a popular algorithm for construct-

Algorithm 1 RS-Adapted-Decision-Tree(Ratings)

```
create root node
if (Ratings.size < threshold)

    root.recommendations←recommendation_list(Ratings)
    return root

else

    Let A be the user attribute that best classifies
    the input
    For each possible value v of A

        Add a branch b below root, labeled (A=v)
        Let Ratings_v be the subset of Ratings that have
        the value v for A
        add RS-Adapted-Decision-Tree(Ratings_v) below this
        new branch b

return root
```

ing such decision trees. It uses the criterion of normalized information gain [Mit97] to pick the attribute by which to split each node. The attribute with the largest normalized information gain is picked, as it provides the largest reduction in entropy. Since this is a heuristic, it does not guarantee the smallest possible tree.

In recommender systems the input set for building the decision tree is composed of *Ratings*. *Ratings* can be described as a relation $< ItemID, UserID, Rating >$ (in which $< ItemID, UserID >$ is assumed to be a primary key). The intuition behind the heuristic proposed in the present paper is as follows. Consider a split into two subsets, *A* and *B*. The less *ItemIDs* are shared between the sets *A* and *B*, the better the split is since it forms a better distinction between the two groups. However, different splits may result in different group sizes. Comparing the size of the intersection between splits of different sub-relation sizes would not be good since an even split for example (half the tuples in group *A* and half in group *B*) would probably have a larger intersection than a very uneven split (such as one tuple in group *A* and all the rest in group *B*). Instead, we look at the probability of our split's item-intersection size compared to a random (uniform) split of similar group sizes. A split which is very likely to occur even in a random split is considered a bad split (less preferred) since it is similar to a random split and probably does not distinguish the groups well from each other. A split that is the least probable to occur is assumed to be a better distinction of the two subgroups and is the split selected by the heuristic.

More formally let us denote:

- $items(S) = \pi_{ItemID}(S)$, where π is the projection operation in relational algebra, the

set of all *ItemIDs* that appear in a set *S*.

- $O_i(S) = |\sigma_{ItemID=i}(S)|$, where σ is the select operation in relation algebra, is the number of occurrences of the *ItemID i* in the set *S*.

Let S_q (q denotes the number of ratings) be a random binary partition of the tuples in *Ratings* into two sub-relations *A* and *B* consisting of *k* and *q-k* tuples, respectively. We are interested in the probability distribution of $S_q \equiv |(items(A) \bigcap items(B)|$.

First, let us find the probability of an item belonging to the intersection. The probability of all o_i occurrences of any item *i* to be in the set *A* is $\left(\frac{k}{q}\right)^{o_i}$. Similarly the probability for all o_i occurrences of item *i* to be in the set *B* is $\left(\frac{q-k}{q}\right)^{o_i}$. In all other cases item *i* will appear in the intersection, thus the probability P_i that item *i* belongs to $items(A) \bigcap items(B)$ is:

$$P_i = 1 - \left(\frac{k}{q}\right)^{o_i} - \left(\frac{q-k}{q}\right)^{o_i} \tag{1}$$

Next, we can construct a random variable x_i which takes the value 1 when item *i* belongs to the intersection of *A* and *B*, and the value 0 otherwise. Using the above equation, the variable x_i is distributed according to a Bernoulli distribution with a success probability P_i. Thus, S_q is distributed as the sum of $|items(Ratings)|$ non-identically distributed Bernoulli random variables which can be approximated by a Poisson distribution [Cam60]:

$$Pr(S_q = j) = \frac{\lambda^j \cdot e^{-\lambda}}{j!} \tag{2}$$

where

$$\lambda = \sum_{i \in items(Ratings)} P_i \tag{3}$$

The cumulative distribution function (CDF) is therefore given by:

$$Pr(S_q \leq j) = \frac{\Gamma(\lfloor k+1 \rfloor, \lambda)}{\lfloor k \rfloor !} \tag{4}$$

where $\Gamma(x, y)$ is the incomplete gamma function and $\lfloor k \rfloor$ is the floor function.

To summarize, given a binary split of *Ratings* into two sub-relations *A* and *B*, of sizes *q* and *q-k* respectively. Our proposed heuristic first computes the size of the item-intersection, $|(items(A) \bigcap items(B)| = j$. Next, we compute the probability of receiving such intersection size in a similar-size random split using the probability$Pr(S_q \leq j)$. Out of all possible splits of *Ratings*, our heuristic picks the one with the lowest probability $Pr(S_q \leq j)$ to be the next split in the tree.

4 Experimental Evaluation

4.1 Evaluation Measures

To assess the quality of the resulting recommendations list, we evaluated the *half-life utility* metric ([BHK98]) of the movies that were ranked by the user but were not used in the profile construction. The above metric assumes that successive items in the list are less likely to be viewed with an exponentially decreasing rate. The utility is defined as the difference between the user's rating for an item and the "default rating" for an item. The grade is then divided by the maximal utopian grade. Specifically, the grade of the recommendations list for user a is computed as follows:

$$R_\alpha = \sum_j \frac{\max(r_{a,j} - d, 0)}{2^{(j-1)/(\alpha-1)}} \tag{5}$$

where $r_{a,j}$ represents the rating of user a on item j of the ranked list, d is the default rating, and α is the viewing half-life that in this experiment was set to 10. The overall score for a dataset across all users (R) is

$$R = 100 \frac{\sum_a R_a}{\sum_a R_a^{\max}} \tag{6}$$

where $R_a^{\max}$ is the maximum achievable utility if the system ranked the items in the exact order that the user ranked them.

4.2 Experimental Setup

For our experimental evaluation we used the MovieLens ([RL10]) data set, which holds 1 million ratings of 6040 users rating 3900 distinct movies. For the training set, about 10% of the movies were selected, and all ratings associated with them. These ratings accumulate to roughly 50,000 ratings for half the user set (3063 in number). The users' attributes consisted of age, occupation and gender. The same attribute can serve as a split attribute at multiple junctions in the tree as we confine the tree to use only binary splits. The RS is asked to provide recommendations for all remaining users (2977 in number), and the *half-life utility* metric is used to evaluate the results. In order to evaluate the new heuristic, we compare two recommendation systems identical in all details except for the split heuristic used. One system is using the standard information gain heuristic, and the other is using the proposed least probable intersection size heuristic. The same training set is given as input to both systems.

4.3 Results

Figure 3 compares the performance of the new intersection size heuristic criterion with the well known information gain. The horizontal axis of this plot indicates the depth of the decision tree, as the tree is being constructed. The vertical axis indicates the average utility of recommendations produced by the tree. The solid line shows the utility of the decision tree using the new criterion, whereas the broken line shows utility obtained by the information gain criterion. Both lines follow the well-known over-fitting pattern, in which the utility first increases, then decreases. Note that the new criterion dominates the information gain criterion over all depths. Specifically, the best performance of the two criteria are 75.88% and 72.71% respectively which implies a 4.35% relative improvement.

To examine the effects of the tree's depth and of the splitting criteria, a two-way analysis of variance (ANOVA) with repeated measures was performed. The dependent variable was the mean utility. The results of the ANOVA showed that the main effects of the tree's depth $F = 34.2, p < 0.001$ and the splitting criteria $F = 7.24, p < 0.001$ were both significant.

The Post-Hoc Duncan test was conducted in order to examine when the proposed criterion outperforms the information gain criterion. With $\alpha = 0.05$, starting from tree of four levels and up to the a tree of 21 levels the proposed method is significantly better than the information gain.

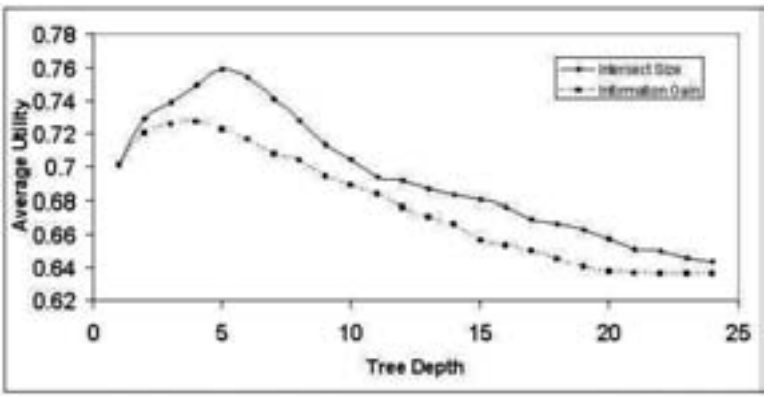


Figure 3: Comparing the average utility of the two splitting criteria on different tree's sizes.

5 Conclusions

A new decision tree based recommender technique was presented. The new technique requires only a single traversal of the tree for producing the recommendation list, by holding lists of recommended items in the tree's nodes.

A new splitting criterion for guiding the induction of the decision tree is also proposed. The

experimental study shows that the new criterion outperforms the well known information gain criterion.

Additional issues to be further investigated include: (a) Add a pruning phase which will follow the growing phase and will help to avoid over-fitting; (b) Compare the proposed technique to other decision tree based techniques; and (c) Examine the proposed method on other benchmark datasets.

Acknowledgements

The project was funded and managed by Deutsche Telekom Laboratories as part of the Context-Aware Service Offering and Usage (CASOU) project.

References

[BHK98] J. S. Breese, D. Heckerman, and C. Kadie. Emperical analysis of predictive algorithms for collaborative filtering. In *Fourteenth Conference on Uncertainty in Artificial Intelligence*, 1998.

[BRBG08] A. Bouza, G. Reif, A. Bernstein, and H. Gall. Semtree: ontology-based decision tree algorithm for recommender systems. In *International Semantic Web Conference*, 2008.

[Cam60] L. Le Cam. An Approximation Theorem for the Poisson Binomial Distribution. *Pacific Journal of Mathematics, volume 10*, pages 1181–1197, 1960.

[LY04] P. Li and S. Yamada. A Movie Recommender System Based on Inductive Learning. In *IEEE Conference on Cybernetics and Intelligent Systems*, 2004.

[Mit97] T. M. Mitchell. *Machine Learning (The McGraw-Hill Companies, Inc.)*. 1997.

[Net] NetFlix. The NetFlix prize, www.netflixprize.com.

[Qui86] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, pages 81–106, 1986.

[Qui93] J. R. Quinlan. *C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning)*. Morgan Kaufmann, January 1993.

[RL10] GroupLens Research-Lab. "The MovieLens Data Set", http://www.grouplens.org/node/73, February 2010.

[ZI02] T. Zhang and V. S. Iyengar. Recommender Systems Using Linear Classifiers. *Journal of Machine Learning Research, volume 2*, pages 313–334, 2002.

Classifying Business Types from Twitter Posts Using Active Learning

Chanattha Thongsuk, Choochart Haruechaiyasak, Phayung Meesad

Department of Information Technology
Faculty of Information Technology
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pibulsongkarm Rd., Bangsue, Bangkok 10800

Human Language Technology Laboratory (HLT)
National Electronics and Computer Technology Center (NECTEC)
Thailand Science Park, Pathumthani 12120, Thailand

Department of Teacher Training in Electrical Engineering
Faculty of Technical Education
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pibulsongkarm Rd., Bangsue, Bangkok 10800

s4970290054@kmutnb.ac.th
choochart.haruechaiyasak@nectec.or.th
pym@kmutnb.ac.th

Abstract: Today, many companies have adopted Twitter as an additional marketing medium to advertise and promote their business activities. One possible solution for organizing a large number of posts is to classify them into a predefined category of business types. Applying normal text categorization technique on Twitter is ineffective due to the short-length (140-character limit) characteristic of each post and a large number of unlabeled data. In this paper, we propose a text categorization approach based on the active learning technique for classifying Twitter posts into three business types, i.e., airline, food and computer & technology. By applying the active learning, we started by constructing an initial text categorization model from a small set of labelled data. Using this text categorization model, we obtain more positive data instances for constructing a new model by selecting the test data which are predicted as positive. As shown from the experimental results, our proposed approach based on active learning helped increase the classification accuracy over the normal text categorization approach.

1 Introduction

Twitter, a well-known micro-blogging website, has recently gained a lot of popularity among the Web 2.0 community. Increasingly, many businesses use *Twitter* as a new channel to promote their products and services including other related activities. For example, many airlines use Twitter to post special flight discount or promotions for their

followers. As with many social networking websites, Twitter is considered an important part of Web 2.0 community. Web 2.0 is a departure from traditional websites, and represents a large Internet social networking group which is constantly collecting a lot of online information. In a social networking website, people are allowed to follow other users based on their personal interests. Advertising on social networking websites is growing and interesting because it can reach a lot of customers with low overhead costs.

Twitter provides an attractive platform for advertisers to promote their brands. The customer will get information and promotions from the companies. Moreover, the customers can post their opinions or complaints to the companies. Therefore, Twitter acts as a third-party provider where partners may place advertisements on their products and services. Today many brands and companies are using Twitter to advertise, get feedback from the customers and gain more revenue.

With a large number of posts, one approach for organizing them is to apply a text categorization model. Previous research works on text categorization considered textual documents such as news articles, publications and web pages. These documents typically contain a large number of words in the range of hundreds or thousands of words. Applying a text categorization model for Twitter is very challenging due to the following reasons.

1. Twitter is a micro-blogging website which allows only short posts of no more than 140 characters. The average number of terms in each post from our corpus is approximately equal to 12.
2. Most posts are often colloquialism and consist of acronym.
3. There are a lot of the junk posts.

To construct a text categorization model, training data set is required. However, to prepare a large labeled data set (assigning each document with a class label) is time consuming and expensive. One approach to improve the performance of traditional supervised learning is by applying the active learning technique. In this paper, we apply the active learning to automatically increase the number of labelled training instances for classification. We use Twitter posts from three business types, i.e., airline, food and computer & technology. Starting with a small size of training data set, an initial classification model is constructed. To increase the training data size, we iteratively

accumulate more posts which are classified with positive label. Once more posts are automatically obtained, we construct a new improved model. The final text categorization model can be used to organize the posts into different business types.

The rest of this paper is organized as follows. Section 2 gives a review of previous research works related to different machine learning approaches especially the active learning technique. Section 3 describes our solution based on the active learning approach. Section 4 presents the evaluation results of our experiments. Finally, Section 5 depicts our conclusions.

2 Related Works

Text categorization is a well-known and widely applied machine learning technique for classifying textual documents into a predefined set of categories. Previous text categorization approaches were applied on document corpora such as news articles and web pages. The typical document corpus contains a large labelled data set in which each document instance contains hundreds or thousands of terms. One of the problem issues in text categorization is the preparation of labelled data set. To construct an effective model, a large number of documents is needed. However, manually labelling each document instance into a predefined category set is very time consuming.

There have been some solutions proposed in previous works. Cabrera Rafael Guzman et al. [CG08; CRG09] proposed the automatic extraction of unlabeled examples from the web using self-training approach to classify documents without requiring a predefined set of unlabeled data. In addition, they also proposed a method using the semi-supervised learning to solve an ambiguous word to the correct sense using unlabeled examples extracted from the web. Cheng Yong et al. [CZ09] proposed self-training classifier based on local consistency eliminating the noise and discovering the unlabeled data to join the training set by classifier to be consistent with the local neighborhood. Watson Rebecca et al. [WBC07] applied the statistical parser with sentences partial-bracketing self-training. Mao Ching-Hao et al. [MLP09] proposed a co-training method for multi-view intrusion detection and identify unlabeled data in ambiguous parts. Mojdeh Mona et al. [MC08] applied the semi-supervised learning to spam filtering. Wu Xianchao et al. [WOT09] to mining Chinese-English lexicons from large amounts of Chinese Web pages but the algorithm still had the problem of appending new mined entries into the existing seed lexicon. Yang Bishan et al. [YSW09] predicted the possible labels of the unlabeled data and the expected loss multi-label data according to the confident result of label prediction. Zheng Yabin et al. [ZTL08] discovered the constant common knowledge and built a model to fit the distribution test set by adding confident instance of unlabeled test set to training set until convergence.

We apply the active learning concept to improve the performance of classification algorithm using a small initial training set and built a better classifier.

3 The Proposed Active Learning Solution

To construct an effective classification model, a large sample size of labelled training data set is usually required. To perform data selection, normal sampling methods are not very effective, since some instances with incorrect class labels could be added into the corpus. In this paper, we focus on finding a better solution for effectively selecting high-quality training data instances. We apply the active learning concept which can help increase the performance of classification algorithm using a small initial training set to iteratively increase more training data instances.

The details of the proposed active learning technique are as follows.

1. Build the elementary classifier of each corpus using a small initial training set.
2. Classify the unlabeled set using a classification algorithm.
3. Select N instances (posts) per class from the unlabelled data set pool.
4. Append the selected instances into the initial training set.
5. Build the classifier using the improved training set and evaluate the classification accuracy at that point. (The points refer to amount of training instances after combined training set.)
6. Repeat Step 2 until the combined training set is complete.

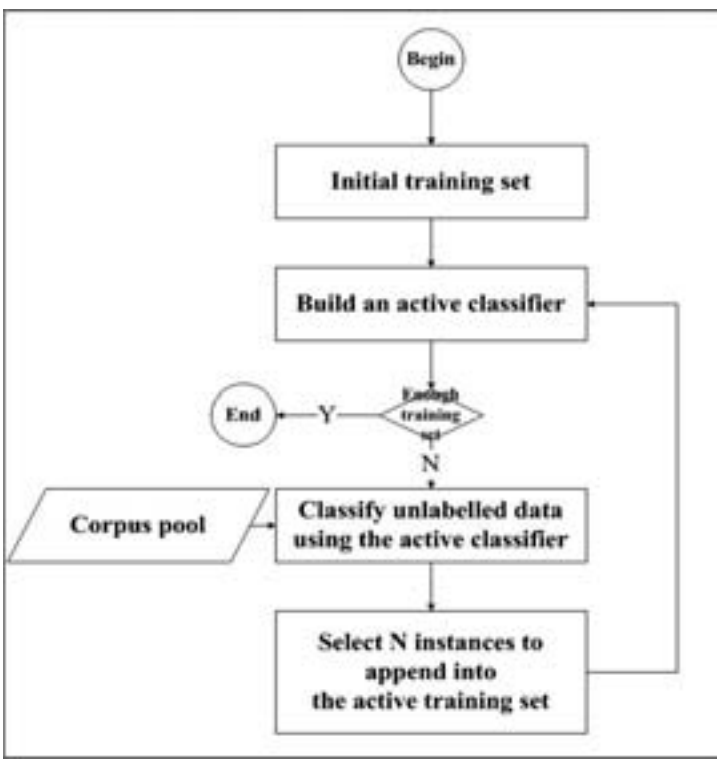


Figure 1: The Proposed Active Learning Process.

In this paper, we observed six iterations for evaluating the trend of the classification accuracy of each business type, e.g., 500, 600, 700, 800, 900 and 1,000 instances per class.

4 Experiments and Discussion

This section presents the experimental evaluation of the proposed method. This evaluation was carried out in three different business types in Twitter such as “airline”, “food”, and “computer & technology”.

Data Collection. We performed experiments using a collection of direct posts in micro-blogging of followers obtained from a selection of ten companies, which are screen names in Twitter, of the three business types: airline, food, and computer & technology. Direct Post is composted of a screen name of the receiver, which appears between the symbols “@” and wrote post together, for example, if the screen name is Google then the screen name of the receiver is @Google@. We collected the follower list of each company using Twitter API and collected direct posts from follower’s blogs using java application. In this paper, we collected the direct posts from only ten companies. And we prepared the initial training set for each corpus consisting of 1,000 posts (500 direct posts for the class label to itself, e.g., airline, food, and computer & technology and 500 normal posts labeled with “other” class). The initial training set will be plotted in the first iteration.

Data Pre-Processing. We removed all punctuation marks, numerical symbols and screen name of the receiver (consists with “@”symbol and receiver’s screen name) i.e., “@Google@”. After that, we converted all words to lowercase.

Learning Algorithm. We selected Support Vector Machines (SVM) to classify business type from the original corpus and the manual selected corpus, because, based on state-of-the-art literature, the SVM algorithm is an appropriate algorithm for many applications. We used the WEKA machine-learning environment as the experiment tool.

Active Learning Experiment. We separated the experiment into two sections. The *first* is the experiment for the manual method and the *second* is the experiment for the active learning technique.

For the initial iteration, we selected 500 instances for the individual class and 500 instances for the “other” class in the initial training set. We inputted the initial training set to the manual method of each corpus and separate our experiment into two experiment sections. Both experiments received the same data (the initial training set) then built the elementary classifier of each corpus using a small initial training set. Therefore, the classification accuracy values of the first iteration from the manual method and active learning technique all have the same values.

For the next iteration, additional instances are increased in each iteration. The manual method, we selected 100 instances per class from the corpus pool that have the relevant terms related to the business type from each corpus, and then these are inserted into the manual method experiment corpus. But in the active learning, we classify the unlabelled instances from the corpus pool using the elementary classifier and selected 100 instances per class which are predicted as positive to insert into the active learning experiment corpus. After that, we rebuild classifier and used the SVM algorithm to classify the instances in both active learning and manual experiments to evaluate the classification accuracy of each iteration. We repeat the same process until the number of positive training instances reaches 1,000 instances per class. (Each completed corpus has 1,000 instances of the "positive" class and 1,000 instances of the "other" class)

The results of the both experiment sections are as follows.

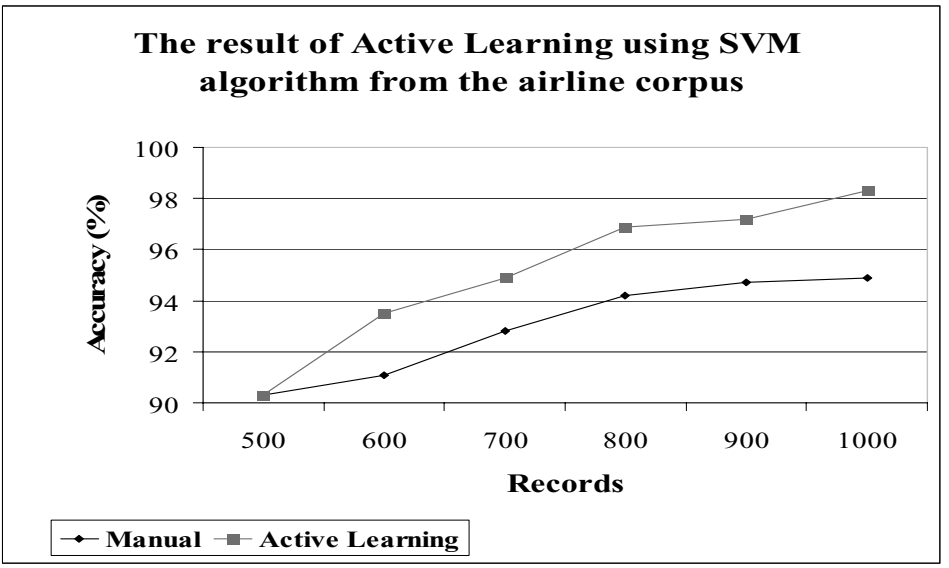


Figure 2 : The graph presents the classification accuracy comparison between the manual method and active learning method of the airline corpus using SVM Algorithm.

The trend of the airline corpus using the highest accuracy with the SVM algorithm was stable. Although the future, we will increase the new instances to appended to the initial training set continuously and using active learning technique. The classification accuracy was not lower.

Airline	Accuracy (%)		
	DT	NB	SVM
Original	65.8	78.5	**85.4**
Active Learning	78.4	85.8	**98.3**
Manual	83.1	87.8	**94.9**

Table 1: The classification accuracy (F1-Measures) of the six iteration (1,000 instances per class) of the airline corpus.

The classification accuracy comparisons of three algorithms (DT, NB and SVM) in the airline corpus are shown in Table 1. The classification accuracy of the active learning using SVM Algorithm is higher than the original method up to 12.9% (The original method was the random selection) which is still higher than the manual method, up to 3.4%. The results of Active Learning using Decision Tree and Naïve Bayes of each corpus shows high percentage of classification accuracy based on the last iteration (at 1,000 instances per class) from Active Learning.

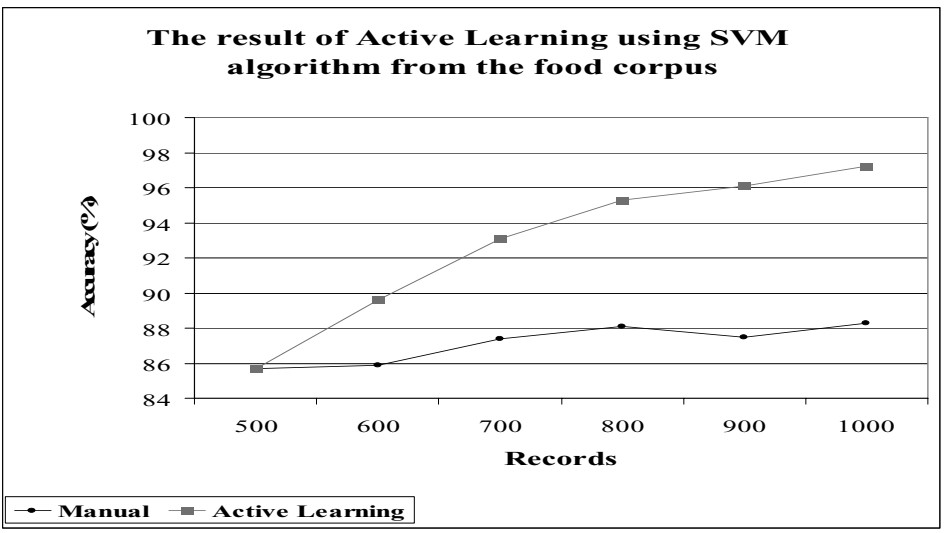


Figure 3: The graph presents the classification accuracy comparison between the manual method and active learning technique of the food corpus using SVM Algorithm.

The trend of the active learning of food corpus is very fast growing when the data set has a lot of instances.

Food	Accuracy (%)		
	DT	NB	SVM
Original	66.8	73.9	**78.8**
Active Learning	74.7	80.3	**97.2**
Manual	74.1	81.2	**88.3**

Table 2: The classification accuracy (F1-Measures) the six iteration (1,000 instances per class) of the food corpus.

Table 2 shows the classification accuracy comparisons among the original, active learning, and the manual methods in the food corpus. The classification accuracy of the active learning is higher than the original method up to 18.4% and higher than the manual method which is up to 8.9%.

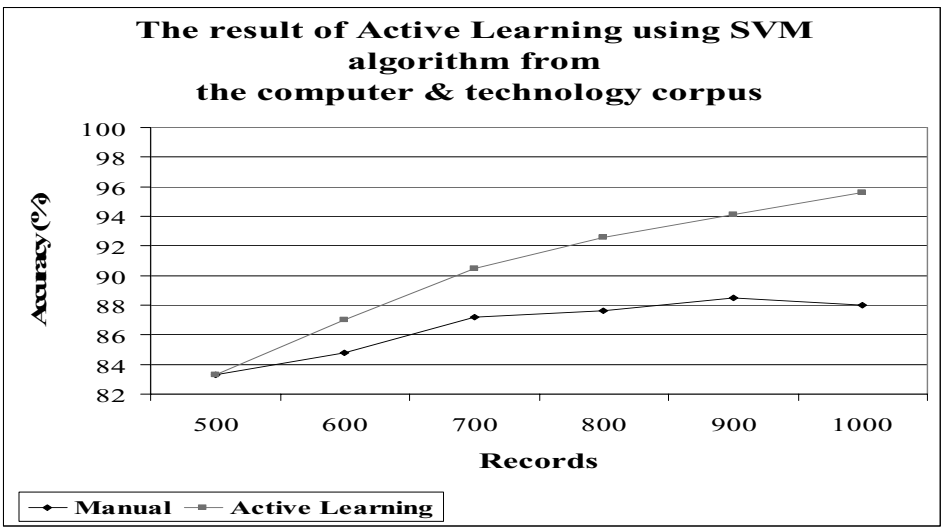


Figure 4: The graph presents the classification accuracy comparison between the manual and active learning technique of the computer & technology corpus using SVM Algorithm.

The trend of the active learning of the computer & technology corpus was steadily growing like the airline corpus but smooth and continues unlike the airline corpus.

Computer & Technology	Accuracy (%)		
	DT	NB	SVM
Original	64.6	70.2	**79.3**
Active Learning	74.8	77.1	**95.6**
Manual	77.5	80.1	**88.0**

Table 3: The classification accuracy (F1-Measures) of the six iteration (1,000 instances per class) of the computer & technology corpus.

Table 3 shows the classification accuracy comparisons of the three methods perform on the computer & technology corpus. The classification accuracy of the active learning using the SVM algorithm is higher than the original method up to 16.3% and higher than the manual method up to 7.6%.

From the results of the experiments, the active learning technique of direct posts in Twitter can improve the classification accuracy higher than the original method (random selecting) and still higher than the manual method by humans selection by observing the relevant words that appear in the posts.

5 Conclusion and Future Works

In this paper, we focus on the problem issue of data selection for building a classification model. The problem is due to, firstly, the extremely small labelled training set and, secondly, acquiring a large high-quality training set with high overhead costs. We applied the active learning technique to solve those problems. The additional training instances were acquired using the proposed active learning method to append the initial training set with 500 instances per class for improving the classification accuracy. From the experimental results, the performance of active learning method, based on the F1 measure, is higher than the original method (random selection). The three data sets: airline, food, and computer & technology, active learning method improves 12.9%, 18.4%, and 16.3%, respectively, the active learning method performs better with higher accuracy than the manual method (manually selection) up to 3.4%, 8.9%, and 7.6%.

For future works, we plan to change the number of initial training set, i.e., 100 instances per class, initial training set is a key factor to grow the classification accuracy rate, and maybe use the Latent Dirichlet Allocation (LDA) algorithm to build the topic model classifier and the training set from the active learning technique to improve the classification accuracy.

References

[CG08] Cabrera, R. G.; Gomez, M. M.: Using the Web as corpus for self-training text categorization, Springer Science, 2008.

[CRG09] Cabrera, R. G.; Rosso, P.; Gomez, M. M.: Semi-supervised Word Sense Disambiguation Using the Web as Corpus, *Springer-Verlag Berlin Heidelberg*, 2009.

[CZ09] Cheng, Y.; Zhao, R.: Self-Training Classifier via local learning regularization, *Proceedings of the Eighth International Conference on Machine Learning and Cybernetics*, 2009.

[DZL09] Ding, X.; Zhao, Y.; Li, Y.: A global Optimization of SVM batch active learning, 2009.

[JST07] Java, A.; Song, X.; Finin, T.; Tseng, B. : Why we Twitter: understanding micro-blogging usage and communities, *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis,* 2007.

[JZS09] Jansen, B.J.; Zhang, M.; Sobel, K.; Chowdury, A.: Micro-blogging as Online Word of Mouth Branding, *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, 2009.

[MC08] Mojdeh, M.; Cormack, G.V.: Semi-Supervised Spam Filtering: Does it Work?, *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, Singapore, 2009; s. 745 – 746.

[MLP09] Mao, C.H.; Lee, H.M., Parikh, D.; Chen, T. Huang, S.Y.: Semi-Supervised Co-training and Active Learning based Approach for Multi-view Intrusion Detection, *SAC'09, Homolulu, Hawaii, U.S.A*., 2009.

[WBC07] Watson, R.; Briscoe, T.; Carroll, J.: Semi-supervised Training of a Statistical Parser from Unlabeled Partially-bracketed Data, *Proceedings of the 10th International Conference on Parsing Technologies, Praque, Czech Republic,* 2007; s. 23 – 32.

[WOT09] Wu, Z.; Okazaki, N.; Tsujii, J.: Semi-supervised Lexicon Mining from Parenthetical Expressions in Monolingual Web Pages, *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Computational Linguistics,Boulder, Colorado,* U.S.A., 2009; s. 424 – 432.

[YSW09] Yang, B.; Sun, J.T.; Wang, T.; Chen, Z.: Effective Multi-Label Active Learning for Text Classification, *KDD'09, Paris, France,* 2009.

[ZR09] Zhao, D.; Rosson, M.B.: How and Why People Twitter: The Role that Micro-Blogging Plays in Informal Communication at Work, *Proceedings of the ACM 2009 international conference on Supporting group work,* 2009.

[ZTL08] Zheng, Y.; Teng, S.; Liu, Z.; Sun, M.: Text Classification Based on Transfer Learning and Self-Training, *Natural Computation, ICNC'08, Fourth International Conference, Volume 3*, 2008; s.363 – 367.

The Limbic Characteristic and El-Farol Games

Coskun Akinalp, Herwig Unger

Fernuniversität in Hagen
Fakultät für Mathematik und Informatik
Universitätsstraße 27, PRG
D-58084 Hagen, Germany
coskun@akinalp.com, herwig.unger@FernUni-Hagen.de

Abstract: Human decision behaviour with there modelling has been an area of research for a number of years in many different disciplines. The idea of this paper is to model human characteristics in a computer based game environment (minority games). In order to prove our hypotheses of non rational behaviour decision in game theory, a simulation of the well known El-Farol game is introduced and analysed. The simulation results provided new and interesting results for our approach using the limbic characteristics for analysing the behaviour of players. The simulations proof that certain characteristic more successful than others.

1 Introduction and Motivation

In today saturated markets successful companies need to differentiate themselves from other companies. Success is more related to the customer than in the past and this can only be achieved by knowing and understanding the customer – and being fully aware of his/her behaviour and motives for making selections [BM07,HJ05].
Driven by the challenge of offering customers the right product and solutions, today, offerings to customers can be understood as well under the aspects of game theory [RC08]. Is it possible to identify different characteristics based on a marketing model for complex popular problem in economic systems battling for limiting resources without communications of attendee?
The prime question is: is it possible to identify which characteristic is more successful than another one? The limbic characteristic description uses the marketing approach to convert it into a computer agent model, which can be simulated, analyzed, discussed and compared against real life.

2 Characteristics and Minority Games

2.1 Limbic Characteristics

The limbic characteristics were introduced by Häusel in [HH07] and have the aim to systemize customer motives, emotions and values and to segment target groups for marketing purposes.

The central pillar of the limbic characteristic is modelling on three key motivational and emotional systems.
These three limbic instructions are:

- The **balance** instructions (wish for security, stability, warmth; avoidance of fear and uncertainty)
- The **dominance** instructions (wish for self-assertion, power, status, autonomy; avoidance of helplessness, heteronomy and oppression)
- The **stimulant** instructions (wish for variety, novelty and reward, avoidance of boredom and lack of stimuli)

Each person is been identified by the characteristics which are based on the limbic instruction (balance, dominance and stimulant). These 3 parameters values are defining the 8 main characteristics, presented in the following table, where as a 1 expresses in the instructions a relevance and 0 not relevance for this instruction.

Balance	*Dominance*	*Stimulant*	Characteristic Type	*Type No.*
0	0	0	*Apathetic person*	0
0	0	1	*Hedonist*	1
0	1	0	*Technocrat*	2
0	1	1	*Entrepreneur*	3
1	0	0	*Harmoniser / The scared person*	4
1	0	1	*Epicure / The Enjoyer*	5
1	1	0	*Stress-Type*	6
1	1	1	*Eccentric*	7

Table I: Limbic Characteristics

2.2 The Classical El-Farol

The El-Farol bar problem is created by Arthur [AB94] to investigate limited resource problems in economics see also [DW08]. It was inspired by the El-Farol bar in Santa Fe, New Mexico, which offered Irish music on Thursday nights. The problem is set out as follows: there is a finite population of people and every Thursday night all of them want to go to the El-Farol bar. However, the El-Farol bar is quite small, and it is not enjoyable to go there if it is too crowded. So much so, in fact that the following rules are in place:

- If less than 60% of the population goes to the bar, those who go have a more enjoyable evening at the bar than they would have had, had they stayed at home.
- If 60% or more of the population goes to the bar, those who go have a worse evening at the bar than they would have had, had they stayed at home.

Unfortunately, it is necessary for everyone to decide at the same time whether they will go to the bar or not. They cannot wait and see how many others go on a particular Thursday before deciding to go themselves on that Thursday.
Formally the El-Farol Problem uses N players where N (indexed by i) represents the inhabitants of the village.
Arthur mentions the use of a set of predictors in which each agent can select out of the portfolio his strategy, but the single characteristics were not part of the discussions. In several papers different strategies for predicators were been introduced [OJS06, ST06]

and discussed. The mathematical formalization of the set of predictors has been analyzed and discussed by Challet and Marsili [CD05]. Results has been achieved and presented but in a real life web experiment of a minority game it was discovered, that humans perform better than mathematical approaches. Which lead us to investigate further in this direction.

2.3 Modified El- Farol

The basic El-Farol game was modified in such a manner that it reflects the basic idea of the minority game. Where as minority games are been defined by the rule that the minority of the players are the winner in the game. The main idea was to model limited resources in a similar to stock exchange environment.
The next table represent the decision and the results for the modified El-Farol game. The table contains 3 players represent by the left column 0 and 1 for not visiting and visiting the bar. Where as the right side of the table represents the results of the decisions per player.

Player (1/2/3) ;0=Home,1=Bar	*Player 1*	*Player 2*	*Player3*
0 / 0 / 0	*0*	*0*	*0*
0 / 0 / 1	*-1*	*-1*	*+2*
0 / 1 / 0	*-1*	*+2*	*-1*
0 / 1 / 1	*+2*	*-1*	*-1*
1 / 0 / 0	*+2*	*-1*	*-1*
1 / 0 / 1	*-1*	*+2*	*-1*
1 / 1 / 0	*-1*	*-1*	*+2*
1 / 1 / 1	*0*	*0*	*0*

Table II: Modified El-Farol Results for 3 Players

The game which is played in the simulation can be described as follows: *N=(1..N)* are the set of players where $i \in N$. The actions are $D = (0,+1)$ and describe the set of possible actions. Where 0 is **not to go** to the bar and 1 is the decision **to go** to the bar. Each simulation step is represented by *t* that is incremented after each simulation until it reaches the end. The conclusion is defined by the number of games and the rounds per each game. All players have an account which contains a starting value which is zero for every beginning of the game. For each game simulation step every player pays 1 game unit. The total sum of charges for the game unit is equal to N . Each player decides the actions for joining or leaving the bar. The players who choose the minority receive the payoff where as the players in the majority receive 0. Where *L* is number of players in the minority and K is the number of players in the majority. *N=L+K*. The payoff for each player in the minority is given by *Payoff-*L_i*=N/L*. The majority payoff is given by *Payoff-*K_i*=0*.
For the two cases in which no minority is been defined a) everybody goes to the bar b) nobody goes to the bar all players will get back there game charges of one unit.
The modifications do follow the rules of a zero sum game and can be named as zero sum minority game.

3 Simulation

3.1 Modeling of the Characteristics

As already discussed in section II the basic limbic instructions are been model in the simulation software as follows:

The **balance instruction** primarily describes the wish for stability and fear. This is mostly represented by the account value of the game. In order to make a decision the balance introduction is checking the account value if it is greater than zero or not. Afterwards due to the wish for stability it compares the software parameter of the basic orientations (to stay or to go). If the last n rounds were lost, the balance type changes his strategy compared to the previous used.
The **stimulant instruction** describes the wish for variety. If this value is 0 every decision is changed. If this value is 100 once made, a decision is kept. Another parameter in this configuration allows as well the definition of the degree of risk tolerance. This parameter is been used for the relation with other limbic instructions
The **dominant instruction** expresses self-assertion, power and status. The dominant function is calculated out of several prediction functions, which give the best probability for the win in the next round. These functions are

- Average of the n rounds of visitors
- Median of the last n rounds
- Cycle calculation of visitors over n rounds and
- Trend function of the last n rounds.

Based on this prediction functions an estimate is calculated as to which function would be the best. This information is used to build a profit margin afterwards and is used for the next selection criteria.
The decision logic is implemented in simulation Software as follows:

Type 0: This type does not follow any method. The type is deciding randomly
Type 1: This type use the decision method described in stimulant instruction
Type 2: This type use the decision method described in dominant instruction. Average/ median/trend function last 4 to 7 rounds, Cycle last 2 to 5 rounds
Type 3: This type use the decision method described in dominant instruction with Average/median/trend function last 2 to 5 rounds, Cycle last 2 to 4 rounds and the stimulant combined with the wish of risk value
Type 4: This type use the decision method described in balance instruction.
Type 5: This type use the decision method described in balance and stimulant instruction and with the wish of balance weighted value.
Type 6: This type use the decision method described in dominant instruction with Average/median/trend function last 2 to 4 rounds, Cycle last 2 to 4 rounds and the balance combined with the wish of balance weighted value.
Type 7: This type uses the decision method described in dominant instruction with Average/median/trend function for the last 2 to 3 rounds, Cycle last 2 to 3 rounds and the balance + stimulant instruction. The decision to go to the bar will be made from the majority of results between the instructions.

3.2 Simulation Results

In order to compare the results of each game per characteristic the results are presented in a matrix in which every characteristic is played against any other characteristic. The entries in the matrix can contain two entries - win and undefined. Win is indicated as standard deviation distance to the other character, which is not overlapping.

Winning / Types	0	1	2	3	4	5	6	7	Wins	Percentage
0		N/A	0	0	0	0	0	0	6	21.43%
1			1	1	1	1	1	1	6	21.43%
2				3	4	5	6	7	0	0.00%
3					3	5	N/A	7	2	7.14%
4						5	6	7	1	3.57%
5							5	5	5	17.86%
6								N/A	2	7.14%
7									3	10.71%
Undefined									3	10.71%

Table III: Results of Winning Type Matrix

Summarizing this information suggests that characteristic Type 0 and characteristic Type 1 are dominating and successful in this game and simulation environment.
This result indicates a clear strategy for the simulation environment in the modified El-Farol game. Random behaviour representatives in Type 0 and stimulant behaviour representatives in Type 1 are the winners in the game. Where as Type 1 can be as well interpreted as a modified random player. Type 5 is a stimulant player who is driven by the account information (balance instruction). This means if the character is successful do not change by behaviour which is stimulant. Type 3 which also contains stimulant instruction is been controlled by the dominance instruction which stops the stimulant instruction with the overruling parameter (setting 65% overruling of dominance).

From this simulation results it is clear, that
- there is a significant between the characteristics playing the game and
- there is a strategy for players to win the game which is to play randomly.

It is also interesting that these results do not stand in contradiction to experiments in stock markets where monkeys or other random players frequently perform better than professional brokers and analysts [WR10].As well do not stand in contradiction to the theoretical results in game theories which point out through the NASH equilibrium that the mixed strategy is a successful strategy for this game setting.

4 Conclusions and Future Work

The motivation and assumption of this work is to understand and prove that humans are NOT deciding rational. We introduced a computer environment for playing the modified El-Farol game with limbic characteristics for the players and analyzed the results. We

have proved that some characters are in special situation consistently more successful than others. These results also give answers for a successful strategy, which is random behaviour.
It seems that our model may support more complex decision processes than before. The approach of using the limbic characteristics was implemented in a non-marketing area for the first time. The modelling and results of this kind of emotional characteristics is a valid and useful approach in the area of computer science.

Future works will include a more detailed discussion of game results of groups consisting of different and more complex player combinations. Namely we think about a simulation with a character distribution derived from the German population as it can be obtained from [BC08]. In addition, the result seems to be quiet interesting for deriving strategies for the stock market extending the previously published results in [WR10]. Last but not least, for weak real time systems with a learning or adaptive scheduling strategy our results may be also applicable. In general, it is intended to answer the question, which impact limbic characters may have on the area of resource scheduling in complex networks. Consequently, it must be considered, if limbic characters maybe detected form the behaviour of the users in front of a computer system (e.g. by measuring his keystrokes or mouse movements) instead of a long questionnaire.

Bibliography

[AB94] Arthur W.B., (1994) Amer. Econ Rev. 84:406-411

[BM07] Buchanan, Mark (2007), The Social Atom: Why the Rich Get Richer, Cheaters Get Caught, and Your Neighbor Usually Looks Like You, Bloomsbury

[BC08] Burda Community Network Gmbh (2008), Typologie der Wünsche, Offenburg

[CD05] Challet D.; Marsili M.; Zhang Y-C (2005) Minority Games: Interacting Agents in Financial Markets (Oxford University Press)

[DW08] Whitehead D. (September 17, 2008), The El Farol Bar Problem Revisted: Reinforcement Learning in a Potenital Game

[HH07] Häusel, H. G. (2007), Think Limbic, Rudof Haufe Verlag Gmbh

[HJ05] Jungermann, H; Pfister, H.-R., Fischer K.; (2005), Die Psychologie der Entscheidungen, 2. Auflage, Spektrum Akademischer Verlag

[LA07] Lautin A. (2007), The Limbic Brain, Springer Verlag, Berlin

[NG10] Nymphenburg Gruppe : Citing web resource : http://www.nymphenburg.de/referenzen.html, January 25, 2010

[OJS06] Oh J., Smith S.F., (2006) A few good agents: multi-agent social learning , International Conference on Autonomous Agents , page 339-346, Hakodate, Japan

[RC08] Rieck, C. (2008), Spieltheorie – eine Einführung, 8. überarbeitete und erweiterte Auflage, Christian Rieck Verlag, Eschborn

[ST06] Schlegel T., Braun P., Kowalczyk P., (2006) Towards autonomous mobile agents with emergent migration behaviour, International Conference on Autonomous Agents , page 585-592, Hakodate, Japan

[WR10] Wirtschaftsmagazin Ruhr, Affe schlägt Börsenmakler Dr. Walter Krämer : Citing web resource:http://www.wirtschaftsmagazin-ruhr.de/fileadmin/wmr/pdf/wmr_archiv/wmr0408/Aktientipp_WMR0408.pdf , January 29, 2010

Enhancing Communities by Social Interactions in Mobile Environments

Gerald Eichler[1], Christian Erfurth[2], and Volkmar Schau[2]

[1]T-Systems Enterprise Services GmbH, IBU Telco, LoB Products and Services
Deutsche-Telekom-Allee 7
D-64295 Darmstadt, Germany

[2]Friedrich-Schiller-University Jena, Department of Computer Science
D-07740 Jena, Germany

gerald.eichler@t-systems.com, {christian.erfurth|volkmar.schau}@uni-jena.de

Abstract: The progress in www-technologies and the trends toward ubiquitous computing have led to web-based user communities in recent years. Community members meet in virtual space and share information. Member interactions are mainly based on centralised services. This paper introduces Agent Assisted Communities, an approach to enhance member interactions and community forming in mobile environments. Based on domain-specific user-defined profiles held on mobile devices, mobile software agents can look for individuals in the vicinity, who have shared interests. If found, an initial contact can be established via the mobile devices in peer-to-peer mode and then used in real life.

1 Introduction

"Are you always mobile? - If not, you are out, of the community …" Web 2.0 is knocking on your door, but how does it look like in practice? Powerful multimedia handsets support on the fly activities, which were limited to "fixed"/stationary computers, in the past. Pure web consumptions moves into user content creation and sharing, either publicly on the World Wide Web, or in restricted mode by a closed user group – the community (see fig.1). While the past was mostly driven by technology affine applications followed by service offerings, in today's networks human beings become the most important part of the game.

Online content itself is more diverse. Plain text messages are enriched by audio and video items, photographs are combined with geo-coordinates and discussion forums are established [BE01]. Public projects, like all the Wiki derivates, exceed the critical mass to be simply ignored by established content service providers and publishers.

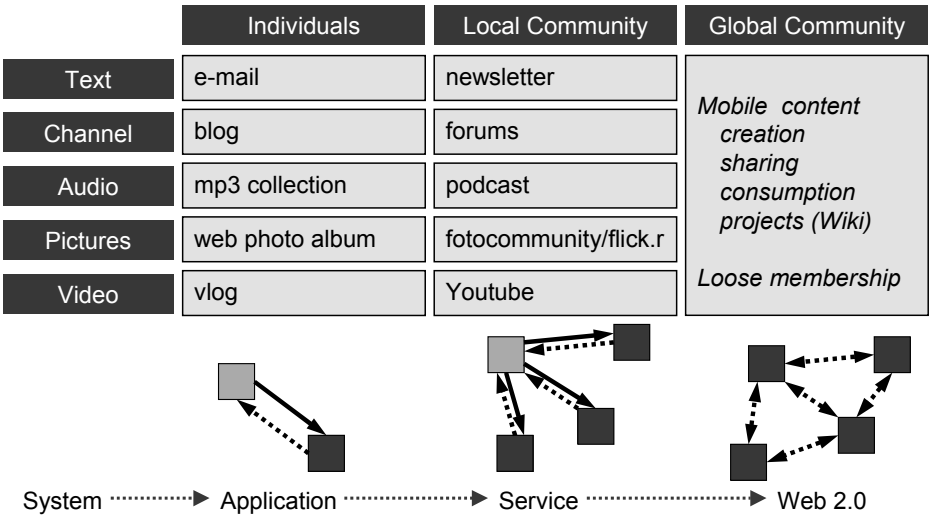


Fig. 1. Media driving mobile communities.

1.1 Application Scenarios

For several of years, our research group in the department of computer science at Friedrich-Schiller-University (FSU) in Jena, Germany, has focused on the development of loosely coupled distributed systems. In May 2005, as a result of ongoing research, the MobiSoft project [Mob] [BKR07] started to investigate future usage scenarios of mobile agent technology [Tra] [BR04] combined with mobile environments. MobiSoft is a joint research project between three partners – GODYO AG [God], the agent factory GmbH [Taf], and FSU Jena – partly funded by the local government.

Based on the identification of future application scenarios involving mobile users, about three dozen such scenarios were defined, ranked, and finally – if chosen as relevant – clustered into four integrated, more complex scenarios. Additionally, T-Systems provides a profile matching-based personalized mobile recommendation system.

- Supply chain management
  - Personal assistants support production workflows
- Project Assistant
  - Assistance for project managers
- Campus.NET
  - Platform for students and employees
- Socio-Mobile Assistant
  - Interaction in ad-hoc networks
- MediaScout
  - Recommended application for video contents from several sources

During the project, Campus.NET and Socio-Mobile Assistant have gotten stronger community aspects. Despite the fact that these scenarios have fewer business aspects,

they received more attention during presentation of the project results, especially on the CeBIT Germany 2007.

Campus.NET is intended to be a field-test scenario at FSU and benefits from a sophisticated infrastructure and potentially large number of users interested in new kinds of mobile applications. Information from different administrative areas of the university are ad-hoc integrated into more complex scenarios. So, one can imagine that a student receives on her/his mobile phone a message from his personalised assistant like this (underlined words imply more detailed information):

"Next semester, there is an interesting course on distributed systems. I've reserved for you one exemplar of the book at the library which is proposed by the course leader. Do you want to enrol for the course? And by the way, today is your favourite meal available at the student's cafeteria!"

The information is not picked up from a central point but is collected by mobile software agents migrating through the network in order to collect selected information (see fig. 2). Campus.NET will integrate community functions to form student learning groups or support team networks.

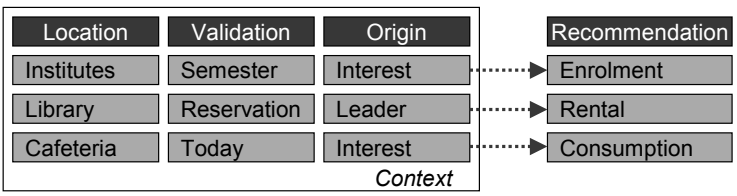


Fig. 2. Information from several origins is combined having the context in mind and leading to a recommendation.

The *Socio-Mobile Assistant* has a human-centred approach. It connects people sharing the same interests. Again, a personalised mobile software agent on a mobile device visit other mobile devices in the vicinity when reachable. Usually, the communication is done using Bluetooth. Profiles are compared by agent-to-agent communication (see chapter 2.2). In case of matches the device owners are notified. Sample scenarios are to share a cab or to find business partners at a fair.

1.2 Challenges in Mobile Communities

While implementing applications on mobile devices we faced important challenges. The current generation of mobile phones has excellent communication facilities. Unfortunately, the access via APIs does not exploit the full function set, or it is not accessible from Java-based applications. Different display resolutions and different operating systems also acquire additional effort in realisation.

Beside these implementation restrictions, there is a set of general restrictions in using mobile devices as an interface for mobile access to distributed information. These devices have limited resources. CPU performance, available memory and limited battery power demand for optimisations in applications. The most challenging limitations for usage are input and display capabilities of mobile devices. Standard desktop applications – like an internet browser even with software feature extensions like a lens for better readability – are not practically applicable to these small devices. It makes sense to have these applications on a mobile device for basic information retrieval, but they cannot be as sophisticated as their desktop counterparts. As our experiences from MobiSoft indicate, assistant-based applications taking user profiles and the current context into account help to simplify and to improve integrated usage of mobile devices.

2 User-Centred Community Approach

With the new possibilities provided by modern mobile devices and the experiences from MobiSoft, community issues and mobility can be combined more easily. We target a user-centred approach of connecting people sharing similar interests and exchanging information.

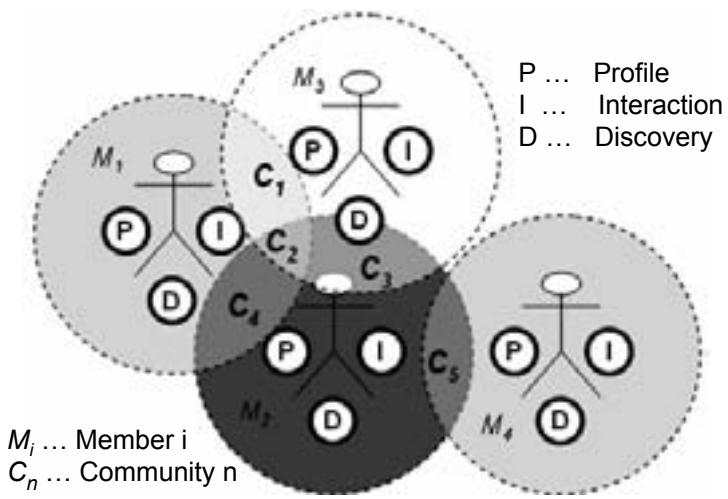


Fig. 3. Basics for an human-centered approach to form communities.

The mobile device is used for discovery of (D) and interaction with (I) people in the vicinity. People with overlapping profiles P will get connected and can enlarge or start to form communities C with two or more members M (see fig. 3). A similar approach for information sharing and collaboration without considering profiles can be found in [HKLM03]. There, information Clouds – iClouds – are introduced as the set of people sharing the same communication horizon and providing information on the basis of offer (have)- and search (wish)-lists.

2.1 Profile Handling

Profiles, usually stored in central community systems, have to be carried on mobile devices. A central user profile is just a copy of the user-centred profile. If a user has enabled her/his device to join communities (scan mode) or search for interested people the software works autonomously in the background and communicates with other devices (discovery). For the autonomous communication between these networked devices *mobile agents* will be used as a preferred solution. Their ability to migrate autonomously, compare profiles with each other and proactively inform people are a very good base for the realisation of such a mobile community – an *Agent Assisted Community*.

> *Agent Assisted Communities are defined as user-centred communities where members are supported by personalised substitutes – the mobile agents/assistants – acting as autonomous connectors to form communities including social interactions especially in mobile environments. A mobile agent knows the preferences of its owner using its profile. It acts as an enabler for the real life communication between peoples.*

Fig. 3 shows the establishment of communities. Circles around people indicate their vicinity. People sharing the same vicinity may get informed by their agents and may get in touch.

2.2 Profile Vocabulary and Structure

Quite often people search for information or look for a person who shares an interest. Thus, it is necessary to structure personal information and to introduce common terms to have a base for identifying similar interests.. Imagine that there is software which helps you to meet people you are interested in. We accept without qualification the use such software for business, for example on a conference. Next time the software scans a new contact it has to check whether you are interested or not. Now it starts a question-and-answer game with intention of finishing the game with a happy ending, namely the discovery of a common interest. During each contact you rerun the game. If you remember the questions for all interviews, there are common topics like personal information, qualifications or interests. Thus, a good approach is to build a database with all kinds of data you are interested in as well what you want to share with each other while protecting your privacy. Now we have a personal *profile* as a private view of the world.

Let's start our conference scenario again. The software relies on a profile database and a module to match a profile. It seems to be possible to use an automatic matching process for pairs of profiles. One of the assumptions is that all profiles use the same vocabulary. If there is an update, you have to extend the profile to use the same vocabulary. As a second pre-condition, we assume there is an entire contact profile which covers all topics in the world. Hence, such a profile defines all objects using a well formed vocabulary. A real instance of the profile is filled by the person who owns the profile. A comparison of two profiles could be done based on the same vocabulary. Everything outside the contact profile we have to model as knowledge of the world and requires a world view.

Fig. 4. Personalization control on a mobile phone for MobiSoft by FSU (a), personal interest selection (b) and MediaScout by T-Systems (c).

Now we have to discuss the question for the border established between knowledge of the world and the private contact profile. Where is the limit? As a second point there is the question how to define knowledge of the world. At the end we are not able to give an answer for the border question. Moreover, we cannot present a definition of knowledge of the world as well as a profile which covers all possible topics.

2.3 Domain based Subprofiling

As a next step we want to introduce our approach for profiling as a *domain based profile*. The main idea is to use the origin of each information item. The composition of a profile is a set of different *subprofiles*. Information in a subprofile is assigned according domains. A *domain* represents a group of related items, topics, or subjects belonging to a certain field. The group itself is established by related items with a common origin e.g., a surname is part of the personal information. Personal information as the origin for surname establishes a domain with the same name.

An item can be assigned to different domains. We accept without qualification there is a main topic for an assignment into one domain. Dividing the profile into pieces of domains (subprofiles) we get a distributed approach. Each part is generated by its own area if necessary. The area specific information is left in the domain and we are able to manage or extend the domain in its own context. Hence, we do not need to be proficient

in each part of the profile. Each domain could be managed by its own experts – mostly the users themselves.

Based on the idea low-dimensional domains the domain vocabulary can be distributed by sharing XML definitions. In cases of different domain vocabulary a second approach establishes the mapping process which is introduced next. Profiling based on domain profiles breaks down the complexity of an entire profile into straightforward small-sized domain profiles (divide and conquer). An entire profile is a set of domain profiles which can be extended or adapted by adding a new domain profile. Each user builds up their own personal profile.

2.4 Interaction and Information Discovery

Following the user-centred approach, a set of domain specific profiles is spanned by mobile users. Via profile matching, contacts between users can be established. Looking at the communication between two participants the first step is to *identifying the shared domains* and the vocabulary used to perform the matching. As a first approach a common communication protocol for syntactic comparison is applicable.

Once we have found "compatible" domains represented by their subprofiles S in the first step, an *in-depth look at the semantics* is necessary to check whether used vocabulary terms have the same meaning. In [WR01], the DOGGIE approach enables agents to locate similar semantic concepts. This promising approach can also be helpful to identify homonymous words with different meaning and synonymous words.

After passing these two steps, the matching process can be started in order to identify shared interests. Fig. 5 indicates an overlap of subprofiles S1 and S4. As the timeframe is unknown during which an intercommunication between two mobile devices occurs, information exchange will follow three phases:

1. fast ID tag exchange – for later re-identification
2. chunk exchange – to support re-linking via the community backend
3. information exchange – to support local P2P ad-hoc communication

[KM01] proposes an information-theoretic approach to compare interests of users. The interests are captured in a weighted ontology of keywords.

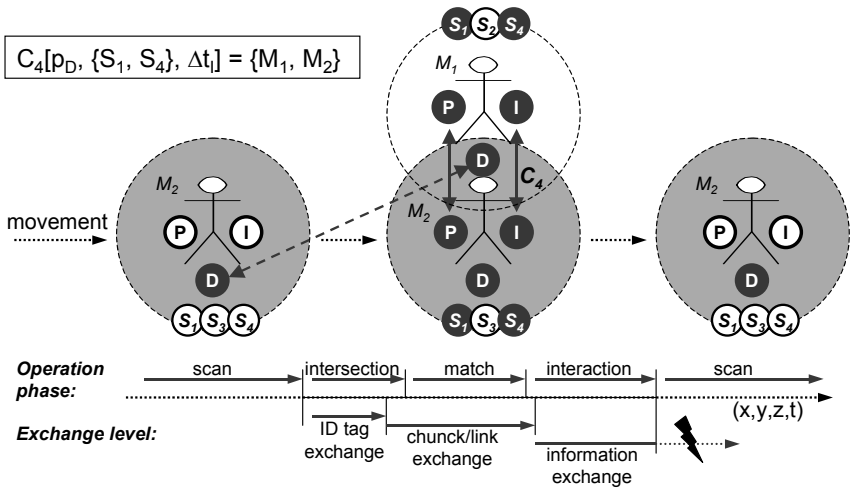


Fig. 5. Ad-hoc connectivity between moving mobile devices with a probability p_D over a limited time slot Δt.

Every established contact with other users as potential community members is important to form a community and to discover the (shared) interests of others, even if only one compatible domain has found by the contact and the other mentioned steps were not successful. Of course the contact is more interesting when shared interests could be identified.

As a result of this approach, communities are formed in a user-based manner. In contrast to a traditional client-server approach, the user is put into the middle. In addition, this user-centred approach integrates mobile and non-mobile users. Due to characteristics like autonomy, personalisation and adaptability, mobile agents are an adequate connector between users and their communities.

3 Integration of Traditional Community Systems

Mobile communities equipped with functions to involve new partners as proposed in the last section can not live without a stable backbone containing centralized services and knowledge. Typically, an internet community uses community centred services (Wiki, Forum, Blog, etc.) or community supporting services (del.icio.us, digg, slashdot, reddit, studiVZ, etc.) to interact, to exchange information, and to gain benefits.

Beside these content-oriented services there are also meta services: (central) directory of services, community explorer (which communities are available), or member areas. Usually, such services are installed on stable server infrastructures. In spite of the central characteristic of such services, there is no need to have a single central server which limits scalability. In most cases a community maintains their own infrastructures.

A central question is how to combine mobile devices with traditional community software systems. Internet browsers are typically used to participate in a community. As mentioned in chapter 1.2, this method of using community software from mobile devices is not feasible due to strong restrictions in human-device interactions. Extensive content contribution can not be expected from a mobile device.

Fig. 6 gives a schematic overview of the connection between a potential mobile community member and her/his community. With the help of an "intelligent" peer-to-peer middleware (P2P MW) like mobile agent technology, the integration of mobile community members is possible in an efficient manner. Simple tasks can be handed-over to assistants acting autonomously and proactively within a networked environment. The personalised assistant can also act as a proxy for the member and informs her/his on relevant events (see chapter 1.1).

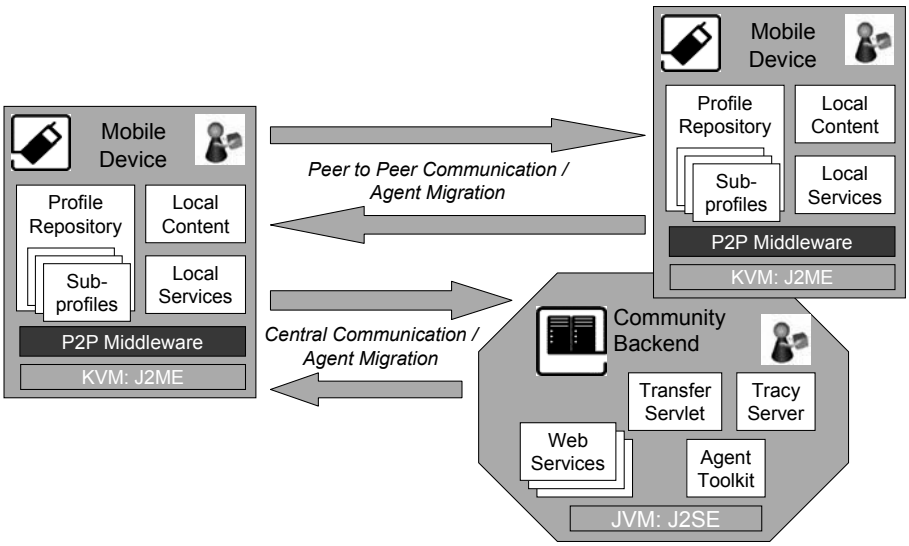


Fig. 6. Local and central communication.

In fig. 7, the combination of mobile devices and traditional community software systems is shown to form a Human-Centered Mobile Community (HCMC). Thereby, mobile clients can also extend the possibilities within the community by creating a virtual pool of services and information distributed on the mobile part of the network.

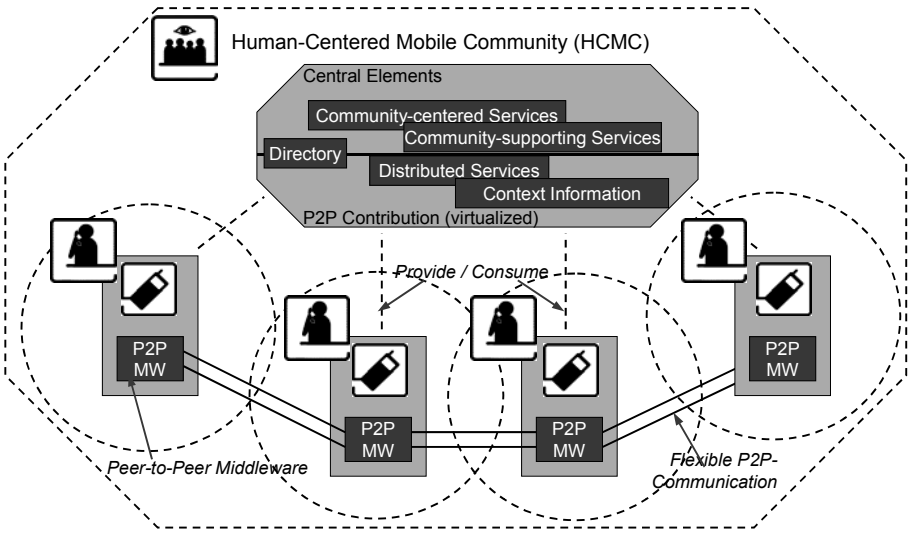


Fig. 7. Integration of mobile elements and central community services.

4 Summary and Further Work

This paper proposes an Agent Assisted Community to support mobile community members, integrate them into existing community infrastructure, and extend the community by social interaction in mobile environments. The focus of the approach is the mobile user and not centralized community services. Domain specific profiles are held on mobile devices. Mobile agents running on mobile devices are ready to compare these profiles in a 3-step process with profiles of people in the vicinity of the user. Overlapping interests will be fed back to the users to establish a contact. The defined process will be investigated in more detail as part of the research at FSU.

The paper also comes up with challenges for mobile device usage for communities. Experiences from device handling in projects are reflected. From both MobiSoft scenarios and T-System's trials addressing topics like "Personal Shopping Assistant" or the mobile recommendation application "MediaScout" the following lessons we learned:

1. *Bilateral cross matching* between *user offers* and *search requests* is essential.
2. Basic *community services* could be modelled as offers.
3. A profile could be seen as an instance of an *ontology*.
4. *User profiles* constructed of domain specific subprofiles must be easily manageable, namely switchable.
5. *Context awareness* can efficiently assist the subprofile management.
6. Beside others, the use of any kind of *tags* e.g., Bluetooth tags, visual codes, or a GPS location can be a *profile activation trigger*.

More interesting aspects which one might look at are even more in the social area of potential users e.g.,

- Is the usage of mobile devices as a community interface accepted?
- Has usability reached an appropriate level for ad-hoc mobile device usage?
- Is there a correlation between age and gender of potential users and the acceptance of usage?

Consolidated findings in this interdisciplinary area are essential for a successful application of future technologies in the mobile sector. The user is a central element in ubiquitous computing.

References

[BE07] Böhm, A.; Eichler, G.: *Examining the future potential and possibilities of emerging mass market location-based services.* Contribution to Multimedia Handsets (MMH'07), Amsterdam, The Netherlands, March 2007.

[BKR06] Braun, P.; Kern, S. and Rossak, W.: *Mobisoft: An agent-based middleware for social-mobile applications.* In Robert Meersman, editor, First International Workshop on MObile and NEtworking Technologies for social applications (MONET'06), Montpellier, France, In conjunction with OnTheMove Federated Conferences (OTM'06), Lecture Notes in Computer Science. Springer Verlag, Oct 29 2006.

[BR04] Braun, P. and Rossak, W.: *Mobile Agents: Basic Concepts, Mobility Models, and the Tracy Toolkit*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[God] GODYO AG. URL http://www.godyo.de/

[HKLM03] Heinemann, A.; Kangasharju, J.; Lyardet, F. and Mühlhäuser, M.: *iClouds - Peer-to-Peer Information Sharing in Mobile Environments*. In: Proc. Euro-Par 2003. Int. Conf. on Parallel and Distributed Computing, Klagenfurt, Austria, 26-29 August 2003, LNCS Series. Harald Koch, László Börszörményi, Hermann Hellwagner (Hrsg.). pp. 1038-1045, Springer Verlag, Heidelberg etc., Germany, 2003.

[KM01] Koh, W. and Mui, L.: *An Information Theoretic Approach to Ontology-based Interest Matching*. International Joint Conference on Artificial Intelligence (IJCAI) Ontology Learning Workshop, Seattle, WA, 2001.

[Mob] MobiSoft – Project Home. URL http://mobisoft.informatik.uni-jena.de/

[Taf] the agent factory GmbH. URL http://www.the-agent-factory.de/

[Tra] Tracy2 – the mobile agentsystem. URL http://www.mobile-agents.org/

[WR01] Williams, A. B. and Ren, Z.: *Agents teaching agents to share meaning.* In Proceedings of the Fifth international Conference on Autonomous Agents. Montreal, Quebec, Canada. AGENTS '01. ACM Press, New York, NY, 465-472, 2001.

Actors–media–qualities: a generic model for information retrieval in virtual communities

Gregor Heinrich

Natural Language Processing Group
Department of Computer Science
University of Leipzig
gregor@arbylon.net

Abstract: The article presents a model of the structural properties of virtual communities and the information they can access. It argues that a large part of the information – and actually knowledge – present in virtual communities can be identified by a graph structure that consists of three node types – actors, media and qualities – as well as the relations that connect them. Based on these relations, information retrieval and other inference mechanisms can be mapped into the model.

1 Introduction

Virtual communities have become a major factor for the design of information systems and Web-based applications, giving rise to community-based information infrastructure. Recently, this development is apparent for instance in the emergence of social computing and the "Web 2.0" [WCZM07], as well as in the paradigm shift in enterprise knowledge management from techno-centric approaches towards tacit knowledge and social capital [HWW03, Got05]. It even extends to the increasing importance of peer-to-peer systems [WB05] and collaboration grids [Sto07] for information management and processing, where peers or grid nodes act in lieu of humans and can be considered members of generalised virtual communities.

A major reason for this shift towards community-based systems is that the information available from in such approaches includes an added value that is impossible to generate or capture using classical, purely content-based approaches, because it directly gains from the intelligence, creativity and social behaviour of people. On the "supply" or authoring side, this added value consists of processes to generate content interesting for the community, which is done by providing infrastructure to easily author or make available contributions, to ask or answer questions (Web 2.0, peer-to-peer), to supply and exchange explicit and tacit knowledge (knowledge sharing infrastructure) or to offer computational services (grid). On the other, the "demand" or retrieval side, the added value consists of processes to provide social decision support, which is done by collaborative filtering, feedback mechanisms and importance measures that often use the relational structure of the community (social recommenders, reputation systems). Further, by offering suitable

infrastructure to facilitate and amplify social processes, community-based systems tend to reinforce the identification of their users as members of a community and thus create the motivation to contribute to the community.

Objective. In this article, we investigate the question of how to capture some of the added values that community-based approaches offer and how to combine them with content-based approaches in a reasonably compact model. Such a model is envisioned to describe the structure of community-based information either for comparison and classification of existing systems, or as a structural basis for new developments.

The final goal that this work contributes to is to build infrastructure for *access* to the knowledge that exists within a community, i.e., the demand side of the infrastructure. However, one of the features that makes users of community-based systems unique compared to those of others, is that they are on both sides of the system, supply and demand, i.e., are contributors and retrievers alike. Looking at retrieval, or, more generally, inference approaches cannot therefore completely exclude the content creation processes in communities.

Outline. This paper continues with a more detailed discussion of the requirements needed for the envisioned model in Section 2. As the core contribution of this article, the model itself is proposed in Section 3 and applied to a practical example in Section 4. Finally, after a short statement on related work in Section 5, the present approach and future research is discussed in Section 6.

2 Requirements

This section derives the properties of the envisioned model of virtual communities by specifying requirements. There are three major types of requirements: First we need to identify the scenarios that should be covered (Section 2.1) and need to define how specific the model should be to them (Section 2.2). Finally, the types of community knowledge of the scenarios that should be addressed in the model need to be considered (Section 2.3).

2.1 Requirement 1: Usage scenarios covered

The main purpose of the envisioned model is to *represent a virtual community to support information access scenarios and associated inference tasks*. Such tasks are for instance to find community members and documents according to certain criteria. The scenarios for information access include:

- Expert finders: Systems that allow to find people who have expert knowledge in a given topic, based on profile information, document content and authoring information (e.g., MITRE [MDH00], AnswerGarden [AM96], XperT [Hei04]);

- Digital libraries: Systems that allow to access documents where the community consists of the authors who mutually cite their articles or monographies (CiteSeer [GBL98][1], the ACM Digital Library [Whi01]);
- Collaborative authoring: Systems that allow community members to contribute to a collaborative information repository, either *ad hoc* asynchronous communication (mailing lists, forums) or as "Web 2.0" tools like blogs, wikis [WCZM07]) and other approaches that make accessible and allow users to contribute content (Twitter, flickr and YouTube) or meta-content (structured data as in IMDB, or tags as in CiteULike and del.icio.us);
- Social network platforms: Systems that offer self-authored personal profiles and connections as dynamic contact and friends lists (Xing, myspace);
- Peer-to-peer systems: Systems that distribute content over a community of peer modules (that can themselves represent a community of people) and can be considered "generalised communities" (SemPIR [WB05]).

2.2 Requirement 2: Scope and specificity

The model is primarily used in early stages of system design where the working concepts are decided and basic algorithmic considerations are undertaken (cf. [BFHV03]). Therefore, the model should be generic enough to be *independent of scenario specifics* like particular types of persons or documents. This also makes it suitable for representation and comparison of a wide variety of existing and new systems and scenarios. In fact, the result may be a form of data model or ontology whose structure can be specialised for particular scenarios in question, similar to meta-modelling methodologies (cf. [Bez06]).

2.3 Requirement 3: Information and knowledge types addressed

Many systems for community-based information infrastructure are not only used to retrieve information but actually knowledge. For these cases, it is inevitable to not only *represent documented information as explicit knowledge*, but to also *integrate tacit knowledge* [NT95, Boi99] and possibly *social capital* [Les00] in the model. This way, a great part of the added value can be captured that is ascribed to community-based approaches compared to purely content-based ones, as suggested by the literature (see, e.g., [Wen98, HWW03] covering Communities of Practice).

The access to knowledge needs a few more remarks. By definition, tacit knowledge – or "knowing" as a process rather than a state [Pol74] – is restricted to individuals or groups of individuals and it is in most cases difficult to write down (to "externalise" [NT95]) because it depends on intangible factors like experience, procedural knowledge, special

[1] Registration of articles with CiteSeer is actually a community-based process, as well.

talents, cultural background and norms that can only be made available to others by direct interaction. Social capital as a form of collective tacit knowledge [DGKT03] supports this interaction by holding together communities and being the basis for offers of help needed to achieve shared goals or to solve problems [Put00].

To give an example, in a newsgroup the experience of an expert answering a complex question cannot be written down in its entirety; it is tacit. Further, the fact that such exchange works at all is often a result of the social capital established within the community, which is tacit as well. The predominant way to approach the problem of making available such tacit "assets" is to identify the expert, e.g., from a profile, from previous answers or articles, by explicit recommendation, possibly confirmed by the location within a social network that reflects the communication within the community.

Therefore, "cues" to tacit knowledge (and social capital) in the community are important auxiliaries. Locating both tacit and explicit knowledge extends the notion of information retrieval, and in the following we use the term information retrieval to refer to this more general form, avoiding new definitions like "knowledge retrieval" because at their core, the basic approaches are those of information retrieval [BYRN99], and tacit assets should be represented in the model as explicit cues that point to them. The types of such cues are manifold, and we take an "inductive" approach and summarise sources that are commonly used and need to be represented in the envisioned model:

- Authoring and reference information: Tacit knowledge leaves traces in documents that are created in the community, either by experts themselves ("authoring") or via reference in documents by other authors, such as in reports and in scientific citations ("reference"). This is not retricted to text content, like scientific authoring, but also in non-textual media, for instance in the way a movie is edited by an expert editor. Authoring and reference information is captured "en passant" from the existing processes in the community.

- Profile information: The existence of expertise can be catalogued using questionnaires, interviews, structured CVs and other means that are "actively" or explicitly applied to capture the existence of tacit knowledge, as in many knowledge and skills managment approaches. However, many tacit skills are unknown even to the expert, and in these cases, only by interaction and problem solving can tacit knowledge be located and may be "implicitly" captured by tracking collaboration. Both active and implicit ways to capture specific traits and properties of users contribute to profiles.

- Social network information[2]: Important pieces of tacit community knowledge are identified from the structure of the community, i.e. the position of individuals and groups in the social network. Depending on the type of relations available as a representation of the real social network, this may allow identification of experts by their embedding into clusters of other experts, as well as possibly capture cues of social capital, such as trust, recommendation and reputation, which are important prerequisites to sharing of tacit knowledge through collaboration.

[2] In "genereralised communities" like peer-to-peer networks, social network information does not represent social capital proper but similarly, relational properties of the network are used as cues to identify items of interest.

3 The actors–media–qualities model

The requirements collected in Section 2 yield a set of qualitative input factors to develop the model. With the focus on information access, Requirement 1 implies the need for a semantic representation of the items in the model that can be used to evaluate the relevance to a query, i.e., some sort of profile or set of "qualities" that can be associated with the items. The generic scope (avoidance of scenario specifics) from Requirement 2 implies what in ontology design is called "minimal ontological commitment", i.e., the restriction of the model to a minimum of elements [Gru94]. Requirement 3 implies on one hand the representation of explicit knowledge items, which can be modelled as a documents or, more general, "media", on the other cues of tacit knowledge, i.e., the types of sources in the list in Section 2 need to be included. Authoring and reference demand for a connection of documents or media with community members, and profiles can be considered special cases of documents. Finally, social network information indicates the appropriateness of a graph-based representation of the community.

Fortunately, such a graph representation is flexible enough to be the structural basis for the entire model. Authoring information etc. can be expressed by including into the network media items and connections with the authors. Moreover, semantic or other qualities associated with the items can be directly included as nodes in the graph representation.

In the next subsections, we define the model in terms of a graph structure. We first introduce node types in Section 3.1, its edge types in Section 3.2 and finally the complete model structure in Sec.3.3.

3.1 Defining entities

We define AMQ entity types similar to classes in ontology or software analysis and denote them by calligraphic letters like $\mathcal{A}$. Our model supports subsumption/inheritance (*is_a*) relationships, i.e., entity types may have a hierarchy of subtypes that are denoted in italic type $A \in \mathcal{A}$ etc. Further, it supports aggregation (*has_a*) relationships. Instances, i.e., objects that represent the actual data, will then be denoted in lower case $a \in A \in \mathcal{A}$. Considering type hierarchy, for simplicity we will use the shorthand $a \in \mathcal{A}$ to denote that a is an instance of some type $A \in \mathcal{A}$. Three root entity types are proposed to model a community according to the above requirements: actors, media and qualities.

Actors, $a \in \mathcal{A}$**,** are entities that represent everyone/everything acting in an autonomous way, which implies actions like to *write*, *collaborate*, *query*, *study* or to *assess*, in addition to explicit (verbal) knowledge (to *know*). These actions will later be defined as relations. Actors bear tacit knowledge, and naturally represent people and groups of people that engage in knowledge sharing and interaction. As a special case, intelligent agents can be considered actors although they are often represented via explicit rule sets. Subsumption of actors is defined (an author *is a*n actor), as is aggregation (a group *has a* number of authors).

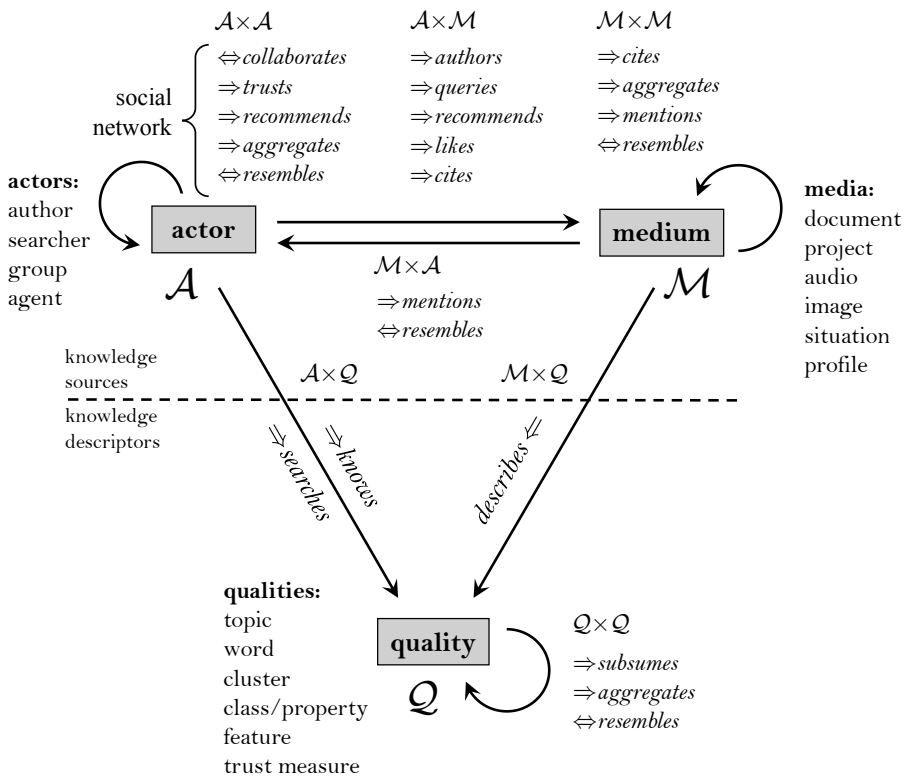


Figure 1: Structure of the AMQ model, with example entity and relation (italics) types. Directed relations between classes denoted with ⇒, undirected ones with ⇔.

Media, $m \in \mathcal{M}$**,** are entities that contain information and "react" to actors. Media bear explicit, i.e., verbal or numerical knowledge, and can be thought of as a generalisation of documents to all formats that can contain explicit knowledge or serve as cues to knowledge, including for example project and course documentation, audio tracks and images, video clips and other artefacts. In addition, user queries fit into this scheme as "reciprocal" media (requesting rather than providing knowledge). Further, user profiles behave like media. Like actors, media allow subsumption (a document *is a* medium) and aggregation (a book *has a* number of chapters).

Qualities, $q \in \mathcal{Q}$**,** are entities that provide a set of attributes to describe actors, media and other qualities in a way that inference can be performed on them. This inference includes comparison using a distance or similarity metric and consequently retrieval. Qualities as the units of knowledge representation are one key to mapping existing inference and information retrieval methods into the model and can be used develop new models. Considering existing metadata frameworks, full-text approaches like [AM96] use inverted indices as representations of qualities. In Semantic Web-based information retrieval, reasoning requires qualities to be defined with formal semantics, i.e., $\mathcal{Q}$ is the representation of an

ontology that can be queried or reasoned upon. And when using latent-semantic models like latent Dirichlet allocation [BNJ02], automatically inferred latent variables (commonly referred to as latent topics) are the qualities to describe the entities in the model. Finally, when dealing with multimedia data, content-based features can be used as qualities.

For a schematic view of these entities, see Fig. 1, where they are displayed as rectangles, along with examples of typical classes they subsume. With regard to its three root entity types, we call the model the *actors–media–qualities model* or *AMQ model*.

3.2 Defining relations

With the entities allowing to represent real-world items in the model as instances of one of the actors, media and qualities types (or subtypes), relations between them provide the actual information used for inference and retrieval. Relations in the AMQ model are quantified by some weighting function $w(x, y, R) : \mathcal{X} \times \mathcal{Y} \to I_R \subset \mathbb{R}$ with I_R the set of allowed weighting values for relation of type $R \in \mathcal{R}$ and $\mathcal{R}$ being the set of all relation types. Depending on the relation type, this interval can be discrete and binary, $I_R = \{0, 1\}$, probablistic, $I_R = [0, 1]$, some distance or similarity measure, $I_R = [0, \infty)$, or any other set depending on the semantics of the relation. Typically, relations restrict the node types $\mathcal{X}$ and $\mathcal{Y}$ they map to each other (domain and range), and may be directed or undirected. For brevity, we represent relations by their weighting function, using the entities a, m and q as defined above. Typical relation types as given in Fig. 1 will be discussed in the next paragraphs.

Media–quality relations, $w(m, q, R) : \mathcal{M} \times \mathcal{Q} \to I_R$, describe the semantics of media. Explicit knowledge cues in media items use the media–quality (*mq*-) relation *describes*, denoting explicit knowledge in a particular subject. Here, $I_R = [0, 1]$ is a function that maps to a relevance value for the subject, and to describe one this subject, weights are established from the medium to all possible qualities q_i that this subject is composed of. Combining the weightings of the *mq*-relation over several qualities $\vec{q} = \{q_i\}_i, i \in [1, K]$ can be expressed by introducing a shorthand for a vector weighting function $w(m, \vec{q}, R) : \mathcal{M} \times \mathcal{Q} \to (I_R)^K$ over all qualities q_i. This can for instance represent a vector of topic probabilities or a set of binary association functions with the range of ontology classes. When trying to retrieve relevant documents, the subject must be expressed as a set of qualities, which themselves are extracted from a query text or other object (as a "reciprocal" medium).

Actor–quality relations, $w(a, q, R) : \mathcal{A} \times \mathcal{Q} \to I_R$, describe the semantics of knowledge associated with an actor. The central relation with respect to knowledge cues is the actor–quality (*aq*-) relation *knows*, which, however, is not directly observable. In most approaches in the literature, the *knows* relation is inferred, for instance from *author*ship: For example, in the MITRE [MDH00], AnswerGarden [AM96] and XperT [Hei04] systems, knowledge cues from documents are used via the *describes* relation, and experts are inferred via the actor–media (*am*-) relation *authors*. The Author-Topic Model [RZGSS04],

however, directly extracts latent topics for actors, implementing the *knows* relation directly. Opposite to this "supply" dimension of knowledge, the "demand" of specific knowledge can be evaluated for actors, which is reflected by the actor–quality relation *searches* that can be inferred via the *describes* and *queries* relations (for all explanations, see Fig. 1). Comparing the qualities of both supply and demand dimensions enables the functionality of matchmaking systems like that in [RSW05].

Actor–media relations, $w(a,m,R) : \mathcal{A} \times \mathcal{M} \rightarrow I_R$ or $w(m,a,R) : \mathcal{M} \times \mathcal{A} \rightarrow I_R$, describe the association of an actor with a medium or vice versa. Actor–media (*am*-) relations usually derive from authoring and reference information (*authors*, *cites*, *recommends* etc.). Further, query actions by actors are special types of AM relations (*queries*). Typically, such information is often explicit and can be extracted a priori as a basis for inference. Inferred *am*-relations are used in collaborative approaches to express recommendation (*recommends*) and preference (*likes*).

Media–media relations, $w(m,m',R) : \mathcal{M} \times \mathcal{M} \rightarrow I_R$, describe mutual relationships between media. Media–media (*mm*-) relations play an important role in citation networks and digital libraries, both as references and aggregation. Like *am*-relations, they are often explicit and can be used as basis for inference. An important inferred relation for retrieval is similarity (*resembles*).

Actor–actor relations, $w(a,a',R) : \mathcal{A} \times \mathcal{A} \rightarrow I_R$, describe social structure of the community and other relations between actors. Actor–actor (*aa*-) relations can represent information on social capital in the community. ReferralWeb [KSS97], for example, uses relational cues such as friends, colleagues, and co-workers, Opal [HKJ+05] in addition ratings between collaboration candidates [DM04]. Depending on the application case, different types of inference are possible. An example is to find an actor who is an expert in a topic and trusted by reputable actors. The inference based on explicit cues is the same as described for *aq*-relations, but the set of relevant actors now is filtered via appropriate network criteria, such as shortest path or reputation measures that aggregate weighted ratings (see [KSS97, PSD03] and references therein). An alternative way is to perform inference in an integrated manner is to use statistical relational learning techniques that integrate semantic and relational steps of inference (see, e.g., [Nev06]).

Quality–quality relations, $w(q,q',R) : \mathcal{Q} \times \mathcal{Q} \rightarrow I_R$, map knowledge description frameworks into the AMQ model. For instance, for ontologies quality–quality (*qq*-) relations may include RDFS or OWL relations (e.g., *rdfs:subClass*, properties or aggregations). The AMQ model does not make any commitment on the formalism for *qq*-relations, allowing to include axiomatic descriptions of qualities, for instance to use the results of Semantic-Web inference, or hierarchical relations between latent topics. Further, *qq*-relations are the place in the model where distance measures fit in to compare actors and media as the knowledge sources in the model, with ontology approaches on one hand (mostly leading to binary results) and real-valued retrieval functions modelling relevance on the other. For instance, two actors in an expert finder system may be similar in terms of their knowledge if the qualities they are described with are similar.

3.3 AMQ graphs and inference

In order to complete the AMQ model, all data are joined in a graph structure, which is the basis for inference algorithms. More specifically, taking ideas from ontology modelling, e.g., [MvH04], the schema and instance structures are distinguished.

Schema graph. The structure that combines the entities and relations discussed in the last sections is defined as an AMQ schema graph, $\mathcal{G}(\mathcal{V}, \mathcal{E})$, with the vertex set consisting of the three entity types (possibly their subtypes), $\mathcal{V} = \mathcal{A} \cup \mathcal{M} \cup \mathcal{Q}$, and the edge set mapping to the various relation types between them, $\mathcal{E} : \mathcal{V} \times \mathcal{V} \rightarrow \mathcal{R}$, with $\mathcal{R} \ni R$ denoting the set of relation types. Fig. 1 can be understood as a simplified example of a schema graph where the different node and edge types are collapsed into the clique of root entity types, which will be extended by a more expressive graphical notation in Section 4.

Instance graph. While the schema graph reflects the structure of the data about the community, an instance graph $G(V, E)$ fills this AMQ schema with data, leading to a kind of generalised co-citation graph or social network [WF94]. The instance graph contains typed objects as vertices, $v \in V$, where each vertex v has a type that is a member of $\mathcal{V}$ in the schema, as well as edges between the objects, $e \in E$, where each e maps to a relation type $R \in \mathcal{R}$ and a weight, i.e., $E : V \times V \rightarrow \mathcal{R} \times \mathbb{R}$ with $\mathbb{R} \supset I_R$ subsuming the range of all weighting functions $w(x, y, R)$. Note that this definition can be easily extended to hypergraphs by allowing edge sets with different vertex counts per edge.

In order to represent a virtual community for a retrieval scenario, typically only a small set of entity and relation types need to be included in the schema, depending on the available information on the community and the retrieval mechanisms required to fulfill a particular set of retrieval tasks.

Inference in AMQ models is the process of identifying or creating entities or relations in the AMQ instance graph by analysis of its semantic or structural properties. Semantics here refers to the qualities associated with entities (e.g., topics associated with a document), and structure to the general topology of the AMQ graph (e.g., co-citation, social network) spanned by the different data available.

More formally, inference algorithms can be defined as transformations from a given instance graph structure $G(V, E)$ to another structure that adds the inferred items to G: $G' = G(V, E) \cup G(\hat{V}, \hat{E})$.

In this way, standard methods of information retrieval and inference may be expressed in the AMQ model, providing a method to classify or unify existing algorithms, possibly creating a library of standard algorithms for re-use. Further, the method allows to define novel inference schemes that may use combinations of existing algorithms or lead to completely new approaches. As the AMQ model itself makes no commitment on the type of inference used, the range of possibilities is wide. In the next section, this will be explained with an example.

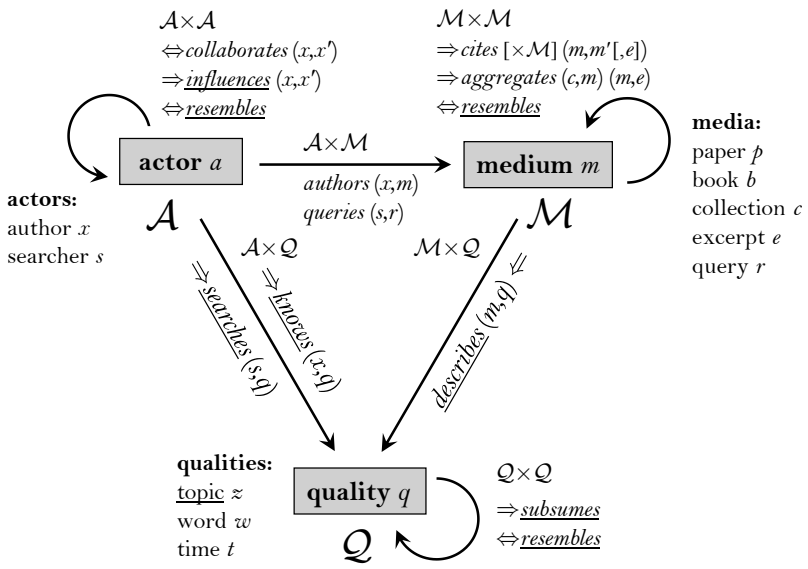


Figure 2: AMQ schema considered for CiteSeer and other digital library corpora.

4 Example: CiteSeer as an AMQ model

An illustrative application for an AMQ model is the CiteSeer digital library [GBL98], which offers research publications online. The authors of these publications can be considered to form a scientific virtual community whose members venture to create new knowledge by using and extending existing sources.

Being only one example among a set of scientific digital libraries (cf. the CORA dataset[3] or the digital libraries in Section 2), beside the semantic content of titles, abstracts and full-texts, CiteSeer gives access to authorship, co-citation, publication time (and partly venue) information and thus can be used to track the knowledge creation process in the community or to identify relevant and influential papers. However, the CiteSeer portal has only limited retrieval features, allowing document and author name search while restricting structural analysis to importance measures in the co-citation graph. On the other hand, a snapshot of the CiteSeer portal data exists that may be used to extend this: the CiteSeer corpus[4] with approx. 564k document abstracts and titles, 229k authors and 1272k intra-corpus citations.

Before we analyse inference and retrieval tasks possible on these data in Section 4.2, we characterise the structure of the CiteSeer data as an AMQ model in Section 4.1.

[3] `http://www.cs.umass.edu/~mccallum/code-data.html`
[4] `http://citeseer.ist.psu.edu/oai.html`.

4.1 Schema

The AMQ schema structure is shown in Fig. 2. Compared to Fig. 1, only the entities and relations relevant to CiteSeer are displayed, but on the other hand, Fig. 2 adds some additional information on the schema:

- Entities are added symbols, like x for an author as a special type of actor, a.
- These symbols are used to specify the domain and range of the different relations, like $\mathit{authors}(x, m)$, which means that this relation applies to authors (as opposed to searchers, s) but to all possible media types, m. This notation allows to retain the collapsed simple graphic representation of the AMQ schema. If no constraints are specified, a relation applies to the root types.
- Underlined types are considered inferred, whereas the others can directly be extracted from the available data.

Regarding actors, there are authors and searchers. The data in the CiteSeer corpus in fact do not include any "demand" data; the searcher entity $s \in \mathcal{A}$ is rather included to show the querying process of someone retrieving data from the corpus than to represent data contained in it. Looking at media entities seems self-explanatory, an excerpt $e \in \mathcal{M}$ being part of a medium ($\mathit{aggregates}(m, e)$) that can serve as context for a reference to another medium. Then optionally the binary $\mathit{cites}(m, m')$ relation between two documents is extended to a ternary $\mathit{cites}(m, m', e)$ relation with additional excerpt e as part of m (leading to a hypergraph structure). Finally, there are three different types of quality: topic, word and time, and while the latter two can be directly read from the corpus by indexing or from metadata, topics are themselves inferred entities, e.g., extracted by latent Dirichlet allocation or the author–topic model (see qualities definition in Section 3.1). The qq-relation *subsumes* can be applied to a topic for a hierarchical approaches, to a word when using some semantic hierarchy and to nest periods of time.

4.2 Inference tasks

On the CiteSeer data, various existing and novel inference and retrieval methods can be applied, and here we view them from the perspective of how they are expressed in terms of the AMQ model rather than the actual algorithms.

Media retrieval is to find documents etc. for a given query. This requires initial indexing, which creates *describes* relations between media and qualities (words for inverted index or vector-space models [BYRN99], inferred topics for latent semantics, etc.). During search, the same is done for a query, completing the graph with the respective weighting. The actual ranking of relevant documents is then based on an appropriate distance measure between the weightings of the *describes* relation, using vector weighting function $w(m, \vec{q}, \text{“}\mathit{describes}\text{”}) : \mathcal{M} \times \mathcal{Q} \to \mathbb{R}^K$ with K qualities for a given medium m. In the AMQ

schema, such distances may be mapped to a *resembles* relation as a basis for ranking:

$$w(m, m', \text{“}resembles\text{”}) = f(\text{distance}\{w(m, \vec{q}, \text{“}describes\text{”}), w(m', \vec{q}, \text{“}describes\text{”})\}) \ .$$

In the example, both words and topics as qualities allow to combine literal and latent-semantic search in an appropriate retrieval function or distance measure. For latent semantics, the weighting function $w(m, \vec{q}, \text{“}describes\text{”})$ represents the probability distribution $p(z|m)$ over K topics and implies the existence of topic distributions $p(w|z)$ that map words to topics (cf. [BNJ02]).

Expert finding. Extending document retrieval to scenarios like expert finding (see Section 2) is simple: Ranking for retrieval is then done via distances between *knows aq*-relations or *describes mq*-relations of documents authored by a particular person, identified via the *authors am*-relation that infers a *knows* relation:

$$w(a, \vec{q}, \text{“}knows\text{”}) = f(\{w(m, \vec{q}, \text{“}describes\text{”})\}_m) \ \forall \ \{m : w(a, m, \text{“}authors\text{”}) > 0\} \ .$$

For the actual implementations of the associated algorithms, numerous possibilities exist in the literature, e.g., the mentioned [RZGSS04] or [Hei04] that make use of latent topic distributions $p(z|x)$ for authors.

Advanced tasks. Beyond this, various other inference and retrieval tasks can be performed with the CiteSeer schema, for instance:

- Semantic matching: For a given document, the distance to other documents is inferred based on the *describes* relation, inferring the $resembles(m, m')$ relation.
- Co-citation matching: The similarity between the subgraph structures spanned by *cites* relations around two documents is inferred, e.g., based on intersection.
- Document citation influence: The influence of a document along $cites(m, m')$ relations is inferred, e.g., using graph importance measures like PageRank.
- Author influence: Document citation influence may be mapped to their authors using $authors(a, m)$.
- Actor matching: Combinations of searcher and author are ranked by their *knows* or *searches* relations, inferring *resembles* relations.
- Sub-community detection: Similar interests *searches* and/or knowledge *knows* can be clustered into communities with an entity group $g \in \mathcal{A}$ and $aggregates(g, a)$.
- Semantic citation influence: Combining the influence of citations with their semantics ($describes(m, q)$), possibly exploiting citation context via $cites(m, m', e)$.
- Dynamic models: Combining the above models with temporal information and its temporal derivatives, e.g., to analyse the evolution of *describes* or *searches* relations.

Moreover, with relevance or preference information included in the data, this list could be extended by collaborative approaches like recommender systems where actors rate media via $recommends(s, m)$ and inference yields a $likes(s, m)$ relation, possibly creating profile clusters similar to the groups g above.

5 Related work

Regarding previous approaches to define some generic structural basis of information access that uses the typical data available in virtual communities, existing work turned out to be surprisingly scarce.

Nevertheless, several research strands are relevant, mostly considering the way data is represented. From this perspective, the AMQ model can be viewed as an extension of social networks [WF94] by documents and items of knowledge representation. On the other hand, there are close relationships with ontology modelling [Gru94], particularly the Web Ontology Language OWL [MvH04]: Regarding entities and their types, OWL defines individuals that belong to classes that themselves support subsumption and aggregation as well as other relations called properties to link individuals to each other (object properties) or to data values (datatype properties). The AMQ model takes up the individual and class concepts but is limited to the object properties as the basis for its relations. Because OWL and other ontology languages are focussed on logical reasoning, the concept of weighted relations cannot be modelled in a simple and expressive way but rather requires workarounds like reification. Because the possibility of weighted relations is at the core of the AMQ model and it does not restrict inference methods to logic reasoning, the AMQ model has been defined as a more generic graph structure.

Moreover, the AMQ model can be considered an application of entity-relationship modelling [Che76] to community-based information retrieval tasks, providing a "template" to designing domain-specific database schemes. In a similar direction, the approach has some relations to meta-modelling [Bez06] as it can be used to derive models from a template model structure.

All of these viewpoints focus on the data structure that the AMQ model defines. Considering its objective to classify inference tasks and retrieval algorithms for community knowledge reveals no specific work beyond the general treatments of information retrieval methods like [BYRN99].

6 Conclusions

In this article, the "AMQ model" was developed, a representation of virtual communities that can be used as the basis for information systems to support retrieval and inference on their data. The model can be considered an attempt to characterise the domain of virtual communities from a data structure viewpoint considering the most important factors of explicit and tacit knowledge. Yet the model stays conceptually simple and does not claim to cover all imaginable scenarios. Rather, it attempts to pragmatically characterise a domain of applications of community knowledge access, namely such that infer similarity, relevance, classifications and other information from relations between people, documents and semantics.

By formalising information structure of community-based retrieval and inference tasks, the proposed model opens a new perspective on how to develop such approaches, and re-

search can depart from this into various directions. First, as this paper only discussed the structure of the data and associated tasks, one of the foremost future research topics is to fill the tasks with concrete algorithms. Here it is of special interest to explore combinations of existing approaches to obtain better retrieval tools, e.g., merging semantic with collaborative approaches. Second, an empirical study of the properties of the AMQ graphs of real-world scenarios may reveal interesting features that may be exploited for novel inference methods, e.g., based on suspected small-world properties of different relations and their combinations.

References

[AM96] M.S. Ackerman and D.W. McDonald. AnswerGarden 2: Merging organizational memory with collaborative help. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work*, pages 97–105, 1996.

[Bez06] J. Bezivin. On the Unification Power of Models. *Software and System Modeling (SoSym)*, 4(2):171–188, 2006.

[BFHV03] Peter Barna, Flavius Frasincar, Geert-Jan Houben, and Richard Vdovjak. Methodologies for Web Information System Design. In *Proc. ITCC*, 2003.

[BNJ02] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. In *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[Boi99] Max H. Boisot. *Knowledge assets – securing competitive advantage in the information economy*. Oxford University Press, 1999.

[BYRN99] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press & Addison-Wesley, 1999.

[Che76] Peter P. Chen. The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems*, 1(1):9–36, 1976.

[DGKT03] E. Davenport, M. Graham, J. Kennedy, and K. Taylor. Managing Social Capital as Knowledge Management – Some Specification and Representation Issues. In *Proc. American Society for Information Science and Technology (ASIS&T)*, pages 101–108, 2003.

[DM04] Elisabeth Davenport and Leo McLaughlin. *Trust in knowledge management and systems in organizations*, chapter Interpersonal Trust in Online Partnerships: The Challenge of Representation, pages 107–123. Idea Group, 2004. ISBN:1-59140-220-4.

[GBL98] C. Lee Giles, Kurt Bollacker, and Steve Lawrence. CiteSeer: An Automatic Citation Indexing System. *Proc. 3rd ACM Conf. on Digital Libraries*, pages 89–98, June 23–26 1998.

[Got05] Petter Gottschalk. *Strategic knowledge management technology*. Idea Group, 2005.

[Gru94] Thomas Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human and Computer Studies*, 43(5/6):907–928, 1994.

[Hei04] Gregor Heinrich. Teamarbeit nach Mass – Expertisemanagement in Organisationsnetzwerken (in German). In Anette Weisbecker, Thomas Renner, and Stefan Noll, editors, *Electronic Business – Innovationen, Anwendungen und Technologien*, pages 52–59. Fraunhofer IRB-Verlag, Stuttgart, Sep 2004. ISBN 3-8167-6621-8.

[HKJ+05] Gregor Heinrich, T. Keim, C. Jung, U. Krafzig, and S. Noll. Smart collaboration networks – a toolkit and a vision for creating and predicting partnership. In *Proc. Int. Conf. eChallenges*, 2005.

[HWW03] Marleen Huysman, Etienne Wenger, and Volker Wulf, editors. *Communities and Technologies*. Dordrecht: Kluwer, 2003.

[KSS97] H. Kautz, B. Selman, and M. Shah. ReferralWeb: Combining Social Networks and Collaborative Filtering. *Communications of the ACM*, 40(3), March 1997.

[Les00] E. Lesser, editor. *Knowledge and social capital: foundations and applications*. Oxford: Butterworth-Heinemann, 2000.

[MDH00] M. Maybury, R.D. D'Amore, and House. *Beyond Knowledge Management: Sharing Expertise*, chapter Automated Discovery and Mapping of Expertise. Cambridge: MIT Press, 2000.

[MvH04] Deborah L. McGuinness and Frank van Harmelen. OWL Web Ontology Language Overview. W3c recommendation, W3C, Feb. 2004.

[Nev06] Jennifer Neville. *Statistical Models and Analysis Techniques for Learning in Relational Data*. PhD thesis, Stanford University, 2006.

[NT95] Ikujiro Nonaka and Hirotaka Takeuchi. *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford Univ. Press New York/Oxford, 1995.

[Pol74] Michael Polanyi. *Personal knowledge: Towards a Post-Critical Philosophy*. University of Chicago & Press, Chicago, 1974. Originally published in 1958.

[PSD03] J.M. Pujol, R. Sangüesa, and J. Delgado. *Web Intelligence*, chapter A Ranking Algorithm Based on Graph Topology to Generate Reputation or Relevance, pages 382–395. Springer, 2003.

[Put00] R.D. Putnam. *Bowling Alone: The Collapse and Revival of American Community*. New York: Simon & Schuster, 2000.

[RSW05] Tim Reichling, Kai Schubert, and Volker Wulf. Matching Human Actors based on their Texts: Design and Evaluation of an Instance of the ExpertFinding Framework. In *Proceedings of GROUP 2005*. New York: ACM-Press, 2005.

[RZGSS04] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The Author-Topic Model for authors and documents. In *20th Conference on Uncertainty in Artificial Intelligence*, 2004.

[Sto07] Heinz Stockinger. Defining the grid: a snapshot on the current view. *J Supercomput*, (to be published), 2007.

[WB05] H.F. Witschel and T. Böhme. Evaluating profiling and query expansion methods for p2p information retrieval. In *Proc. of the 2005 ACM Workshop on Information Retrieval in Peer-to-Peer Networks (P2PIR)*, 2005.

[WCZM07] Fei-Yue Wang, Kathleen M. Carley, Daniel Zeng, and Wenji Mao. Social Computing: From Social Informatics to Social Intelligence. *Intelligent Systems, IEEE*, 22:79–83, 2007.

[Wen98] E.C. Wenger. *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, 1998.

[WF94] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[Whi01] John White. ACM Opens portal. *Commun. ACM*, 44(7):14–ff, 2001.

Use of genetic algorithms for a user specific reduction of amounts of interesting association rules

Birgit Wenke

Fakultät für Informatik
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39, D-85577 Neubiberg, Germany
email: birgit.wenke@unibw-muenchen.de

Abstract: The huge amounts of stored digital data, which are nowadays available in many domains contain lots of previously unknown coherences. For the user it is often difficult or unfeasible to find those coherences he is interested in without any technical support. One kind of these coherences are association rules.
This paper presents an iterative procedure which supports the user in finding these association rules he is interested in, by considering his interests without the explicit formulation of these interests by the user in advance. The procedure presents iteratively association rules to the user, who has to value each of them as interesting or uninteresting. With the help of a genetic algorithm the procedure learns interactively the interests of the user and formulates classification rules, which are used to classify the not yet presented association rules in the classes interesting and uninteresting so that only interesting classified association rules are presented to the user in the following.
The procedure was evaluated on a standard dataset and a dataset of the web2.0 application flick. The evaluation results show, that the developed procedure is useful for both standard database applications and innovative web2.0 applications. Different genetic methods and scenarios of interests were evaluated. The most interesting evaluation results will be presented in this paper.

1. Introduction

The huge amounts of digital data, which are available in many different domains due to the growing use of the internet and other digital applications contain lots of previously unknown coherences. For the user it is often difficult or unfeasible to find these coherences he is interested in without any technical support.

For example 2.5 mil. people suffer worldwide from multiple sclerosis. The cause and course of this illness are as far as possible unexplored, although a huge amount of data is available. This shows the need for new methods to find previously unknown coherences, which make it possible to formulate new research projects. This paper concentrates on

association rules which is one of several different kinds of patterns in data, generally known as data mining methods.

In this paper an iterative procedure will be presented, which supports the user in finding those associations rules of a dataset, he is interested in. The procedure consists of three main phases (Figure 1).

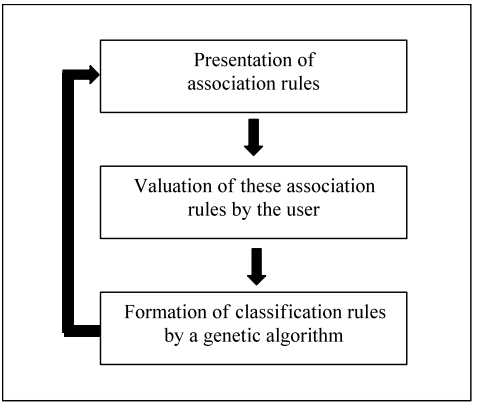


Figure 1: main phases of the developed procedure

The first phase is the presentation of about five association rules to the user. In the next phase the user has to valuate each of the presented rules if it is interesting or not for him. A genetic algorithms uses this subjective information of the user to formulate classification rules which classify the not yet presented association rules is the classes interesting or uninteresting. Starting from the second run only association rules will be presented to the user, which were classified as interesting in the previous run.

Main characteristic of this procedure is the consideration of the user's interests without an explicit formulation of these interests by this user. This is an advantage as the formulation of interests is often too difficult for the user, or the interests change by time or the user is not aware of all his interests.

The remainder of this paper is organised as follows. In the next section a brief introduction to data mining and genetic algorithms is given. Section 3 presents the most important phases of the developed procedure in detail and Section 4 presents the proceedings of the evaluation and the most interesting evaluation results. Also a summarisation of the results of the interviews with experts is given. The last section concludes with a summary.

2. Data mining and genetic algorithms

The task of knowledge discovery, especially data mining, is to discover new, potentially useful, nontrivial, unexpected and comprehensible knowledge from collected data of databases. [FPS96] Two commonly used methods are association rules, first introduced by [AIS93], and classification rules, which describe patterns hidden in the data.

In the following the data mining methods association rules and classification rules will be explained, two approaches of the literature will be presented, which support the user in finding interesting association rules and genetic algorithms will be described shortly.

2.1 Association rules

An association rule describes the co-occurrence of two sets of attribute-values, X and Y. Whenever the attribute-values of X occur in a tuple, the attribute-values of Y are likely to be also found there. Given is a set of attributes of a relational database D, $atts(D) = \{A_1, A_2, ..., A_n\}$, where each attribute consists of a set of attribute-values, $dom(D) = dom(A_1) \times dom(A_2) \times ... \times dom(A_n)$,. An association rule is an expression of the form $X \Rightarrow Y$, where X and Y are tow sets of attribute-values with $dom(X) \in dom(D)$, $dom(Y) \in dom(D)$ and $X \cap Y = \varnothing$ and which satisfy the user defined thresholds for support and confidence.

Support expresses the statistical significance of an association rule, i.e. the percentage of tuples in the database D, in which X and Y occur together.

$Sup(X \Rightarrow Y) = \frac{|X \cup Y|}{|D|}$, where |D| is the number of tuples in the database.

All itemsets which meet the user defined support threshold are called large itemsets.
Confidence is a measure of the strength of an association rule and is calculated by the conditional probability. It expresses the percentage of tuples containing X that are also containing Y.

$Conf(X \Rightarrow Y) = \frac{|X \cup Y|}{|X|}$.

All association rules which meet the user defined confidence threshold are called valid association rules.

A vast number of different algorithms can be found in the literature for the computation of association rules. [cp. AP98, CL02, GB03, HGN02, HMGN02, NDD99] They were analysed if they consider user's interests during the computation of these rules. The result shows, that only few of them do and if they do, they need user defined queries in advance through which the mentioned problems of formulating interests arise.

2.2 Classification rules

Classification rules are used for the sorting of objects in distinct and so far unknown classes. They use attribute values of tuples given in the dataset to decide about the class memberships of these tuples. The quality of the classification rules is determined by the number of faultless classifications [cp. BS01].

At first glance there might be no big difference between association rules and classification rules. But each of this methods has a different task. The task of association rules is to describe coherences hidden inside the data whereas classifications are used for the prediction of class membership with the help of attribute values in the database [cp. F00].

2.3 Other approaches that support the user's search

The literature was analysed for other approaches that support the user in his search for interesting association rules. In the following the two most interesting approaches 'interestingness measures' and 'redundancy of association rules' will be presented and compared with the developed procedure.

Interestingness measures

Interestingness measures determine the interestingness of association rules by statistical ratios or beliefs of the user. They were analysed, if and how they consider interests of the user. Some of them do. They are called subjective measures and need user formulated interests in advance [cp. KMRTV94, LHCM00, PT98, ST95]. Thereby they are linked with the problems of formulating interests as mentioned before.

Redundancy of association rules

In the literature exist several definitions about redundancy of association rules. They all are highly non uniform. The application of different definitions of redundancy on the same dataset leads to different, sometimes even conflicting results. Moreover only few of them are generally applicable on any amount of association rules as only some of them deal with redundancy in a logical purpose. None of them consider user interests [cp. BAG99, CS02, LHM99, SA95, SLRS99].

2.4 Genetic algorithms

Genetic algorithms are stochastically, intelligent search methods that imitate the natural evolution. They are characterised by an iterative run of several phases. For each of these phases different genetic methods can be used. This leads to a vast number of method combinations [cp. N97].

An analysis of the literature was done to examine the use of genetic algorithms for the computation of association rules and classification rules. The result of this analysis shows, that the computation of classification rules is a main field of application of genetic algorithms. Lots of different approaches can be found in the literature [AMR01, EJ93, EKK04, FLF00, IF03, K94, KK05, MVFN01]. However, for the computation of association rules genetic algorithms are only used for the consideration of special cases, for example like the computation of frequent amounts of attribute values by the computation of association rules [MAR02].

3. The developed procedure in detail

In the following the most important steps of the developed procedure will be explained. Figure 2 gives a review of the procedure. Starting point of the procedure is the amount of valid association rules of the considered dataset. This amount is given and has not to be computed by the procedure. Firstly some association rules have to be selected and presented to the user. Secondly the user has to value each of these rules as interesting or uninteresting. A genetic algorithms uses this subjective information of the user to form classification rules, which are then used to classify the not yet presented association rules in the classes 'interesting' and 'uninteresting'. Starting from the second run, only those association rules can be selected for the presentation, which were classified as interesting in the previous run.

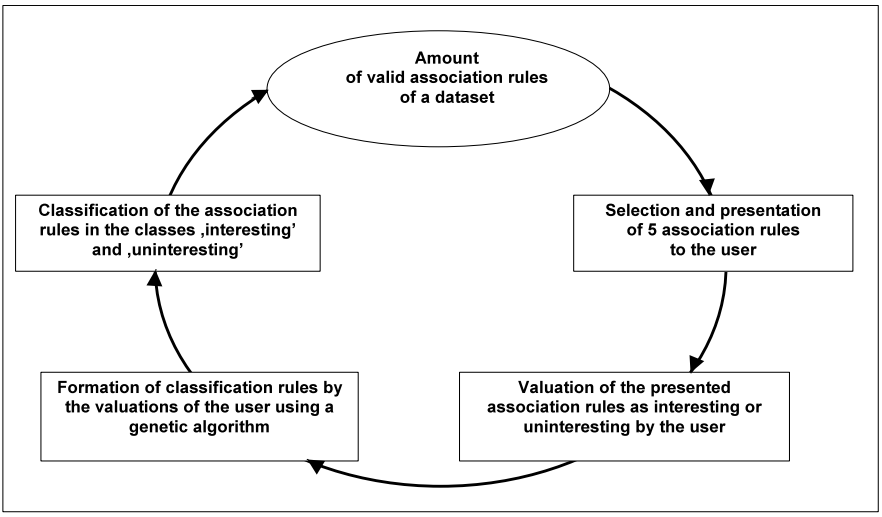


Figure 2: Review of the developed procedure

First important step is the selection of the presented association rules. Basis of this selection is the following distance measure:

$$Dist(X{:}x \Rightarrow Y{:}y, V{:}v \Rightarrow W{:}w) = |(X{:}x, Y{:}y)\ \Phi\ (V{:}v, W{:}w)| \text{ [cp. DL98)]}$$

This measure computes the distance of two association rules X⇒Y und V⇒W by the number of attribute values they are different in. This distance is computed for every association rule to the others. Those association rules will be selected for the presentation which have the highest distances.

Second important step is the valuation of the user. To illustrate the user valuation the following association rule of a dataset about patients that suffer from multiple sclerosis is used: gender:man, handicap:impaired vision, age:older than 50 years ⇒ illness:no multiple sclerosis. For example, if a user is interested in association rules that give an information about the gender of a patient, the given association rule would be interesting. Otherwise, if a user is for example interested in association rules about patients between 30 and 40 years, the given association rule would be uninteresting. This valuation has to be done for every presented association rule.

In the first run an initialisation strategy is needed to create the first classification rules, which will be the parents for the genetic operators. In the following runs the classification rules of the previous run will be used instead. The initialisation strategy of the procedure uses the valued association rules and turns them into classification rules as their classes are known due to the valuation of the user. To make these classification rules applicable on other association rules a generalisation is done by the random removal of one attribute value of each classification rule.

Now the genetic operators recombination and mutation are applied to create new classification rules. The recombination combinates the parents to form new classification rules whereas the mutation uses duplicates of the parents and alters them randomly. Together with the evaluation results of the procedure will be explained which genetic operators were used for the evaluation.

To make a selection of classification rules possible an evaluation of these rules has to be done. This is done by the sum of the following two ratios:

$$laplacefunction = \frac{tp+1}{tp+fp+2} \text{ and } completeness = \frac{tp}{tp+fn} \text{ [cp. LK00]}$$

Both ratios base on the results of the application of the classification rules on the yet presented and user-valued association rules. The laplacefunction is the percentage of association rules that were classified as interesting and are interesting for the user on all association rules that were classified as interesting. The completeness expresses the percentage of association rules that were classified as interesting and are interesting for the user on all association rules that are interesting for the user. This sum is computed for every classification rule (parents, recombination rules and mutation rules).

With these results a selection of classification rules is possible. This selection determines, which classification rules will be used for the classification of the not yet presented association rules in the classes ‘interesting’ und ‘uninteresting’ in the current run and as parents of the genetic operators in the following run. Together with the

evaluation results of the procedure will be explained which genetic operators were used for the evaluation.

4. Evaluation and interviews with experts

The evaluation of the developed procedure was done to analyse the general performance of the procedure and the influence of different genetic parameters and different scenarios of interest. In the following the datasets, used for the evaluation, the proceeding of the evaluation, the most interesting evaluation results and a summary of the interviews with experts will be presented.

4.1 Evaluation proceeding

Two highly different datasets were used for the evaluation. First dataset is the heart disease dataset. It is a standard dataset, which is available on the internet (www.liacc.up.pt/ML/statlog/datasets/heart/heart, download 18.10.2004). 4053 association rules where computed for this dataset.

The second dataset is the flickr dataset, an individual dataset of tag tuples of the web2.0 application flickr. As is data is not available as complete dataset on the internet, it first had to be collected and 5932 association rules were computed for it.

Flickr is an internet portal which allows users to store and present their photos. One important characteristic of this portal is, that every photo is tagged by his owner with up to four tags. These tags are used by the search engine of the portal. As the users are completely free in choosing their tags, for other user it is often very difficult to find those tags which lead to the photos they are interested in. To enhance this situation, the developed procedure supports the user in finding those tags, which lead to the photos he is interested in.

These tow datasets are characterised by two main differences. First difference is about the attributes. In the flickr dataset every attribute consists of exactly one attribute value as it is not possible to eliminate any tag combination, because there are not restrictions for the user's tag choices. This fact leads to about 70 attributes for the flickr dataset whereas the heart disease data dataset has only 14 attributes. Second difference is about the presentation of association rules. In the flickr application it does not make sense to present different association rules about the same amount of attribute values to the user. Therefore only one association rule for each amount of attribute values was chosen for the presentation. This fact makes the learning of the user's interests much more difficult.

The evaluation was done with 150 runs of the procedure, each with 50 iterations, in which six users with different scenarios of interests were simulated.

Evaluation criterion was the average predictive accuracy, which is the average number of presented association rules that agree to the interests of the simulated users.

4.2 Evaluation results

In the following, evaluation results of different genetic methods and different scenarios of interest will be presented for both datasets.

Genetic methods

For each of the three important genetic algorithm phases recombination, mutation and selection the two most different methods were chosen (Table 1).

	Method 1	**Method 2**
Recombination	Diagonal crossover (several parents are used for the formation of a new individual)	Single-point crossover (exactly two parents are used for the formation of a new individual)
Mutation	Mutation probability: 1	Mutation probability: ≤ 1
Selection	Stochastic universal sampling (probabilistic selection)	Truncation selection (deterministic selection)

Table 1: genetic methods

The combination of these six methods leads to eight different alternatives (Table 2) which were evaluated for both datasets (Figure 3).

Alternative	**Recombination method**	**Mutation method**	**Selection method**
1	Diagonal crossover	Mutation probability ≤ 1	Stochastic universal sampling
2	Diagonal crossover	Mutation probability ≤ 1	Truncation selection
3	Diagonal crossover	Mutation probability = 1	Stochastic universal sampling
4	Diagonal crossover	Mutation probability = 1	Truncation selection
5	Single-point crossover	Mutation probability ≤ 1	Stochastic universal sampling
6	Single-point crossover	Mutation probability ≤ 1	Truncation selection
7	Single-point crossover	Mutation probability = 1	Stochastic universal sampling
8	Single-point crossover	Mutation probability = 1	Truncation selection

Table 2: genetic method alternatives

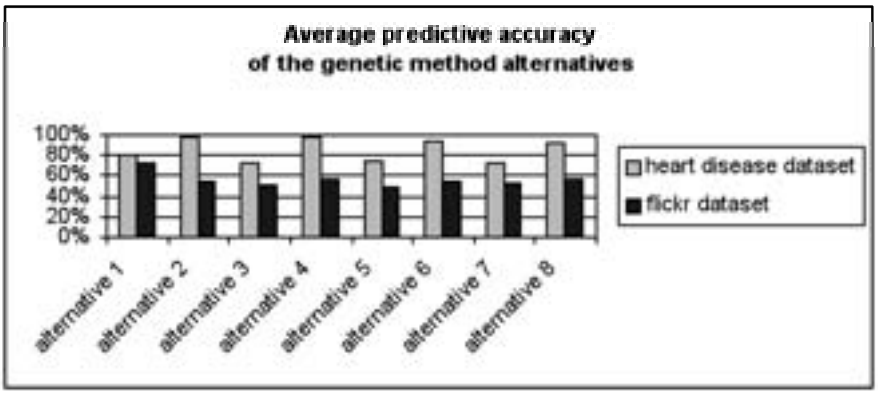


Figure 3: Evaluation results for the genetic method alternatives

The results of the heart disease dataset show, that the alternatives with the deterministic selection method truncation selection achieved the best results, whereas the flickr dataset achieved the best results with the first alternative, the recommendation of the literature (diagonal crossover, mutation probability ≤ 1 and stochastic universal sampling). This illustrates, that for a small dataset with relatively easy learning conditions like the heart disease dataset the need for stochastic elements is low as the deterministic selection leads to better results. However in case of a bigger data set with more difficult learning conditions like the flickr dataset the influence of stochastic methods is needed to achieve better results.

The comparison of the genetic method results of both datasets shows, that the results of the heart disease dataset are always better than those of the flickr dataset. This is due to the characteristics of the datasets, which were mentioned before.

Scenarios of interest

Six scenarios of interest, which differ in complexity and the number of interesting association rules were evaluated for both datasets. An scenario of interest with a low complexity is for example the interest in association rules that give an information about the age of a person, whereas the interest in association rules about men older than 50 years with an impaired vision is more complex. The numbers of interesting association rules are the numbers of association rules that agree to the different scenarios of interest.

Figure 4 shows the results for the heart disease dataset. The scenarios of interest are ordered increasingly after their complexity and the numbers of interesting association rules for each scenario are given in brackets.

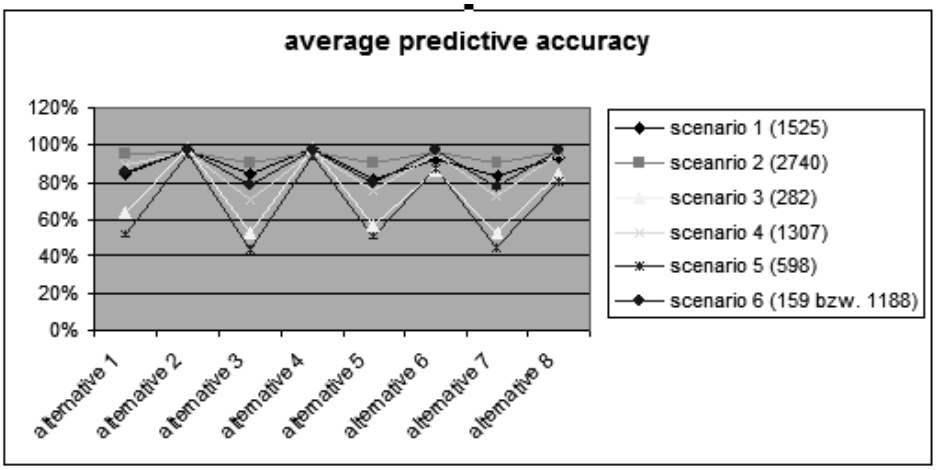


Figure 4: Results of the scenarios of interest for the heart disease dataset

Referring to the complexity of the scenarios of interest no influence on the performance can be derived by the results, because for example the most complex scenario 6 achieved always better results than scenario 3 which is less complex. Referring to the numbers of interesting association rules an influence on the performance can be derived from the results. For example scenario 2 with the highest number of interesting association rules achieved for all eight alternatives better results than scenario 3 and 5, both with much lower numbers of interesting association rules. As the results correspond not for all scenarios directly to the numbers of interesting association rules (scenario 6 has sometimes better results than scenario 1), the influence is moderate.

Figure 5 shows the results for the flickr dataset. The scenarios of interest are ordered increasingly after their complexity and the numbers of interesting association rules for each scenario are given in brackets.

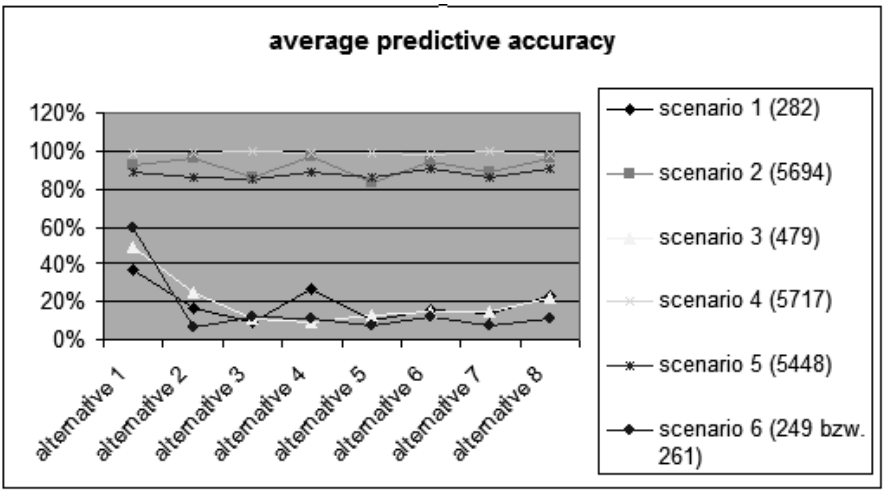


Figure 5: Results of the scenarios of interest for the flickr dataset

Referring to the complexity of the scenarios of interest no influence on the performance can be derived by the results, because for example the very complex scenario 5 achieved always better results than scenario 1 and 3 which are less complex. Referring to the numbers of interesting association rules an influence on the performance can be derived from the results. The scenarios 2, 4 and 5 with much higher numbers of interesting association rules than the other three scenarios achieved for all eight alternatives much better results. As the results correspond almost directly to the numbers of interesting association rules, the influence is strong.

The comparison of the scenarios of interest results for the complexity do not indicate an influence on the performance of the procedure for both datasets. However for the numbers of interesting association rules an influence can be derived from the results of both datasets. In case of the heart disease dataset the influence is moderate whereas in case of the flickr dataset the influence is strong.

4.3 Interviews with experts

To analyse the main field of application of the developed procedure and the relevancy of practice, four experts of medicine, statistics and biochemistry where interviewed in partial structured interviews. As main field of application the research domain was mentioned and the experts affirmed the need and relevancy of practice of the procedure.

5. Summary

The developed procedure allows the user an interactive and explorative search for previously unknown coherences hidden in huge amounts of digital data. The proceeding

of the procedure is related to browsing the internet. Main characteristic of the developed procedure is, that the user does not have to formulate his interests explicitly in advance although his interests are considered by the procedure. This is an important fact as often the formulation of interests is difficult or unfeasible for the user, the interests of the user change by and by or the user is not aware of all his interests as he does not know all the coherences, which are hidden inside the data. Huge amounts of digital data can be found in many different domains whereby the developed procedure is applicable in various fields of applications.

The evaluation results have shown that the use of genetic algorithms for an user specific reduction of amounts of interesting association rules leads to good results. The maximal predictive accuracy of the heart disease dataset is 99% whereas for the flickr dataset the maximal predictive accuracy is 73%. The analysis of different genetic methods has shown that they have an influence on the performance of the procedure and that the genetic methods which achieved the best results are dependent on the dataset they are applied on. The analysis of different interest scenarios has led to two different findings. For the degree of complexity of the interest scenarios no influence on the performance can be derived from the evaluation results of both datasets. In contrast the number of interesting association rules of an interest scenario had an influence on the performance for both datasets. For the heart disease dataset a light influence was indicated by the evaluation results as the for the flickr dataset the evaluation results showed a strong influence on the performance.

5. References

[AIS93] Agrawal, R. / Imielinski, T. / Swami, M.: Mining association rules between sets of items in very large databases in: Proc. of the ACM SIGMOD Conf. on Management of data, p. 207-216, 1993

[AMR01] Alvarez, J. L. / Mata, J. / Riquelme, J. C.: Oblic: Classification Systems using Evolutionary Algorithm, in: Proc. of the 6th Int'l. Work-Conference on Artificial and Natural Neural Networks 2001, p. 644-651

[AP98] Aggarwal, Charu C. / Philip S. Yu: Online generation of association rules, in: Proc. of the 4th Int'l. Conf. on Knowledge Discovery and Data Mining 1998, p. 129-133

[BAG99] Bayardo, Roberto J. Jr. / Agrawal, Rakesh / Gunopulos, Dimitrios: Constraint-based rule mining in large dense databases, in: Proc. of the IEEE Int'l. Conf. on Data Engineering 1999, p. 188-197

[BS01] Bäck, Thomas / Schütz, Martin: Evolutionäre Algorithmen im Data Mining, in: Handbuch Data Mining im Marketing, hrsg. v. Hippner, Hajo / Küsters, Ulrich / Meyer, Matthias / Wilde, Klaus, Braunschweig, Wiesbaden 2001, p. 403-426

[CL02] Cong, Gao / Liu, Bing: Speed-up iterative frequent itemset mining with constraint changes, in: Proc. of the 2nd IEEE Int'l. Conf. on Data Mining 2002, p. 107-114

[CS02] Cristofor, Laurentiu / Simovici, Dan: Generating an informative cover for association rules, in: Proc. of the IEEE Int. Conf. on Data Mining 2002, p. 597-600

[DL98] Dong, Gouzhu / Li, Jinyan: Interestingness of discovered association rules in terms of neighborhood-bases unexpectedness, in: Proc. of the 2nd Pacific-Asia Conference on Knowledge Discovery and Data Mining 1998, p. 72-86

[EJ93] Eick, C. F. / Jong, D.: Learning bayesian classification rules through genetic algorithms, in: Proc. of the 2nd Int'l. Conference on Information Knowledge Management 1993, p. 305-313

[EKK04] Eggermont, J. / Kok, J. N. / Kosters, W. A.: Genetic programming for data classification: paritioning the search space, in: Proc. of the 2004 ACM Symposium on Applied Computing 2004, p. 1001-1005

[F00] Freitas, Alex A.: Understanding the crucial difference between classification and discovery of association rules - a position paper, in: ACM SIGKDD Explorations 2(1) 2000, p. 65-69

[FLF00] Fidelis, M. V. / Lopes, H. S. / Feitas, Alex A.: Discovering comprehensible classification rules with a genetic algorithm, in: Proc. of the Congress on Evolutionary Computation 2000, p. 805-810

[FPS96] Fayyad, Usama M. / Piatetsky-Shapiro, Gregory / Smyth, Padhraic: From Data Mining to Knowledge Discovery: An Overview, in: Advances in knowledge discovery and data mining, hrsg. v..Fayyad, Usama M. et. al., Menlo Park, California /Cambridge, Massachusetts / London, England, p. 1-34, 1996

[GB03] Goethals, Bart / Bussche, Jan Van den: On supporting interactive constrained association rule mining, in: Proc. of the 2nd Int'l. Conf. on Data Warehousing and Knowledge Discovery 2000, p. 307-316

[HGN02] Hipp, Jochen / Güntzer, Ulrich / Nakhaeizadeh, Gholamreza: Data mining of association rules and the process of knowledge discovery in databases, in: Proc. of the Industrial Conf. on Data Mining 2002, p. 15-36

[HMGN02] Hipp, Jochen / Mangold, Christoph / Güntzer, Ulrich / Nakhaeizadeh, Gholamreza: Efficient rule retrieval and postponed restrict operations for association rule mining, in: Proc. of the Pacific-Asia Conference on Knowledge Discovery and Data Mining 2002, p. 52-65

[IF03] Isasi, P. / Fernandez, F.: Evolutionary approach to overcome initialization parameters in classification problems, in: Proc. of the 7th Int'l. Work-Conference on Artificial and Natural Neural Networks 2003, p. 254-261

[K94] Konstam, A.: N-Group classification using genetic algorithms, in: Proc. of the 1994 ACM Symposium on Applied Computing 1994, p. 212-216

[KK05] Kshetrapalapuram, K. K. / Kirley, M.: Mining classification rules using evolutionary multi-objective Algorithms, in: Proc. of the 9th Conference on Knowledge-based Intelligent Information and Engineering Systems 2005, p. 959-965

[KMRTV94] Klemettinen, Mika / Mannila, Heikki / Ronkainen, Pirjo / Toivonen, Hannu / Verkamo, A. Inkeri: Finding interesting rules from large sets of discovered association rules, in: Proc. of the 3rd Int'l. Conf. on Information and Knowledge Management 1994, p. 401-407

[LHCM00] Liu, Bing / Hsu, Wynne / Chen, Shu / Ma,Yiming: Analyzing the subjective interestingness of association rules, in: IEEE Intelligent Systems 15(5) 2000, p. 47-55

[LHM99] Liu, Bing / Hsu, Wynne / Ma, Yiming: Pruning and summarizing the discovered association rules, in: Proc. of the 5th ACM SIGKDD Int'l. Conference on Knowledge Discovery and Data Mining 1999, p. 125-134

[LK00] Liu, Juliet Juan / Kwok, James Tin-Yau: An extended genetic rule induction algorithm, in: Proc. of the Congress on Evolutionary Computation 2000, p. 458-463

[MAR02] Mata, Jacinto / Alvarez, José-Luis / Riquelme, José-Cristobal: An evolutionary algorithm to discover numeric association rules, in: Proc. of the 2002 ACM Symposium on Applied Computing 2002, p. 590-594

[MVFN01] Mendes, R. R. F. / Voznika, F. de B. / Freitas, A. A. / Nievola, J. C.: Discovering Fuzzy Classification Rules with Genetic Programming and Co-evolution, in: Proc. of the 5th European Conference on Principles of Data Mining and Knowledge Discovery 2001, P. 314-325

[N97] Nissen, Volker: Einführung in Evolutionäre Algorithmen: Optimierung nach dem Vorbild der Evolution, Braunschweig, Wiesbaden 1997

[NDD99] Nag, Biswadeep / Deshpande, Prasad M. / DeWitt, David J.: Using a knowledge cache for interactive discovery of association rules, in: Proc. of the 5th ACM SIGKDD Int'l. Conference on Knowledge Discovery and Data Mining 1999, p. 244-253

[PT98] Padmanabhan, Balaji / Tuzhilin, Alexander: A belief-driven method for discovering unexpected patterns, in: Proc. of the 4th Int'l. Conf. on Knowledge Discovery and Data Mining 1998, P. 94-100

[SA95] Srikant, Ramakrishnan / Agrawal, Rakesh: Mining generalized association rules, in: Proc. of the 21st Int'l. Conf. on Very Large Data Bases 1995, P. 407-419

[SLRS99] Shah, Devavrat / Lakshmanan, Laks V. S. / Ramamritham, Krithi / Sudarshan, S.: Interestingness and pruning of mined patterns, in: Proc. of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery 1999

[ST95] Silberschatz, Avi / Tuzhilin, Alexander: On subjective measures of interestingness in knowledge discovery, in: Proc. of the 1st Int'l. Conf. on Knowledge Discovery and Data Mining 1995, p. 275-281

Chapter 4: Content Management

Contributions to 10th I2CS 2010, Bangkok, Thailand

Wongkot Sriurai, Phayung Meesad, Choochart Haruechaiyasak
Improving Web Page Classification by Integrating Neighboring Pages via a Topic Model

Maleerat Sodanil, Supot Nitsuwat, Choochart Haruechaiyasak
Improving ASR for Continuous Thai Words Using ANN/HMM

Nivet Chirawichitchai, Parinya Sa-nguansat, Phayung Meesad
A Comparative Study on Feature Weight in Thai Document Categorization Framework

Contributions to 7th I2CS 2007, Munich, Germany

Florian Holz, Hans-Friedrich Witschel, Gregor Heinrich, Gerhard Heyer, Sven Teresniak
An Evaluation Framework for Semantic Search in P2P Networks

Improving Web Page Classification by Integrating Neighboring Pages via a Topic Model

Wongkot Sriurai, Phayung Meesad, Choochart Haruechaiyasak

Department of Information Technology
Faculty of Information Technology
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pibulsongkarm Rd., Bangsue, Bangkok 10800

Department of Teacher Training in Electrical Engineering
Faculty of Technical Education
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pibulsongkarm Rd., Bangsue, Bangkok 10800

Human Language Technology Laboratory (HLT)
National Electronics and Computer Technology Center (NECTEC)
Thailand Science Park, Pathumthani 12120, Thailand

s4970290021@ kmutnb.ac.th
pym@kmutnb.ac.th
choochart.haruechaiyasak@nectec.or.th

Abstract: This paper applies a topic model to represent the feature space for learning the Web page classification model. Latent Dirichlet Allocation (LDA) algorithm is applied to generate a probabilistic topic model consisting of term features clustered into a set of latent topics. Words assigned into the same topic are semantically related. In addition, we propose a method to integrate the additional term features obtained from neighboring pages (i.e., parent and child pages) to further improve the performance of the classification model. In the experiments, we evaluated among three different feature representations: (1) applying the simple BOW model, (2) applying the topic model on current page, and (3) integrating the neighboring pages via the topic model. From the experimental results, the approach of integrating current page with the neighboring pages via the topic model yielded the best performance with the F1 measure of 84.51%; an improvement of 23.31% over the BOW model.

1 Introduction

Today, the amount of Web documents (e.g., Web pages, blogs, emails) is increasing with an explosive rate. Text categorization is a widely applied solution for managing and organizing those documents. For example, a text categorization model can be used to assist the information retrieval process in filtering the documents for a specific topic. Text categorization process usually adopts the supervised machine learning algorithms for learning the classification model [Du98], [YP97]. To prepare the term feature set, the bag of words (BOW) is usually applied to represent the feature space. Under the BOW model, each document is represented by a vector of weight values calculated from, for example, the term frequency–inverse document frequency (TF-IDF), of a term occurring in the document. The BOW is very simple to create, however, this approach discards the semantic information of the terms, i.e., synonym. Therefore, different terms whose meanings are similar or the same would be represented as different features. As a result, the performance of a classification model learned by using the BOW model could become deteriorated.

In this paper, we apply a topic model to represent the feature space for learning the Web page classification model. Words (or terms), which are statistically dependent under the topic model concept, are clustered into the same topic. Given a set of documents D consisting of a set of terms (or words) W, a topic model generates a set of latent topics T based on a statistical inference on the term set W. In this paper, we applied the Latent Dirichlet Allocation (LDA) algorithm to generate a probabilistic topic model from a Web page collection. A topic model can help capture the hypernyms, hyponyms and synonyms of a given word. For example, the words "vehicle" (hypernym) and "automobile" (hyponym) would be clustered into the same topic. In addition, the words "automobile" (synonym) and "car" (synonym) would also be clustered into the same topic. The topic model helps improve the performance of a classification model by (1) reducing the number of features or dimensions and (2) mapping the semantically related terms into the same feature dimension.

In addition to the concept of topic model, our proposed method also takes an advantage of hyperlink structure of the Web. Given a Web page (denoted by *current page*), there are typically incoming links from *parent* pages and outgoing links to *child* pages. Both parent and child pages are collectively referred to as the *neighboring* pages. Using the additional terms from the neighboring pages could help increase more evidence for learning the classification model. However, the terms from current page should be weighted higher than terms from neighboring pages. Therefore, the proposed method for integrating neighboring information provides a function for varying the weight values of terms coming from the parent pages and the ones from the child pages. Using the Support Vector Machines (SVM) as the classification algorithm, the experimental results showed that by integrating the additional neighboring information via a topic model, the classification performance under the F1 measure was significantly improved over the simple BOW model.

The rest of this paper is organized as follows. In next section we provide a brief review of related works. Section 3 presents the proposed framework of feature representation via the topic model for learning the Web page classification models. Section 4 presents experiments with some discussion on the results. In Section 5, we conclude the paper and put forward the directions of our future works.

2 Related Works

Text categorization (also known as *document classification*) is a supervised learning task, concerning the assigning of category labels to new documents based on the information learned from a labeled training data [Du98], [YP97]. Text categorization is a well-studied research area related to information retrieval, machine learning and text mining. A number of machine learning algorithms have been introduced and applied for the task of text classification including the Support Vector Machines (SVM) [Jo98], [Va95]. The SVM has been shown to yield the best performance compared to other classification algorithm in many previous works. In this paper, we adopt the SVM in our experiments.

In the domain of the Web, text categorization has been applied for classifying Web pages. Recent works in Web page classification proposed some methods to include the information from neighboring Web pages to learn the model. The information of neighboring pages is, for example, title and surrounding text of anchor text [AGS99], [SLN02], [Zh07]. Furnkranz [Fu99] proposed a classification method using anchor text, surrounding text of anchor text that precedes the hyperlink. Shen et al. [Sh06] proposed an approach to compare of implicit and explicit links for Web page classification. The experimental results showed that the use of the implicit links is better than using explicit links in classification.

Qi and Davison [QD06] proposed a method to improve Web page classification by utilizing the class information from neighboring pages in the link graph. The categories represented by four kinds of neighbors (parents, children, siblings and spouses) are combined to help with the page in question. Experiments showed that sibling pages are the most important type of neighbor to use. Qi and Davison [QD08] proposed a method by utilizing a weighted combination of the contents of neighbors to generate a better virtual document for classification. Their experimental results showed that including a weighted value from neighboring pages helps improve the performance of Web page classification. Chen and Choi [CC08] presented an automatic genre-based Web page classification system, which can work either independently or in conjunction with other topic-based Web page classification system.

In this paper, we also apply the neighboring information for improving the classification model. However, we adopt the topic model to represent the feature space. There have been many studies on discovering latent topics from text collections [SG06]. Recently, the Latent Dirichlet Allocation (LDA) has been introduced as a generative probabilistic model for a set of documents [BNJ03]. The basic idea behind this approach is that documents are represented as random mixtures over latent topics. Each topic is represented by a probability distribution over the terms. Each article is represented by a probability distribution over the topics. LDA has also been applied for identification of topics in a number of different areas such as classification and collaborative filtering [BNJ03].

The process to generate a topic model can be explained as follows. The input data for the LDA algorithm consists of an article collection which is a set of m documents denoted by $D = \{D_0, \ldots, D_{m-1}\}$. The LDA algorithm generates a set of n topics denoted by $T = \{T_0, \ldots, T_{n-1}\}$. Each topic is a probability distribution over p words denoted by $T_i = [w^i_0, \ldots, w^i_{p-1}]$, where w^i_j is a probabilistic value of word j assigned to topic i. Based on this topic model, each document can be represented as a probability distribution over the topic set T, i.e., $D_i = [t^i_0, \ldots, t^i_{n-1}]$, where t^i_j is a probabilistic value of topic j assigned to document i [HD08].

3 Feature Representation via the Topic Model

Figure 1 illustrates the proposed framework of feature representations for learning the Web page classification models. In our proposed framework, we evaluated among three different feature representations: (1) applying the simple BOW model on current page, (2) applying the topic model on current page, and (3) integrating the neighboring pages via the topic model. Each approach is described in details as follows.

Approach 1 (BOW): Given a Web page collection, the process of text processing is applied to extract terms. The set of terms is then filtered by using the feature selection technique, information gain (IG) [DL97]. Once the term features are obtained, we apply the Support Vector Machines (SVM) to learn the classification model. The model is then used to evaluate the performance of category prediction.

Approach 2 (TOPIC_CURRENT): Given a Web page collection, the process of text processing is applied to extract terms. The set of terms is then generated by using the topic model based on the LDA algorithm. The output from this step is the topic probability representation for each article. The Support Vector Machines (SVM) is also used to learn the classification model.

Approach 3 (TOPIC_INTEGRATED): The main difference of this approach from Approach 2 is we integrate the additional term features obtained from the neighboring pages to improve the performance of Web page classification. The process of integrating the neighboring pages is explained as follows.

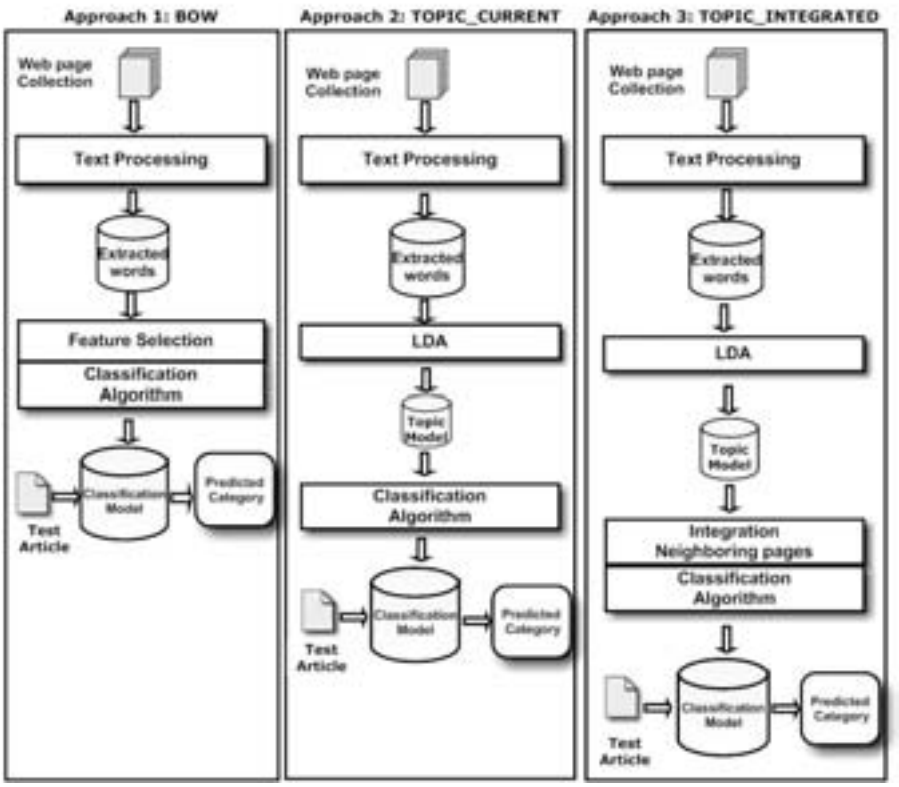


Figure 1: The proposed feature representation framework

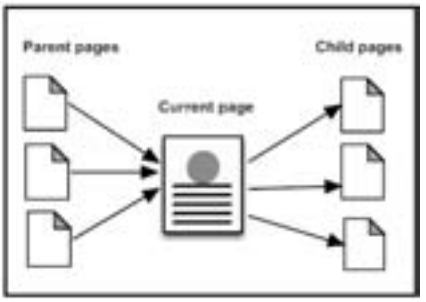


Figure 2: A Current Web page with two types of neighboring pages

Figure 2 shows two types of neighboring pages, parent and child pages. Given a Web page (denoted by *current page*), there are typically incoming links from *parent* pages and outgoing links to *child* pages. Both parent and child pages are collectively referred to as the *neighboring* pages. Using the additional terms from the neighboring pages could help increase more evidence for learning the classification model.

In this paper, we vary a weight value of neighboring pages from zero to one. A weight value equals to zero means the neighboring page is not included in the feature representation. Under this approach, terms from different page types (i.e., current, parent and child) are first transformed into a set of n topics denoted by $T=\{T_0,...,T_{n-1}\}$ by using the LDA algorithm. The weight values from 0 to 1 are then multiplied to the topic dimension T_i of parent and child pages. The combined topic feature vector by integrating the neighboring topic vectors with adjusted weight values can be computed by using the following equation:

$$T(Integrated) = T(current\ page) + w_p \times T(parent\ pages) + w_c \times T(child\ pages) \quad (1)$$

where $T(Integrated)$, $T(current\ page)$, $T(parent\ pages)$ and $T(child\ pages)$ are topic sets of integrated page, current page, parent pages and child pages, respectively. w_p and w_c are the weights of parent pages and child pages, respectively. The values of w_p and w_c are varied from 0.0 to 1.0 with 0.1 interval.

4 Experiments and Discussion

In our experiments, we use a collection of articles obtained the Wikipedia Selection for Schools, which is available from the SOS Children's Villages Web site[1]. There are 15 categories: art, business studies, citizenship, countries, design and technology, everyday life, geography, history, IT, language and literature, mathematics, music, people, religion and science. The total number of articles is 4,625.

We used the LDA algorithm provided by the linguistic analysis tool called LingPipe[2] to run our experiments. LingPipe is a suite of Java tools designed to perform linguistic analysis on natural language data. LingPipe tools include a statistical named-entity detector, text classification and clustering. In this experiment, we apply the LDA algorithm provided under the LingPipe API and set the number of topics equal to 200 and the number of epochs to 2,000.

For text classification process, we used WEKA[3], an open-source machine learning tool, to perform the experiments. The standard performance metrics for evaluating the text classification used in the experiments are precision, recall and F1 measure [Du98]. We tested all algorithms based on the 10-*fold* cross validation.

[1] SOS Children's Villages Web site. http://www.soschildrensvillages.org.uk/charity-news/wikipedia-for-schools.htm
[2] LingPipe. http://alias-i.com/lingpipe
[3] Weka. http://www.cs.waikato.ac.nz/ml/weka/

We start by evaluating the weight values of neighboring pages under Approach 3. Table 1 shows the results of weight value adjustment on parent pages and child pages based on Eq. (1). The best weight value of parent pages is equal to 0.4 with the F1 measure of 0.8463. For the child pages, the maximum value of F1 measure is 0.8463 with the weight value of 0.2. The results showed that using information from parent pages is more effective than child pages for improving the performance of a classification model. The reason is due to the parent pages often provide terms, such as in the anchor texts, which provide additional descriptive information of the current page.

Neighbors	w_p	P	R	F1
Parent pages	0.1	0.8587	0.8319	0.8440
	0.2	0.8575	0.8321	0.8435
	0.3	0.8575	0.8323	0.8439
	0.4	**0.8569**	**0.8313**	**0.8463**
	0.5	0.8566	0.8318	0.8428
	0.6	0.8533	0.8279	0.8391
	0.7	0.8541	0.8274	0.8391
	0.8	0.8543	0.8251	0.8378
	0.9	0.8527	0.8225	0.8358
	1	0.8524	0.8225	0.8355

Neighbors	w_c	P	R	F1
Child pages	0.1	0.8590	0.8343	0.8455
	0.2	**0.8577**	**0.8335**	**0.8463**
	0.3	0.8576	0.8332	0.8442
	0.4	0.8548	0.8305	0.8416
	0.5	0.8560	0.8333	0.8437
	0.6	0.8576	0.8331	0.8444
	0.7	0.8549	0.8285	0.8404
	0.8	0.8539	0.8286	0.8400
	0.9	0.8548	0.8293	0.8407
	1	0.8547	0.8289	0.8403

Table 1: Weight value adjustment of parent pages and child pages under Approach 3

The experimental results of three feature representation approaches are summarized in Table 2. From the table, the approach of integrating current page with neighboring pages via the topic model (TOPIC_INTEGRATED) yielded a higher performance compared to applying the topic model on current page (TOPIC_CURRENT) and application of the BOW model. Applying the TOPIC_INTEGRATED approach yielded the best performance with the F1 measure of 84.51%; an improvement of 6.11% over the TOPIC_CURRENT approach and an improvement of 23.31% over the BOW model. Applying the TOPIC_CURRENT approach helped improve the performance over the BOW by 17.2% based on the F1 measure. Thus, integrating the additional neighboring information, especially from the parent pages and child pages, via a topic model could significantly improve the performance of a classification model.

Approaches	P	R	F1
1. BOW	0.6000	0.6610	**0.6120**
2. TOPIC_CURRENT	0.7960	0.7710	**0.7840**
3. TOPIC_INTEGRATED	0.8571	0.8299	**0.8451**

Table 2: Evaluation of different feature representation approaches

5 Conclusions and Future Works

To improve the performance of Web page classification with the bag of words feature representation, we proposed a method based on a topic model to integrate the additional term features obtained from the neighboring pages. We applied the topic model approach based on the Latent Dirichlet Allocation algorithm to cluster the term features into a set of latent topics. Words assigned into the same topic are semantically related. From the experimental results, the approach of integrating current page with the neighboring pages via the topic model yielded the best performance with the F1 measure of 84.51%; an improvement of 6.11% over applying the topic model on current page approach and an improvement of 23.31% over the BOW model. Thus, integrating the additional neighboring information, especially from the parent pages and child pages, via a topic model could significantly improve the performance of a classification model.

In our future work, we plan to evaluate our proposed method on other interesting data sets such as social media contents. Other idea is to include other types of neighboring pages such as sibling and spouse pages, in addition to the parent and child pages.

References

[AGS99] Attardi, G.; Gulli, A.; Sebastiani,F.: Automatic Web page categorization by link and context analysis: Proc. of European Symposium on Telematics, Hypermedia and Artificial Intelligence (THAI), 1999, pp. 105-119.

[BNJ03] Blei, D. M.; Ng, A. Y.; Jordan, M. I.: Latent dirichlet allocation. Journal of Machine Learning Research, 2003. vol. 3, no. 5, pp. 993-1022.

[CC08] Chen, G.; Choi, B.: Web page genre classification : Proc. of 2008 ACM symposium on Applied computing , 2008, pp. 2353-2357.

[DL97] Dash, M.; Liu, H.: Feature Selection for Classification. Intelligent Data Analysis, 1997, vol. 1, no. 1–4, pp. 131-156.

[Du98] Dumais, S. et al.: Inductive Learning Algorithms and Representations for Text Categorization: Proc. of CIKM1998, 1998, pp. 148-155.

[Fu99] Furnkranz, J.: Exploiting structural information for text classification on the WWW: Proc. of the 3rd Symp. On Intelligent Data Analysis (IDA), Springer, 1999, vol. 1642, pp. 487-497.

[HD08] Haruechaiyasak, C; Damrongrat C.: Article Recommendation Based on a Topic Model for Wikipedia Selection for Schools: Proc. of the 11th International Conference on Asian Digital Libraries, 2008, pp. 339-342.

[Jo98] Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features: Proc. of the European Conference on Machine Learning (ECML), Berlin, 1998, pp. 137-142.

[QD06] Qi, X.; Davison, B. D.: Knowing a Web Page by the Company It Keeps: Proc. of the 15th ACM Conference on Information and Knowledge Management (CIKM), Arlington, Virginia, USA, 2006, pp. 228-237.

[QD08] Qi, X.; Davison, B. D.: Classifiers Without Borders: Incorporating Fielded Text From Neighboring Web Pages: Proc. of the 31st Annual International ACM SIGIR Conference on Research & Development on Information Retrieval, Singapore, 2008, pp. 643-650.

[SG06] Steyvers, M.; Griffiths, T.: Probabilistic topic models, In T., Landauer, D., McNamara, S., Dennis, and W., Kintsch, (eds), Latent Semantic Analysis: A Road to Meaning, Laurence Erlbaum, 2006.

[Sh06] Shen, D. et al.: A comparison of implicit and explicit links for web page classification: Proc. of the 15th international conference on World Wide Web table of contents, 2006, pp. 643-650.

[SLN02] Sun, A.; Lim, E.-P.; Ng, W.-K.: Web classification using support vector machine: Proc. of the 4th Int'l Workshop on Web Information and Data Management (WIDM), ACM Press, 2002, pp. 96-99.

[Va95] Vapnik, V.: The Nature of Statistical Learning Theory. In Springer-Verlag, New York, 1995.

[YP97] Yang, Y.; Pederson, J.P.: A comparative Study on Feature Selection in Text Categorization: Proc. of the 14th International Conference on Machine Learning, 1997.

[Zh07] Zhu, S. et al.: Combining content and link for classification using matrix factorization: Proc. of the 30th Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval, 2007, pp. 487-494.

Improving ASR for Continuous Thai Words Using ANN/HMM

Maleerat Sodanil [1], Supot Nitsuwat[1] , Choochart Haruechaiyasak[2]

[1]Department of Information Technology
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pibulsongkarm Rd., Bangsue, Bangkok 10800

[2]Human Language Technology Laboratory (HLT)
National Electronics and Computer Technology Center (NECTEC)
Thailand Science Park, Pathumthani 12120, Thailand

msn@ kmutnb.ac.th
sns@kmutnb.ac.th
choochart.haruechaiyasak@nectec.or.th

Abstract: The baseline system of an automatic speech recognition normally uses Mel-Frequency Cepstral Coefficients (MFCC) as feature vectors. However, for tonal language like Thai, tone information is one of the important features which can be used to improve the accuracy of recognition. This paper proposes a method for building an acoustic model for Thai-ASR using a combination of MFCC and tone information as an input feature vector. In addition, we apply Artificial Neural Network (ANN) multilayer perceptrons to estimate the posterior probabilities of a class model given a sequence of observation input. The performance of the ANN approach is compared with the Gaussian Mixture Model (GMM) used in the Hidden Markov Model Toolkit (HTK). The experiments were carried out with 2-grams and 3-grams of language model. The training and test data sets were prepared from reading speech of ten Aesop's stories from 5 male and 5 female speakers. The results showed that the proposed method can be used to improve the performance of Thai-ASR in term of reducing word error rate.

1 Introduction

The challenge in Automatic speech Recognition (ASR) is how to improve the accuracy of speech recognition in term of performance of the algorithm. There are three main parts of ASR, the first one is feature extraction that extracts distinguished feature of speech utterance, the second is training model which is typically based on the Hidden Markov Model (HMM) framework and the third is decoder which finds the best probabilistic match between speech utterance and text transcription. For tonal language like Mandarin or Thai in which tone is important for specifying the meaning of speech utterance, therefore, tone information could be considered in the system in order to improve the accuracy of speech recognition. There have been some researches proposing tone recognition or classification [SM1999, SY1995, NB2002] for improving the accuracy of speech recognition [CT2006]. Although well-known Mel-frequency cepstral coefficients (MFCC) features and HMM are widely used as feature vectors, there are some concern about testing, combining and adapting them to improve the accuracy or performance of the system which may not depend on speakers or languages [XM2006, PA2008].

The HMM is a very powerful statistical method for characterizing the observed data samples of a discrete time series. In HMM, the states are not directly visible, but variables influenced by the state are visible. Each state has a probability distribution over the possible output observation. The state transitions are also probabilistic in nature. The complete HMM model is denoted as $\lambda = (A,B,\pi)$. The HMM training procedure tries to estimate the value of state transition probability distribution (A), observation symbol probability density or emission probability (B), and initial state distribution (π). The emission probability distribution function (PDF) estimates the probability with which a given observation has been generated. However, the standard HMM based on maximum likelihood criteria (ML) has some weakness caused by several assumptions which reduce discriminative power in classification. PDF is mostly computed by gaussian mixture distribution function as baseline system in order to reduce the number of trainable parameters and lower the computational costs [DA1994]**.** However, neural network based on the conventional forward-backward algorithm has also been used to estimate the posterior probability of the state distribution given an observation sequence of speech utterance [YM1997].

Recently, Reynolds and Antoniou [TC2003] investigated the use of a layered modular/ensemble neural network architecture for acoustic modeling. This architecture decomposes the task of acoustic modeling by phone. Pavelka and Ekštein [TK2009] used the hybrid of neural network (NN) to estimate the state emission probabilities which reduce word error rate compared with GMM/HMM. These probabilities are used as the HMM state emission probabilities to perform the Viterbi decoding to find the result path of word which the system can be recognized.

Accoding to the advantage of tone information and the HMM framework which used extensively in speech recognition to model the temporal information in speech and neural network which more powerful tool for a clssification tasks due to their discriminant nature of speech manner [AT1989]. In this paper, the combination of tone and MFCC was used as input of acoustic model. The neural network multilayer perceptron is used to estimate the state emission probabilities for all classes which corresponding to phonetic units.

The remaining of this paper is organized as follows. The feature extractions will be introduced in Section 2. In Section 3, an acoustic model will be described with two methods of GMM/HMM and ANN/HMM. The experimental results and conclusion were summarized in Section 4 and Section 5, respectively.

2 Features Extraction

The objective of features extraction is to extract characteristics from the speech signal that are unique to each word which will be used to differentiate between a wide set of distinct words. In this paper, the combination of MFCC and tone features was used as an input of acoustic model.

2.1 MFCC Features

Mel-frequency cepstral coefficients (MFCC) is considered as the standard method for feature extraction in speech recognition systems. The MFCC computational starts with pre-emphasis. Then the continuous time signal (speech) is sampled at discrete time points to form a sample data signal representing the continuous time signal. The samples are quantized to produce a digital signal. Next step is framing using hamming window. In this paper, the input speech signal is segmented into frames of 25 ms of frame size with optional overlap of 10 ms. Each frame has to be multiplied with a hamming window in order to keep the continuity of the first and the last points in the frame. The discrete Fourier transform (DFT) is normally computed via the fast fourier transform (FFT) algorithm to evaluate the frequency spectrum of speech. FFT converts each frame of N samples from the time domain into the frequency domain and obtain the magnitude frequency response of each frame. Then the magnitude coefficients of the fourier transform for the speech segment is binned by correlating it with each triangular filter in the filterbank. In this paper, 24 filterbanks are used. The logarithm state simply converts the multiplication of the magnitude of the fourier transform into addition such as log energy within a frame. The final procedure for the mel frequency cepstral coefficients computation consists of performing the inverse of DFT on the logarithm of the magnitude of the mel filterbank output referred to as signal's mel cepstrum. In our experiments, 13 MFCC features plus deltas and double-deltas parameters are extracted using HTK.

2.2 Tone Features

For Thai language, the syllable consists of three parts: initial consonant, vowel and final consonant respectively. Each syllable has its tone. The fundamental frequency (F_0) or pitch can be extracted from voiced part of the time unit in the utterance. Normally, in vowel position of syllable. Therefore, the F_0 needs to be interpolated in unvoiced regions to avoid variance problems in recognition using a smoothed log-pitch estimate and its two temporal derivatives [XM2006]. In this paper, F0 is extracted and smoothed , then combined to the standard MFCC to be used as an input of acoustic model. The average magnitude difference function is used instread of autocorrelation function to extract pitch period. It computes the difference between the signal and time shifted version of itself. The average magnitude difference function [MH1974] is defined as :

$$AMDF(\tau) = \frac{1}{N}\sum_{n=0}^{N-1-\tau} |x(n) - x(n-\tau)| \quad (1)$$

Where $x(n)$ are the samples of analyzed speech frame
$x(n+\tau)$ are the samples time shifted τ seconds and N is the frame size.
The smoothing using the moving average smoothing is used as the following equation.

$$\widehat{F_n} = \frac{1}{N}\sum_{i=n-N/2}^{n+N/2} F_i \quad (2)$$

where F_i is the order i of F_0, $\widehat{F_n}$ is the smoothed F_0 of frame n, and N is the frame size.
In order to solve the end-effect problem, a simple first order differences at the start and end of the speech was used as following

$$delta_n = \begin{cases} \frac{f_{n+\theta} - f_{n-\theta}}{2\theta}, & \theta < n < N-\theta \\ f_{n+1} - f_n, & n < \theta \\ f_n - f_{n-1}, & n \geq N-\theta \end{cases} \quad (3)$$

where $delta_n$ is a delta coefficient at time n, θ is the internal distanace between two F_0, f_n is a smoothed F_0 value at time frame n and N is to total frame. The total of tone feature equal 3 feature vectors.

3 Acoustic Models

Acoustic modeling plays a critical role in improving accuracy of any speech recognition system. For the given acoustic observation O = $O_1 O_2 ... O_n$ the goal of speech recognition is to find out the corresponding word sequence that has the maximum posterior probability P(W|O) as expressed by Eq. (4).

$$\widehat{W} = argmax_w\ P(W|O) = \ argmax_w\ \frac{P(W)P(O|W)}{P(O)} \tag{4}$$

Since the maximization of Eq. (4) is carried out with the observation O fixed, the above maximization is equivalent to maximization of the following equation:

$$\widehat{W} = argmax_w P(W)P(O|W) \tag{5}$$

where P(O| W) is acoustic models and P(W) is language models, that can truly reflect the spoken language to be recognized. In this paper, an acoustic model is considered to improve the accuracy of speech recognition using neural network multilayer perceptron. The combination of MFCC standard feature vectors and tone features are given as input to ANN which will be described. To measures speech recognition error and evaluate the performance of the system. The word recognition error rate is widely used as one of the most important measures. The *Word Error Rate* is defined as:

$$WER = \frac{N - S - D - I}{N} \times 100\% \tag{6}$$

where N is the total number of words. S, D and I are number of word substitutions, deletions and insertions respectively.

3.1 GMM/HMM

A gaussian mixture model (GMM) which parameterized by a mean and a variance often modeled to estimate the emission probability density of an HMM framework. In the hidden markov toolkit (HTK), the parameter estimation was done by a flat start embedded training which required the phonetic transcriptions of training utterances to be available. HTK allows each observation vector at time t to be split into a number of S independent data streams o_{st}. The formula for computing $b_j(o_t)$ is then defined as

$$b_j(o_t) = \left[\prod_{s=1}^{S}\sum_{m=1}^{M_s} b_{jm}(o_{st}; \mu_{jm}, \Sigma_{jm})\right]^{\gamma_s} \tag{7}$$

where M_s is the number of mixture components in stream s, c_{jm} is the weight of the m'th component and $N(o; \mu, \Sigma)$ is a multivariate Gaussian with mean vector μ and covariance matrix Σ, that is

$$N(o; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} e^{-\frac{1}{2}(o-\mu)'\Sigma^{-1}(o-\mu)} \tag{8}$$

where n is the dimensionality of o. The exponent γ_s is a stream weight. It can be used to give a particular stream more emphasis.

3.2 ANN/HMM

There are two basic approaches to speech classification using neural networks: static and dynamic. In static classification, the neural network sees all of the input speech at once, and makes a single decision. By contrast, in dynamic classification which we used in this paper, the neural network sees only a small window of the speech, and this window slides over the input speech while the network makes a series of local decisions, which have to be integrated into a global decision at a later time [AT1989]. The neural network computes the weighted sum of its input and the passed this sum to a nonlinear function, most commonly a threshold or sigmoid function [RP1987]. The advantages of using neural network in HMM are the ability for discriminative training, no strong assumptions about the statistical distribution of the acoustic space, beter robustness to insufficient training data and ability to model acoustic correlation. It has been applied successfully to perform static pattern recognition or speech recognition [TK2009]. The neural network in our system is used as state emission probability estimator for HMM from a posterior probabilities.

A multilayer perceptron with fully connected neural network was used to model an output class which corresponding to phonetic unit. These posterior probabilities were used directly as the HMM state emission probabilities to perform a standard viterbi decoding which obtained the best phone sequence. Figure 1 shows the framework of acoustic model using MFCC and tone features as an input vectors. A feed-forward multilayer perceptron neural network we used is shown in Figure 2. There are 42 feature vectors per frame for the input layer with one hidden layer and all neurons use the non-linear sigmoid activation functions. The output of the neural network is a vector of posterior probabilities, with one class for each phone which is generated from input feature vectors.

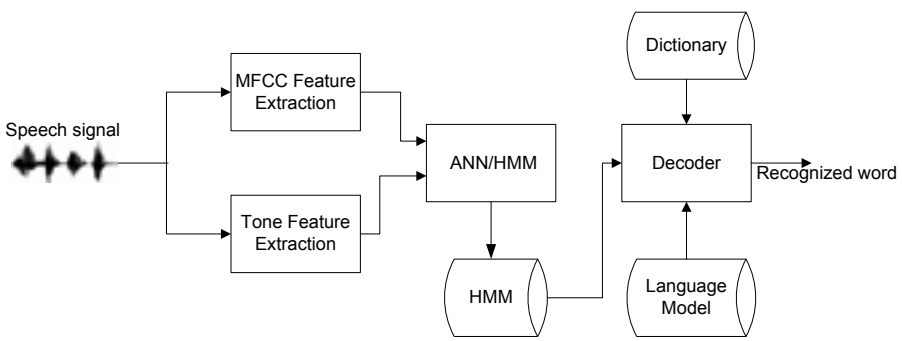


Figure 1. ANN/HMM acoustic training for HMM

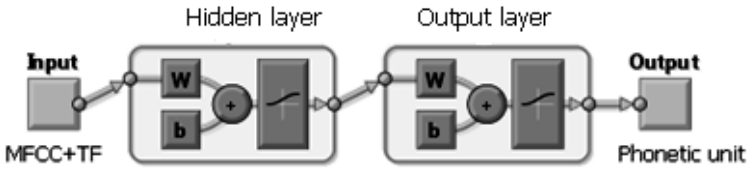


Figure 2. Neural network multilayer perceptrons

4 Experiment and Results

In this paper, the experiment was conducted using a speech corpus of ten Aesop's stories (translated into Thai) recorded from adult speakers. These stories contain all five tone levels in Thai language. The total number of words is 809 and the total number of distinct syllables is 2787. The data was collected from 10 native Thai speakers (5 male speakers and 5 female speakers), with different ages from 24 to 35 years old. The speech signals were sampled at 22 kHz and digitized with a 16 bit A/D converter. All speech data was recorded using Audacity program and stored as one sentence per file. We randomly split 80 percent of data to be used for training the model and 20 percent to be used for testing. Table 1 and Table 2 show the statistics of syllables and tone levels from the corpus. A GMM model was trained by using the Hidden Markov Toolkit [YS2002]. The parameter estimation was done by a flat start embedded training which requires the phonetic transcriptions of the available training utterances, while neural network feed-forward multilayer perceptrons was trained by Matlab. Both acoustic MFCC and Tone feature vectors are served as input vectors. The neural networks was trained on the same training data as the GMM/HMM systems with different input vectors and language model.

Story #	Number of syllable	Number of unique syllable
1	230	165
2	286	153
3	296	142
4	230	119
5	285	153
6	289	162
7	319	144
8	292	163
9	301	128
10	256	121

Table 1. Number of syllables of Aesop's stories

Tone	Number of syllable
Mid	687
Low	598
Falling	544
High	583
Rising	372

Table 2. Number of syllables for each tone of Aesop's stories

Configuration	WER(%)
MFCC + GMM + 2-gram	41.46
MFCC + GMM + 3-gram	31.15
MFCC + TF+GMM + 2-gram	26.35
MFCC + TF+GMM + 3-gram	24.76
MFCC + TF + ANN + 2-gram	25.20
MFCC + TF + ANN + 3-gram	23.35

Table 3. Experiment Results in term of Word Error Rate(%)

The results are summarized in table 3. The word error rates (WER) were reduced from the baseline HTK for both 2-grams and 3-grams of the language model. For the configuration of the combination of MFCC and tone features input, word error rates were reduced when ANN was applied to estimate the state emission probabilities in acoustic model. As the results, there are two difference things: Firstly, the different input features of acoustic model which we used pure MFCC and the combination of MFCC and tone feature vectors. Secondly, the method that uses to estimate the state emission probabilities. In this experiment, GMM and ANN were applied to estimate the state emission probabilities. Normally, speech recognition will gives some error especially in continuous speech because of the difficulty to segment the speech signal and the speaking speed. It shows more than 30 percent of word error rate for our corpus and around 20-25 percent for isolated word. When tone features were applied to be an input features, the recognition performance was improved more than 6% as shown in Table 3. Also the performance improved when ANN was applied with tone features. According to training data that we used in the experiments is not big enough, then the error might be occurred in most past of the adjacent syllable event there were difference tone. However, the language model is considered to greatly increase the performance of the continuous speech system as used in this experiment. Although, the different WER between ANN and GMM is not big enough, at least it showed that the proposed system can be improved the performance of speech recognition by reducing the word error rate.

5 Conclusions

In this paper, an approach based on the Artificial Neural Network (ANN) multilayer perceptrons is proposed to score the state emission probabilities under the HMM framework. A combination of the Mel-Frequency Cepstral Coefficients (MFCC) and tone information is used as input feature vectors to train an acoustic model. The total of 42 input vectors were normailzed and classified by multilayer perceptron neural network with 62 target outputs, each represents the phonetic units. The experiments were carried out to compare the performance between the ANN approach and the Gaussian Mixture Model (GMM) used by HTK with different language models. The results showed that the ANN approach with MFCC with tone features yielded a higher accuracy, i.e., lower word error rate (WER), for speech recognition compared to the GMM approach.

References

[AT1989] A. Waibel, T. Hanazawa, G. Hinton, K. Shiano, and K.Lang, Phoneme recognition using time-delay neural networks, In IEEE Trans. on Acoust., Speech, and Signal Processing, volume 37(3), pp. 328-339,1989.

[CT2006] Chutima Pisarn and Thanarak Theeramunkong, Improving Thai Spelling Recognition with Tone features, Springer-Verlag Berlin Heidelberg pp.388-398, 2006.

[DA1994] David M. Lubensky, Ayman O. Asadi, and Jayant M. Naik. Connected digit recognition using connectionist probability estimators and mixture-gaussian densities. In ICSLP, pp. 295--298.

[JR1997] John-Paul Hosom and Ronald A.Cole, A diphone-based digit recogition system using neural network, ICASSP-97,1997.

[MH1974] M. J. Ross, H. L. Shaffer, A. Cohen, R. Freudberg, and H. J. Manley, “Average magnitude difference function pitch extractor,” IEEE Transactions on Acoustics, Speech, Signal Processing, vol. ASSP22, pp. 353–362, 1974.

[NB2002] N. Thubthong and B. Kijsirikul, An empirical study for constructing Thai tone models, in Proc. the 5th Symposium on Natural Language Processing and Oriental COCOSDA Workshop, pp. 179–186, 2002.

[PA2008] Poonam Bansal, Anuj Kant, Sumit Kumar, Akash Sharda, Shitij Gupta. Improved model of HMM/GMM for speech recognition, "Intelligent Information and Engineering Systems" INFOS, pp.69-74, 2008.

[SM1999] S. Potisuk, M. P. Harper, and J. Gandour, Classification of Thai tone sequences in syllable-segmented speech using the analysis-by synthesis method, IEEE Transactions on Speech Audio Processing, vol. 7, no.1, pp. 95–102, 1999.

[SY1995] S.-H. Chen and Y.-R.Wang, Tone recognition of continuous Madarin speech based on neural networks, IEEE Transactions on Speech audio Processing, vol.3, no. 2, pp. 146–150, 1995.

[TC2003] T. Jeff Reynolds, Christos A. Antoniou, Experiments in speech recognition using a modular MLP architecture for acoustic modelling, Inf. Sci. 156 pp. 39-54, 2003.

[TK2009] Tom´aˇs Pavelka and Kamil Ekˇstein , A Comparison of Acoustic Models Based on Neural Networks and Gaussian Mixtures, Springer-Berlin / Heidelberg, pp.291-298, Volume 5729/2009.

[XM2006] X. Lei, M. Siu, M. Ostendorf, and T. Lee, Improved Tone Modeling for Mandarin Broadcast News Speech Recognition. Interspeech, 2006.

[YM1997] Younghong Yan, Mark Fanty and Ron Cole, Speech Recognition using neural networks with forward-backward probability generated targets, ICASSP-97,Vol.4 1997.

[YS2002] Young, S., et al.: The HTK Book, Cambridge University Engineering Dept, 2002.

A Comparative Study on Feature Weight in Thai Document Categorization Framework

Nivet Chirawichitchai[(1)] , Parinya Sa-nguansat[(2)], Phayung Meesad[(3)]

[(1)]Department of Information Technology, Faculty of Information Technology
[(3)]Department of Teacher Training in Electrical Engineering, Faculty of Technical Education
King Mongkut's University of Technology North Bangkok
[(2)]Faculty of Information Technology, Rangsit University
nivet99@hotmail.com, sanguansat@yahoo.com, pym@kmutnb.ac.th

Abstract: Text Categorization is the process of automatically assigning predefined categories to free text documents. Feature weighting, which calculates feature (term) values in documents, is one of important preprocessing techniques in text categorization. This paper is a comparative study of feature weighting methods in statistical learning of Thai Document Categorization Framework. Six methods were evaluated, including Boolean, tf, tf×idf, tfc, ltc, and entropy weighting. We have evaluated these methods on Thai news article corpus with three supervised learning classifiers: Support Vector Machine (SVM), Decision Tree (DT), and Naïve Bayes (NB). We found that ltc weighting method is most effective in our experiments with SVM and DT algorithms, while entropy and Boolean weighting is more effective than the weighting with NB algorithms. Using ltc weighting with a SVM classifier yielded a very high classification performance with the F1 measure equal to 96%.

1 Introduction

In recent years we have seen an exponential growth in the volume of text documents available on the Internet. While more and more textual information is available online, effective retrieval is difficult without organization and summarization of document content. Text categorization is one solution to this problem. A growing number of statistical classification methods and pattern recognition techniques have been applied to text categorization in recent years, including nearest neighbor classification, Naïve Bayes, decision trees, neural networks, boosting methods, and Support Vector Machines.

Vector Space Model (VSM) [SL68] is a major method for representing documents in text categorization. In this model, each document d is considered to be a vector in the feature space. For a document d, VSM represents it by vector $V_d = (v_{d1}, v_{d2}, \ldots, v_{dn})$, where v_{di} stands for the value of ith feature (term) according to d. Thus, one major characteristic of VSM is calculation of feature values in document vectors. The processing that yields feature values is called feature weight.

A widely used method for feature weight is tf×idf [SB88]. tf is the abbreviation for term-frequency, which stands for the capacity of features expressing document content. idf is the abbreviation for inverse document frequency, which stands for the capacity of features discriminating similar documents. The motivation behind idf is that terms appearing frequently in many documents have limited discrimination power. Because methods of feature selection evaluate feature by scores, we can also adopt these methods for feature weight [YLZ04]. In this paper, we study several excellent weighting methods, including Boolean, tf, tf×idf, tfc, ltc, and entropy weighting, and compare performance of these methods on Thai news article corpus [AE99].

2 Feature Extraction

2.1 Preprocessing

The first step in text categorisation is to transform documents, which typically are strings of characters, into a representation suitable for the learning algorithm and the classification task. The text transformation usually involves of the following processes: removing HTML tags, removing stopwords, and performing word stemming. The stopwords are frequent words that carry no information (i.e. pronouns, prepositions, conjunctions etc.). By word stemming we mean the process of suffix removal to generate word stems. This is done to group words that have the same conceptual meaning, such as walk, walker, walked, and walking. The Porter stemmer [SL68] is a well-known algorithm for this task.

2.2 Weighting Scheme

The perhaps most commonly used document representation is the so called vector space model [SL68]. In the vector space model, documents are represented by vectors of words. Usually, one has a collection of documents which is represented by a word-by-document matrix A, where each entry represents the occurrences of a word in a document, i.e.,

$$\mathbf{A} = (\mathbf{a}_{ik})$$

where $\mathbf{a}_{ik}$ is the weight of word i in document k. Since every word does not normally appear in each document, the matrix A is usually sparse. The number of rows, M , of the matrix corresponds to the number of words in the dictionary. M can be very large. Hence, a major characteristic, or difficulty of text categorization problems is the high dimensionality of the feature space. In Section we discuss different approaches for dimensionality reduction. There are several ways of determining the weight $\mathbf{a}_{ik}$ of word i in document k, but most of the approaches are based on two empirical observations regarding text: The more times a word occurs in a document, the more relevant it is to the topic of the document. The more times the word occurs throughout all documents in the collection, the more poorly it discriminates between documents.

Let f_{ik} be the frequency of word i in document k, N the number of documents in the collection, M the number of words in the collection after stopword removal and word stemming, and n_i the total number of times word i occurs in the whole collection. Next we describe 6 different weighting schemes that are based on these quantities [AE99].

Boolean weighting

The simplest approach is to let the weight equal to 1 if the word occurs in the document and 0 otherwise:

$$a_{ik} = \begin{cases} 1 & \text{if } f_{ik} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Term frequency weighting (tf)

Another simple approach is to use the frequency of the word in the document:

$$a_{ik} = f_{ik}$$

tf × idf-weighting

The previous two schemes do not take into account the frequency of the word throughout all documents in the collection. A well-known approach for computing word weights is the tf × idf-weighting, which assigns the weight to word i in document k in proportion to the number of occurrences of the word in the document, and in inverse proportion to the number of documents in the collection for which the word occurs at least once.

$$a_{ik} = f_{ik} * \log\left(\frac{N}{n_i}\right)$$

tfc-weighting

The tf × idf-weighting does not take into account that documents may be of different lengths. The tfc-weighting is similar to the tf × idf-weighting except for the fact that length normalisation is used as part of the word weighting formula.

$$a_{ik} = \frac{f_{ik} * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{j=1}^{M}\left[f_{jk} * \log\left(\frac{N}{n_j}\right)\right]^2}}$$

ltc-weighting

A slightly different approach uses the logarithm of the word frequency instead of the raw word frequency, thus reducing the effects of large differences in frequencies.

$$a_{ik} = \frac{\log(f_{ik} + 1.0) * \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{j=1}^{M}\left[\log(f_{jk} + 1.0) * \log\left(\frac{N}{n_j}\right)\right]^2}}$$

Entropy weighting

Entropy-weighting is based on information theoretic ideas and is the most sophisticated weighting scheme. In it turned out to be the most effective scheme in comparison with 6 others. Averaged over five test collections, it was for instance 40 % more effective than word frequency weighting. In the entropy-weighting scheme, the weight for word *i* in document *k* is given by:

$$a_{ik} = \log(f_{ik} + 1.0) * \left(1 + \frac{1}{\log(N)} \sum_{j=1}^{N}\left[\frac{f_{ij}}{n_i} \log\left(\frac{f_{ij}}{n_i}\right)\right]\right)$$

2.3 Dimensionality Reduction

A central problem in statistical text classification is the high dimensionality of the feature space. There exists one dimension for each unique word found in the collection of documents, typically hundreds of thousands. Standard classification techniques cannot deal with such a large feature set, since processing is extremely costly in computational terms, and the results become unreliable due to the lack of sufficient training data. Hence, there is a need for a reduction of the original feature set, which is commonly known as dimensionality reduction in the pattern recognition literature. Most of the dimensionality reduction approaches can be classified into feature selection. Feature selection attempts to remove non-informative words from documents in order to improve categorisation effectiveness and reduce computational complexity. In their experiments, the authors found the three first to be the most effective. Below a short description of these methods is given [AE99,YP97].

χ^2 -statistic (Chi-square)

The χ^2 -statistic measures the lack of independence between word *w* and class c_j . It is given by:

$$\chi^2(w, c_j) = \frac{N \times (A\,D - C\,B)^2}{(A + C) \times (B + D) \times (A + B) \times (C + D)}$$

Here A is the number of documents from class c_j that contains word w, B is the number of documents that contains w but does not belong to class c_j, C is the number of documents from class c_j that does not contain word w, and D is the number of documents that belongs to class c_j nor contains word w. N is still the total number of documents. Two different measures can be computed based on the χ^2 -statistic

Information gain (IG)

Information Gain measures the number of bits of information obtained for category prediction by knowing the presence or absence of a word in at document. Let . Let c_1, ..., c_K denote the set of possible categories. The information gain of a word w is defined to be:

$$IG(w) = -\sum_{j=1}^{K} P(c_j) \log P(c_j) + P(w) \sum_{j=1}^{K} P(c_j|w) \log P(c_j|w) + P(\bar{w}) \sum_{j=1}^{K} P(c_j|\bar{w}) \log P(c_j|\bar{w})$$

Here $P(c_j)$ can be estimated from the fraction of documents in the total collection that belongs to class c_j and $P(w)$ from the fraction of documents in which the word w occurs. Moreover, $P(c_j/w)$ can be computed as the fraction of documents from class c_j that have at least one occurrence of word w and $P(c_j/w)$ as the fraction of documents from class c_j that does not contain word w. The information gain is computed for each word of the training set, and the words whose information gain is less than some predetermined threshold are removed.

Document Frequency Thresholding (DF)

The document frequency for a word is the number of documents in which the word occurs. In Document Frequency Thresholding one computes the document frequency for each word in the training corpus and removes those words whose document frequency is less than some predetermined threshold. The basic assumption is that rare words are either non-informative for category prediction, or not influential in global performance.

3 Classification Algorithms

The goal of classification is to build a set of models that can correctly predict the class of the different objects. The input to these methods is a set of objects, the classes which these objects belong to (i.e., dependent variables), and a set of variables describing different characteristics of the objects (i.e., independent variables). Once such a predictive model is built, it can be used to predict the class of the objects for which class information is not known *a priori*. The key advantage of supervised learning methods over unsupervised methods (clustering) is that by having an explicit knowledge of the classes the different objects belong to, these algorithms can perform an effective feature selection if that leads to better prediction accuracy. This section gives a brief introduction to three well-known algorithms that are widely used for text classification.

3.1 Naive Bayes (NB)

NB algorithm has been widely used for document classification, and shown to produce very good performance. The basic idea is to use the joint probabilities of words and categories to estimate the probabilities of categories given a document. NB algorithm computes the posterior probability that the document belongs to different classes and assigns it to the class with the highest posterior probability. The posterior probability of class is computed using Bayes rule and the testing sample is assigned to the class with the highest posterior probability. The naive part of NB algorithm is the assumption of word independence that the conditional probability of a word given a category is assumed to be independent from the conditional probabilities of other words given that category [LE98].

3.2 Support Vector Machine (SVM)

SVM algorithm is based on the structure risk minimization principle. It has been shown in previous works to be effective for text categorization. SVM divides the term space into hyperplanes or surface separating the positive and negative training samples. An advantage of SVM is that it can work well on very large feature spaces, both in terms of the correctness of the categorization results and the efficiency of training and categorization algorithm. However, a disadvantage of SVM training algorithm is that it is a time consuming process, especially training with a large corpus [JO98].

3.3 Decision Tree (DT)

DT algorithm is a common method used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables. Each interior node corresponds to one of the input variables; there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf [QU86].

4 Thai Document Categorization Framework

Figure 1 illustrates the Thai Document Categorization framework. The inputs are news articles pre-classified into a set of categories. The news articles are first pre-processed by the text processing module. For Thai language, the main task of text processing is the segmentation of texts into word tokens. Thai texts are naturally unsegmented, i.e., words are written continuously without the use of word delimiters. Due to this distinct characteristic, preparing a feature set for Thai text categorization is more challenging than Latinbased languages such as English, French, and Spanish. With Latin-based languages, a text string can easily be tokenized into terms by observing the word delimiting characters such as spaces, semicolons, commas, quotes, and periods.

To prepare a feature set for Thai news article corpus, we must first apply a word segmentation algorithm to tokenize text strings into series of terms. We use the state-of-the-art word segmentation program called LexTo are dictionary based on a longest matching algorithm [HKD08] as a tokenizer in this Bag-Of-Words approach. Once a set of extracted words is obtained from the training news corpus, the collected words are to removing the HTML tags, stopword, stemming from dictionary list. The output from this step we use the weighting scheme for assigning the feature values as described in Section 2.2, we reduce the number of word features by applying the dimensionality reduction technique as described in Section 2.3.

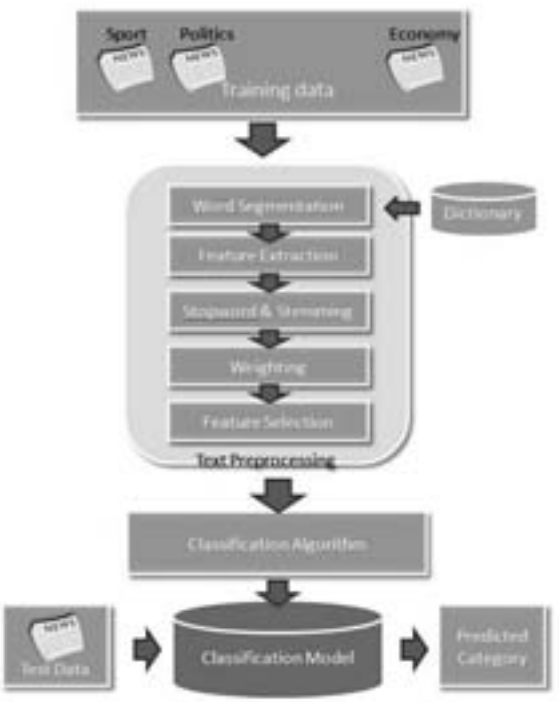


Figure 1: Thai Document Categorization Framework

The output from this step is a set of feature vectors representing news articles from the corpus. A classification algorithm could learn from the feature vector to build a classification model. To predict a category, a test news article is first transformed into a feature vector. The classification model is used to calculate the scores for each category based on the test feature vector. The test article is assigned to the category whose score is the maximum among all other categories [HA08].

Our main contribution for this paper is to perform a comparative study on feature weighting scheme with different feature selection methods for Thai document categorization.

5 Experiments and Results

We performed experiments using a collection of news articles obtained the Web. There are ten news categories: economics, education, entertainment, international, politics, society, sports, farming, Bangkok, and technology. The total number of training is 12,000 articles. We used WEKA [HA09] an open-source machine learning tool, to perform the experiments. We used the default setting for all algorithms. For SVM, the default kernel function is Linear kernel. Classification effectiveness is usually measured by using precision (p) and recall (r). Precision is the proportion of truly positive examples labelled positive by the system that were truly positive and recall is the proportion of truly positive examples that were labelled positive by the system. The F1 function which combines precision and recall is computed as:

$$F_1 = \frac{2.p.r}{p+r}$$

We tested all algorithms by using the 10-fold cross validation method. The results in terms of precision, recall and F1 are the averaged values calculated across all 10-fold cross validation experiments. The experimental results of these six feature weighting methods with respect to F1 measure on Thai news article corpus in combination with three feature selection and three learning algorithms are reported from Figure 2 to Figure 4.

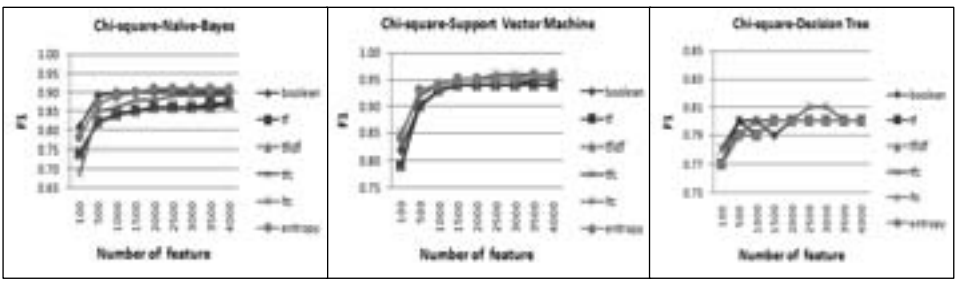


Figure 2: Results of different weighting methods on chi-square using three learning algorithms

Figure 2 summarizes the categorization on the chi-square feature selection method results for using three learning algorithms on Thai news article corpus after feature weight using Boolean,tf, tf×idf,tfc,ltc and entropy weighting, respectively. Two observations from the categorization results are as follows. First, ltc weighting is more effective than Boolean, tf, tf×idf, tfc, and entropy weighting with SVM and DT, While entropy weighting more effective than another weighting with NB. Second, all term weighting methods reach a peak at the full feature and the best F1 points on ltc-chi-SVM is 96%, the best F1 points on entropy-chi-NB is 91% and the best F1 points on ltc-chi-DT is 80%, respectively.

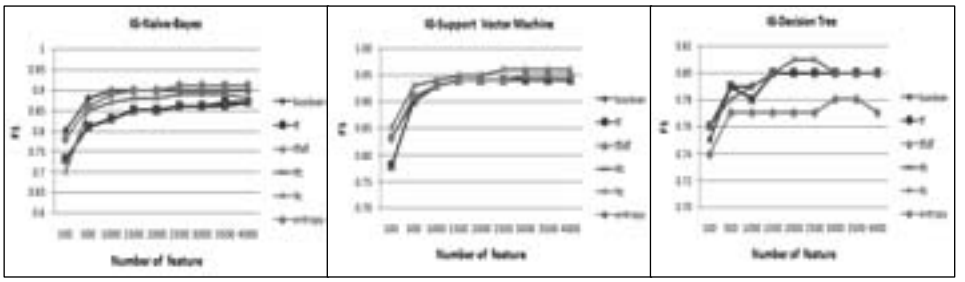


Figure 3: Results of different weighting methods on Information gain using three learning algorithms

Figure 3 summarizes the categorization on the Information gain feature selection method results when using the three learning algorithms on Thai news article corpus after feature weight using Boolean,tf, tf×idf, tfc, ltc and entropy weighting, respectively. Two observations from the categorization results are as follows. First, ltc weighting is more effective than Boolean, tf, tf×idf, tfc, and entropy weighting with SVM and DT, While entropy weighting is more effective than another weighting with NB. Second, all term weighting methods reach a peak at the full feature and the best F1 points on ltc-IG-SVM is 96%, the best F1 points on entropy-IG-NB is 91%, and the best F1 points on ltc-IG-DT is 81%, respectively.

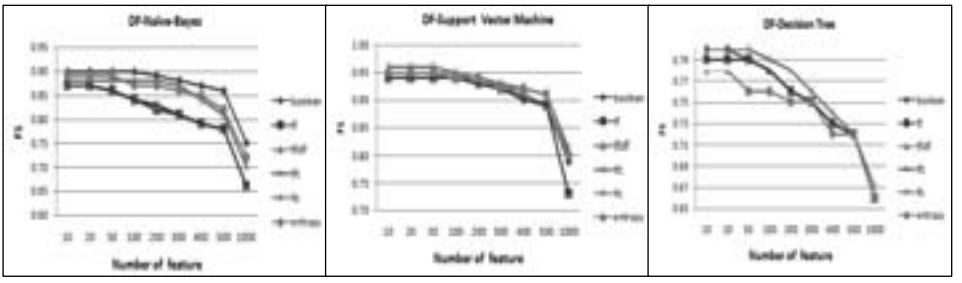


Figure 4: Results of different weighting methods on Document Frequency Thresholding using three learning algorithms

Figure 4 summarizes the categorization on the Document Frequency Thresholding feature selection method results for using three learning algorithms on Thai news article corpus after feature weight using Boolean, tf, tf×idf, tfc, ltc and entropy weighting, respectively. Two important key points from the categorization results were observed. First, Boolean weighting is more effective than tf, tf×idf, tfc, ltc and entropy weighting with NB, While ltc weighting is more effective than the weighting with SVM and DT. Second, all term weighting methods reach a peak at the full feature and the best F1 points on ltc-DF-SVM is 96%, the best F1 points on entropy-DF-NB is 89%, and the best F1 points on ltc-DF-DT is 80%, respectively.

6 Conclusion

This is an evaluation of feature weighting methods for Thai document categorization framework. We found ltc weighting most effective in our experiments with SVM and DT algorithms, while entropy and Boolean weighting are more effective than the weighting with NB algorithms. We also discovered that the ltc weighting is suitable to combination with all feature selection methods. The ltc weighting with SVM algorithm yielded the best performance with the F1 measure of all algorithms. Our experimental results also reveal that feature weight methods react on the effectiveness of Thai document categorization.

References

[SL68] G. Salton and M. E. Lesk. Computer evaluation of indexing and text processing. Journal of the ACM, 1968, 15(1): 8-36.

[SB88] G. Salton, C. Buckley. Term-weighting approaches in automatic text retrieval. Information Processing and Management, 1988, 24 (5): 513-523.

[YLZ04] J. X. Yu, X. Lin, H. Lu, and Y. Zhang: A Comparative Study on Feature Weight in Text Categorization , APWeb 2004, Springer-Verlag Berlin Heidelberg , 2004, p. 588–597.

[AE99] K. Aas and L. Eikvil. Text Categorization: a Survey.Report No.1Norwegian Computing Center. 1999.

[YP97] Y. Yang and J. P. Pedersen. A comparative study on feature selection in text categorization. Processing of the Fourteenth International Conference on Machine Learning, 1997, p.412–420.

[LE98] D. Lewis. Naive bayes at forty: The independence assumption in information retrieval. Processing of European Conference on Machine Learning, 1998, p.4–15.

[JO98] T. Joachims. Text categorization with support vector machines:Learning with many relevant features. Processing of the 10th European Conference on Machine Learning, 1998, p.137–142.

[QU86] J. R. Quinlan. Induction of decision trees. Machine Learning,1986, p.81–106.

[HKD08] C. Haruechaiyasak, S. Kongyoung and M. Dailey, “A comparative study on Thai word segmentation approaches”, Processing of the ECTI-CON 2008, p.125-128.

[HA08] C. Haruechaiyasak, W. Jitkrittum, C. Sangkeettrakarn and C. Damrongrat. Implementing News Article Category Browsing Based on Text Categorization Technique. International Conference on Web Intelligence and Intelligent Agent Technology - Volume 03, 2008, p.143-146.

[HA09] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and Ian H. Witten. The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter – Volume 11, 2009, p.10-18.

An Evaluation Framework for Semantic Search in P2P Networks

Florian Holz Hans-Friedrich Witschel Gregor Heinrich Gerhard Heyer
Sven Teresniak

Abteilung ASV, Institut für Informatik, Universität Leipzig
{holz|witschel|heyer|teresniak}@informatik.uni-leipzig.de
gregor@arbylon.de

Abstract:
We address the problem of evaluating peer-to-peer information retrieval (P2PIR) systems with semantic overlay structure. The P2PIR community lacks a commonly accepted testbed, such as TREC is for the classic IR community. The problem with using classic test collections in a P2P scenario is that they provide no realistic distribution of documents and queries over peers, which is, however, crucial for realistically simulating and evaluating semantic overlays. On the other hand, document collections that can be easily distributed (e.g. by exploiting categories or author information) lack both queries and relevance judgments.

Therefore, we propose an evaluation framework, which provides a strategy for constructing a P2PIR testbed, consisting of a prescription for content distribution, query generation and measuring effectiveness without the need for human relevance judgments. It can be used with any document collection that contains author information and document relatedness (e.g. references in scientific literature). Author information is used for assigning documents to peers, relatedness is used for generating queries from related documents. The ranking produced by the P2PIR system is evaluated by comparing it to the ranking of a centralised IR system using a new evaluation measure related to mean average precision. The combination of these three things – realistic content distribution, realistic and automated query generation and distribution, and a meaningful and flexible evaluation measure for rankings – offers an improvement over existing P2PIR evaluation approaches.

1 Introduction

Semantic peer-to-peer information retrieval (P2PIR) offers important advantages to the design of information infrastructures, among them scalability, localisation of information and localisation of access control. However, for evaluation and comparison of P2PIR systems, no sufficiently objective methodologies exist yet as they do for "classical" information retrieval (IR) where (1) test collections are available (e.g., TREC [TRE]) and (2) methods to evaluate retrieval performance are scientifically well-established [VH06].

The classical evaluation method in information retrieval is to train the IR system with a test collection and determine the precision and recall values for a set of queries and

relevance judgements defined by the test collection. Transferring this method to P2PIR is not a straightforward task, however. Although in principle, precision and recall can be defined on a P2PIR result set analogous to a centralised IR system, the features of the test collections available are not sufficient for practical evaluations. The main problems arise from the distributed character of the required test collection: There are no data available that provide realistic distributions of the test collection over peers, which – in order to be sufficient data for a methodology analogous to classical IR – must include distributions of documents, queries and associated relevance judgements over the P2P network.

This lack of available data has become a serious obstacle in the advancement of P2PIR research. To alleviate this problem, an evaluation framework is proposed in this paper that draws from two approaches: First, a plausible distribution of documents and queries is elaborated for the case of P2PIR, and second, an evaluation method is proposed to overcome the lack of relevance judgements for queries. Using these two contributions, we present an evaluation plan for a P2PIR system.

We continue this article by reviewing the technical background on P2PIR in Section 2 and review the state of the art in evaluation methods in Section 3. Subsequently, we present the parts of our framework in Section 4 and finally describe how to parameterise the testbed in Section 5.

2 Background

Since the evaluation framework presented in this paper is designed for unstructured, "non-flooding" P2P systems with semantic overlays, it is necessary to describe the characteristics of such systems:

A P2P system is called *unstructured* if the topology of the overlay network is not fixed in any way and if content can be stored anywhere in the system. This is in contrast to structured systems such as distributed hash tables [RFH+01, SMK+01, ZKJ01] where each peer is responsible for storing data that corresponds to a certain range of hash values. Unstructured systems are more flexible than structured ones because no control over data placement is assumed.

On the other hand, the systems we are interested in avoid flooding the network with queries. Flooding imposes a great load on the underlying network (cf. [Gnu]).

In order to avoid flooding and still guarantee good recall, such systems have to provide solutions to the following three tasks:

- *Peer description*: For making predictions about which peer is likely to be capable of contributing to a certain topic, there must be descriptions or *profiles* of peers' contents.
- *Query routing*: Assuming that peers have both the address and the profile of their neighbours, these profiles can be used for *informed search* (rather than flooding): they must be matched against queries in order to decide where the queries should

be forwarded. Queries that are sent to the wrong peers will fail to retrieve relevant documents.

- *Neighbour selection*: in order to facilitate query routing, peers should choose their neighbours in a way that maximises the probability of always finding a good neighbour to forward queries to. Networks in which peers choose neighbours according to semantic criteria are called *semantic overlay networks* [CGM02b].

Regarding the last task, there is an increasing agreement in the research community that peers should be organised in clusters of semantic similarity [SMZ03]. Additionally, some random shortcuts have been shown to be beneficial, resulting in a small world graph [Kle00] structure of the overlay [AWMM06, Sch05, WB05, LLS04, ZGG02].

In the following, we will shortly describe our own approach to semantic search and overlay structuring. It has been developed as part of a project supported by DFG[1] and was described and evaluated in more detail in [Wit05, WB05].

The algorithm for *building overlay structure* is based on gossiping: by searching for its own profile – i.e. sending out queries that consist of the profile – and receiving answers from other peers, each participant in the network fills up a part of its routing table with addresses and profiles of neighbours that offer content similar to its own. Peers thus organise into clusters of semantic similarity.

Another part of the routing table is reserved for some arbitrarily chosen neighbours (random shortcuts) that provide links between different clusters of peers. The combination of these two neighbour selection strategies is intended to result in a small world network structure (see [Kle00] for a theoretical model) that can be exploited for an efficient search algorithm.

The *search mechanism* works as follows: each peer that receives a query, first scans its local index for matching documents and then forwards the message to just one (or at least only a small number) of its neighbours: the one whose profile best matches the query, i.e. the one deemed most likely to have an answer to it. This continues until the time-to-live (TTL) of the query expires.

Since peers are organised into clusters of semantic similarity, queries will find many relevant results once they have reached the right cluster because peers that can contribute answers to a given query are likely to know others that also can.

Because in this and most other systems with semantic overlays, structure is built based on peers' shared content, it should be clear that the evaluation of such systems requires realistic models of content and query distribution. These problems are tackled in the remainder of this paper.

[1] DFG project grant nr. 255712

3 Related Work

In order to choose a feasible approach for getting a *testbed*, one might have first a look at the requirements a P2PIR test collection must satisfy: it has to provide the usual features of a IR test collection, i.e. documents, queries and relevance judgments, and in addition, there needs to be a prescription of how to distribute documents and queries among peers. Distribution of documents is either done in a way that springs naturally from the collection, e.g. via author information [BMR03] or built-in categories [CGM02a, SCK03], or it is established in less natural ways via clustering [NBMW06], latent concept analysis [HTW06], or domains of web pages [LC03, KJPD05]. In [Coo04], documents are even artificial which facilitates their distribution, but doesn't result in a real testbed.

In [BMR03], a number of different testbeds is proposed. In each, authors are identified with peers and all the papers an author has written are assigned to the corresponding peer. One testbed uses TREC, the other two (Reuters and CiteSeer) do not have queries or relevance judgments. Therefore, queries are generated randomly from documents and documents are considered relevant w.r.t. a query if they contain all query terms.

In [LC03, KJPD05], web pages are used as a test collection and the prefixes of their URLs define the peers. In [LC03], the TREC WT10g web test collection is used, together with queries generated from the documents. Evaluation is done by comparing results to that of a centralised system using plain precision and recall. Klampanos et al. [KJPD05] also use the WT10g collection and define and compare various testbeds. In addition to domains of URLs, they also use link information and textual similarity to enlarge the document set on a peer. Queries and relevance judgements are taken from the original WT10g collection (100 queries).

The approach of [CGM02a, SCK03] uses freely available copora (like OpenDirectory or CiteSeer) which contains classified documents. The distribution of documents over peers follows the topic structure in the corpus.

Without a preclassification of documents one can get a structuring by clustering the corpus. Therefore, Neumann et al. [NBMW06] use Wikipedia, with a clustering based on the internal linking structure between the wikipedia articles. To distribute the documents among the peers, the clusters are seperated into chunks and documents are chosen from these chunks using a sliding window in order to get overlap. Neumann further proposes the usage of google zeitgeist archive for generating queries, which, however, come without relevance judgements. In addition, the construction of overlapping content between peers is not as natural as e.g. a fuzzy clustering would be.

An approach based on statistics is proposed in [Coo04]. There, histograms over the counts of relevant documents per query, the degree of document replication, and the mean count of relevant documents for a query on one peer are used to generate an artificial testbed, which refactors real statistical relations. The disadvantage of this approach is that one doesn't get a real testbed, but one which merely behaves similar in certain circumstances, and that real data of P2P communities and therefore feasible relevance judgments are hard to get.

Heinrich et al. propose in [HTW06] to use a latent concept analysis to estimate relevant

parameters of the distribution of topics over peer and documents and to use these parameters to distribute another document collection over peers in the same manner. That means that it is only necessary to have a little statistics of a P2P network which can be scaled up to a much bigger set of documents and peers. An alternative to create P2P evaluation data without data from a real P2P community was proposed in [Hei06] based on corpora with author and citation data.

One can see that often a natural content distribution means that there are no queries and relevance judgments, in which case artificial queries are generated from the collection's documents [LC03, BMR03] or taken from other sources [NBMW06]. In that case, relevance judgments are not available and performance is compared to a centralised setting via simple precision and recall measures. The next section presents a new idea of designing a P2PIR testbed and a new evaluation measure.

4 The proposed evaluation framework

Our framework for constructing a concrete testbed consists of two distinct parts: a well-defined distribution of content (documents and automatically generated queries) among peers, which can be applied to various existing corpora, and an evaluation measure, which respects the ranking of the retrieved documents and has no need for given relevance judgements.

4.1 Distributing content and queries

Since nodes in peer-to-peer networks correspond most often to single persons, a P2PIR testbed should provide a content distribution that realistically reflects the interests of persons running peers. One way to achieve this is to identify peers with authors of documents, as done in [BMR03].

We propose to use corpora that provide relations between documents and authors (authoring relation) as well as between documents and other documents (citation relation). In [Hei06] various freely available examples of such corpora are analysed, e.g. the CiteSeer corpus [GBL98][2] or the Cora corpus [MNRS00][3].

We propose to map the entities and relations available in these corpora to those in a P2PIR testbed in the following way:

- *Peers* map to *authors* in a relational corpus.
- *Documents* associated with a peer trivially map to documents within a relational corpus via the authoring relation. As an extended approach, the documents cited in authors' documents can be additionally associated with the peer. After all, people

[2] http://citeseer.ist.psu.edu/oai.html
[3] http://www.cs.umass.edu/∼mccallum/code-data.html

are likely to be competent in the field of expertise that their citations are concerned with. Using this extended document mapping allows a higher degree of overlap between the peers. The plain set of authored documents restricts document overlap between peers to co-authorship relations.

- *Queries* associated with a peer map to documents that are cited by the associated author's documents. For this, only citations are valid that refer to documents included in the corpus. This makes sense because, in our opinion, it is realistic to assume that people ask questions concerning issues which will extend their knowledge of the things within their focus of interest (as reflected by their documents).

Generating queries automatically out of the collection is considered necessary in order to have a sufficient number of queries, which are semantically associated to and issued by the persons running the peers. However, this means that there will be no relevance judgments for these queries. The next section presents an approach to evaluation without the need for any human relevance judgements in this setting.

4.2 Evaluation measure

Since there are no relevance judgments for queries, the performance of distributed retrieval algorithms will be measured by comparing it to a centralised setting. This means that the optimal value of the evaluation measure is reached if the P2P system retrieves the same documents and in the same order as a centralised system. It is assumed that both systems use the same retrieval function for ranking documents and the same global basis for estimating term weights (cf. [Kro02, WB05]) so that the difference between rankings is only attributable to the failure of the P2P system to retrieve certain documents. This, in turn, depends on the routing strategy: a good strategy routes queries to those peers that can contribute the most relevant documents (i.e. those ranked most highly by the centralised system) and leaves aside peers that can only contribute documents ranked lowly by the centralised system.

For this, we would like to propose a new measure that reflects the capability to retrieve the highest ranked documents, and does so better than naïve approaches that only use simple precision and recall on some defined sets of returned documents [LC03, NBMW06]. Thus, a system that retrieves the 20 documents ranked highest by the centralised system receives a better evaluation score than a system that retrieves the 200 lowest ranked documents.

The measure is closely related to mean average precision (cf. [VH06], chapter 3) and can be computed as follows:

- We assume that a query in the P2PIR system returns a ranked list A of the best k documents it can find (after merging results, that is). The value of k is assumed to be set by the user who tells the system how many (namely k) documents he/she is willing to look at.
- This will be compared to the ranked list C of *all* documents returned by a centralised

search engine.

- We now mark the positions of all documents in A within C. As an example, let's assume that the user has requested to view the best $k = 3$ documents and that $A = [L, M, O]$ and $C = [K, \mathbf{L}, \mathbf{M}, N, \mathbf{O}, P]$.

- Now we compute

$$\sum_{i=1}^{k} \frac{m(A_i)prec(A_1, ..., A_i)}{min(k, |C|)} \tag{1}$$

where $m(D)$ is 1 if D is marked (see above), else 0. This means that at each document found in the distributed case, we calculate precision and we average this over $min(k, |C|)$, i.e. over the k documents the user requested or $|C|$, if even the centralised system retrieves less than k documents. In our example, this yields $\frac{1}{3}(\frac{1}{2} + \frac{2}{3} + \frac{3}{5}) = 0.59$. In contrast, if the system retrieves $B = [N, O, P]$, then we get $\frac{1}{3}(\frac{1}{4} + \frac{2}{5} + \frac{3}{6}) = 0.38$

Note that if we use a reference corpus for weight estimation as proposed in [WB05], then rankings are global and hence the scores of documents within A, B, and C will not differ. The measure then tells us how high the best k documents that the distributed search finds are ranked – on average – by the centralised search engine.

5 Parameterising the system

The above evaluation framework leaves a number of parameters free, which need to be chosen in order to make it operational:

- *Which corpus to use*: see [Hei06] for a comparison of freely available corpora.

- *Details of document distribution*: as mentioned above, there are two main possibilities for choosing the documents of a peer: either assigning only the documents the author has written himself, or additionally assigning those that are referenced in these papers. The advantage of the second approach is that it is maybe more realistic to assume that people store more documents from their field of expertise than their own. On the other hand, with the proposed query generation mechansim, this would mean that peers search for a subset of their shared documents.

- *Query generation*: the exact way to form queries was not detailed above because there may be different views on how to realistically choose a query's length and the exact key terms. For example, a librarian is likely to phrase longer and more detailed queries than a casual internet user. A simple possibility would be to use the titles of referenced documents as queries. Alternatively, one can choose the most relevant keywords from the entire document, the distribution of query lengths being

determined by examining e.g. query logs of a web search engine. In general, the query length distribution also is linked to the retrieval function: Given a boolean retrieval more key terms lead to less retrieved documents, but given a vector space retrieval so more key terms lead to more retrieved documents.

- k: to fix the number of documents that should be retrieved, one might again use an empirical study of user behaviour: how many result documents do users request on average from e.g. a web search engine? This question can be easily answered from query logs and the distribution can be used for varying k in a guided way.
- *Retrieval function*: the exact measure of similarity between queries and documents, but also between other entities such as peer profiles, can be chosen individually for each experiment and varied for comparison.

As a conscious decision, we decided not to include network dynamics into the evaluation framework. However, these may be easily be added at a later stage. These aspects of dynamics include, for example:

- *Churn at several time scales*: short-term churn – i.e. peers joining and leaving the network during the day – or long-term churn – i.e. new peers being introduced into the system or leaving it terminally. The latter may require not to use the full set of author-generated peers in the beginning of a simulation, but let the network grow.
- *Document distribution*: one may take the testbed only as a starting setup for the simulation and model the download behaviour of users.
- *Querying activity*: determine in what time intervals peers should ask which of their queries. The query sets associated to the peers may also change over time according to changes in the document distribution.

Because it leaves a great deal of freedom for modeling different aspects of P2PIR, the framework allows to be used in many contexts, but still guarantees comparable results within the P2PIR community.

6 Conclusions

In this paper, we have presented an evaluation framework, providing a strategy for the construction of a P2PIR testbed, that consists of a realistic prescription for content distribution, query generation and distribution, and for measuring a retrieval system's effectiveness in ranking documents. We rely only on a freely availbale relational corpus and there is no need for hand-crafted queries or human relevance judgments. The framework offers a great amount of flexibility via free parameters but still ensures comparability of results.

In the future, the framework will be used in our research, in order to evaluate the quality of the peer description, query routing and neighbour selection strategies that have been described above in Section 2.

References

[AWMM06] R. Akavipat, L.-S. Wu, F. Menczer, and A.G. Maguitman. Emerging semantic communities in peer web search. In *P2PIR '06: Proceedings of the international workshop on Information retrieval in peer-to-peer networks*, pages 1–8, 2006.

[BMR03] M. Bawa, G. S. Manku, and P. Raghavan. SETS: search enhanced by topic segmentation. In *Proc. of SIGIR '03*, pages 306–313, 2003.

[CGM02a] A. Crespo and H. Garcia-Molina. Semantic Overlay Networks for P2P Systems. Technical report, Computer Science Department, Stanford University, 2002.

[CGM02b] Arturo Crespo and Hector Garcia-Molina. Semantic Overlay Networks for P2P Systems, 2002.

[Coo04] B. F. Cooper. A content model for evaluating peer-to-peer searching techniques. In *ACM/IFIP/USENIX 5th International Middleware Conference*, Toronto, 2004.

[GBL98] C. Lee Giles, Kurt Bollacker, and Steve Lawrence. CiteSeer: An Automatic Citation Indexing System. *Proc. 3rd ACM Conf. on Digital Libraries*, pages 89–98, June 23–26 1998.

[Gnu] Gnutella. www.gnutella.com, last visited 04/03/2007.

[Hei06] Gregor Heinrich. Free text corpora and their application to community retrieval evaluation. Technical report, Arbylon & University of Leipzig, http://www.arbylon.net/publications/corpora.pdf, 2006.

[HTW06] G. Heinrich, S. Teresniak, and H. F. Witschel. Entwicklung von Testkollektionen für P2P Information Retrieval. In Christian Hochberger and Rüdiger Liskowsky, editors, *Workshop P2P Information Retrieval, 36. Jahrestagung der Gesellschaft für Informatik*, volume P-93 of *Lecture Notes in Computer Science (LNI)*, pages 20–27, 2006.

[KJPD05] I. A. Klampanos, J. M. Jose, V. Poznanski, and P. Dickman. A Suite of Testbeds for the Realistic Evaluation of Peer-to-Peer Information Retrieval Systems. In *27th European Conference on IR Research, ECIR 2005*, pages 38–51, 2005.

[Kle00] J. Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.

[Kro02] A. Z. Kronfol. FASD: A Fault-tolerant, Adaptive, Scalable, Distributed Search Engine, 2002.

[LC03] J. Lu and J. Callan. Content-based retrieval in hybrid peer-to-peer networks. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 199–206, 2003.

[LLS04] M. Li, W.-C. Lee, and A. Sivasubramaniam. Semantic Small World: An Overlay Network for Peer-to-Peer Search. In *Proceedings of the International Conference on Network Protocols (ICNP)*, pages 228–238, 2004.

[MNRS00] Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the Construction of Internet Portals with Machine Learning. *Information Retrieval Journal*, 3:127–163, 2000. www.research.whizbang.com/data.

[NBMW06] T. Neumann, M. Bender, S. Michel, and G. Weikum. A Reproducible Benchmark for P2P Retrieval. In *Proc. of the First International Workshop on Performance and Evaluation of Data Management Systems, ExpDB 2006 at ACM SIGMOD*, pages 1–8, 2006.

[RFH+01] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker. A Scalable Content Addressable Network. In *Proceedings of the ACM SIGCOMM*, 2001.

[Sch05] C. Schmitz. Self-Organization of a Small World by Topic. In *Proceedings of 1st International Workshop on Peer-to-Peer Knowledge Management*, 2005.

[SCK03] M. Schlosser, T. Condie, and S. Kamvar. Simulating A File-Sharing P2P Network. In *Proc. of 1st Workshop on Semantics in Grid and P2P Networks*, 2003.

[SMK+01] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In *Proceedings of the 2001 ACM SIGCOMM Conference*, pages 149–160, 2001.

[SMZ03] K. Sripanidkulchai, B. Maggs, and H. Zhang. Efficient Content Location Using Interest-Based Locality in Peer-to-Peer Systems. 2003.

[TRE] Text REtrieval Conference. http://trec.nist.gov/.

[VH06] E.M. Voorhees and D.K. Harman. *TREC – Experiment and Evaluation in Infromation Retrieval*. The MIT press, Cambridge, Massachusetts, 2006.

[WB05] H.F. Witschel and T. Böhme. Evaluating Profiling and Query Expansion Methods for P2P Information Retrieval. In *Proc. of the 2005 ACM Workshop on Information Retrieval in Peer-to-Peer Networks (P2PIR)*, 2005.

[Wit05] H.F. Witschel. Content-oriented Topology Restructuring for Search in P2P Networks. Technical report, University of Leipzig, http://wortschatz.uni-leipzig.de/~fwitschel/papers/simulation.pdf, 2005.

[ZGG02] H. Zhang, A. Goel, and R. Govindan. Using the Small-World Model to Improve Freenet Performance. In *Proc. of IEEE Infocom, 2002. 14*, 2002.

[ZKJ01] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. Tapestry: An infrastructure for fault-resilient wide-area location and routing. Technical Report UCB//CSD-01- 1141, U. C. Berkeley, 2001.

Chapter 5: Community Structure Building

Contributions to 10th I2CS 2010, Bangkok, Thailand

Sunantha Sodsee, Maytiyanin Komkhao, Zhong Li, Wolfgang A. Halang, Phayung Meesad
On the Convergence of a Leader-Following Discrete-Time Consensus Protocol

Panchalee Sukjit, Herwig Unger
HexaGrowth: a new Grid Generation with a Local Algorithm

Daniel Berg, Herwig Unger
n-Dimensional Border Growth

Contributions to 7th I2CS 2007, Munich, Germany

Vincent Levorato, Marc Bui
Modeling the Complex Dynamics of Distributed Communities of the Web with Pretopology

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Alessandro E. P. Villa, Javier Iglesias, Solange Ghernaouti-Helie
OpenAdap.net: a Community-Based Sharing System

On the Convergence of a Leader-Following Discrete-Time Consensus Protocol

Sunantha Sodsee[1,2], Maytiyanin Komkhao[1], Zhong Li[1]
Wolfgang A. Halang[1], Phayung Meesad[2]

Faculty of Mathematics and Computer Science
Fernuniversität in Hagen, Germany[1]
Faculty of Information Technology
King Mongkut's University of Technology North Bangkok, Thailand[2]
Email: sunantha.sodsee@fernuni-hagen.de

Abstract: A discrete-time consensus protocol with the leader-following control is proposed, which is for agents to follow either static or time-varying reference state of the leader. An agent updates its state based on only the current information available from its neighbors. To estimate the consensus convergence, the eigenvalues are investigated by the Gershgorin's theorem, and simulations will be given to show the effectiveness of the proposed consensus protocol.

Keywords: Discrete-time consensus, Leader-following, Gershgorin's theorem, Static and time-varying state.

1 Introduction

Consensus problem has a long history in computer science, particularly in distributed computing [Ly97][De74]. However, in the context of networked multi-agent systems, consensus problems is related to the group coordination, which is for a group of agents to make a decision or to reach an agreement regarding a certain quantity of interest that depends on the states of all agents. The consensus problem is also related to the cooperative control problem, in which the global objective is to reach a consensus of all agents [SFS07]. Under a certain topology of the multi-agent systems, the consensus problem is to design the consensus algorithm or protocol, or say, the interaction rule, that specifies the information exchange between the agents on the network, so as for the agents to reach a consensus [FM04][OFM07][LC08][RBA05].

In 1995, Vicsek first proposed a discrete-time consensus model, and simulated the phase transitions in a self-driven particles [Vi95]. Then, Jadbabie *et al*. provided a theoretical explanation for Vicsek's observation by proposing the nearest neighbor rules to coordinate the group behavior of mobile autonomous agents [JLM03]. Fax and Murray [FM04] considered the information flow and cooperative control for unmanned vehicles. Ren and Beard [RB05] studied the consensus seeking in multi-agent systems under dynamically changing interaction topologies. Olfati-Saber and Murray proposed the consensus protocols for the networks of dynamic agents [OM04][OM03]. So far, most

consensus protocols just guarantee the agents to converge to an emergent consensus value, but not to reach a specific value [FXD08][KB06][XWW06][RB08]. For this purpose, leader-following consensus models have been proposed [Re07][CRL09][SK08] [HH07][Li08], where an external control signal acting as the leader who is connected to agents is used to drive the group behavior of the system, and a follower updates its state based on the previous or current information available from its neighbors and the leader, so as for the agents/followers to reach a consensus associated with the leader's state.

The leader-following continuous-time consensus model was proposed in the form of [RB08]

$$\dot{\xi}_i = -\sum_j a_{ij}(\xi_i - \xi_j) - a_{i(n+1)}(\xi_i - \xi^r), \tag{1}$$

where ξ_i represents the state of follower i (i=1,2,…,n), the leader is labelled as n+1, ξ^r is the reference state acting as the state of the leader, and $(a_{ij})_{n\times n}$ is the adjacency coefficient matrix and $a_{i(n+1)} \equiv 1$ for all i if each follower is connected with the leader. With this model, the followers can track the static state of the leader but fail with the time-varying state of the leader. The corresponding discrete-time consensus model was then proposed and is called P-like discrete-time consensus algorithm [CRL09], which is described as following,

$$\xi_i[k+1] = \xi_i[k] - T\sum_j a_{ij}(\xi_i[k] - \xi_j[k]) - Ta_{i(n+1)}(\xi_i[k] - \xi^r[k]), \tag{2}$$

where T is the sampling period, $\xi_i[k]$ and $\xi^r[k]$ are the state of follower i and the state of the leader at time step k, respectively.

To determine the stability properties of the consensus protocol, the location of the eigenvalues of the network is concerned. Which the second smallest eigenvalue of Laplacian matrix [OFM07] or the second largest eigenvalue of Perron matrix [OFM07] of the network topology is a measure of performance or speed of consensus algorithms. As well the Gershgorin's theorem is an one way to estimate the eigenvalues for finding the trace of the network matrix [Br07].

Normally, the reference state of the target can be static or changed dynamically (time-varying). Therefore, the question of how to develop a consensus protocol so as for the agents to follow both static and time-varying state of the leader needs to be addressed; and how to analyze the convegence speed of the proposed protocol are the main concern of this paper. A leader-following discrete-time consensus protocol is first proposed in this paper, with which the agents can follow both the static and time-varying state of the leader and the follower updates its state by using only the current exchanged information from its neighbors and the leader; As well as, the Gershgorin's theorem is applied to estimate the eigenvalues on this proposed consensus protocol.

The rest of this paper is organized as follows. Background and preliminaries are introduced in Sec.II. Sec.III presents the proposed model and the numerical simulation result is presented in Sec.IV. Finally, the work is concluded in Sec.V.

2 Background and Preliminaries

An interactive topology of network of agents is represented by a directed graph $G_n = (V_n, E_n)$, where V_n is the set of vertices v_i, i={1,2,…,n},and E_n the set of edges $\rho_{ij} = (v_i, v_j), i, j = \{1,2,..., n\}$. Herein, the edge from v_j to v_i donates that v_i receives information from v_j. The adjacency matrix $A_n = [a_{ij}] \in R^{n\times n}$ associated with G_n is defined as $a_{ij} = \begin{cases} 1, & \text{if } v_i, v_j \in E_n, \\ 0, & \text{Otherwise.} \end{cases}$. The directed G_n is called balanced if $\sum_{i\neq j} a_{ij} = \sum_{i\neq j} a_{ji}$ for all $i, j \in V_n$. $N_i\{v_j \,|\, a_{ij} \neq 0 \text{ and } i \neq j\}$ is the set of neighbors of v_i, and $|N_i|$ the number of neighbors of v_i or in-degree of node $i, \deg_{in}(v_i) = \sum_j a_{ij}$. The degree matrix of digraph G_n is a diagonal matrix $D_n = [d_{ij}]$ where $d_{ii} = \sum_{i\neq j} a_{ij}$ or $\deg_{in}(v_i)$. Further, the graph Laplacian L_n is defined as $L_n = D_n - A_n$. It is It is obvious that all row-sums of L_n are zero, hence, L_n always has a zero eigenvalue λ_1=0.

The original discrete-time consensus protocol is describe by [OFM07]

$$x_i[k+1] = x_i[k] - \varepsilon \sum_{j\in N_i} a_{ij}(x_i[k] - x_j[k]), \tag{3}$$

where $x_i[k]$ denotes the state of agent i at time step k, and $0 < \varepsilon \leq 1/\Delta$ is the sampling period, in which Δ is the maximum degree of agents.

Further, Eq.3 can be recast as $x[k+1]=P_n x[k]$, where $P_n=I_n- \varepsilon L_n$ is the Perron matrix of graph G_n, and I_n is the identity matrix. Assume that P_n is a primitive-nonnegative matrix, and denote w as the nonnegative left eigenvector associated with eigenvalue 1, *i.e.*, $w^{\mathrm{T}} P_n = w^{\mathrm{T}}$.

A group of agents is said to reach a global consensus if $x_j[k] = x_i[k]$ for each pair (i,j),~i,j =1, 2,…, n and $i \neq j$, and the common agreement value of all agents is called the group decision value, denoted by $\alpha = \sum_i w_i x_i[0]$, where w_i is the left eigenvalue associated with eigenvalue $\lambda_i (i = 1,2,..., n)$, satisfying $\sum_i w_i = 1$, and $x_i[0]$ is the initial state [OFM07][FXD08][KB06]. For a balanced digraph, one has $w = (\frac{1}{n})\,\mathbf{1}$, $\alpha = (\frac{1}{n})\,\mathbf{1}^{\mathrm{T}} x_i[0]$, and

$$\lim_{t\longrightarrow\infty} \alpha(t) = \frac{\sum_i x_i[0]}{n}, \tag{4}$$

where $t = k\varepsilon$ [OFM07][KB06].

3 Leader-following Control

In this section, the leader-following discrete-time consensus algorithms with the time-varying leader state is concerned.

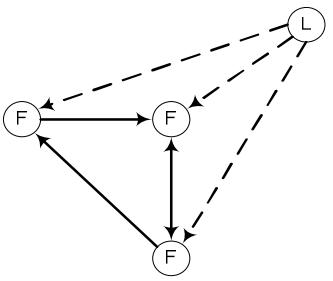


Figure 1: Leader-follower interaction topology

A leader is added to the system, and is connected to all followers. Denote the followers as ${x_i}^f, i = \{1,2,..,n\}$ and the leader as ${x_{n+1}}^l$ or x^l. The multi-agent system can be represented as the directed graph or digraph $G_{n+1} = (V_{n+1}, E_{n+1})$, where $V_{n+1} = \{v_1, v_2, ..., v_{n+1}\}$ is the set of nodes and $E_{n+1} \subseteq V_{n+1} \times V_{n+1}$ is the set of edges.

Here, only the leader sends the information to the followers. Fig.1 depicts the interaction topology between the leader and the followers. The dash line connecting between the leader and follower means that the leader sends the information to the follower. In additional, the thick line presents the communication between followers. Because the reference state of leader can be static or time-varying, a discrete-time consensus protocol is given as

$$x_i^f[k+1] = x_i^f[k] - a_{i(n+1)}(x_i^f[k] - x^l[k]) - \varepsilon \sum_j a_{ij}(x_i^f[k] - x_j^f[k]), \qquad (5)$$

Let $\varepsilon = \frac{1}{n+1}$. It is obvious that $a_{i(n+1)} \equiv 1$, $i = \{1,2,...,n\}$. The consensus is said to be reached if $x_i^f[k] = x^l$ as $k \longrightarrow \infty$. (5) can be recast in the matrix form

$$x^f[k+1] = (P - B)x^f[k] + bx^l[k], \qquad (6)$$

where the matrix $P = I - \varepsilon L$ is the Perron matrix, $B = diag[a_{1(n+1)}, a_{2(n+1)}, ..., a_{n(n+1)}]$, and the vector $b = [a_{1(n+1)}, a_{2(n+1)}, ...a_{n(n+1)}]^{\mathrm{T}}$. (6) can be further written in the compact form

$$x^f[k+1] = Fx^f[k] + bx^l[k], \qquad (7)$$

where *F=P-B*.

An irreducible stochastic matrix *P* is primitive if it has one singular eigenvalue with maximum modulus. *P* is a row stochastic non-negative matrix; row-sums of *P* are 1, hence *P* always has a one eigenvalue $\lambda_1 = 1$. The convergence analysis of the discrete-

time consensus algorithm relies on the second largest eigenvalue of P. It reachs with a speed that is faster or equal to $\lambda_2 = 1 - \varepsilon\mu_2(L)$, where $\mu_2(L)$ is the second smallest eigenvalue of L and λ_2 is the second largest eigenvalue of P [OFM07]. The set of eigenvalues of P can be ordered sequentally as $1 = \lambda_1 > \lambda_2 \geq \ldots \geq \lambda_{n+1}$. According to the Gershgorin's theorem [Br07], all eigenvalues of the network matrix (P) are located on a unit circle. Each eigenvalue lies within the Gershgorin disc centered at (r_i,0), where r_i is the P_{ii}. The radius of each disc is calculated by $d_i = \sum_{i \neq j} | P_{ij} |$.

Theorem 1

All eigenvalues of P must lie in a unit circle centered at (0,0).

Proof

A leader is connected to all followers in the system, and only the leader sends the information to the followers (there is no feedback information from the followers). The eigenvalues of followers are $0 < \lambda_i < 1$ and these eigenvalues lie in the Gershgorin disc at (r_i,0), where $0 < r_i < 1$ and the radius $0 < d_i < 1$. On the other hand, the eigenvalue of the leader is always one, it is centered at (r_{n+1},0) , where r_{n+1}=1 then the radius d_{n+1} is zero.

4 Numerical Simulations

In this section, we present an example, the unbalanced digraph, to show the effectiveness of the proposed work.

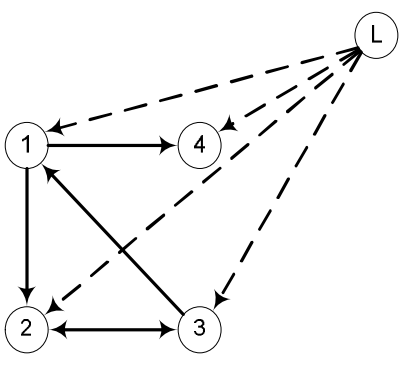


Figure 2: Example of unbalanced digraph

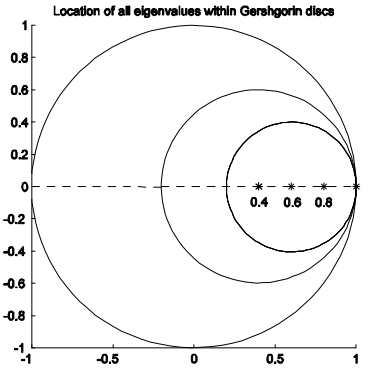


Figure 3: Location of eigenvalues

Fig. 2 depicts the example of unbalanced digraph [RB08], there is one leader connected to four followers. This interaction graph corresponds to the Parron matrix as follow

$$P = \frac{1}{5}\begin{bmatrix} 3 & 0 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 & 1 \\ 0 & 1 & 3 & 0 & 1 \\ 1 & 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 0 & 5 \end{bmatrix},$$

The eigenvector λ of P is $\lambda = [0.6 \quad 0.8 \quad 0.4 \quad 0.4 \quad 1]^T$. Fig.3 shows the Gershgorin discs of all eigenvalues. The centers of each disc are (0.6,0), (0.4,0), and (0,0). The bigest disc is a unit circle.

To estimate the location of eigenvalues, followers' eigenvalues are 0.6, 0.8, 0.4, and 0.4. They are located in Gershgorin discs centered at (0.6,0), (0.4,0), (0.6,0), and (0.6,0). Their discs' radius are 0.4, 0.6, 0.4, and 0.4 respectively. On the other hand, the eigenvalue of the leader is one, it is centered at (1,0). Thus, the location of all eigenvalues are within the unit circle that centered at (0,0), as well as they are located within every Gershgorin disc.

To show the effectiveness of the proposed protocol, the initial state of followers is $x^f[0] = [-0.5 \quad 0 \quad 0.5 \quad 1.5]^T$ and the reference state of the leader are static with $x^l = 1$ and time-varying with $x^l = \cos(t)$. Presenting the global consensus, Fig. 4 and 5 are depicted the simulation results that the followers can reach the consensus following the static state of the leader (Fig.4) and the time-varying state of the leader (Fig.5), respectively.

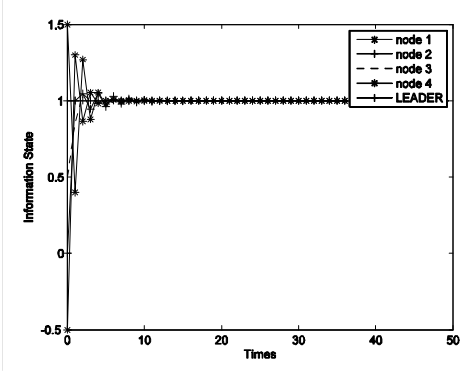


Figure 4: Static state of leader

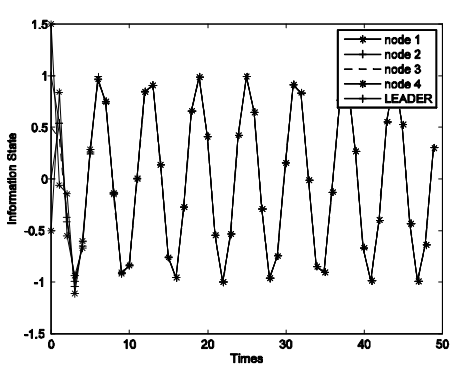


Figure 5: Time-varying state of leader

5 Conclusions

In this paper, the discrete-time consensus protocol based on a leader-following behavior of multi-agent system has been proposed, suitable for both the static and time-varying states of the leader. It uses only the current state of agents to identify the updated state of

followers, which reduces the memory storage. The simulation result has shown that this proposed protocol is efficient. The states of followers can converge to the consensus following the leader's state, which their eigenvalues location analyzed by the Gershgorin's theorem.

References

[Br07] Brakken-Thal, S.: Gershgorin's theorem for estimating eigenvalues, http://buzzard.ups.edu/courses/2007spring/projects/brakkenthal-paper.pdf, 2007.

[CRL09] Cao, Y.; Ren, W.; Li, Y.: Distributed discrete-time coordinated tracking with a time-varying reference state and limited communication, Automatica, vol. 45, pp. 1299-1305, 2009.

[De74] DeGroot, H. M.: Reaching a consensus, Journal of American Statistical Association, vol. 69, no. 345, pp. 118-121, 1974.

[FM04] Fax, A. J.; Murray, M. R.: Information flow and cooperative control of vehicle formations, IEEE Trans. Autom. Control, vol. 49, pp. 1465-1476, 2004.

[FXD08] Fu-Xiao, T.; Xin-Ping, G.; De-Rong, L.: Consensus protocol for multi-agent continuous systems, Chinese Phys. B, vol.17, no. 10, October, 2008.

[HH07] Hu, J.; Hong, Y.: Leader-follower coordination of multi-agent systems with coupling time delays, Physica A: Statistical Mechanics and its Applications, vol. 374, iss. 2, pp. 853-863, 2007.

[JLM03] Jadbabaie, A.; Lin, J.; Morse, S. A.: Coordination of groups of mobile autonomous agents using nearest neighbor rules, IEEE Trans. Autom. Control, vol. 48, no. 6, pp. 988-1001, 2003.

[KB06] Kingston, B. D.; Beard, W. R.: Discrete-time average-consensus under switching network topologies, in Proc. American Control Conf., 14-16 June 2006.

[LC08] Liu, X.; Chen, T.: Consensus problems in networks of agents under nonlinear protocols with directed interaction topology, http://arxiv.org/abs/0804.3628, 2008.

[Li08] Liu, B.; Chu, T.; Wang, L.; Xie, G.: Controllability of a leader-follower dynamics network with switching topology, IEEE Trans. Autom. Control, vol. 53, no. 4, pp. 1009-1013, 2008.

[Ly97] Lynch, A. N.: Distributed algorithms, Morgan Kaufmann Publishers, Inc., 1997.

[OM03] Olfati-Saber, R.; Murray, M. R.: Consensus protocols for networks of dynamic agents, in Proc. Am. Control Conf., pp. 951-956, 2003.

[OM04] Olfati-Saber, R.; Murray, M. R.: Consensus problems in networks of agents with switching topology and time-delays, IEEE Trans. Autom. Control, vol. 49, pp. 1520-1533, 2004.

[OFM07] Olfati-Saber, R.; Fax, A. J.; Murray, M. R.: Consensus and cooperation in networked multi-agent systems, in Proc. IEEE, vol. 95, pp. 215-233, 2007.

[RB05] Ren, W.; Beard, W. R.: Consensus seeking in multi-agent systems under dynamically changing interaction topologies, IEEE Trans. Autom. Control, vol. 50, pp. 655-661, 2005.

[RBA05] Ren, W.; Beard, W. R.; Atkins, M. E.: A survey of consensus problems in multi-agent coordination, in Proc. American Control Conf., 8-10 June 2005.

[RB08] Ren, W.; Beard, W. R.: Distributed consensus in multi-vehicle cooperative control theory and applications, Springer, 2008.

[Re07] Ren, W.: Multi-vehicle consensus with a time-varying reference state, Systems & Control Letters, vol. 56, pp. 474-483, 2007.

[SFS07] Schmalz, M.; Fujita, M.; Sawodny, O.: Directed gossip algorithms, consensus problems, and stability effects of noise trading, in Proc. MASHS2007, 2007.

[SK08] Semsar-Kazerooni, E.; Khorasani, K.: Optimal concensus algorithms for cooperative team of agents subject to partial information, Automatica, vol. 44, pp. 2766-2777, 2008.

[Vi95] Vicsek, T.; Czirok, A.; Jocob, E.; Cohen, I.; Schochet, O.: Novel type of phase transitions in a system of self-driven particles, Physical Review Letters, vol. 75, pp. 1226-1229, 1995.

[XWW06] Xiao, F.; Wang, L.; Wang, A.: Consensus problems in discrete-time multiagent systems with fixed topology, J. Math. Anal. Appl., vol. 332, pp. 587-598, 2006.

HexaGrowth: a new Grid Generation with a Local Algorithm

Panchalee Sukjit, Herwig Unger

Communication Network
Faculty of Mathematics and Computer Science
Universitätsstraße 27 – PRG
58084 Hagen
{panchalee.sukjit | herwig.unger}@fernuni-hagen.de

Abstract: In various previous works about pattern formation methods on top of P2P networks have been discussed. Most of structures had done by the rectangular grids building methods, which could later be used to improve the efficiency of routing and data search algorithms. In this contribution, another new, local working algorithm is introduced, which allows the formation of other than rectangular grids. Mostly, the generation and use of hexagonal and triangular grids is introduced. Apart from the formal description of the algorithm, its performance is evaluated using the P2PNetSim tool. Finally, it has been observed, that different structures may have distinct advantages over the different purposes/applications. In order to cover such different requirements, structure transformations may be a useful tool. Therefore, some structure transformation methods will be discussed in the last section of this paper.

Keyword: P2P Systems, Grid, Hexagonal, Structured Overlay Network, Local Algorithm, Decentralized Systems

1 Introduction

The appearance of structures and special pattern in various kinds of systems has fascinated and attracted not only computer scientists. In [KVG02] and other publications, the authors give explanations, how the respective structures may appear and why and how animals may get benefit from them. Bird flocking, fish schooling, ant street and bee foraging have been - for instance -discovered and investigated by scientists of different disciplines [BM09, DB07, GGT07, Mo09] and even in computer science those systems give ideas to create several algorithms [KA09, LAK03]. However, also artificially created, simple examples from computer science have been found, like the definition of "*The game of life*" from J. Conway [Ga70] approximately 40 years ago. The quite simple definition was the background for the definition and research about cellular automata.
Finally it has been investigated that even in large distributed systems, namely P2P-systems, structures and special properties of graphs play an important role [CCR04]. Furthermore, [Kl00] and [LU09] have investigated that grids structured P2P systems (see figure 1) could proof that the performance of search and routing procedures can be

significantly improved, if the anarchically grown small world structures are overlayed with regular, mostly rectangular grid structures. From these considerations the question came up, whether other grid types like “hexagonal” grids may have similar, useful properties.

Physical Networks P2P Networks Overlay Networks

Figure 1: A system hierarchy using P2P overlay networks

In chemistry [Gl08, YYL09], biology [Wv04] and even in computer science and engineering works hexagons seems to have a special importance (see figure 2). In addition, some publications already give construction mechanisms for hexagon grids under very special conditions [LAK03]. Consequently, the question arises, if such hexagonal grids could also be built up by local working, distributed algorithms on top of P2P systems and may give advantages, when used (figure 1).

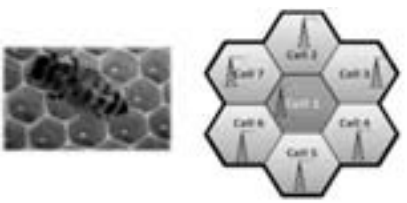


Figure 2: Hexagonal grid structure in nature and technology.
(honey comb and mobile phone cell structures)

In this contribution, it is intended to discuss this question in more detailed. Therefore, section 2 presents the formal definition of the hexagon grid building method. After that, following this; section 3 describes its performance evaluation using a simulation and section 4 presents possibilities for grid transformations. Section 5 summarizes and concludes the paper.

2 HexaGrowth-a new Grid-Building Method

2.1 Requirements

As discussed in previous publications as [Be07, BSU09a, BSU09b, SBU09 and SU09], for rectangular grids also the hexagonal grid structures can be generated from a single, initial cell, without any global knowledge, predefined coordinate system or cell universe (as it is used for instance in cellular finite state machines). Nevertheless, it is necessary to avoid the appearance of holes or overlapping parts during the whole growth process.

Therefore, all new nodes must be properly connected to all their already existing neighbors and it must be ensured that later added neighbors can connected to them with local information (i.e. data from their direct neighbourhood nodes), only. Consequently, all grid building methods must meet the following requirements for a structure building method, which have been formulated in [BSU09b]:

- The structure must be built fast and with non-complex algorithms.
- The structure does not appear in a previously fixed coordinate or cell space system.
- It must be easy to repair in case any changes in the network appear.
- The algorithm is running locally on each peer and only can use the information available on this peer and eventually on its neighborhood peers (since global information is not available).
- The generated overhead shall be minimal and the achieved efficiency maximal.

Different to rectangular grids, hexagonal grids offer two principle possibilities for a building process (see figure 3).

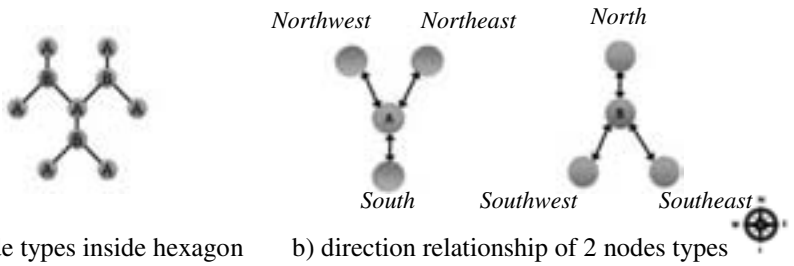
a) 6 peers as corner point of hexagon b) a center peer represent a full hexagon

Figure 3: Building hexagon based grids

In a first approach, every corner point of a hexagon may be represented by a peer, while two neighboring hexagons always share two corner peers. Another possibility is obtained, if a peer represents a whole hexagon. In this case, this peer must be connected to all its neighbors, what results in 6 neighborhood connections. As it is easy to be seen, the respective generation method would result in the setup of a grid consisting of triangles while the hexagon structure is only implicitly present. Therefore, the first approach is chosen for the algorithm design in the following section.

2.2 The Algorithm

If the basic form of a hexagonal grid pattern is considered, then two different node types can be found. Figure 4 shows these two types of node, the so-called *"A"* and *"B"* nodes distinguished from the direction of the connections of the edges. Node *"A"* has three neighbors, in upper direction two nodes and one neighbor in the lower direction. Conversely, node *"B"* (having also 3 neighbors as *"A")* has in the upper direction one and two neighbors in the lower direction (see figure 4a).



a) 2 node types inside hexagon b) direction relationship of 2 nodes types

Figure 4: 2 node types in the hexagonal grid and relationship within the structure

As the consequence of that, in the below grid building method two types of nodes (*"A"* and *"B"*) will be used (figure 4b). While nodes of type *"A"* allow two upper and one lower neighbor, node *"B"* allow neighbors in exactly the opposite manner: 2 lower and only 1 upper neighbor node. In order to allow a concatenation of the nodes in the right manner, the respective links get names, derived from the compass direction in which the respective links point. It is clear that in the algorithm

- Nodes *"A"* have 3 neighbors of *"B"* and also "*B"* have 3 neighbors of *" A"*
- A *South* link of node *"A"* link can only be connected with a *North* link of node *"B"*
- Only *Northeast* and *Southwest* as well as *Northwest* and *Southeast* links can be connected with each other.

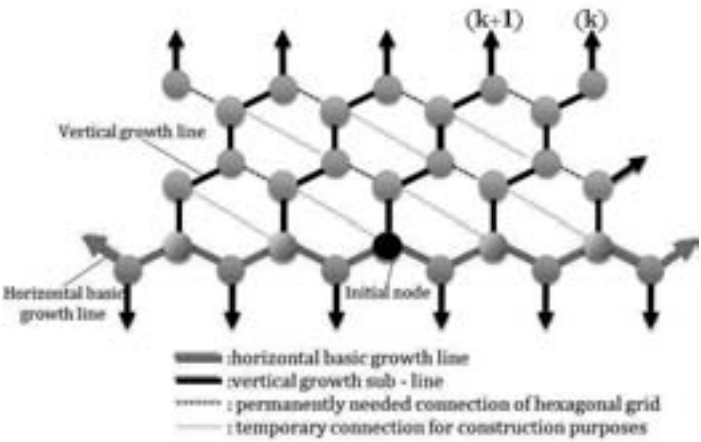


Figure 5: Overview on the construction mechanisms

Figure 5 shows the main principle of the structure building method. In an initial step, the so called *horizontal*, basic growth line is created. This building process starts from the initial cell and includes a growth to both sides. Alternating, type *"A"* and type *"B"* cells are connected using their southeast-northwest and northeast-southwest connections. As a result of this building process, the red basic line is created, whose cells later give also the basic orientation for all growth processes. Each of the cells/nodes of the basic line start active after its full connected to two neighbours along the horizontal basic line like *"angle"* the so called vertical growth processes. In this vertical growth process

- Cells of type *"A"* create vertical growth lines into the north direction
- Cells of type *"B"* create a vertical growth in south direction
- In the north direction, the concatenation north-south followed by northeast-southwest is used in a repeated manner while in the south direction the same is done with the sequence south-north and southwest-northeast.

The horizontal and vertical growth processes work independently and parallel from each other. The more nodes are included in the structure, the higher the parallelity will become, since more nodes will try to add other nodes to the already existing structure.

In a final process the vertical growth lines must be connected as it can be seen in Fig. 5. In order to keep the algorithm local (i.e. use only information, which the node itself or its direct neighbours have) a temporary connection is used, which will be deleted later. Figure 6 shows the interaction diagram for the respective communication needed among the different nodes. For simplicity and due to the existing symmetry, only the interconnection in one direction is considered.

We number the positions along the horizontal growth lines by *"k"* and the respective numbers along the vertical growth lines by *"r"*. Along the basic horizontal line, the parameter *"r"* is equal to zero.

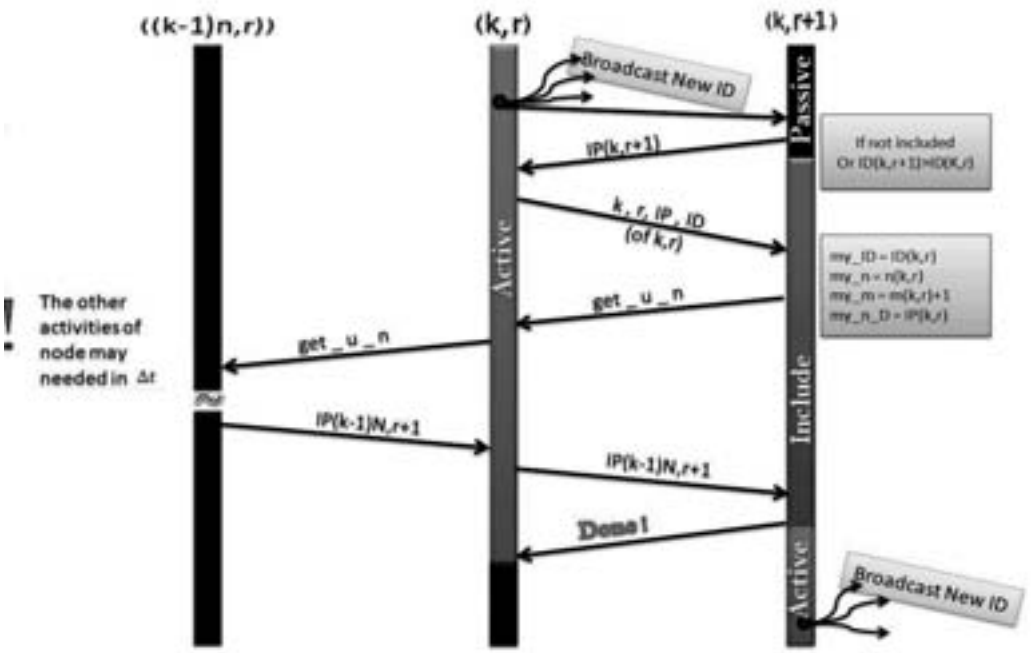


Figure 6: Generation of the temporary and permanent interconnections between the *k-th* and *(k-1)*-th vertical growth line

At the beginning, it is assumed that the *(k-1)-th* and *k-th* cells from the horizontal basic line know each other and the *(k-1)* cell has communicated the address of its newly generated cell from the vertical growth line to the neighbour *k.* Neighbor *k* may submit this information to its "child" in the vertical direction *(k,1).* So the node *(k,1)* may connect to *(k-1,1)* and build up the first temporary connection which is later needed to communicate the position of *(k-1,2)* to *(k,1)* and result in a building of the first permanent connection finishing the first hexagon in this part.

In general, the algorithm generates alternatingly a temporary and a permanently connection and finally builds the hexagonal structured. The steps of this algorithm are the following.

- The node on position *(k, r)* will generate the node *(k, r + 1)*. On the left side, the node *((k − 1), r)* can be found (and its address is known to *(k, r)).* It will consequently generate the new node *((k − 1), (r + 1)).*
- From the step before, the nodes *(k, r)* and *((k − 1), r)* are already connected by a horizontal link (In the case of *r = 1*, the horizontal basic line of course fulfills this task as described).
- Due to this connection the node (k, r) may obtain the IP address of the node *((k−1), (r+1))* and communicate this to its new generated node *(k, r + 1).*
- The latter node can now connect its future neighbor (*(k−1), (r+1))* by the use of this IP address and the new connection may be established.
- Alternating temporary and permanent connections are generated and the generation of a permanent connection result in a message to remove the before temporary line.

This way, the network construction is also synchronized among the neighbor nodes, i.e. the network will grow almost equally.

3 Performance Evaluation

3.1 Simulation Setup

To observe how this algorithm performs under real network conditions, P2PNetSim [RU08] – a distributed network simulator - was utilized. P2PNetSim allows large scale network simulations and analyse on cluster computers (up to 2 million peers can be simulated on up to 256 computers). The behaviour of all nodes can be implemented in Java and then be distributed over the nodes of the simulated network. At simulation start-up the peers are interconnected small-world-like in order to simulate the typical physical structure of computers interconnected in the Internet. On top of this structure, an overlay-network is built using the grid-algorithm. In order to explore the behaviour of our grid generation algorithm, any search times for new nodes in the P2P system have not been considered, i.e. a fixed constant time is assumed for that. Thus we avoid any interference in the duration of the search (fast in the beginning, slower when almost all nodes of the P2P community are already assembled in the grid) using grid building algorithm. This is more important, since the parallelity of the growth process is rapidly increasing with the number of nodes included in the grid.

3.2 Results

Using the above described setting, several simulations were executed. The most interesting question is of course, how fast the respective structures are growing, i.e. the number of nodes included into the hexagonal grid over the time. The second important question, how many nodes are able to add new nodes within the structure in each time step. This number of nodes stands in direct relation to the maximal number of parallel growth processes which may exist.

The authors of this contribution have introduced in [BSU09a, BSU09b and SU09] the skeleton-, border growth and inheritance growth method. Therefore figure 7 and 8 shows the respective results in comparison with that one achieved from the previous works.

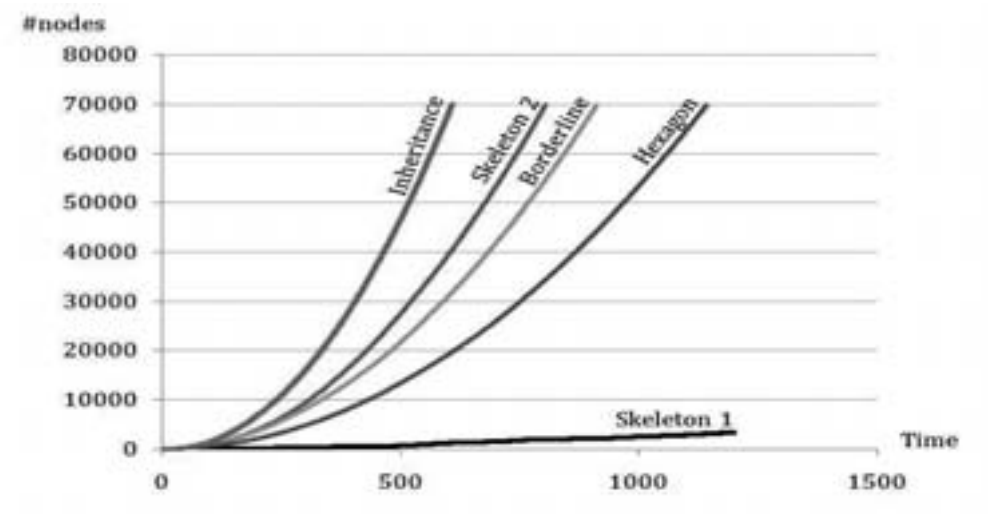


Figure 7: Nodes included in the structure over time

It is clearly be seen, that the new, hexagon growth method shows a qualitative similar, fast behavior as all methods developed before. The structure growth is quite fast, i.e. approximately quadratic with the time (for the simulated 2D grids, figure 7). Since the hexagonal grid is a more complex structure than a rectangular grid, a faster growth than the skeleton and borderline growth method could not be expected. The numbers of growable nodes are to be seen in figure 8.

Although the permanently growing basic horizontal line generates new vertical growth lines in each step, the number of growable nodes is increased following approximately a linear time function, only. This can be explained with the synchronization mechanisms which have been introduced for the construction of the cross connections between the vertical growth lines. Different to rectangular grids, 3D or n-D hexagonal grids seem not to have any applications and have not been considered so far. However, various grid structures may have advantages for different applications and so the question of grid transformation came up and shall be discussed in the next section.

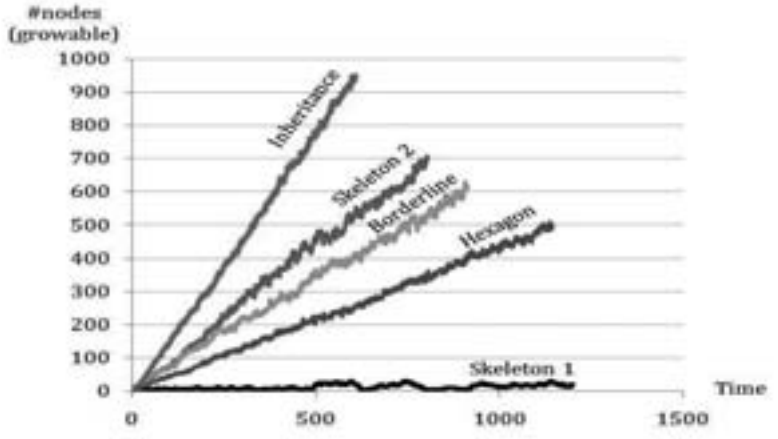
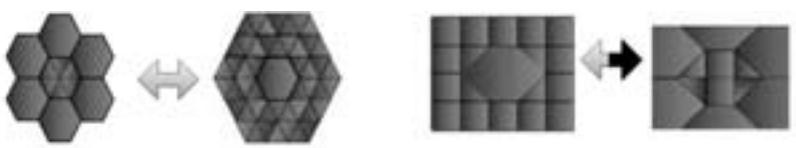


Figure 8: Number of growable nodes at the periphery

4 Grid Transformations

A tessellation of a 2D plane can be done with all grids generated so far, no matter if they are based on rectangles or hexagons.
By simple rules, a transformation from one grid to the other may be possible, if it is started form a single point in the grid, which can be chosen by a simple setup (using the underlaying, implicitly generated coordinate system, where the initial cell has the unique coordinates (0,0)) or voting among the participating.



a) hexagon transforms into triangle b) rectangular transforms into hexagon

Figure 9: Possible grid transformations

The following cases can be of interest:

1. The transformation from triangular grids to hexagonal grids and vice versa can easily be done. This simplicity comes from the fact, that a hexagon may be constructed form 6 triangles by removing the center point/peer of these 6 parts, while in opposite direction just request the addition and right connection of an additional node/peer, which might be easily taken from the periphery of the structure.

2. Several rectangular grids may also be easily transferred into hexagonal grids, as the second part of figure 9 demonstrates. From this picture, it also becomes clear that both directions of the transformation cannot always be implemented in an easy and local executable manner.

Random walkers starting from an initial point might be one good realization for the described transformation and should be considered in later publications.

5 Conclusion and Outlook

In this paper a new structure building algorithm for hexagonal grids have been introduced. This method demonstrates that even more complex structures like hexagons can be built on top of an anarchic grown P2P system with local working algorithms with a good performance.

The contribution extended and completed the series of publications of the authors on 2D local structure building mechanism, which mostly have considered rectangular grids. In addition to the previous works, a performance evaluation for all developed methods has been presented. Finally, a few grid transformation methods have been introduced. Later publications will deal with the application of these structures to improve the efficiency of the routing and search in decentralized P2P systems.

Acknowledgment

The authors would like to thank the internship students *Mr. Thongchai Woraphong* and *Mr. Sirisak Jaroonrojwong* from the King Mongkut's University of Technology North Bangkok, Thailand, for their support in developing simulation software.

References

[Be07] Berg, D.; Coltzau, H.; Sukjit, P.; Unger, H. and Nicolaysen, J.: Passive RFID tag Processing using a P2P architecture, In: Malaysian Software Engineering Conference, Striving for high quality software (MYSEC'07), December 2007.

[BSU09a] Berg, D.; Sukjit, P. and Unger, H.: Borderline growth: a new method to build complete grids with local algorithms, In: World Academy on Science, Engineering and Technology: WCSET 2009 Bali, Indonesia, 2009.

[BSU09b] Berg, D.; Sukjit, P. and Unger, H.: Grid generation in decentralized systems, In: Proceedings of the International Workshop on Nonlinear Dynamics and Synchronization 2009 (INDS), Klagenfurt, Austria, 2009; pp. 95–99.

[BM09] Bonabeau, E. and Meyer,C.: Swarm Intelligence A Whole New Way to Think About Business, Harvard Business Review, 2009.

[CCR04] Castro, M.; Costa, M.; and Rowstron, A.:Peer-to-peer overlays: structured, unstructured, or both, Technical Report MSR-TR-2004-73, Microsoft Report and Cambridge Systems and Networking, 2004.

[DB07] Dorigo , M. and Birattari, M.: Swarm Intelligence" Scholarpedia ,the free peer-reviewed encyclopedia, 2007; p.22437.

[Ga70] Gardner, M.: Mathematical Games: The fantastic combinations of John Conway's new solitaire game of life, In Scientific American 223, 1970; pp 120–123.

[GGT07] Garnier, S.; Gautrais, J. and Theraulaz, G.: The biological principles of swarm intelligence", Swarm Intelligence, Springer Science + Business Media, LLC, 1(1):3–31,2007.

[Gl08] Glettner, B.; Liu, F.; Zeng, X.; Prehm, M.; Baumeister, U.; Ungar, G. and Tschierske, C.: Liquid-crystal engineering with anchor-shaped molecules: Honeycombs with hexagonal and trigonal symmetries formed by polyphilic bent-core molecules, Journal of the Gesellschaft Deutscher Chemiker (Angewandte Chemie:International edition, 2008.

[KA09] Karaboga, D. and Akay, B.: A survey: algorithms simulating bee swarm intelligence, In: Artificial Intelligence Review. Springer Netherlands, 2009.

[Kl00] Kleinberg, J.: The small-world phenomenon: An algorithmic perspective, In: Proceedings of the 32nd ACM Symposium on Theory of Computing, 2000; pp 163–170.

[LU09] Lertsuwanakul, L. and Unger, H.: A Thermal Field Approach in a Mesh Overlay Network, In: 5th National Conf. on Computing and Information Technology (NCCIT09), May 2009; pp 610–615.

[Mo09] Moallem, A.: Using swarm intelligence for distributed job scheduling on the grid, Master of Science thesis of the Department of Computer Science University of Saskatchewan, Saskatoon, Canada, 2009.

[KVG02] K. Parrish, J.; V. Viscido, S. and Grünbaum, D.: Self- organized fish schools,An examination of emergent properties. Biol. Bull, 202,2002; pp 296–305.

[RU08] Rojas Gonz´alez, M. and Unger, H.: A preliminary performance evaluation of P2PNetSim simulator, In: IADIS International Conference WWW Internet, 2008.

[LAK03] L. Stewart, R.; A. Russell, R. and Kleeman, L.: Generating a Honeycomb Structure using Cellular Automata with Applications for Swarm Robotics, Academic Research Forum of Department of Electronic and Computer Systems Engineering, 2003.

[SBU09] Sukjit, P. Berg, D. and Unger, H.: A New, Fully Decentralized Grid Generation Method, Information Technology Journal, 5th National Conf. on Computing and Information Technology (NCCIT09), 2009; pp. 35–39.

[SU09] Sukjit, P. and Unger, H.: Inheritance growth: a biology inspired method to build structures in P2P, In: Journal of the World Academy on Science, Engineering and Technology 59: WCSET 2009 Bali, Indonesia, 2009; pp. 339–343.

[YYL09] Yao, X.; Yao, H.; Li, Y. and Chen, G.: Preparation of honeycomb scaffold with hierarchical porous structures by core-crosslinked corecorona nanoparticles, Journal of Colloid and Interface Science , 2009.

[Wv04] West Virginia Delaware, Maryland New Jersey Pennsylvania and the USA Cooperating, : Basic Bee Biology For Beekeepers, Mid-Atlantic Apicultural Research and Extension Consortium(MAAREC), 2004.

n-Dimensional Border Growth

Daniel Berg, Herwig Unger
Department of Communication Networks,
FernUniversität Hagen, Germany
{daniel.berg|herwig.unger}@fernuni-hagen.de

Abstract: Peer-To-Peer (P2P) networks become more and more present in the consumer area as well as in industrial applications. Especially in the industrial- and the business area, reliable and scalable protocols are needed, that produce low network-overhead and react quickly on any network-changes. In this paper a generalization of the Border-Growth-algorithm is introduced, that improves the network's scalability, its connectivity, and decreases its diameter by providing multiple dimensions, rather than just two of them.

Keywords: P2P, Grid, Scalability, Decentralized Algorithms, Self-Organization

1 Introduction

Decentralized overlay topologies provide characteristics that cannot be achieved with classical client/server architectures. Due to their completely decentralized architectures they can provide wide scalability and fault-tolerance [LCP04, SUL04]. However, maintaining those structures is not trivial. When participants join or leave the network, it has to be ensured that the structure is kept consistent after those operations. This can become a complex task, depending on the complexity of the network's structure [RFH01, CHORD01, STRUCT02].

In [STRUCT02] the Border-Growth-Algorithm was introduced. This algorithm builds a complete, contradiction- and hole-free two-dimensional lattice. Regular lattice-like overlays are very useful for xy-routing based algorithms [RFID07, THERM09, 3DIOS09]. Compared to existing algorithms for overlay structures, like CAN, Chord, Koorde, Pastry or Tapestry the Border-Growth-Algorithm can manage Join- and Leave operations very quickly. As long as the lattice is not broken into two distinct clusters, there are always multiple available paths from any node to another.

This paper aims to improve the scalability of this algorithm by increasing its dimensions from two to n dimensions. It will be showed that this can be achieved without increasing the complexity of the growth-algorithm itself. (The complexity of updating a new node's neighborhood increases linearly with the dimension n.)

Section 2 provides the requirements and a brief description for the 2D-Border-Growth algorithm. In section 3 follows a detailed discussion for the new, n-dimensional version, starting with the motivation in section 3.1 and the modifications made to the new version

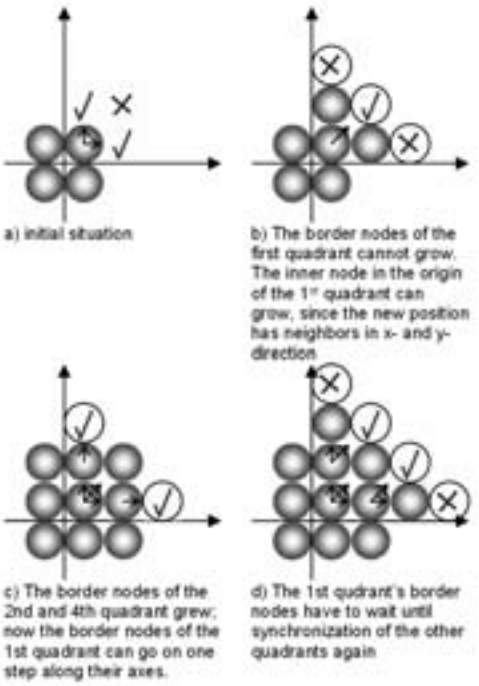


Figure 1: The rules defined by the two-dimensional Border-Growth-algorithm

in section 3.2 Section 3.3 gives the mathematical description, and section 3.4 gives some information about the simulation environment that was used to run the algorithm. Section 4 discusses the simulation results and compares them to the results of the two-dimensional version. Section 5 finally gives a conclusion and a outlook for further work based in the new algorithm.

2 The Border-Growth-Algorithm

2.1 Two-Dimensional Border-Growth

This section gives a brief description for the two-dimensional border-growth-algorithm. The algorithm defines three node types: The **Root Nodes**, that reside at a discrete, virtual coordinate-system's origin in all quadrants, the **Border Nodes** that are positioned along an axis of the virtual coordinate system, and finally the **Inner Nodes** that are all nodes, which are neither Border Nodes nor Root Nodes. The Root Nodes are special cases of the Border Nodes.

Nodes in the two-dimensional lattice have knowledge about their direct neighbors in north-

, west-, south-, and east-direction. To simplify the algorithm, and to increase connectivity, all nodes additionally posses links to their direct neighbors in the diagonal directions northwest, southwest, southeast, and northeast.

The initial state of the structure is illustrated in Fig. 1a): Each quadrant of a virtual, discrete coordinate system has one Root Node in its origin. A Root- or a Border Node is allowed to grow along its axis, if it has a neighbor at the other side of the corresponding axis. This condition is true for the first quadrant's Root Nodes in Fig. 1a and for the first quadrant's Border Nodes in Fig. 1c.

An Inner Node, a Root Node, or a Border Node can grow away from the coordinate system's origin, if the position to which it wants to grow already has neighbors into the x- and y-direction. This condition is fulfilled by the Root Node in Fig. 1b and for two of the first quadrant's Border Nodes in Fig. 1d.

When a new node is accepted as the neighbor of a node in the structure, the new node's neighborhood has to be updated. That means that the surrounding nodes need to update their neighbor-links to accept the new node.

For a detailed mathematical description of this algorithm, refer [BORDER09] and section 3.3 of this paper, which gives a detailed description of the n-dimensional Border-Growth algorithm. Though some modifications had to be made to the new version (see section 3.2), the two-dimensional case is just a special case of the n-dimensional structure.

3 n-Dimensional Border-Growth

3.1 Motivation

Though it is easy to replace a failed node within the structure by another (new or existing) node, such a structure is only able to grow at its borders, which is one reason for why join-operations are very simple and quick. However, growth-scalability in terms of the ratio of the number of all nodes and the number of nodes that can grow, decreases with growing structure size.

In order to lower this effect, the algorithm was modified in that way, that it grows into an arbitrary number of dimensions. Increasing the dimensions of the lattice leads to more advantages: The connectivity of the structure increases, any (inner) node is connected to 2^n other nodes, rather than just to $2^2 = 4$ nodes. Therefore there are more possible paths between any two nodes. Another effect implied by the former one is the reduction of the structure's diameter; the paths between any two nodes of the structure becomes shorter in average.

While the connectivity (which corresponds to the maximum number of neighbors) - and therefore the complexity for updating the neighborhood of a new node - increases exponentially, the complexity of finding a potential position for a neighbor stays constant, independently of the number of dimensions. Even in a n-dimensional grid with $n > 2$, a node just has to consider two dimensions to make a decision, if it can grow or not. The two

dimensions can be selected randomly. A single node gets many more potential positions into which it can grow. If a node recognizes that it cannot grow considering the two chosen dimensions, it can just choose other dimensions and check if the growth-rules are fulfilled for these dimensions.

3.2 Modifications

Some modifications were to be made to the n-dimensional version of the Border-Growth-Algorithm. The first modification is the elimination of diagonal links. Since the number of those diagonal links grows exponentially with the number of dimensions it would not be useful to maintain them within a higher dimensional grid. In 5 dimensions there would be $2^5 = 32$ additional links. The main purpose for doing so was to increase connectivity. But now, better connectivity is achieved by the higher number of dimensions. The mathematical model was relaunched and generalized and now works without diagonal links at all.

To simplify the mathematical description of the algorithm, only the positive quadrants are considered. This makes it necessary to adapt the Border Node growth rule. In the new version a Border Node does not synchronize with a neighbor-node of the neighbor-quadrant, but with its neighbor on the other side within the quadrant. The potential growth-rate of all nodes is equal, and the growth-behavior it symmetric to the origin, so these modification will not significantly change the structure's qualitative growth behavior.

3.3 The n-Dimensional Border-Growth Algorithm

This section gives a detailed mathematical description of the new n-dimensional border-growth-algorithm. Let $n \in N, n > 1$ be the grid's dimension and $t \in N_0$ the discrete time needed, to clearly identify multiple join-operations that occur at the same time. To refer to direct and indirect neighbors of a node, linear combinations of normalized unit vectors are used: $J_+ = \{\vec{j}_1, .., \vec{j}_n |\ \vec{j}_i = (x_1, .., x_n), x_k = \begin{cases} 1, if\ k = i \\ 0, if\ k \neq i \end{cases}\}$ is the set of unit vectors along the positive main axes and $J_- = \{-\vec{j}_i |\ \forall 1 \leq i \leq n\}$ the set of unit vectors along the negative main axes. $J = J_+ \cup J_-$ is the set of unit vectors along all main axes.

The network itself can be described as a graph G with a vertex-set V, and an edge-set E. Since the graph describes a network that changes over time, V and E, and therefore G depend on the discrete time t: $G(t) = (V(t), E(t)), E(t) \subset V^2(t)$. Every vertex of the graph describes a network-node. The function pos associates a n-dimensional position to every node $v \in V(t)$: $pos |\ V(t) \rightarrow \mathbb{N}_0^n,\ \ pos(v_i) = (x_1, ..x_n)$. The function N associates a node $v \in V(t)$ with a neighbor $N(v, \vec{j}_i, t)$ for each $\vec{j}_i \in J$ at time t. If v does not have a neighbor in $\vec{j}_i$, then $N(v, \vec{j}_i, t)$ is 0: $N |\ (V(t), J, t) \rightarrow V(t) \cup \{0\},\ \ N(v, \vec{j}_i, t) = \begin{cases} \text{neighbor of } v \text{ in direction } \vec{j}_i \\ 0,\ if\ \ \nexists w \in V(t) \text{with } pos(v) + \vec{j}_i = pos(w) \end{cases}$

The growth-process starts at $t = 0$. The only node at start-up is a Root Node v_0 residing at the coordinate-system's origin. Since v_0 is the only node, no neighbor-relationships are established yet: $t := 0,\ V(t) := v_0,\ E(t) := \{\},\ pos(v_0) = (0, .., 0)$. We classify the nodes that joined the structure into three types. The **Root Nodes** that reside at the center of the virtual, discrete coordinate system: $V_0(t) = \{v_0|\ v_0 \in V(t) \wedge pos(v_0) = \vec{0}\}$ The **Border Nodes**, that grow along the axes: $V_B(t) = \{v_B|\ v_B \in V(t) \wedge pos(v_B) = m * \vec{j_i}\ \forall \vec{j_i} \in J_+\}$ The **Inner Nodes**, that are all nodes which are neither Root Nodes nor Border Nodes: $V_I(t) = \{v_I|\ v_I \in V(t) \setminus (V_0(t) \cup V_B(t))\}$

A node that wants to accept a new node, must fulfill at least one growth-rule. The node-type specifies which rules can be used to determine if it can grow, and in which directions it can grow. There are three growth-rules. For a given node they specify a set of directions into which this node could grow according to the used rule. All rules contain the basic condition $N(v, \vec{j}, t) = 0$, which ensures that nodes only can grow into directions$\vec{j}$ that are currently free. Root Nodes use the Root-Growth-Rule L_0 to determine into which directions they could grow:

Root-Node-Growth-Rule L_0: Root Nodes can grow into any free positive direction: $L_0(v_0 \in V_0(t)) = \{\vec{j_k} \in J_+|\ N(v_0, \vec{j_k}, t) = 0\}$. **Border-Node-Growth-Rule L_B:** A Border Node v_B can grow along its positive axis, if the position into which it wants to grow is free, and if it has a neighbor, which is an Inner Node: $L_B(v_B \in V_B(t)) = \{\vec{j_k} \in J_+|(N(v_B, \vec{j_k}, t) = 0) \wedge (\exists w \in V(t), \vec{j_l} \in J_+ \setminus \{\vec{j_k}\}$ with $pos(w) = pos(v_B) + \vec{j_l})\ \}$ **Inner-Node-Growth-Rule L_{IB}:** A Border Node v_B or an Inner Node v_I can grow, if the position in direction $\vec{j_i}$ is free, and if the new position has a neighbor in at least one further dimension, which has a common neighbor with the growing node: $L_{IB}(v_{IB} \in V_I \cup V_B) = \{\vec{j_k} \in J_+|(N(v_{IB}, \vec{j_k}, t) = 0) \wedge (\exists w, z \in V(t), \vec{j_l} \in J \setminus \{\vec{j_k}, -\vec{j_k}\})|\ (pos(w) = pos(v_{IB}) + \vec{j_l}) \wedge (pos(z) = pos(v_{IB}) + \vec{j_l} + \vec{j_k})\ \}$ Following these rules it is possible that a node can accept multiple new nodes at the same time. All possible grow directions for a node $v \in V(t)$ are given by: $L(v) = L_0(v) \cup L_B(v) \cup L_{IB}(v)$.

It is possible (and for neighboring nodes even likely) that two different nodes v, w want to grow to the same position at the same time: $pos(v) + \vec{l_j} = pos(w) + \vec{l_k},\ \vec{l_j} \in L(v),\ \vec{l_k} \in L(w)$. This must be avoided, since it leads to overlapping, inconsistent structures. The way to do this, is to provide a locking-mechanism. Nodes can be locked. A lock is related to a certain growth-position. If a node wants to grow into a certain direction, it first has to lock all nodes that belong to the neighborhood of the new position. If at least one node is already locked by another growing node, the growth process must be canceled. This locking-mechanism is not part of the mathematical model described here. The simulation uses a simplified way to avoid those situations by taking advantage of having a global view to all nodes that want to grow. See section 3.4 for further details.

Once a node $v \in V(t)$ with a join-request from a new node v_N has a rule which allows it to grow into a direction $\vec{j_i} \in L(v)$ of an unlocked environment, it will initiate the growth process. The first step of this process is to initiate the new node v_X by giving it a position within the grid: $pos(v_X) := pos(v) + \vec{j_i}$. v_X has to be added to the graph $G(t+1)$'s vertex set. The edge set is appended by two new tuples since all neighbor-links are bidirectional: $V(t+1) := V(t) \cup \{v_X\}$, $E(t+1) := E(t) \cup \{(v, v_X), (v_X, v)\}$. The neighbor-links between v and v_X have to be established:$N(v, \vec{j_i}, t+1) := v_X$, $N(v_X, -\vec{j_i}, t+1) := v$.

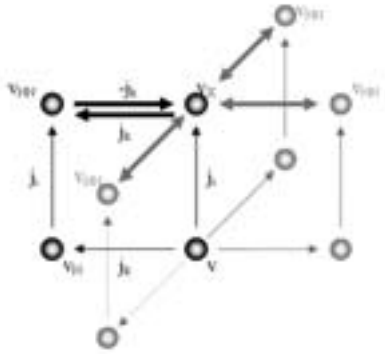


Figure 2: Illustration of the neighborhood-update in a 3d-lattice for a new node v_X, that was accepted by node v in direction $\vec{j_i}$

Now, the neighbor-links of all nodes in v_X's neighborhood have to be updated. To make v find these nodes, all neighbors v_N of v in all dimensions, except that given by $\vec{j_i}$ are requested to make their neighbors v_{NN} in direction $\vec{j_i}$ being a neighbor of v_X (see Fig. 2):
$\forall \vec{j_k} \in J \setminus \{\vec{j_i}, -\vec{j_i}\} | (v_N := N(v, \vec{j_k}, t) \neq 0) \ \wedge (v_{NN} := N(v_N, \vec{j_i}, t) \neq 0) :$
$N(v_{NN}, -\vec{j_k}, t+1) := v_X, \ N(v_X, \vec{j_k}, t+1) := v_{NN}$
$E(t+1) := E(t) \cup \{(v_{NN}, v_X), (v_X, v_{NN})\}$

Summarizing, a node that got a join-request performs the following steps:

- check the rule(s) applicable for this node type.
- if there's no direction to which to grow, reject or forward join-request, else choose one possible direction.
- try to lock the environment, which would be involved in the growth-process
- if locking failed, choose another direction and try again.
- if an unlocked environment could be found, lock it. Iif there's no unlocked environment for any grow-direction, reject or forward join-request.
- perform growth-process (see below) and unlock environment.

The algorithm described here does not yet take any node failures into account. It assumes 'perfect' nodes and communication channels. Detailed strategies on how to deal with node failures in real-life networks is part of further work.

3.4 Simulation

For simulating the 2D-algorithm from [BORDER09] P2PNetSim [P2PNETSIM06], a distributed Java-based network simulator was utilized. A protocol suite for node-(un)locking, neighborhood establishment, and join requests was implemented. It could directly be used in real network applications.

To simplify performance- and efficiency analysis a much more compact, non-distributed Java application was provided for the n-dimensional border-growth implementation. It focuses on the algorithm itself, rather than on "real-life's" technical issues. Since the implementation was intended to be a proof-of-concept, the focus of the implementation was to provide a quick, stable realization of the algorithm. The implementation makes use of the fact, that there is a global view on the whole network in the implementation: No local locking mechanism was implemented. Instead, in each discrete time-step t a data structure is built that contains lists of those nodes, that want to grow to the same position. From this list, only one randomly selected node is allowed to grow. From the 'local point of view', this would be the node that first got the chance to lock the new position's environment. The advantage is, that the simulation needs fewer resources and can simulate growth-processes with millions of nodes on a single machine. Comparisons between this code running with two dimensions with the 2d-simulation from [6], which uses (un-)locking-protocols show that there are no qualitative differences between the global and the local way of resolving conflicts. Later versions will be distributed again, and will implement the lock-/unlock-protocol again in order to get closer to a real networking scenario.

Fig. 3a, 3b, 3c, and 3d show the output of four small simulations with 40 nodes that grew in two, three, four and five dimensions. Fig. 3e shows the output of a larger simulation that built 20.000 nodes into a five-dimensional lattice. Remember, that, for simplicity, the algorithm described here, just considers the positive quadrants. The dark lines are the five positive axes of the lattice - projected to a two-dimensional circle.

4 Simulation results

Fig. 4 shows the growth-rates in the first 100 time steps for simulations with two, three, four, and five dimensions. In the simulation of two dimensions the growth-rate has roughly linear characteristics (notice, that the growth-rate axis has a logarithmic scale.) After 100 time steps the growth-rate is 17 nodes/time-step. The five dimensional case shows an exponential course which leads to a growth-rate of about 300.000 nodes/time-step after 100 time steps. As expected, the growth-scalability can be strongly improved by increasing the structure's dimension. Assuming that the number of nodes and the growth rate follow these equations: $\#nodes = a_{\#nodes} * t^{n_{\#nodes}}$ and $growthRate = a_{growthRate} * t^{n_{growthRate}}$ The coefficients a and exponents n were computed, based on the simulation data in Fig. 4, as follows: The tables in show that growth-scalability is improved exponentially with the increasing dimensions, though the fractions of growable nodes decrease with more dimensions. The nodes' ability to grow into several dimensions within one time step

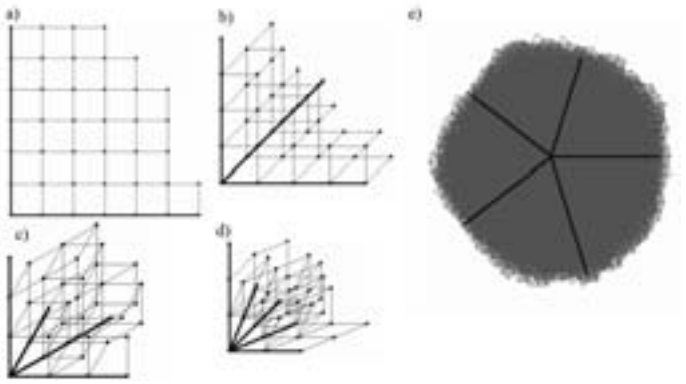


Figure 3: a-d: simulation with 40 nodes in a 2d-, 3d-, 4d- and 5d-lattice, e: simulation with 5 dimensions and 20k nodes

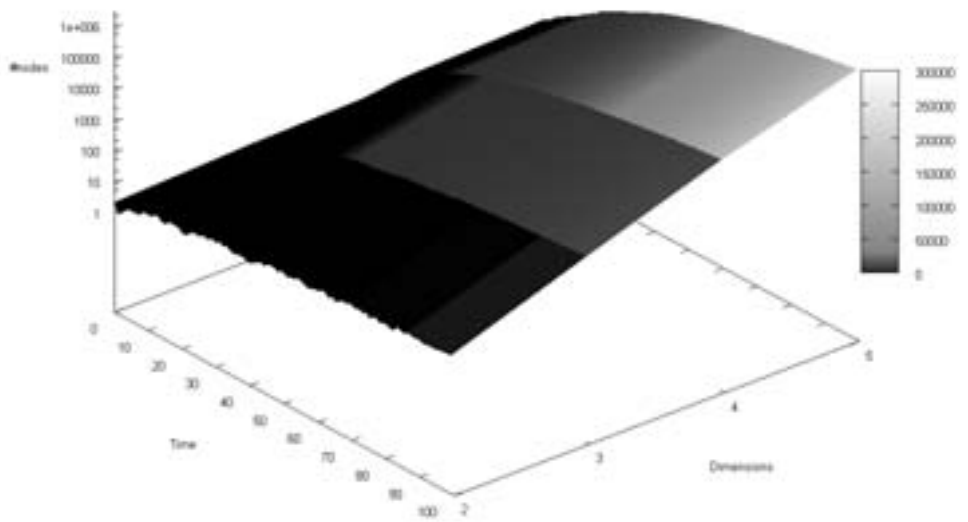


Figure 4: a: growth rate depending on time and dimension

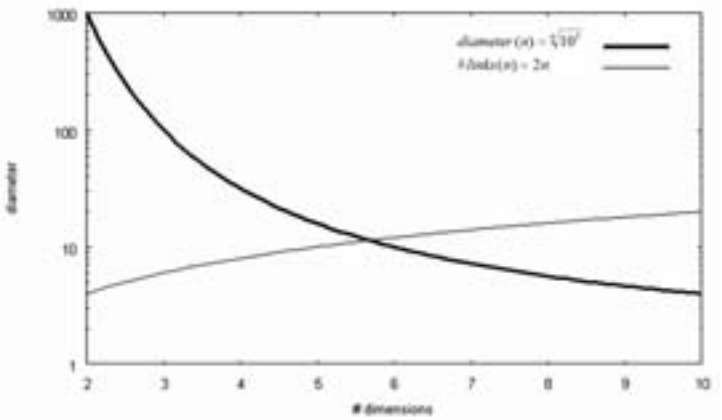


Figure 5: The diameter (bold graph) and the number of links per node (thin graph) of a network with 1.000 nodes depending on the number of dimensions

	2d	3d	4d	5d
a	0,269	0,087	0,025	0,006
n	1,912	2,869	3,807	4,736

	2d	3d	4d	5d
a	0,563	0,470	0,303	0,169
n	0,892	1,729	2,548	3,348

Figure 6: left table: growth equation parameters, right table: growth-rate-equation parameters

does obviously not affect the growth-behavior in a positive way. Detailed analysis of the growth-behavior with increasing dimensions will be subject of further research.

For estimating the network's diameter it is assumed as a first approximation that the structure has a n-dimensional cubic shape, that grow uniformly into each possible direction. The diameter of a network with 1000 nodes depending on the dimension then would be: $diameter(n) = \sqrt[n]{1000}$ (see bold graph in Fig. 5). While the network's diameter decreases exponentially with the number of dimensions, the number of neighbor-links that have to be managed by each node, grows only linearly: $\#links(n) = 2n$ (thin graph in Fig. 5).

By providing more than two dimensions a node has multiple directions in which it could grow within a single time step. Originally it was expected that this will improve growth-behavior especially in higher dimensions. Since this is valid for all nodes, this leads to the situation that more nodes compete for the same position, which lowers the advantage of the nodes' possibilities of multi directional growth. That might be the reason, why the algorithm can't really take advantage of this effect. This might change in much later time steps of the simulation, especially when higher dimensions (> 7) are used.

Beside that there are other factors that hinder growth, the more so as more dimensions are used. Detailed investigations have to be made to find exact formulas to predict the n-dimensional growth behavior.

5 Conclusion & Outlook

By increasing the structure's dimension, the scalability can be significantly improved. The structure's diameter decreases exponentially. The complexity of the growth-process stays constant independently of the number of dimensions, the complexity of the update-process of a new node's environment increases linear with the number of dimensions. Since there is no limit for the number of dimensions, topologies with small diameters can be established. [RFID07] introduces an algorithm that uses EPCs (Electronic Product Codes) to manage product information in a decentralized network spanning across multiple organizations. The two-dimensional address-space of such a network is defined by splitting a 28-bit part of the EPC into two components representing the coordinates in the network. It could be improved by mapping the 28-bit code to a, for example, seven-dimensional border-growth-structure. This would result into a network with a diameter of $2^{\frac{28}{7}} = 16$.

References

[LCP04] *Lua, E.; Crowcroft, J.; Pias, M.; Sharma, R.; Lim, S.*: "A Survey and Comparison of Peer-to-Peer Overlay Network Schemes", IEEE Communications Survey and Tutorial, March 2004.

[SUL04] *Sakarian, G.; Unger, H.; Lechner, U.*: "About the value of Virtual Communities in P2P networks.", Proc. ISSADS 2004, Guadalajara, Mexico, Lecture Notes in Computer Science (LNCS) 3061, Guadalajara, Mexico, 2004.

[RFH01] *Ratnasamy, S.; Francis, P.; Handley, M.; Karp, R. M.; Shenker, S.*: "A Scalable Content Addressable Network", Proc. ACM SIGCOMM 2001.

[CHORD01] *Stoica, I.; Morris, R.; Karger, D.; Kaashoek, M. F.; Balakrishnan, H.*: "Chord: A scalable peer-to-peer lookup service for internet applications.", In Sigcommâ01, Proc. of the 2001 conference on Applications technologies architectures and protocols for computer communications, pp. 149-160.

[STRUCT02] *Unger, H.; Unger, H.; Titova, N.*: "Structure Building in Distributed Communities", A. Tentner (ed.) "High Performance Computing (HPC) '2002",San Diego, 2002

[BORDER09] *Berg, D.; Unger, H.; Sukjit, P.*: "Borderline-growth a new method to build complete grids with local algorithms", 2nd International Workshop on Nonlinear Dynamics and Synchronization 2009 (INDSâ09), S. 95-99, Klagenfurt, Austria, July 2009,Shaker Verlag, Aachen, 2009

[RFID07] *Berg, D.; Coltzau, C.; Sukjit, P.; Unger, H.; Nicolaysen, J.*: "Passive RFID tag Processing using a P2P architecture.", Malaysian Software Engineering Conference 2007, S. 169-179, 2007

[THERM09] *Lertsuwanakul, L.; Unger, H.*: "A Thermal Field Approach in a Mesh Overlay Network.", 5th National Conference on Computing and Information Technology (NCCIT'09),Bangkok, Thailand, 2009

[3DIOS09] *Coltzau, C.; Unger, H.*: "3DIOS â Konzept eines Internet Operating Systems.", Gemeinschaften in Neuen Medien (GeNeMe) '09, 2009

[P2PNETSIM06] *Coltzau, C.*: "Specification and Implementation of a Simulation Environment for Large P2P-Systems", Diploma, University Of Rostock, 2006

Modeling the Complex Dynamics of Distributed Communities of the Web with Pretopology

Vincent Levorato, Marc Bui

Laboratoire d'Informatique et des Systèmes Complexes (LaISC)
41 rue G. Lussac, F-75005, Paris, France.
Email : {vincent.levorato, marc.bui}@laisc.net

Abstract: The aim of this article is to present a methodological approach for problems encountered in structural analysis of web communities. This approach is based upon the pretopological concepts of pseudoclosure and the searching of equivalent nodes. The advantage of this approach is that it provides a framework needed to pass through the actual limits of graph theory modeling. The problem of modeling and understanding web communities is described, then a review of the existing models and their limits, and we finish by an example of a structuring algorithm.

1 Introduction

The study of complex networks dynamics is a domain which is still topical, and specially on the Web communities aspect [JPS02]. These communities continuously evolve in some evolutionary process starting with just a few individuals and the resulting set of interrelated community members is generally called the *social network* of a community. While the number of individuals in a community can grow very fast, the single individual needs only little information about other individuals to still be able to potentially interact with a large number (or all) of the community members [JVJ02]. The six degrees of separation property illustrates this in the case of human communities [DJW98]. Moreover, communities are often characterized by a highly self-organizing behavior.
Computer networks or distributed systems in general may be regarded as communities ; most obviously, the Internet or Web forms entities that can be characterized as communities. Nowadays, we are able to recover large amount of data on the Web using *logs files* in a lot of well known Web communities such as LinkedIn (professional relations), Second Life (virtual life game) or political blogs. It is a very important point because of the few amount of data usually used by sociologists [MNW06]. In this article, we want to analyze social networks on the Web ; thus, we must focus on two points:

- *identifying and working with appropriate datasets*: one needs a large, realistic social network containing a significant collection of explicitly identified groups, and with sufficient time-resolution that one can track their growth and evolution at the level of individual nodes.

- *developing new theoretic models* to pass through the limits of tools provided until now and to build a new theoretical one, adapted to real-world networks, and especially in this case, web communities.

First, we will make a review of the existing network models that had been studied many years ago, in the second part, we propose our own model based on pretopology with an algorithm and an associated example, then we'll finish by a conclusion opening on new ideas and future work.

2 Limits of tools and theory

2.1 Study of Social Network Analysis

If we would like to represent a social network, we use generally graph theory: sociologists choose one property (friendship connection, associate connection, ...) [Deg04] [DV94] to study despite of the others. In the meantime, networks, not only in social sciences, are evolving structures and dynamical systems [MNW06]. And graph theory seems to be not enough complete to represent all the properties of such a complex system. Here is a review of few definitions concerning the classical graph theory, followed by a review of existing models using graph theory, showing that they reached their limits. Next, we will focus on the extension of this theory: *hyper-graphs*, that should be an answer to bring new models, but finally, we'll explain later in the article that hyper-graphs are a special case of pretopology, so they are included into.

2.1.1 Graph Theory

A graph is a mathematical abstraction that is useful for solving many kinds of problems. We assume the reader familiar with the notions of graph theory, so only a quick review is presented here. Fundamentally, a graph consists of a set of vertices, and a set of edges, where an edge is something that connects two vertices in the graph. More precisely, a graph is a pair (V, E), where V is a finite set and E is a binary relation on V. V is called a vertex set whose elements are called vertices. E is a collection of edges, where an edge is a pair (u, v) with $u, v \in V$. In a directed graph, edges are ordered pairs, connecting a source vertex to a target vertex. In an undirected graph, edges are unordered pairs and connect the two vertices in both directions, hence in an undirected graph (u, v) and (v, u) are two ways of writing the same edge. Here is some few definitions:

A sub-graph is a subset of a graph G where p is the number of sub-graphs. For instance $G' = (v', e')$ can be a distinct sub-graph of G. *Connection* means a set of two nodes as every node is linked to the other. A *Path* is a sequence of links that are traveled in the same direction. For a path to exist between two nodes, it must be possible to travel an uninterrupted sequence of links. A *Chain* is a sequence of links having a connection in common with the other, never mind the direction. The *Length* of a path is the number of

links (or connections) in this path. A *Cycle* refers to a chain where the initial and terminal node is the same and that does not use the same link more than once is a cycle. A *Circuit* is a path where the initial and terminal node corresponds. It is a cycle where all the links are traveled in the same direction. A graph is *symmetrical* if each pair of nodes linked in one direction is also linked in the other. By convention, a line without an arrow represents a link where it is possible to move in both directions. However, both directions have to be defined in the graph.

A graph is *complete* if two nodes are linked in at least one direction. A complete graph has no sub-graph. A complete graph is described as *connected* if for all its distinct pairs of nodes there is a linking chain. Direction does not have importance for a graph to be connected, but may be a factor for the level of connectivity. If $p > 1$ the graph is not connected because it has more than one sub-graph. In a connected graph, a node is an *articulation point* if the sub-graph obtained by removing this node is no longer connected. It therefore contains more than one sub-graph ($p > 1$).

2.1.2 Models of Networks

A lot of applications are using Internet, and a lot of researchers had tried to model it. We make here a review of these most famous models, showing that they are not sufficient for modeling dynamics and structure of real networks, especially web communities, compared to pretopology. Models using classical graph theory can be isolated in three basic classes:

- Random graphs models
- Small-Worlds models
- Scale-Free models

and in another category:

- Hypergraphs model

Random graphs models In 1959, Erdös and Rényi [PE59] published a seminal article in which they introduced the concept of a random graph $G_{n,p}$. A random graph is simple to define. One takes some number N of nodes or *vertices* and places connections or *edges* between them, such that each pair of vertices i, j has a connecting edge with independent uniform probability p. We show example of such random graph in Fig. 1. This model is one of the simplest models of a network there is, and is certainly the most studied. The random graph has become a cornerstone of the discipline known as discrete mathematics, and many hundreds of articles have discussed its properties. However, as a model of a real-world network, it has some serious shortcomings. Perhaps the most serious is its degree distribution (poisson distribution), which is quite unlike those seen in most real-world networks (power-law distribution).

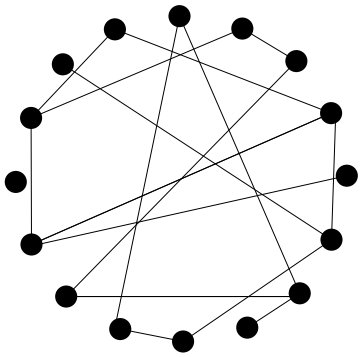


Figure 1: Erdös & Rényi random graph $G_{n,p}$

Molloy & Reed in 1995 [MM95] introduced an example of a mathematically rigorous treatment of random graphs with arbitrary sequences, breaking the distribution degree limitation of the original $G_{n,p}$ model.

Small-world models This model has been introduced first by Watts and Strogatz [DJW98] in 1998 as a simple model of social networks. Although the model has some drawbacks as a model of a real social network, it provides good intuition about the small-world effect as well as demonstrating convincingly the utility of statistical physics techniques in the study of networks. This small-world model (Fig. 2) is motivated by the observation that many real-world networks show the following two properties:

1. The *small-world effect*, meaning that most pairs of vertices are connected by a short path through the network.
2. High *clustering* meaning that there is a high probability that two vertices should connect directly to one another if they have another neighboring vertex in common.

Kleinberg [Kle00] proposed another kind of small-world belonging to the domain of *search networks*. In his model, vertices are connected together on a regular lattice, and a low density of long-range *shortcuts* are added between randomly chosen vertices (Fig. 3). His greedy algorithm finds a random target from a random starting point in time poly-logarithmic in the lattice size. His model is mainly used for navigation rather than for structural and dynamic purpose.

Scale-free Networks models Several of the articles about the models described previously focus on the observed degree distributions of real networks, finding that for a number of systems, including citation networks, the World Wide Web, the Internet, and obviously certain social networks, the degree distribution approximates a power law [MNW06]. The identification of networks with power-law degree distributions has generated a very large

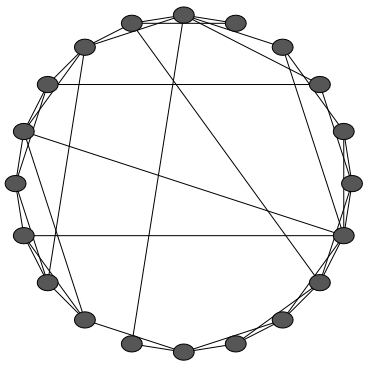


Figure 2: Watts & Strogatz Small-world

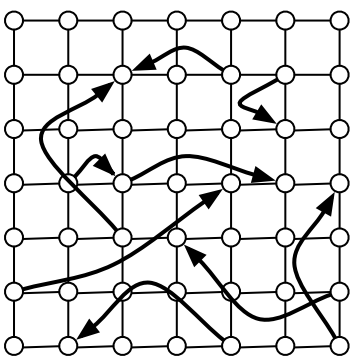


Figure 3: Kleinberg Small-world

number of publications on such networks, *scale-free networks* as they are widely called, a term introduced by Barabási and Albert [ALB99] in 1999.
Efforts at constructing models of scale-free networks have taken network research in new direction. Previous models, such as the random graphs and small-world models do not have power-law degree distributions. Barabási offered a simple generative mechanism called *preferential attachment* that created networks with a power-law degree distribution, where a new node has a higher probability to connect an existing node with a high degree (Fig. 4). The idea is interesting but the resulting graph is a tree, a non realistic structure for real-world networks.

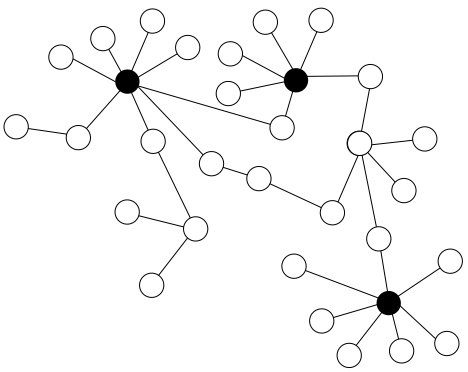


Figure 4: Scale-free network

Hypergraphs In mathematics, a hypergraph is a generalization of a graph, where edges can connect any number of vertices. Formally, a hypergraph is a pair (X, E) where X is a set of elements, called nodes or vertices, and E is a set of non-empty subsets of X called hyperedges. Therefore, E is a subset of $\mathcal{P}(X)\backslash\{\emptyset\}$, where $\mathcal{P}(X)$ is the power set of X. While graph edges are pairs of nodes, hyperedges are arbitrary sets of nodes, and can therefore contain an arbitrary number of nodes.
A hypergraph is also called a set system or a family of sets drawn from the universal set X. Hypergraphs can be viewed as incidence structures and vice versa. Unlike graphs, hypergraphs are difficult to draw on paper, so they tend to be studied using the nomenclature of set theory rather than the more pictorial descriptions (like 'trees','forests' and 'cycles') of graph theory. (Fig. 5)
A transversal or hitting set of a hypergraph $H = (X, E)$ is a set $T \subset X$ that has nonempty intersection with every edge. The transversal hypergraph of H is the hypergraph (X, F) whose edge set F consists of all transversals of H. Computing the transversal hypergraph has applications in machine learning and other fields of computer science.
A hypergraph H is called $k - uniform$ or a $k - hypergraph$ if every edge has cardinality k. A graph is just a $2 - uniform$ hypergraph. The degree $d(v)$ of a vertex v is the number of edges that contain it. H is $k - regular$ if every vertex has degree k. Let $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots e_m\}$. Every hypergraph has an $n \times m$

incidence matrix $A = (a_{ij})$ where

$$a_{ij} = \begin{cases} 1 & \text{if } v_i \in e_j \\ 0 & \text{otherwise} \end{cases}$$

The transpose A^t of the incidence matrix defines a hypergraph $H^* = (V^*, E^*)$ called the dual of H, where V^* is an $m - element$ set and E^* is an $n - element$ set of subsets of V^* . For $v_j^* \in V^*$ and $e_i^* \in E^*$, $v_j^* \in e_i^*$ if and only if $a_{ij} = 1$. The dual of a uniform hypergraph is regular and vice-versa. Considering duals often leads to discoveries.

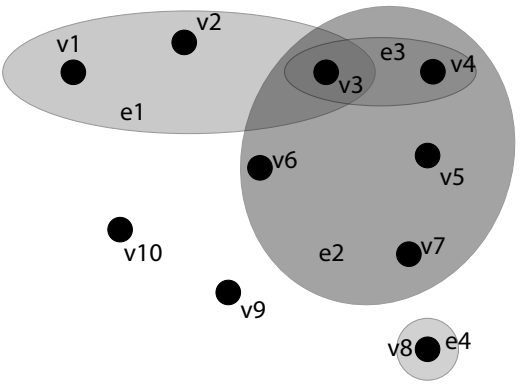


Figure 5: Hypergraph

Hypergraph is a better tool than classic graph to model social networks. We will see in the continuation of this article that hypergraph can be represented with pretopology theory using a certain space. In fact, hypergraph is a particular case of pretopological space.

2.1.3 Why pretopology ?

They are several reasons to make a new network model using pretopology: the models using graph theory have non adequate properties. First, we can't dissociate links: oriented or not, they are all the same. If we want to use n different relations between nodes, we have to construct n different graphs, that is not very practical. Second, all the models using graph theory presented previously have their weakness compared to real networks. Third, the relation are from a node to another, so we can't have relations between a group of node and a node (for example). Hypergraphs should have been the answer but we'll see that it's only a particular case of pretopology. Where others models fail, pretopology theory can bring a real answer.

2.2 Tools for Social Network Analysis

A lot of tools for social network analysis exist. Here is a few :

- *StOCNET* is a project that builds an advanced software system for statistical social network analysis. The software for StOCNET has been developed in collaboration between software engineers of Science Plus and the researchers who contributed the programs that are included in StOCNET.
- *UCINET* is a comprehensive program for the analysis of social networks and other proximity data. The program contains dozens of network analytic routines.
- *Pajek* (Slovenian: spider) is a software for large network analysis which is free for non-commercial use.

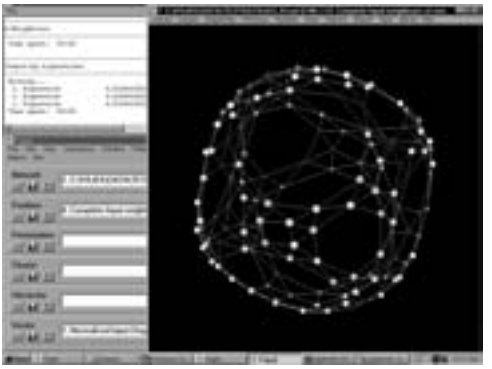


Figure 6: Screenshot of Pajek software

In this softwares, two problems come confirm the problematic described in the introduction:

- The real data sets used here are based on investigations next to the persons (in most cases). This method to collect data is expensive, tedious and can be composed of errors (human behavior is not necessarily natural during investigation).
- The theory used is the *graph theory* which is not the best to represent social networks.

3 Analysis of Structure using Pretopology

The analysis of web communities is a complex task: most of the scientific works and studies in sociology use only description to define the behavior of such networks. Analytic

tools able to *reconstruct* this networks are nonexistent and that's a domain in which there is a lot of requests.
In order to answer this problem, we apply the concepts of pseudoclosure and minimal closed subsets that have been developed in pretopological theory. Pretopology allows the study of parts family of a set to put the obviousness of their structural quality, links, and evolutions. Therefore, dynamic structure of the network can be examined step by step (not feasible with classical topology) for a better understanding [Bel93].
To have interesting results, we propose to apply the theory to a known web community because of its huge amount of data and of the easiness to recover it: *LinkedIn*.

3.1 Pretopology Theory

Let us consider a non-empty finite set E, and $\mathcal{P}(E)$ designates all of the subsets of E.

3.1.1 Pretopological space

Definition 3.1 *A pretopological space is a pair (E, a) where a is a map $a(.) : \mathcal{P}(E) \rightarrow \mathcal{P}(E)$ called pseudo-closure (Fig. 7) and defined as follows : $\forall A$, $A \subseteq E$ the pseudo-closure of A, $a(A) \subseteq E$ such that :*

- $a(\emptyset) = \emptyset$ *(P_1)*
- $A \subseteq a(A)$ *(P_2)*

The pseudo-closure is associated to the *dilation process*. Thus, $a(.)$ can be applied on a set A in sequence, so as to model expansions : $A \subset a(A) \subset a^2(A) \subset ...$. That means we could follow the process *step by step*, which is not possible with topology. Using the pseudoclosure, we can directly model the proximity concept, very useful for aggregation process.

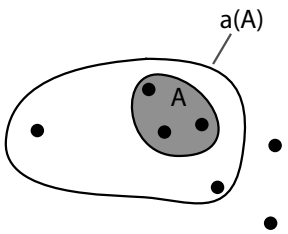


Figure 7: Pseudoclosure of A

We need a taxonomy leading to a better choice of tools, so we have to define a space. The $\mathcal{V}_S$ type is the most useful for our problem.

3.1.2 $\mathcal{V}$, $\mathcal{V}_D$, and $\mathcal{V}_S$ Pretopological spaces

Definition 3.2 *A $\mathcal{V}$ pretopological space (E, a) is defined by :*

$$\forall A, B \subseteq E, A \subset B \Rightarrow\ a(A) \subset a(B)$$

Definition 3.3 *A $\mathcal{V}_D$ pretopological space (E, a) is defined by :*

$$\forall A, B \subseteq E, a(A \cup B) = a(A) \cup a(B)$$

Definition 3.4 *A $\mathcal{V}_S$ pretopological space (E, a) is defined by :*

$$\forall A \subseteq E, a(A) = \bigcup_{x \in A} a(\{x\})$$

3.1.3 *Closure*

Definition 3.5 *$A \in P(E)$ is closed if and only if : $A = a(A)$*
The closure of $A \in P(E)$ is the smallest closed subset containing A, noted $F(A)$ or F_A.

3.1.4 *Connectivities*

We define $\chi - connectivity$ to designate one of the following connectivity:

Definition 3.6 ***Connectivity***
Let (E, a) a $\mathcal{V}$ pretopological space.
(E, a) is connected iff $\forall C \subset E$ and $C \neq \emptyset$,
$F(C) = E$ or $F(E - F(C)) \cap F(C) \neq \emptyset$

Definition 3.7 ***Strong Connectivity***
Let (E, a) a $\mathcal{V}$ pretopological space.
(E, a) is strongly connected iff $\forall C \subset E$ and $C \neq \emptyset$,
$F(C) = E$

Definition 3.8 ***Unilateral Connectivity***
Let (E, a) a $\mathcal{V}$ pretopological space.
(E, a) is unilaterally connected iff $\forall C \subset E$ and $C \neq \emptyset$,
$F(C) = E$ or $\forall B \subset E$ and $B \neq \emptyset$, $B \subset E - F(C) \Rightarrow C \subset F(B)$

Definition 3.9 ***Hyper Connectivity***
Let (E, a) a $\mathcal{V}$ pretopological space.
(E, a) is hyper connected iff $\forall C \subset E$ and $C \neq \emptyset$,
$F(C) = E$ or $\exists B \subset E$ and $B \neq \emptyset$, $B \subset E - F(C) \Rightarrow C \subset F(B)$

Definition 3.10 ***Apo-Connectivity***
Let (E, a) a $\mathcal{V}$ pretopological space.
(E, a) is apo-connected iff $\forall C \subset E$ and $C \neq \emptyset$,
$F(C) = E$ or $\forall B \subset E$ and $B \neq \emptyset$, $B \subset E - F(C) \Rightarrow F(C) \cap F(B) \neq \emptyset$

3.2 Social Network definition with Pretopology

A social network is a social structure made of nodes (which are generally individuals or organizations) that are tied by one or more specific types of binary or valued relations [Deg04].
In pretopology, we can generalized this definition by saying that a network is *a pretopologies family on a given set* E [DV94].

Definition 3.11 *Let* E *be a set.*
Let I *a countable family of indexes.*
Let $\{a_i, i \in I\}$ *a family of pretopologies on* E.
The family of pretopological spaces $\{(E, a_i), i \in I\}$ *is a network on* E.

This definition changes concepts of network models known until now. Take for example a $\mathcal{V}_\mathcal{S}$ pretopological space with entities having binary relations between them. We can redefine the notion of arc in graph theory by this formalism: there is an arc between $\{x\}$ and $\{y\}$ if and only if : $\{y\} \subset a(\{x\})$ (Fig. 8).

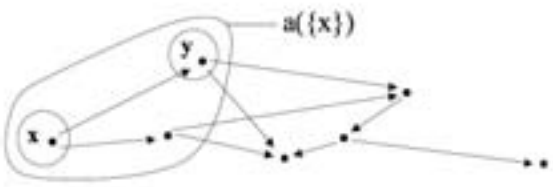


Figure 8: Concept of arc using pretopology

3.3 Equivalent nodes search

We propose in this part what we could call a direct application of the pretopology theory cited previously. In sociology, the problem of *equivalences* between nodes is well known. In mathematics terms, this problem refers to a *classification* problem on discrete data represented in a graph [DV94]. What are the objectives of finding equivalent nodes ?

- to bring together nodes with similar behaviors, meaning *substitutable* nodes regarding of their position in the structure.
- to reduce graph by assimilating *substitutable* nodes and by keeping inter-groups relations.

3.3.1 Definitions

First, we have to redefine what is an *articulation point* and what is a *weak point* with pretopology:

Definition 3.12 ***Articulation point (PA)***
Let (E, a) *a* $\mathcal{V}$ *pretopological space.*
Let $A \subset E$ *with non-empty* A *and* A *a* $\chi - connected$ *subspace of* (E, a).
Let $b \in A$.
b *is an articulation point of* A *in* (E, a) *iff* $(A - \{b\}, a_{A-\{b\}})$ *is not a* $\chi - connected$ *subspace of* (E, a).

Remark: If $A = \{b\}$, then b is not an articulation point of A in (E, a).

Definition 3.13 ***Articulation point with order*** k ***(PAk)***
Let (E, a) *a* $\mathcal{V}$ *pretopological space.*
Let $A \subset E$ *with non-empty* A *and* A *a* $\chi - connected$ *subspace of* (E, a).
Let $b \in A$ *with* b *articulation point of* A *in* (E, a).
Let k *positive integer not null.*
b *is an articulation point of* k *order of* A *in* (E, a) *iff the smallest of the biggest* $\chi - connected$ *subspace of* $(A - \{b\}, a_{A-\{b\}})$ *has a cardinal equal to* k.

Definition 3.14 ***Weak point (PF)***
Let (E, a) *a* $\mathcal{V}$ *pretopological space.*
Let $A \subset E$ *with non-empty* A *and* A *a* $\chi - connected$ *subspace of* (E, a).
Let $c \in A$.
c *is a weak point of* A *in* (E, a) *iff* $\exists\ b \in A - \{c\}$, b *articulation point of* A *of order* 1 *in* (E, a), *and* $\{c\}$ *the biggest* $\chi - connected$ *singleton subspace of* $(A - \{b\}, a_{A-\{b\}})$.

3.3.2 Method

For each biggest $\chi - connected$ subspace, decomposing space allows to determine "layers" of equivalent nodes. For each class created, the algorithm brings together the biggest $\chi - connected$ singleton subspace into one class, and continues decomposing of each biggest non-singleton subspace. This decomposition consists in searching "layers" of weak points, then in assembling articulation points of order 1 not classified yet, and so on (searching weak points...).
When there is no longer weak points neither articulation points of order 1, the algorithm brings together in one class all articulation points not classified before searching again weak points.
The algorithm stops if there is no more articulation points not classified (the rest part is said not decomposable). Here, the analysis is made from periphery to center.
The formalism of the decomposition and the formal definition of equivalency are not presented here because of their heaviness mathematical writing but can be found in [DV94].

3.3.3 Algorithm

We use $\chi - C\ SEP$ notation to define notion of biggest $\chi - connected$ subspace.

Definition 3.15 ***Algorithm***

Choose a (E, a) *a* $\mathcal{V}$ *pretopological space*
Decompose (E, a) *in* $\chi - C\ SEP$
Classify together the $\chi - C$ *singleton* SEP
Consider $\chi - C$ *non-singleton* SEP *set*
While *the* $\chi - C$ *non-singleton* SEP *set is not empty* ***Do***
    Consider one of the $\chi - C\ SEP$
    Take off this $\chi - C\ SEP$ *from the* $\chi - C\ SEP$ *set*
    Search the PA, $PA1$, *and* PF *of this* $\chi - C\ SEP$
    If PF *set is non-empty*
    Then *Classify together the* PF
    Else
        If *the found* $PA1$ *set in this* $\chi - C\ SEP$ *is non-empty*
        Then *Classify together this* $PA1$
        Else
            If *the found* PA *set in this* $\chi - C\ SEP$ *is non-empty*
            Then *Classify together this* PA
            Else *Classify together remaining nodes of the* $\chi - C\ SEP$
    Consider the result subspace of $\chi - C\ SEP$ *after removing classified nodes*
    Notate (A, a_A) *this subspace*
    Decompose (A, a_A) *in* $\chi - C\ SEP$
    If *the* $\chi - C$ *non-singleton* SEP *set of* (A, a_A) *is not empty*
    Then
        Classify together the $\chi - C$ *singletons* SEP *of* (A, a_A)
    Add each $\chi - C$ *non-singleton* SEP *obtained with the* $\chi - C$ *non-singleton* SEP *set*
End While

3.3.4 Example

Taking a small part of LinkedIn personal network (44 nodes), we applied the equivalence algorithm (Fig. 9). In this example, we find two relevant clusters, and seventeen nodes which are alone. When we have a look on the data, people in the same cluster have some

properties in common, like formation in the same establishment, which makes sense to group them together. Standards methods to make such a structure are not efficient (reduced graph for instance).

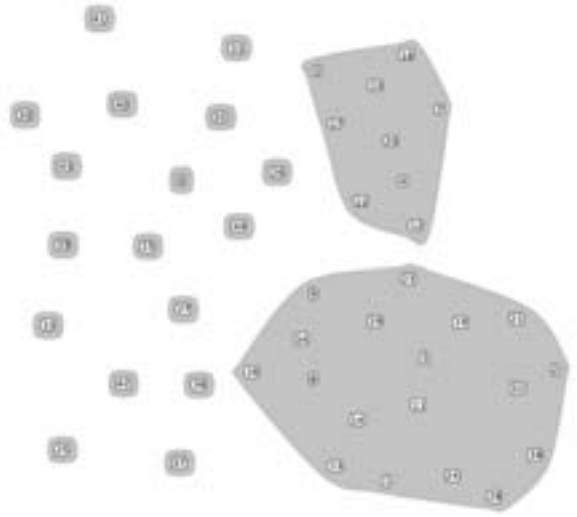


Figure 9: Equivalence algorithm using small part of LinkedIn network

4 Future Works

As shown in previous chapter, we use pretopology theory to build powerful network analysis tools. Next step is to apply theory to a bigger dataset of *LinkedIn*. LinkedIn is an online network of more than 10 million experienced professionals from around the world, representing 130 industries. When you join, you create a profile that summarizes your professional accomplishments. Your profile helps you find and be found by former colleagues, clients, and partners. You can add more connections by inviting trusted contacts to join LinkedIn and connect to you. Your network consists of your connections, your connections' connections, and the people they know, linking you to thousands of qualified professionals.

If I have specifics qualifications and I want to reach the boss of a corporation who is an articulation person of the network for instance, I have to isolate certain parts of this network and tools using pretopology are very useful for this kind of tasks.
We want to answer question such as how is it possible to determine that a node is a "strong" node of the network ? We have to go deeper in investigation to study the behavior of a such social network, by creating new formal methods and algorithms for a better understanding of structure and dynamics of web communities.

References

[ALB99] Réka Albert Albert-Laszlo Barabasi. Emergence of Scaling in Random Networks. *SCIENCE*, Vol 286:509–512, 1999.

[Bel93] Z. Belmandt. *Manuel de prétopologie et ses applications: Sciences humaines et sociales, réseaux, jeux, reconnaissance des formes, processus et modèles, classification, imagerie, mathématiques*. Hermes Sciences Publicat., 1993.

[Deg04] Alain Degenne. Entre outillage et théorie, les réseaux sociaux. *Réseaux Sociaux de l'Internet*, 2004.

[DJW98] Steven H. Strogatz Duncan J. Watts. Collective dynamics of 'small-world' networks. *NATURE*, Vol 393:440–442, 1998.

[DV94] Monique Dalud-Vincent. *Modèle prétopologique pour une méthodologie d'analyse des réseaux: concepts et algorithmes*. PhD thesis, Université Claude Bernard - Lyon 1, 1994.

[JPS02] Peter Schulthess John Plaice, Peter G. Kropf and Jacob Slonim. Distributed Communities on the Web, 4th International Workshop. *DCW*, 2002.

[JVJ02] Gilbert Babin Jean Vaucher, Peter Kropf and Thierry Jouve. Experimenting with Gnutella Communities. *Scientific series*, 55, 2002.

[Kle00] Jon M. Kleinberg. Navigation in a small world. *NATURE*, Vol 406:845, 2000.

[MM95] Bruce Reed Michael Molloy. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 6:161–180, 1995.

[MNW06] Albert-László Barabási Mark Newman and Duncan J. Watts. *The Structure and Dynamics of Networks*. 2006.

[PE59] Alfréd Rényi Paul Erdös. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.

OpenAdap.net: a Community-Based Sharing System

Alessandro E.P. Villa[1,2,3], Javier Iglesias[1,2,3], Solange Ghernaouti-Hélie[3]

[1]Inserm U318, Laboratory of Neurobiophysics,
University Joseph Fourier, Grenoble, France
{Alessandro.Villa, Javier.Iglesias}@ujf-grenoble.fr

[2]Laboratory of Neuroheuristics, University of Lausanne, Switzerland
http://www.nhrg.org/

[3]Information Systems Department, University of Lausanne, Switzerland
sgh@unil.ch
http://www.hec.unil.ch/isi/home/

Abstract: OpenAdap.net is an Open Source project aimed at breaking the barriers existing in the flow of data access and data processing. The infrastructure will make possible the dissemination of resources like knowledge, tools or data, their exposure to evaluation in ways that might be unanticipated and hence support the emergence of communities of users around a specific domain. The architecture is designed by analogy with a virtual distributed operating system in which the dynamic resources are presented as files in a structured virtual file system featuring ownership and access permissions. OpenAdap.net will be open to exploitation during Q3 2006 by networked organisations and alliances taking into account the vital issue of Internet security and privacy.

1 Introduction

The information circulating in the Cyberspace generates a historically unprecedented richness for sharing knowledge and methods of data processing. The software published at Internet in one instance becomes available for the whole society worldwide. However, most of the latest processing and analysis tools developed and provided by research laboratories and businesses are based on specific software and computer platforms.

The current situation generates a lag until these contributions become known within the same field of competence and restrict severely their availability, in particular for cross-fertilization application in other fields. This delay may provoke the re-invention of methods for data processing and, more generally, the re-discovery of the same knowledge. At the societal scale this delay inhibits the development of added-value activities originating from sharing the knowledge and generates repeated unnecessary expenses and even, erroneous applications. The societal benefits in terms of business developments, market diversification and creation of employment generated by the fast circulation and ease of access to the results of the Human Genome Project illustrate the interest in developing an

open and dynamic adaptive network for resource sharing with emergent properties within the Information Society.

Our project is aimed at breaking the barriers existing in the flow of data access and data processing increasing the overall cost of information processing and restricting its availability to developing countries, by avoiding re-invention of existing software and save time and expenses by the whole society and even prevent incorrect applications. The ability to evaluate and to find the best available solution to a given problem may have significant impact on areas such as economics, physics, environmental sciences, meteorology, and health. By adopting the best available method of analysing a set of data, it is possible to test alternative solutions and choose the best from its overall performance.

In the construction industry, which is playing an important role measured by investments in all economical systems, even small enhancements will make resources available to other purposes. Reduction of the energy consumption during the whole lifecycle of a building will decrease pollution and save money. By setting up an environment where people are used to working with state of the art technical solutions, companies throughout the World will be forced to increase their competitiveness work in a completeness way.

We envision the emergence of communities of users sharing resources like knowledge, tools, data, etc. for their specific domain. The key to this vision is the development of a sharing architecture that is independent of a specific type of information: OpenAdap.net. Through programs like web portals and standalone applications interacting with the system, the community will share the set of tools that each of these domains typically use through members contributions.

Individual users of OpenAdap.net can be classified as either contributors of shared resources, or end-users of such. People who develop and provide new methods of data analyses are able to share their contribution and people interested in processing their own data or access data stored elsewhere (e.g. in a community database) can extract the results of their analysis. In addition to individual use, OpenAdap.net is open to exploitation by networked organisations and alliances, by providing an infrastructure with which they can share and integrate resources and develop new business solutions and opportunities.

The philosophy behind OpenAdap.net is that users privacy is as important as contributors traceability. We believe that information sent by an end-user for manipulation by a contributors resource should be anonymized, despite the fact that all the transactions are identified and that the activity is tracked like in any computer system (not to mention the web). In contrast, meta-information concerning the contributed resource like program authorship and version number should be made available to the end-user, as a mark of diligence to the contributor, but also to point the responsibilities in the information processing chain and enhance quality assessment and reproducibility. Each task submitted to the system will be returned to the submitter with a record attesting where the information travelled, which processes were applied and who was responsible for the computers and the programs.

By sharing a program and the computer running it, for example, contributors will keep complete control over their authorship as well as the source and binary codes for the software. At the same time, they will be responsible for maintaining and checking the quality of the results that the community will use and validate. From the viewpoint of contribu-

tors, OpenAdap.net makes possible the dissemination of resources, and their exposure to application and evaluation by a broader users community. The support for broader evaluation of programs and data sets is particularly important in the research arena, as is the ability of other researchers to reproduce computational results.

In a second step, OpenAdap.net will become an environment where new techniques and methodologies will gain access to a wide range of users, possibly beyond the direct community boundaries to the adjacent domains. The system allows the dissemination of resources across domains in ways that might be unanticipated by the original contributor. For example a solution developed by experts in the field of medical statistics could permeate to the field of botany, and become a key tool in the analysis of tree distributions. In this way, OpenAdap.net supports transdisciplinarity by breaking current boundaries to resource sharing.

The OpenAdap.net project is not directly aimed at the production of new methods of analysis, but the platform helps the community to aggregate their already existing tools by dynamically composing new information processing chains using the output of existing programs as the input to others, thus opening the way to cross-fertilization and serendipity.

2 Architecture

OpenAdap.net stands in the area of complexity and aims at providing a distributed environment where tools of all kinds (applications, data, knowledge, etc.) can be accessed transparently via the Internet. At this time, three architectures are used in this field: Grid, Web-services (WS), and Peer-to-Peer (P2P). The peculiar strongholds of these architectures are briefly described here and synthesized in Table 1.

Grid: *Each user has a large dataset to manipulate with one application distributed on a set of computers.* The problem addressed by the Grid is to distribute the processing of large data sets. The required application is copied to distinct computers over a network with each treating a portion of the data. The results are recomposed from all the partial results at the end of the treatment. End-users have control on both data and applications, but little information on remote execution hosts.

Web Services: *Many users exploit the same services permanently provided through a centralized authority.* Web Services provide a secured and controlled access to applications, usually to collaborators in the same company or institution. The goal is to provide distributed access to the same references, databases or applications. The system is articulated around a repository where the service interfaces are published in a self-contained form. The architecture is rather static, as services seldom change and are expected to be permanently available. End-users have no control over the applications and little information on remote execution hosts.

P2P: *Many users exchanging pieces of data in an unsupervised way.* P2P (peer-to-peer) systems address the data-sharing problem. Copies of the applications installed on end-users computers keep open connections from one computer to peers, forwarding queries and results back and forth until a match is found somewhere on the network. The ar-

	Data treatment istribution	Hardware resource allocation	Hidden execution hosts	Application sharing	Published application interface	Data sharing	Highly dynamic system	Transparent user/resource connection
Grid	×	×	×					
WS		×	×	×	×			
P2P					×	×	×	
OAN	×	×	×	×	×	×	×	×

Table 1: Comparison of OpenAdap.net with the other approaches over Internet.

chitecture is open and competing implementations coexist on a self-organized network. End-users have control over their data and information on the peer hosts. It is interesting to note that end-users tolerate incredibly poor service quality and that this architecture raises social as well as technical issues.

OpenAdap.net (OAN) falls somewhere between these three architectures exploiting several of their interesting aspects, but with the intention to address a two-way problem: To provide to a community of users in the same domain a means to interchange their resources in an open, dynamic and secured way and to provide to a community of users the access to the exploitation of information processing solutions contributed by users belonging to other communities. End-users have control over their data, but do not need to manage the resources, nor do they have complete information on remote execution hosts. Collaboration within the OpenAdap.net network allows the dynamic integration of these resources, possibly yielding new or previously unexpected composite resources. This can be summarized as follows: *Many users interchanging resources (data, applications, knowledge,) dynamically provided by interconnected domain-oriented brokers.*

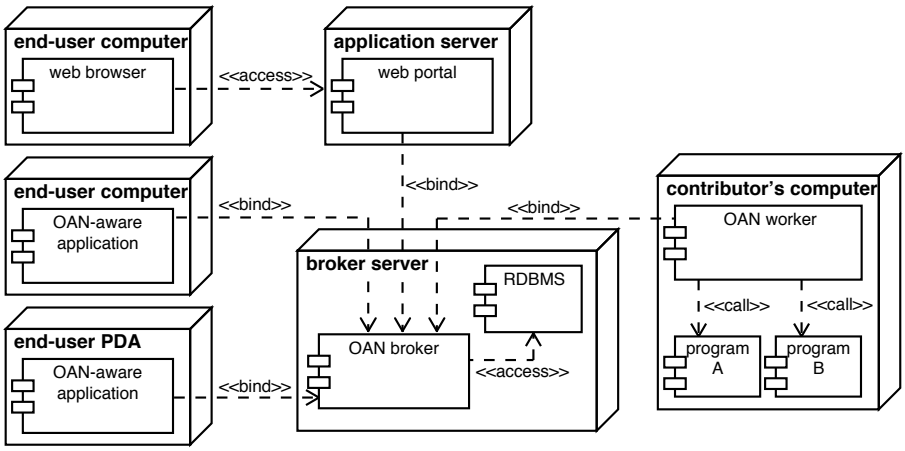


Figure 1: Deployment diagram for the OpenAdap.net components. Boxes represent different computers, rectangles represent processes, and arrows represent inter-process communications over Internet.

OpenAdap.net is a distributed system composed by three types of interconnected compo-

nents: brokers, workers and OAN-aware applications (see Figure 1). A broker is a process permanently running on a server in charge of managing a community of users and dispatching tasks and results on their behalf. Workers are processes shared by community members in charge of giving secured distant access to contributed resources like programs or data. OAN-aware applications are pieces of software (standalone applications, web portals, command line tools, etc.) providing access for an end-user to the community shared resources through identified connections to a broker.

The components are running on a network of computers, each of them defined by their specific CPU architecture, operating system (OS), amount of memory and available programs. Their resources are partially shared in a dynamic way. OpenAdap.net is designed by analogy with a virtual distributed OS in which all the resources are presented in a structured virtual file system. Using this high-level paradigm, resources are assigned to files in the file system tree. Security is enforced through ownership and access permissions defined on a per-file basis. OpenAdap.net goes beyond a traditional OS as the configuration is highly volatile. The file system structure and contents are the result of the constant runtime aggregation of several sources of information: the user database, the inter-process message queues status, the worker status, etc. A dedicated URL name scheme (`oan://`) will be proposed to identify each file in a transparent and interoperable way.

The Java 2 Platform was chosen for the implementation of the project, based on portability and platform neutrality requirements. Brokers, workers and OAN-aware applications are loosely coupled, distributed components that asynchronously communicate through a message-oriented middleware (MOM) as defined by the Java Message Service (JMS) API. The use of established industrial standards such as JMS allowed reusing existing Open Source software as a starting point for the project. It can also be expected that JMS implementations available for other platforms will allow applications written by third parties to connect to OpenAdap.net brokers and interoperate seamlessly.

Internally, brokers are responsible for decomposing and routing end-user tasks to appropriate workers for execution. A key element in the next stage of development consists in making brokers adaptive and dynamically interconnected into an OpenAdap.net network (like a neuronal network). The requests for resources will be processed and dispatched among the components of the system following a set of learning rules dynamically modifying the routing according to, for example, the computing load generated by specific tasks, availability of the resources, or the number of accesses. The rules themselves will evolve and optimize in an unsupervised manner, thus allowing the emergence of unexpected dynamics. In that sense, the required negotiation between brokers (and workers) may be compared to agent interaction. The OpenAdap.net network will also be able to self-adapt via learning processes derived from neural and artificial life learning. Such learning might result in new broker-broker connections, reassessments of the value of such connections, specialisation or generalisation or broker behaviour, etc.

The adaptive and behavioural models for the broker implementation represent major innovations of the OpenAdap.net project. This is definitely a novelty and a plus to the existing architectures for distributed environments like grids and web services that points out the project expected income to the networked computing field. We believe that SMEs and research institutions will be interested in developing novel interdisciplinary solutions

associated to the psychological and technological aspects of evolvable simulation tools, the psychological environment of remote user support and the formal aspects of artificial processing in resource-sharing.

We also expect to interface OpenAdap.net with established distributed systems like grids and clusters. For that purpose, specific workers will be developed to provide OpenAdap.net to Condor or Portable Batch System (PBS) interfaces.

Pushing existing paradigms like neuronal network inspired learning rules for the adaptable information processing or the operating system paradigm for the overall communication layout, and the lessons learned for 10 years on the self, dynamically, openly organized content on the web are key aspects of the OpenAdap.net philosophy and architecture for resource sharing. Besides, OpenAdap.net is an Open Source project designed as an open architecture. Anyone is invited to contribute their own enhancements to the system, via a set of libraries and tools provided by the consortium. Such an initiative is aimed at increasing the impact of the project with all the contributions that competent contributors will imagine and realize within their specific domains.

3 Applications

In the last few years, we have been developing, running and testing a prototype for OpenAdap.net. The concept proof and feasibility have been checked. In the last few months, we have been re-implementing the project from scratch based on the experience we have acquired with the prototype. OpenAdap.net being an Open Source project released under the GPL and LGPL licences, contributions are welcomed from developers and interested professionals. This is encouraged by the availability of publicly accessible development tools like a version control system, mailing lists and a bug tracker. There is enough experience on the impact of such coordinated and distributed development scheme on the quality of the resulting software to embrace it.

During the prototyping phase, the need appeared to have a portable and user-friendly tool that could provide a fast glimpse on the numerical output of unrelated programs. We searched for a generic front-end for plotting a wide range of graphics obtained through the web, and none could be found that was able to run on multiple platforms without requiring complicated installation procedures, and capable of producing journal-quality graphics output for free. XY-Viewer is a Java application that is being developed for that purpose as a by-product of the main project, featuring a dedicated XML data format. This standalone application can be launched directly from the OpenAdap.net portal to visualize files in the appropriate format that can be produced with the help of libraries made available to interested contributors in C and Java under the LGPL licence.

A particular exploitation of the project is the development of web portals tailored to the needs of end-user communities centred on specific domains, possibly by SMEs. Communities will appear first in the domains of competence of the consortium partners. These early adopters will constitute real-life case studies that will assess the validity of the concepts and the usability of the implementation before a broader diffusion. The prototype

has already attracted a few individuals from different research groups to cooperate and share resources in the domain of multivariate time series analysis.

In the future, we expect to give rise to synergies within and between both new and existing communities. The identification of such communities will necessitate further developments such as community specific ontologies and benchmark file repositories. Users will be invited to contribute resources, articles, benchmark data, and to build the domain ontology with the initial help of OpenAdap.net consortium.

Interactions with Publishers and Editors could have an incredible impact on the way scientific dissemination, including visualization, is performed, mainly by facilitating peer review in refereed publications and for results comparison and validation for peer scientific readers. Such interactions could be envisioned once the OpenAdap.net system will be fully available to the scientific community and should be encouraged. One major impact on the scientific community could be obtained by attracting well known scientific editors to encourage the scientific authors to provide source data to the community repositories, and to share their methods through an OpenAdap.net broker, hence promoting reproducibility of results and direct peer review of the published article methodologies and results by the readers.

Citizens of less favoured countries will have access to all shared OpenAdap.net resources with a basic Internet connection, thus benefiting from the knowledge transfer and available assets, and contributing back to the community with their own approaches and resources. The outcome of the expected cross-fertilization is unpredictable. Side effects are expected on the quality and harmonization of the resource documentation, as an important effort is dedicated to the elaboration of tools to enhance them.

4 Conclusion

This paper has presented the main features of OpenAdap.net, which is an intelligent network infrastructure supporting the use of shared resources, such as data, knowledge, tools, expertise, etc. aimed at providing the most advanced tools for data analysis and manipulation to a broad audience over Internet. Individual users can be classified as either contributors of shared resources, or end-users of such. OpenAdap.net will be open to exploitation by networked organisations and alliances taking into account the vital issue of Internet security and privacy.

The ability to tackle a scientific problem from a new perspective relies on both the past experience and new skills adopted by an individual. The OpenAdap.net project is based on the collaboration between information scientists, electronic engineers, computer scientists and neuroscientists having diverse scientific interests and very specialized backgrounds. We feel that such a transdisciplinary approach is a necessary way for the achievement of real advances in producing impacts in the Information Society Technologies.

The OpenAdap.net infrastructure makes possible the dissemination of resources, and their exposure to application and evaluation across domains in ways that might be unanticipated. Simulation processing tools issued from physical sciences could permeate to study

problems as different as the dynamics of societal interactions, linguistic analyses, crops forecast, traffic congestions, and life sciences. OpenAdap.net is aimed at breaking current boundaries to resource sharing and hence supports transdisciplinarity. End-users are provided with the ability to browse and apply shared resources, and dynamically compose and integrate existing resources to leverage new research insights.

OpenAdap.net brokers are responsible for dynamically decomposing and routing end-user tasks to appropriate resource sharers for execution. The negotiation between brokers (and workers) is inspired by the way how the brain processes information. When completed, the OpenAdap.net project network will be able to self-adapt via learning processes that could give rise to modifiable broker-broker connections, specialisation or generalisation of broker behaviour, etc. The nonlinear dynamics that will emerge from our approach makes OpenAdap.net closer to the complexity of a living organism.

Acknowledgments

This work is partially funded by the European research project SECOQC.

References

[EBS06] G. Eisenhauser, F.E. Bustamante, and K. Schwan. Publish-subscribe for high-performance computing. *IEEE Internet Computing*, 10(1):407, 2006.

[GH06] S. Ghernaouti-Hélie. Guide de cybersecurité pour les pays en développement. ITU publication, 2006.

[GHSRG05] S. Ghernaouti-Hélie, M.A. Sfaxi, G. Ribordy, and O. Gay. Using Quantum Key Distribution within IPSEC to secure MAN communications. In *MAN conference*, 2005.

[Lae05] K. Laeufer. A hike through post-EJB J2EE web application architecture. *Computing in Science and Engineering*, 7(5):808, 2005.

[VTI01] A.E.P. Villa, I.V. Tetko, and J. Iglesias. Computer Assisted Neurophysiological Analysis of Cell Assemblies. *Neurocomputing*, 38-40:1025–1030, 2001.

[ZCS05] X. Zhang, S. Chen, and R. Sandhu. Enhancing Data Authenticity and Integrity in P2P systems. *CIEEE Internet computing*, 9(6):429, 2005.

Chapter 6: Security and Theoretic Approaches

Contributions to 10th I2CS 2010, Bangkok, Thailand

Sheikh Ziauddin
An Improved Hwang-Lee-Tang Remote User Authentication Scheme

Dejvuth Suwimonteerabuth
Computing Minimum-Height Certificate Trees in SPKI/SDSI

Sirapat Boonkrong
Some Remarks on Andrew Secure RPC

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Duc Kien Nguyen, Ivan Lavallee, Marc Bui
Generalizing of a High Performance Parallel Strassen Implementation on Distributed Memory MIMD Architectures

Roberto Gómez, Gabriel Ramírez
Using Digital Images to spread Executable Code on Internet

Marco Aurelio Turrubiartes Reynaga, Orlando Ezequiel Rincón Ferrera, Leopoldo Estrada Vargas, Deni Torres Román, David Muñoz Rodríguez, Marlenne Angulo Bernal, Luis Rizo Domínguez
Characterization and Generation of Synthetic Data Traces for IP Traffic Modeling

An Improved Hwang-Lee-Tang Remote User Authentication Scheme

Sheikh Ziauddin
Department of Computer Science
COMSATS Institute of Information Technology, Islamabad, Pakistan
email:sh.ziauddin@gmail.com

Abstract: In this paper, we present a secure and efficient remote authentication scheme by improving Hwang-Lee-Tang's scheme. The security of our scheme is based on the onewayness and collision-resistance properties of the hash functions being used. The proposed scheme is able to withstand all commonly known attacks against remote authentication schemes. In addition, the scheme does not store a password table on the server, provides mutual authentication between the user and the server, does not reveal user's password to the server, allows the user to freely choose a password of her choice, and allows the user to change her password by running a simple protocol with the server.

1 Introduction

The classic technique for remote authentication is based on users' passwords. With the passage of time, it has been realized that the use of password alone is not enough from the security point of view because the typical human-selected passwords have low entropy. Therefore, in modern times, many attempts have been made to build two-factor secure remote authentication systems by combining the passwords and the smart cards [CW93, CH93, HL00, Sun00, LHY02, HLT02, CJT02, SLH03a, AL03, CLH04, LKY04, YRY05, LC05].

Lamport [Lam81] was the first one to propose a remote user authentication scheme. In his scheme, a table of passwords is maintained on the server for users' verification. The major drawback of his scheme is that if the server is compromised, the secret passwords of all the users are disclosed. Subsequently, many password authentication schemes have been presented that do not rely on verification tables stored on the server. Sun [Sun00] presented a two-factor authentication scheme using a password and a smart card. Unfortunately, their scheme provides only uni-directional authentication from the user to the server. In addition, the user's password is known to the server and the scheme does not allow the user to change her password. Hwang, Lee, and Tang [HLT02] presented a scheme that allows for user's password change but no verification check is conducted before committing the password change. Chien et al. [CJT02] proposed another password authentication scheme using hash functions. Their scheme also suffers from the problems of password being known to the server and having no password change option. Lee et al. [LKY04] and

Yoon et al. [YRY05] later presented their respective schemes but they also suffer from the problem of user's password revealed to the server. In addition, in [LKY04], no verification check is performed before password change and in [YRY05], it is easy to retrieve the password if the smart card is stolen.

In addition to the above hash function-based schemes, another research direction is to use public key cryptography for remote authentication [YS99, FLZ02, SLH03b, SLH03a, HL00, AL03]. The main disadvantage of public key schemes is their high computational cost which makes them unsuitable for many practical applications.

In this paper, we present a remote authentication scheme which is constructed from hash functions (e.g., SHA-256). At registration time, the user sends hash of her password to the server and receives a smart card containing some information generated from a combination of the user's password and the server's secret key. At authentication time, the user uses her password and smart card to generate user-to-server messages while the server uses its secret key to generate server-to-user messages. If the messages are successfully validated by the receiving entities, mutual authentication is carried out between the user and the server.

The rest of the paper is organized as follows. In Section 2, we present the assumptions and the threat model for the scheme. Section 3 describes Hwang et al.'s scheme and its weaknesses. In Section 4, we describe the working of the proposed scheme. The security of our scheme is analyzed in Section 5, and finally, we conclude the paper in Section 6.

2 Adversarial Model

In this section, we outline the assumptions made about the proposed scheme and describe the capabilities of the adversary against the scheme. Major points of our model are as follows.

- The user and the server participate honestly in the protocol.
- The adversary cannot steal the server's secret.
- The adversary can steal either the user's password or the smart card, but not both.
- The user and the server use synchronized clocks or they have access to a common trusted time server to get the current time.
- A secure and authenticated channel exists between the communicating parties during the registration phase.
- During authentication and password change phases, the communication takes place in a completely adversarially controlled channel.
- The smart card is tamper resistant. The data can be overwritten but only through a provided interface, e.g., the one used in the password change phase of the proposed scheme.

- Once a smart card is stolen, all the stored information can be extracted by the adversary, e.g., by using reverse engineering techniques.

3 Hwang-Lee-Tang's Scheme

Notation

First we describe the notation that we will use in this paper to denote the elements of both Hwang et al.'s scheme and the proposed scheme. We use ID to denote the identity of the user in a format suitable for the specific application. PW and k denote the user's password and the server's secret key, respectively. We use SC to denote the smart card issued by the server to the user. $\oplus$ denotes an exclusive-or operation. We use $\mathsf{h}(\cdot)$ to denote description of the cryptographically secure hash function being used in the protocol. T, T_1, T_1', T_2, T_2' represent timestamps at different times and Δt denotes the maximum allowed network delay time for a single message passed between the user and the server. PCR denotes a special message in a specific format which we name *password change request*. This message is part of the communication during password change phase only and serves to differentiate between the authentication requests and the password change requests.

3.1 Description of Hwang-Lee-Tang's Scheme

Our scheme is based on Hwang-Lee-Tang [HLT02] scheme. In this section, we briefly describe their scheme and point out its weaknesses. The scheme has three phases: registration, authentication, and password change.

3.1.1 Registration Phase

The user sends the hash of her password $\mathsf{h}(PW)$ and her ID to the server. The server receives the message, calculates $A = \mathsf{h}(ID \oplus k) \oplus \mathsf{h}(PW)$, and personalizes a smart card to the user containing the values $\mathsf{h}(\cdot), ID,$ and A.

3.1.2 Authentication Phase

The user (smart card) calculates $B = A \oplus \mathsf{h}(PW)$ using her password PW and the value A stored on the smart card, gets the current timestamp T, calculates $C = \mathsf{h}(B \oplus T)$, and sends (ID, C, T) to the server. After receiving the message, the server verifies the format of ID and the validity of T. It then calculates $B' = \mathsf{h}(ID \oplus k)$ and $C' = \mathsf{h}(B' \oplus T)$, and verifies that $C \stackrel{?}{=} C'$. If the verification is successful, the user's authentication request is accepted.

3.1.3 Password Change Phase

The user (smart card) calculates $B = A \oplus \mathsf{h}(PW)$ using her old password PW. She next calculates $A' = B \oplus \mathsf{h}(PW')$ using her new password PW'. Stored A on the smart card is then replaced with A'.

3.2 Weaknesses

One weakness of the scheme is its insecure password change phase. Consider an adversary who steals the smart card of a user. The adversary gives any arbitrary password PW' and calculates $B = A \oplus \mathsf{h}(PW')$. The adversary then selects a new password PW'' and calculates $A' = B \oplus \mathsf{h}(PW'')$. Next, stored A on the smart card is replaced with A' without any verification. This shows that it is easy for an adversary to change the password of any user without knowing the original password. Using this new password, the adversary can impersonate the user in the protocol as he possesses both the secrets now.

Another necessary security requirement missing in their scheme is the ability to provide mutual authentication. Mutual authentication is necessary in most remote authentication systems such as those used for electronic commerce where the users want to make sure that they are communicating with the legitimate server before committing any financial transaction.

In next section, we present our scheme which removes the above-mentioned flaws from their scheme. In addition, our scheme provides many other desirable features.

4 Proposed Scheme

4.1 Registration Phase

The registration phase of our scheme is the same as that of Hwang-Lee-Tang's scheme. During this phase, the following steps are carried out.

1. The user freely selects a password PW of her choice along with an arbitrary unique ID and sends her ID along with the hash of her password $\mathsf{h}(PW)$ to the server.
2. After receiving the user's message, the server calculates $A = \mathsf{h}(ID \oplus k) \oplus \mathsf{h}(PW)$ and issues a smart card to the user that contains the values $\mathsf{h}(\cdot), ID,$ and A.

Figure 1 schematically describes the registration phase of the proposed scheme.

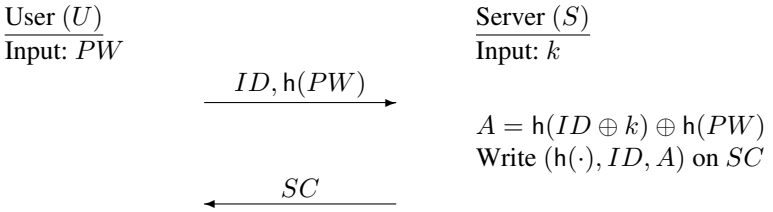


Figure 1: Registration phase of the proposed scheme (Same as that of Hwang-Lee-Tang)

4.2 Authentication Phase

During this phase, the user inserts her smart card in a card reader, keys in her password on a terminal, and then the user (the smart card) and the server communicate with each other for some time. At the end of a successful communication, they authenticate each other. The user's secrets are her password PW and the smart card while the server is in possession of its secret key k. During this phase, the following steps are carried out.

1. The user calculates a hash of her password $\mathsf{h}(PW)$, reads the value A from the smart card, and XORs them to get a value B. Next, she gets the current timestamp T_1, calculates $C_1 = \mathsf{h}(B \oplus T_1)$ and sends C_1, T_1 along with her ID to the server.

2. After receiving the user message, the server verifies the format of user's ID. Next, the server gets the current timestamp T_1' and verifies that $T_1' - T_1$ does not exceed ΔT. It next calculates $B' = \mathsf{h}(ID \oplus k)$ and $C_1' = \mathsf{h}(B' \oplus T_1)$ and verifies that C_1 and C_1' are equal. If any of the verifications described above fail, the request is rejected. Otherwise the request is accepted, i.e., the user is successfully authenticated. Next, the server gets the current timestamp T_2, calculates $C_2 = \mathsf{h}(B' \oplus T_2)$ and sends C_2 and T_2 to the user.

3. After receiving the server message, the user gets the current timestamp T_2' and verifies that $T_2' - T_2$ does not exceed ΔT. The user calculates $C_2' = \mathsf{h}(B \oplus T_2)$ and verifies that C_2 and C_2' are equal. If any of the verifications described above fail, the request is rejected. Otherwise the request is accepted, i.e., the server is successfully authenticated.

If both steps 2 and 3 are successful, this indicates a successful mutual authentication being carried out. Figure 2 schematically describes the authentication phase of the proposed scheme.

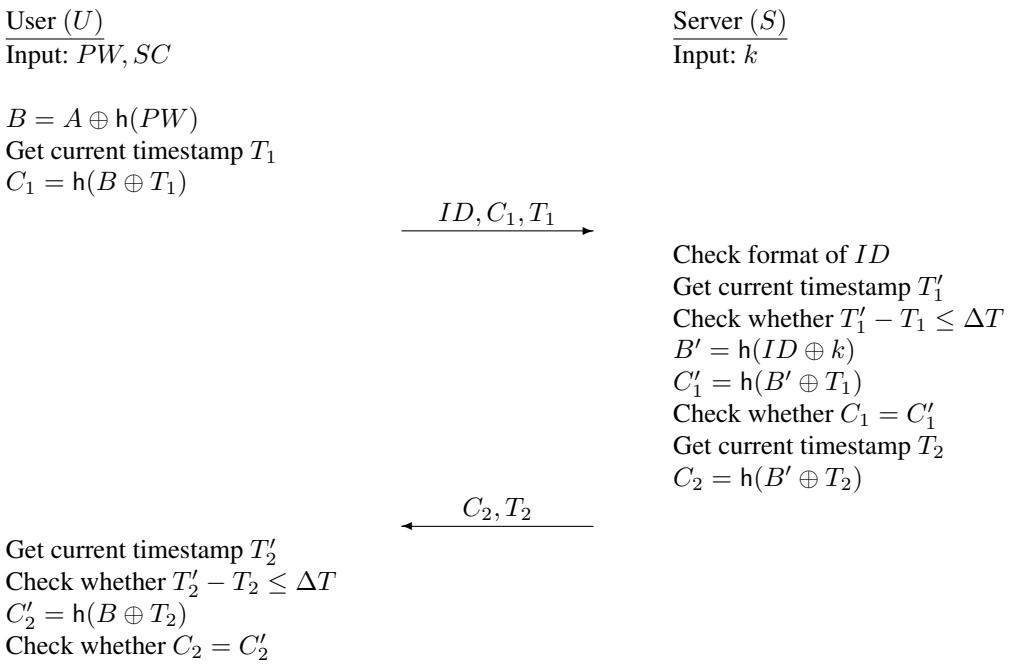


Figure 2: Authentication phase of the proposed scheme

4.3 Password Change Phase

Similar to authentication phase, the user inserts her smart card in a card reader, keys in her password on a terminal, and then the user (the smart card) and the server communicate with each other for some time. At the end of a successful communication, the user changes her password. The user's secrets are her existing password PW and SC while the server has its secret k. During this phase, the following steps are carried out.

1. The user calculates a hash of her password $\mathsf{h}(PW)$, reads the value A from the smart card, and XORs them to get a value B. She gets the current timestamp T_1, calculates $C_1 = \mathsf{h}(B \oplus T_1 \oplus PCR)$ and sends C_1, T_1 and PCR along with her ID to the server.

2. After receiving the user message, the server verifies the formats of ID and PCR. It gets the current timestamp T_1' and verifies that $T_1' - T_1$ does not exceed ΔT. The server next calculates $B' = \mathsf{h}(ID \oplus k)$ and $C_1' = \mathsf{h}(B' \oplus T_1 \oplus PCR)$ and verifies that C_1 and C_1' are equal. If any of the above verifications fail, the password change request is rejected. Otherwise, the request is accepted and the server gets the current timestamp T_2, calculates $C_2 = \mathsf{h}(B' \oplus T_2 \oplus PCR)$ and sends C_2 and T_2 to the user.

3. After receiving the server message, the user gets the current timestamp T_2' and veri-

fies that $T_2' - T_2$ does not exceed ΔT. She next calculates $C_2' = \mathsf{h}(B \oplus T_2 \oplus PCR)$ and verifies that C_2 and C_2' are equal. If any of the above verifications fail, the request is rejected. Otherwise the user calculates $A' = B \oplus \mathsf{h}(PW')$ where PW' is the new password of the user and the value A on smart card is replaced with A'.

Figure 3 schematically describes the password change phase of the proposed scheme.

User (U)		Server (S)
Input: PW, SC		Input: k
$B = A \oplus \mathsf{h}(PW)$		
Get current timestamp T_1		
$C_1 = \mathsf{h}(B \oplus T_1 \oplus PCR)$		
	$\xrightarrow{ID, C_1, T_1, PCR}$	
		Check format of ID
		Check format of PCR
		Get current timestamp T_1'
		Check whether $T_1' - T_1 \leq \Delta T$
		$B' = \mathsf{h}(ID \oplus k)$
		$C_1' = \mathsf{h}(B' \oplus T_1 \oplus PCR)$
		Check whether $C_1 = C_1'$
		Get current timestamp T_2
		$C_2 = \mathsf{h}(B' \oplus T_2 \oplus PCR)$
	$\xleftarrow{C_2, T_2, PCR}$	
Get current timestamp T_2'		
Check whether $T_2' - T_2 \leq \Delta T$		
$C_2' = \mathsf{h}(B \oplus T_2 \oplus PCR)$		
Check whether $C_2 = C_2'$		
$A' = B \oplus \mathsf{h}(PW')$		
Replace A on SC with A'		

Figure 3: Password change phase of the proposed scheme

5 Security Analysis

In absence of a common set of necessary and sufficient security requirements for smart card-based password authentication schemes, we analyze the security of our scheme against an extensive set of security requirements that we consider to be necessary for a two-factor remote authentication scheme.

Replay attack. This attack is not feasible against our scheme. To see this, consider an adversary trying to replay a message (ID, C_1, T_1) sent from the client to the server during

the authentication phase. Clearly, this attack will be detected in Step 2 by the server. Also note that the adversary cannot replace T_1 by a newer time because he cannot generate a valid C_1 for that time without stealing the smart card and knowing the password. Further note that it is highly unlikely for the adversary to get the same C_1 for time $\hat{T} \neq T_1$ due to collision-resistance property of the hash function. The same logic applies for replaying a message from the server to the client.

User impersonation. To impersonate a user, the adversary has to fabricate a valid message (ID, C_1, T_1). As mentioned above, it is not feasible to find C_1 without stealing both the password and the smart card. Also it is not feasible to recover the password (or the value A) from C_1 due to onewayness of hash function.

Server impersonation/Server spoofing. To impersonate the server, the adversary has to fabricate a valid message (C_2, T_2). It is not feasible to find C_2 without stealing the server's secret k. Also it is not feasible to recover k from C_2 due to onewayness of hash function.

Stolen verifier attack. This attack is not possible against our system as the server does not maintain any verification table for users' passwords. Instead, the data needed for verification is stored on the users' smart cards.

Password guessing attack. The password guessing attack is not feasible against our scheme. To see this, first note that the hash of the password is never transmitted over the channel. Next, consider an adversary which intercepts and stores a message (ID, C_1, T_1). There is no way for the adversary to verify the correctness of his password guesses because the value C_1 is a function of not only the password but also of the value A stored on the smart card. This makes it impossible to verify the correctness of a guess without stealing the smart card as well. We point out that password guessing using a compromised smart card is not a valid attack because, in two-factor schemes (using passwords and smart cards), there is no way to stop an adversary from carrying out such an attack. The attack can be countered by replacing low entropy passwords with high entropy secrets such as passphrases or biometrics. Note that this change does not make the above attack unsuccessful, rather it only increases the time complexity of the attack.

Stolen smart card attack. The only secret stored on the smart card is $A = \mathsf{h}(ID \oplus k) \oplus \mathsf{h}(PW)$. Clearly it is not feasible to find either PW or k from A without breaking the onewayness of the cryptographic hash function involved. Also note that the adversary can neither fabricate (ID, C_1, T_1) nor (C_2, T_2) without knowing either the user's password or the server's secret in addition to the smart card data.

Stolen password attack. Stealing the password will not help the adversary. To fabricate a message (ID, C_1, T_1), the adversary has to find $B = A \oplus \mathsf{h}(PW)$. As the adversary already knows $\mathsf{h}(PW)$, clearly the entropy of B is equal to the entropy of A . As A is generated by applying a hash function on the server secret k, it is neither feasible to predict k due to its high entropy, nor it is feasible to find a $\hat{k} \neq k$ such that $A = \mathsf{h}(ID \oplus \hat{k})$ due to the collision-resistance of the hash function.

6 Conclusions

In this paper, we presented a remote mutual authentication scheme. We used cryptographic hash functions as building blocks of our scheme. We presented a thorough security analysis and showed that the proposed scheme is able to withstand many attacks against remote authentication schemes. The proposed scheme does not store a password table on the server, provides mutual authentication and allows the user to easily change her password. The scheme is also efficient as it uses just a few hash operations only.

References

[AL03] A. K. Awasthi and S. Lal. A remote user authentication scheme using smart cards with forward secrecy. *IEEE Transactions on Consumer Electronics*, 49(4):1246–1248, 2003.

[CH93] C. Chang and S. Hwang. Using smart cards to authenticate remote passwords. *Computers and Mathematics with Applications*, 26(7):19–27, 1993.

[CJT02] H. Chien, J. Jan, and Y. Tseng. An efficient and practical solution to remote authentication: Smart card. *Computers and Security*, 21(4):372–375, 2002.

[CLH04] Tzungher Chen, Wei-Bin Lee, and Gwoboa Horng. Secure SAS-like password authentication schemes. *Computer Standards & Interfaces*, 27(1):25–31, 2004.

[CW93] C. Chang and T. Wu. Remote password authentication with smart cards. *IEE Proceedings-E*, 138(3):165–168, 1993.

[FLZ02] L. Fan, J.H. Li, and H.W. Zhu. An enhancement of timestamp-based password authentication scheme. *Computers & Security*, 21(7):665–667, 2002.

[HL00] M. Hwang and L. Li. A new remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, 46(1):28–30, 2000.

[HLT02] M.S. Hwang, C.C. Lee, and Y.L. Tang. A simple remote user authentication scheme. *Mathematical and Computer Modelling*, 36(1):103–107, 2002.

[Lam81] Leslie Lamport. Password Authentication with Insecure Communication. *Communications of the ACM*, 24(11):770–772, 1981.

[LC05] N. Lee and Y. Chiu. Improved remote authentication scheme with smart card. *Computer Standards and Interfaces*, 27(2):177–180, 2005.

[LHY02] C. Lee, M. Hwang, and W. Yang. A flexible remote user authentication scheme using smart cards. *Operating Systems Review (ACM)*, 36(3):46–51, 2002.

[LKY04] S. W. Lee, H. S. Kim, and K. Y. Yoo. Improved efficient remote user authentication scheme using smart cards. *IEEE Transactions on Communications*, 50(2):565–567, 2004.

[SLH03a] J. Shen, C. Lin, and M. Hwang. A modified remote user authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, 49(2):414–416, 2003.

[SLH03b] J.J. Shen, C.W. Lin, and M.S. Hwang. Security enhancement for the timestamp-based password authentication scheme using smart cards. *Computers & Security*, 22(7):591–595, 2003.

[Sun00] H. Sun. An efficient remote use authentication scheme using smart cards. *IEEE Transactions on Consumer Electronics*, 46(4):958–961, 2000.

[YRY05] E. J. Yoon, E. K. Ryu, and K. Y. Yoo. An improvement of Hwang-Lee-Tangs simple remote user authentication schemes. *Computers & Security*, 24(1):50–56, 2005.

[YS99] W.H. Yang and S.P. Shieh. Password authentication schemes with smart cards. *Computers & Security*, 18(8):727–733, 1999.

Computing Minimum-Height Certificate Trees in SPKI/SDSI

Dejvuth Suwimonteerabuth

The Sirindhorn International Thai-German Graduate School of Engineering (TGGS)
dejvuth.s.sse@tggs-bangkok.org

Abstract: SPKI/SDSI is a framework that combines a simple public-key infrastructure and a simple distributed security infrastructure with a means of defining local name spaces. It allows principals, which can be a person or an organization, to locally create groups of principals and delegate rights to other principals or groups of principals by issuing certificates. To prove authorizations, principals need to search for necessary certificates that are, in general, in the form of certificate trees. This paper defines a framework based on SPKI/SDSI which allows principals to give weights to certificates. Weights can be used to address many authorization issues such as access control of limited resources. The paper shows a connection between SPKI/SDSI and the theory of pushdown systems, and presents an algorithm that solves the authorization problem by computing minimum-height certificate trees.

1 Introduction

In access control of shared resources, authorization systems allow to specify a security policy that assigns permissions to principals in the system. The *authorization problem* is, given a security policy, should a principal be allowed access to a given resource? SPKI/SDSI [EFL+99] is a framework which allows a principal to locally create groups of principals by issuing so-called *name certificates*, and grant authorizations or delegate the right to grant authorizations to other principals or groups of principals (even without knowing individuals in the groups) by issuing so-called *authorization certificates*. In [CEE+01], it has been shown that the authorization problem reduces to discovering a *certificate chain* to prove whether a given principal is allowed to access a given resource. The certificate chain might consists of one or more name definitions or authorization grants and delegations.

In general, however, a principal might need to find more than one certificate chain to prove his/her authorization. SPKI/SDSI allows, for instance, Alice to issue a certificate to give an authorization to her relatives who work in her company. Therefore, if Bob wants to prove his authorization, he must find a set of certificates which proves (i) that he is her relative *and* (ii) that he works in her company. This set of certificates forms a *certificate tree* in which each branch represents a certificate chain.

In this paper, we consider a more general system where certificates can have different

weights. The meaning of weights depends on applications of interest. When proving an authorization, a principal does not look for an arbitrary certificate tree, but the tree that has the minimum height, i.e., the tree that involves certificates having the smallest weights. This system can be applied to many applications. For instance, different weights can be interpreted as different degrees of importance. When principals compete for a limited resource, one can control who has the right to access it based on the importance of his/her certificate tree.

Previous works have shown that SPKI/SDSI has a strong connection to the theory of pushdown systems [JR04, SJRS03]. A set of certificates can be seen as a pushdown system, and certificate-chain discovery reduces to pushdown reachability. This paper proceeds in a similar way. We propose an extension to SPKI/SDSI in which one can assign weights to certificates. Then, we present an efficient algorithm for finding minimum-height certificate trees.

We proceed as follows: Section 2 introduces SPKI/SDSI and formally defines the authorization problem. Section 3 presents alternating pushdown systems and other theoretical concepts used in the paper. Section 4 shows the connection between SPKI/SDSI and alternating pushdown systems, and presents an algorithm for solving the authorization problem. Section 5 concludes the paper.

Throughout the paper, we denote by $\mathbb{N}$ the set of non-negative integers and $\mathbb{N}^\infty = \mathbb{N} \cup \{\infty\}$. If n is a positive integer, then $[n] = \{i \in \mathbb{N} \mid 1 \leq i \leq n\}$.

2 SPKI/SDSI

A central notion of SPKI/SDSI are *principals*. A principal can be a person or an organization represented by a public key. Each principal defines his/her own namespace, which assigns *rôles* to (other) principals. For instance, principal *University* can define the rôle `staff` and associate principal `Alice` with its rôle. SPKI/SDSI makes such associations by issuing so-called *name certificates* (*name certs*, for short). Remarkably, principals may reference the namespace of other principals in their certificates. For instance, *University* may state that all *Engineering*'s staffs are also its staffs. In this way, SPKI/SDSI allows to associate a rôle with a group of principals described in a symbolic and distributed manner. SPKI/SDSI then allows to assign permissions to rôles using so-called *authorization certificates* (or *auth certs*).

More formally, a SPKI/SDSI system can be seen as a tuple $S = (P, A, C)$, where P is a set of *principals* (or public keys), A is a set of *rôle identifiers* (or identifiers, for short), and $C = Na \uplus Au$ is a set of certificates. Certificates can be either *name certs* (contained in Na), or *auth certs* (contained in Au).

A *term* is formed by a principal followed by zero or more identifiers, i.e., an element of the set PA^*. A name certificate is of the form $p\ \mathtt{a} \rightarrow t$, where p is a principal, a is an identifier, and t is a term. Notice that p `a` itself is a term. For all terms t, the sets $[\![t]\!]$ are the smallest sets of principals satisfying the following constraints:

- if $t = p$ for some principal p, then $[\![t]\!] = \{p\}$;
- if $t = t'$ a, then for all $p \in [\![t']\!]$ we have $[\![p\ \mathtt{a}]\!] \subseteq [\![t]\!]$;
- if p a $\rightarrow t$ is name cert, then $[\![t]\!] \subseteq [\![p\ \mathtt{a}]\!]$.

For example, if *University*, *Engineering*, *Alice* are principals and `staff` is an identifier, then the certificate c_1 : *Engineering* `staff` $\rightarrow$ *Alice* expresses that Alice is an Engineering's staff, and the certificate c_2 : *University* `staff` $\rightarrow$ *Engineering* `staff` means that all Engineering's staffs are also staffs of the university.

An auth cert is of the form $p\ \square \rightarrow t\ b$, where p is a principal, t is a term, and b is either $\square$ or $\blacksquare$. It means that p grants some authorization to all principals in $[\![t]\!]$. If $b = \square$, then the principals in $[\![t]\!]$ are allowed to delegate the authorization to others; if $b = \blacksquare$, then they are not. Note that auth certs can also contain details about the authorization that they confer. We omit this detail in this paper due to the space constraint.

Formally, auth certs define a smallest relation $aut : P \times P$ between principals such that $aut(p, p')$ holds iff p grants an authorization to p':

- if there is an auth cert $p\ \square \rightarrow t\ b$, for $b \in \{\square, \blacksquare\}$, and $p' \in [\![t]\!]$, then $aut(p, p')$;
- if there is an auth cert $p\ \square \rightarrow t\ \square$, $p' \in [\![t]\!]$, and $aut(p', p'')$, then $aut(p, p'')$.

For example, the certificate c_3 : *University* $\square \rightarrow$ *University* `staff` $\blacksquare$ means that the university grants some right to all university's staffs. They, however, are not allowed to delegate that right to other principals.

The *authorization problem* in SPKI/SDSI is to determine, given a system (P, A, C) and two principals p and p', whether p' is granted authorization by p, i.e., whether $aut(p, p')$. The problem can be solved by finding a *certificate chain* that transforms $p\ \square$ into $p'\ \square$ or $p'\ \blacksquare$. In the example, if Alice wants to prove that she has the right from the university, she needs to find the chain c_3, c_2, c_1, which gives the following proof:

$$\textit{University}\ \square \xrightarrow{c_3} \textit{University}\ \mathtt{staff}\ \blacksquare \xrightarrow{c_2} \textit{Engineering}\ \mathtt{staff}\ \blacksquare \xrightarrow{c_1} \textit{Alice}\ \blacksquare$$

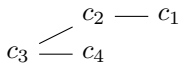
2.1 Intersection certificates

The SPKI/SDSI standard provides for so-called *threshold certificates*, which consists of, say, an auth cert of the form $p\ \square \rightarrow \{t_1\ b_1, \ldots, t_n\ b_n\}$, where $b_1, \ldots, b_n \in \{\square, \blacksquare\}$, and an integer $k \leq n$. The meaning of such a cert is that p grants authorization to principal p' if there is a certificate chain to p' from at least k out of $t_1\ b_1, \ldots, t_n\ b_n$. We restrict ourselves to the case where $k = n$ and use the more suggestive name *intersection certificate* instead. Formally, intersection certificates extend the relation aut as follows:

- if $p\ \square \rightarrow \{t_1\ b_1, \ldots, t_n\ b_n\}$, then $aut(p, p')$, where $p' \in \bigcap_{i=1}^{n} [\![t_i]\!]$; moreover,
- if $b_j = \square$, $p' \in [\![t_j]\!]$, and $aut(p', p'')$ such that $p'' \in \bigcap_{i=1}^{n} [\![t_i]\!]$, then $aut(p, p'')$.

Notice that one could analogously define threshold name certificates in a similar way. However, in [CEE$^+$01, JR04] the use of threshold certificates is restricted to just authorization certificates, claiming that the use of threshold certificates in name certificates would make the semantics "almost surely too convoluted".

In the presence of intersection certificates, proofs of authorizations are in the form of certificate trees, where each branch corresponds to a certificate chain. Continuing the example, if the university instead delegates the right to Bob to grant the authorization to the staffs, we have c_3: $\mathit{University}\ \square \rightarrow \{\mathit{University}\ \texttt{staff}\ \blacksquare, \mathit{Bob}\ \square\}$ and if Bob grants the authorization to Alice c_4: $\mathit{Bob}\ \square \rightarrow \mathit{Alice}\ \blacksquare$, the following certificate tree proves Alice's authorization:



2.2 Min SPKI/SDSI

We extend SPKI/SDSI by assigning weights to certificates. A *min SPKI/SDSI* system is a tuple (S, f), where $S = (P, A, C)$ is a SPKI/SDSI system and $f : C \rightarrow \mathbb{N}^\infty$ is a function that assigns a natural number to each rule. We extend the function f to every node c in certificate trees to signify the height of the node: if c is a leaf, its height is $f(c)$; otherwise, its height is $f(c) + \max_{i=1}^{n} f(c_i)$, where c_i is a child node of c for all $i \in [n]$. The height of a certificate tree is the height of the root.

In the example, if the university issues a similar certificate for Carol c_5: $\mathit{University}\ \square \rightarrow \{\mathit{University}\ \texttt{staff}\ \blacksquare, \mathit{Carol}\ \square\}$, and gives this delegation more priority by assigning, say, $f(c_3) = 3$, $f(c_5) = 1$, and $f(c) = 0$ for any other c. Therefore, assuming that there is a chain from $\mathit{Carol}\ \square$ to $\mathit{Alice}\ \blacksquare$, she would prefer the certificate tree where the certificate c_5 is the root, since it gives her more priority than the previous tree.

The rest of the paper deals with the problem of finding minimum-height certificate trees by using the theory of pushdown systems.

3 Pushdown systems

An *alternating pushdown system* (APDS) is a triplet $\mathcal{P} = (P, \Gamma, \Delta)$, where P is a finite set of *control locations*, Γ is a finite *stack alphabet*, and $\Delta \subseteq (P \times \Gamma) \times 2^{(P \times \Gamma^*)}$ is a set of *transition rules*. A *configuration* of $\mathcal{P}$ is a pair $\langle p, w \rangle$, where $p \in P$ is a control location and $w \in \Gamma^*$ is a *stack content*. If $((p, \gamma), \{(p_1, w_1), \ldots, (p_n, w_n)\}) \in \Delta$, we write $\langle p, \gamma \rangle \hookrightarrow \{\langle p_1, w_1 \rangle, \ldots, \langle p_n, w_n \rangle\}$ instead. If $n = 1$, we write $\langle p, \gamma \rangle \hookrightarrow \langle p_1, w_1 \rangle$ (braces omitted), and call the rule *non-alternating*. We call $\mathcal{P}$ a *pushdown system* (PDS) if Δ consists only of non-alternating rules.

A *min APDS* is an APDS, in which each rule is equipped with a *weight* which is a natural number; formally, $\mathcal{M} = (\mathcal{P}, f)$, where $\mathcal{P} = (P, \Gamma, \Delta)$ is an APDS and $f : \Delta \rightarrow \mathbb{N}^\infty$ is a function that assigns a value from $\mathbb{N}^\infty$ to each rule in Δ. If $f(\langle p, \gamma \rangle \hookrightarrow$

$\{\langle p_1, w_1\rangle, \ldots, \langle p_n, w_n\rangle\}) = a$, we often write $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \{\langle p_1, w_1\rangle, \ldots, \langle p_n, w_n\rangle\}$. We sometimes use the term APDS to refer to its min version when it is clear from the context.

Intuitively, a rule $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \{\langle p_1, w_1\rangle, \ldots, \langle p_n, w_n\rangle\}$ says that, from a configuration c where p is the control location and γ is the top of stack symbol, the computation of the system forks into n parallel computations, each of them starting from the configuration obtained from c by replacing p by p_i and γ by w_i, for all $i \in [n]$. Therefore, a run can be seen as a tree of computations. The height of a run is computed from the weights corresponding to the transition rules by applying $+$ between successive weights and $\max$ on the parallel ones.

Formally, we define the reachability relation $\Rightarrow \subseteq (P \times \Gamma^*) \times \mathbb{N}^\infty \times 2^{P\times\Gamma^*}$ to be the smallest relation such that

- $c \overset{0}{\Rightarrow} \{c\}$, for all $c \in P \times \Gamma^*$,
- if $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \{\langle p_1, w_1\rangle, \ldots, \langle p_n, w_n\rangle\}$ and $\langle p_i, w_i w\rangle \overset{b_i}{\Rightarrow} C_i$ for some $w \in \Gamma^*$, $b_i \in \mathbb{N}^\infty$, and $C_i \subseteq P \times \Gamma^*$, for each $i \in [n]$, then $\langle p, \gamma w\rangle \overset{a+\max_{i=1}^n\{b_i\}}{\Longrightarrow} \bigcup_{i=1}^n C_i$.

Given a configuration c and a set of configurations C, we define

$$H(c, C) = \min\{a \in \mathbb{N}^\infty \mid c \overset{a}{\Rightarrow} C\}$$

to be the minimum height of all runs starting from c and reaching (precisely) the set C. The *reachability problem* is to determine whether $T(c, C) < \infty$. If $T(c, C) < \infty$, we say that c is *backwards reachable* from C. We define $pre^*(C) = \{c \in P \times \Gamma^* \mid \exists C' \subseteq C : T(c, C') < \infty\}$ to be the set of all configurations that are backwards reachable from C.

An APDS is called *simple* if there is a set of bottom stack symbols $\Xi \subseteq \Gamma$, and all transition rules in Δ are in the following forms:

- $\langle p, \gamma\rangle \hookrightarrow \langle p', w\rangle$, where $p, p' \in P$, $\gamma \in \Gamma \setminus \Xi$, and $w \in (\Gamma \setminus \Xi)^*$,
- $\langle p, \bot\rangle \hookrightarrow \{\langle p_1, w_1\bot_1\rangle, \ldots, \langle p_n, w_n\bot_n\rangle\}$, where $\bot, \bot_i \in \Xi$ and $w_i \in (\Gamma \setminus \Xi)^*$ for all $i \in [n]$.

As we shall see later, every APDS in this paper is simple.

Let us fix a min APDS $\mathcal{M} = (\mathcal{P}, f)$, where $\mathcal{P} = (P, \Gamma, \Delta)$. An $\mathcal{M}$-automaton (or simply automaton) is a quintuple $\mathcal{A} = (Q, \Gamma, \delta, P, q_f)$, where $Q \supseteq P$ is a finite set of *states*, $q_f \in Q$ is the *final state*, and $\delta \subseteq Q \times \Gamma \times \mathbb{N}^\infty \times Q$ is a set of transitions. Notice that the *initial states* of $\mathcal{A}$ are the control locations of $\mathcal{P}$. Analogously, a *min $\mathcal{M}$-automaton* equips each rule with a natural number; formally, $\mathcal{B} = (\mathcal{A}, l)$, where $\mathcal{A}$ is an $\mathcal{M}$-automaton and $l : \delta \to \mathbb{N}^\infty$.

We define the *transition relation* $\rightarrow \subseteq Q \times \Gamma^* \times \mathbb{N}^\infty \times Q$ as the smallest relation satisfying:

- $q \xrightarrow{\varepsilon(0)} q$ for all $q \in Q$, and
- if $t = (q, \gamma, q') \in \delta$, $l(t) = a$, and $q' \xrightarrow{w(b)} q''$, then $q \xrightarrow{\gamma w(a+b)} q''$.

Given an initial state p and a word w, we define $W(p, w) = \min\{a \in \mathbb{N}^\infty \mid p \xrightarrow{w(a)} q_f\}$ to be the *minimum weight* of the word w when starting from the state p. $\mathcal{B}$ *accepts* or *recognizes* a configuration $\langle p, w\rangle$ if $p \xrightarrow{w(a)} q_f$ such that $a < \infty$. The set of configurations recognized by $\mathcal{B}$ is denoted by $\mathcal{L}(\mathcal{B})$.

In [SSE06], it has been shown that given a set of configurations C of an (unweighted) simple APDS $\mathcal{P}$, recognized by an automaton $\mathcal{A}$, we can construct another automaton $\mathcal{A}_{pre^*}$ such that $\mathcal{L}(\mathcal{A}_{pre^*}) = pre^*(C)$. (Note that the definitions of reachability relation and transition relation in the unweighted case can be defined analogously.) The procedure has been extended in [Suw09] to handle weighted APDSs where weights are abstractly defined. Min APDSs can be seen as a special case in which weights are instantiated for particular applications. For brevity, we will not elaborate on these correspondences any further, and simply apply the appropriate pushdown theory to SPKI/SDSI. In the next section, however, we propose a novel efficient algorithm which can be directly applied to solve the authorization problem and compute minimum-height certificate trees.

4 Computing certificate trees

Certificates in SPKI/SDSI can be interpreted as prefix rewrite systems. For instance, if p `a` $\rightarrow p'$ `b c` and p' `b` $\rightarrow p''$ `d e` are two certificates interpreted as rewrite rules, then their concatenation rewrites p `a` to p'' `d e c`. In SPKI/SDSI with threshold certificates, a concatenation of two or more certificates forms a *certificate tree*. It is easy to see that the authorization problem, given a principal p and p', reduces to the problem of whether there exists a certificate tree that rewrites $p\ \square$ at the root into $p'\ \square$ or $p'\ \blacksquare$ at its leaves.

It has been observed that the type of rewrite systems induced by a set of SPKI/SDSI certificates is equivalent to that of pushdown systems [JR04] and with the presence of threshold certificates to that of simple alternating pushdown systems [SSE06]. Roughly speaking, principals are interpreted as control locations, rôle identifiers as stack alphabet, and certificates as transition rules. For example, a certificate like p `a` $\rightarrow p'$ `b c` is interpreted as a pushdown rule $\langle p, \mathtt{a}\rangle \hookrightarrow \langle p', \mathtt{bc}\rangle$. The SPKI/SDSI authorization problem then reduces to the pushdown reachability problem.

Notice that the corresponding APDSs are simple, since threshold certificates are limited to authorization certificates. Given a SPKI/SDSI system (P, A, C), we have $\Xi = \{\square, \blacksquare\}$. All alternating rules are of the form $\langle p, \square\rangle \hookrightarrow \{p_1, w_1 b_1, \ldots, p_n, w_n b_n\}$, where $p, p_i \in P$, $w_i \in A^*$, and $b_i \in \{\square, \blacksquare\}$ for each $i \in [n]$.

Reachability algorithms for PDSs have been extensively studied in e.g. [BEM97, EHRS00, Sch02]. The algorithms are based on saturation procedures which, given an automaton $\mathcal{A}$ that recognizes a set of configurations C, repeatedly add transitions to $\mathcal{A}$ according to certain conditions until no more transitions can be added, i.e. it is saturated. We propose a similar saturation procedure for min APDSs.

Let us fix a simple min APDS $\mathcal{M} = (\mathcal{P}, f)$, where $\mathcal{P} = (P, \Gamma, \Delta)$, and a min $\mathcal{M}$-automaton $\mathcal{B} = (\mathcal{A}, l_0)$, where $\mathcal{A} = (Q, \Gamma, \delta_0, P, q_f)$, such that $\mathcal{L}(\mathcal{B}) = C$ for some

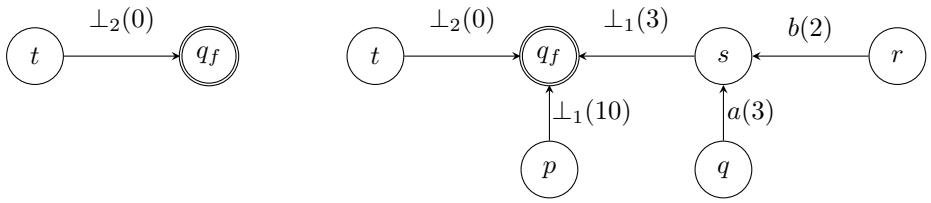


Figure 1: An example: the automata $\mathcal{B}$ (left) and $\mathcal{B}_{pre^*}$ (right)

$C \subseteq P \times \Gamma^*$. Without loss of generality, we assume that $\mathcal{A}$ has no transition leading to an initial state. The following procedure constructs $\mathcal{B}_{pre^*}$ accepting the set of configurations that are backwards reachable from any subset of configurations $C' \subseteq C$ with minimum weights. $\mathcal{B}_{pre^*}$ is defined as $(\mathcal{A}_{pre^*}, l)$, where $\mathcal{A}_{pre^*} = (Q, \Gamma, \delta, P, q_f)$. Initially, $\delta = \delta_0$ and $l(t) = l_0(t)$, if $t \in \delta_0$ and $l(t) = \infty$, otherwise. We iteratively update δ and l according to the following saturation rule until no values can be updated, i.e. until the automaton is saturated

> If $\langle p, \gamma\rangle \overset{a}{\hookrightarrow} \{\langle p_1, w_1\rangle, \ldots, \langle p_n, w_n\rangle\}$ and $p_i \xrightarrow{w_i(b_i)} q$ for all $i \in [n]$, add (p, γ, q) to δ and update $l(t) = \min(l(t), a + \max_{i=1}^{n}\{b_i\})$

Lemma 1 *Given a simple min APDS $\mathcal{M} = (\mathcal{P}, f)$, where $\mathcal{P} = (P, \Gamma, \Delta)$, and a min $\mathcal{M}$-automaton $\mathcal{B} = (\mathcal{A}, l_0)$, where $\mathcal{A} = (Q, \Gamma, \delta_0, P, q_f)$, the saturation procedure constructs $\mathcal{B}_{pre^*} = (\mathcal{A}_{pre^*}, l)$, where $\mathcal{A}_{pre^*} = (Q, \Gamma, \delta, P, q_f)$, such that $\mathcal{L}(\mathcal{B}_{pre^*}) = pre^*(\mathcal{L}(\mathcal{B}))$.*

Given a simple min APDS $\mathcal{M} = (\mathcal{P}, f)$, where $\mathcal{P} = (\{p, q, r, s, t\}, \{a, b, \perp_1, \perp_2\}, \Delta)$, as an example. The rules Δ and the weights f are defined as follows:

$$\langle p, \perp_1\rangle \overset{20}{\hookrightarrow} \langle t, \perp_2\rangle \qquad \langle p, \perp_1\rangle \overset{4}{\hookrightarrow} \{\langle q, a\perp_1\rangle, \langle s, \perp_1\rangle\}$$
$$\langle q, a\rangle \overset{1}{\hookrightarrow} \langle r, b\rangle \qquad \langle r, b\rangle \overset{2}{\hookrightarrow} \langle s, \varepsilon\rangle \qquad \langle s, \perp_1\rangle \overset{3}{\hookrightarrow} \langle t, \perp_2\rangle$$

Figure 1 shows the automaton $\mathcal{B}_{pre^*}$ on the right when applying the saturation procedure to the automaton $\mathcal{B}$ on the left. Notice that the configuration $\langle p, \perp_1\rangle$ is backwards reachable via two possible runs with weights 10 and 20. The weight of the transition $(p, \perp_1, q_f)$ is the minimum one.

An implementation of the saturation procedure is shown in Algorithm 1. The algorithm assumes without loss of generality two restrictions on every rule $\langle p, \gamma\rangle \hookrightarrow R$ in Δ:

(R1) if $R = \{\langle p', w'\rangle\}$, then $|w'| \leq 2$, and

(R2) if $|R| > 1$, then $\forall \langle p', w'\rangle \in R : |w'| = 1$.

In (R1), we call the rule pop rule, normal rule, and push rule when $|w'|$ is 0, 1, and 2, respectively. Note that any APDS can be converted into an equivalent one that satisfies (R1) and (R2) with only a linear increase in size.

Input: Min APDS $(\mathcal{P}, f)$, where $\mathcal{P} = (P, \Gamma, \Delta)$, and min automaton $(\mathcal{A}, l_0)$, where $\mathcal{A} = (Q, \Gamma, \delta_0, P, q_f)$

Output: The saturated min automaton $\mathcal{B}_{pre^*}$

procedure `update` (t, v)
    $\delta := \delta \cup \{t\}$;
    $v := \min(v, l(t))$;
    if $v \neq l(t)$ **then**
        $trans := trans \cup \{t\}$;
        $l(t) := v$;

$\delta := \delta_0$; $trans := \delta_0$; $l := \lambda t.\infty$;
forall $t \in \delta_0$ **do** $l(t) := l_0(t)$;
$\Delta' := \Delta$; $f' := \lambda r.\infty$; $g := \lambda r.\infty$;
forall $r \in \Delta$ **do** $f'(r) := f(r)$; $g(r) := 0$;
forall $r = \langle p, \gamma \rangle \hookrightarrow \langle p', \varepsilon \rangle \in \Delta$ **do** `update` $((p, \gamma, p'), f'(r))$;
while $trans \neq \emptyset$ **do**
    remove $t = (q, \gamma', q')$ from trans;
    forall $r = \langle p, \gamma \rangle \hookrightarrow \langle q, \gamma' \rangle \in \Delta'$ **do**
        `update` $((p, \gamma, q'), f'(r) + \max(g(r), l(t)))$;
    forall $r = \langle p, \gamma \rangle \hookrightarrow \langle q, \gamma'\gamma'' \rangle \in \Delta'$ **do**
        add $r' := \langle p, \gamma \rangle \hookrightarrow \langle q', \gamma'' \rangle$ to Δ';
        $f'(r') := \min(f'(r'), f'(r) + l(t))$;
        forall $t' = (q', \gamma'', q'') \in \delta$ **do**
            `update` $((p, \gamma, q''), f'(r') + l(t'))$;
    forall $r = \langle p, \gamma \rangle \hookrightarrow \{\langle q, \gamma' \rangle\} \cup R \in \Delta'$ *s.t.* $R \neq \emptyset$ **do**
        add $r' := \langle p, \gamma \rangle \hookrightarrow R$ to Δ';
        $f'(r') := \min(f'(r'), f'(r))$;
        $g(r') := \min(g(r'), \max(g(r), l(t)))$;
return $((Q, \Gamma, \delta, P, q_f), l)$;

Algorithm 1: A reachability algorithm for min APDSs

Lines 7–11 initialize the algorithm. Initially, $trans$ contains transitions from δ_0 (line 7), and all rules are copied to Δ' (line 9). The auxiliary functions f' and g are initialized to f and 0, respectively (line 10). Pop rules are dealt with first (line 11). The algorithm then repeatedly removes transitions from $trans$ (line 13) until it is empty (line 12), and examines whether they generate other transitions via the saturation rule (lines 14–24). The algorithm calls the procedure `update` (lines 1–6) when a new weight of a transition is computed. The new transition is added to δ (line 2) before computing the new minimum value (line 3). The if-statement at line 4 makes sure that the transition is added to $trans$ for further computation (line 5) only if its weight changes. As a result, line 6 can change $l(t)$ only to a smaller value.

The idea of the algorithm is to avoid unnecessary operations. Imagine that the saturation rule allows to add transition t if transitions t_1 and t_2 are already present. Now, if t_1 is taken from $trans$ but t_2 has not been added to $\mathcal{B}_{pre^*}$, we do not put t_1 back to $trans$ but store the following information instead: if t_2 is added, then we can also add t. It turns out that this can be done by adding new rules to Δ' and storing information in the auxiliary functions f' and g.

Let us now look at lines 14–24 in more detail. Line 14 handles normal rules where new transitions can be immediately added. Push rules (lines 16–20) and alternating rules (lines 21–24), however, require a more delicate treatment. At line 16 we know that (q, γ', q') is a transition of $\mathcal{B}_{pre^*}$ (because it has been removed from $trans$) and that $r = \langle p, \gamma\rangle \hookrightarrow \langle q, \gamma'\gamma''\rangle$ is a push rule of $\mathcal{P}$. We create the "fake rule" $\langle p, \gamma\rangle \hookrightarrow \langle q', \gamma''\rangle$ and add it to Δ' at line 17. Its f'-value is updated to be the minimum value of its old value (which is initialized to ∞ at line 9) and $f'(r) + l(t)$ at line 18. Later, when a transition (q', γ'', q'') is examined together with this fake rule at line 14, we update the transition (p, γ, q'') with weight $f'(r) + l(t) + l(q', \gamma'', q'')$. On the other hand, if the transition (q', γ'', q''), for any q'', is already in δ (line 19), we need to update the transition (p, γ, q'') accordingly (line 20).

At line 21 we know that $t = (q, \gamma', q'')$ is a transition of $\mathcal{B}_{pre^*}$ and $r = \langle p, \gamma\rangle \hookrightarrow \langle q', \gamma'\rangle \cup R$ is an alternating rule. Therefore, we add the fake rule $\langle p, \gamma\rangle \hookrightarrow R$ to Δ' (line 22), and minimize its f'-value to $f'(r)$ (line 23) and g-value to the maximum value of $g(r)$ and $l(t)$ (line 24). If the set R contains more than one element, then similar processes can take place which result in more fake rules with less elements in the right-hand sets. Line 14 handles the case when the fake rule is non-alternating. Notice that the auxiliary function g is used when constructing fake rules from alternating rules to store maximum values as specified by the saturation rule. The auxiliary function f' extends the function f by including weights of fake rules. Their values are used in line 15.

Lemma 2 *Algorithm 1 implements the saturation procedure.*

The complexity of the algorithm can be derived from the unweighted result in [EHRS00]. The difference is that Algorithm 1 can process transitions several times, while in [EHRS00] each transition is processed exactly once. Thus, the time complexity increases from $\mathcal{O}(|Q|^2|\Delta|)$ in [EHRS00] by a factor that is no more than the number of transitions and the maximum weight of all transitions. Note that weights of transitions can only decrease, and therefore limit the number of times they can be put in $trans$ and subsequently reprocessed.

Theorem 1 *Given a min SPKI/SDSI and two principal p and p', Algorithm 1 can be used to prove whether $aut(p, p')$ by computing the minimum-height certificate tree with p $\square$ at the root and p' $\square$ or p' $\blacksquare$ at its leaves.*

5 Conclusions

We have proposed an extension to the SPKI/SDSI authorization framework. The extension allows principals to assign weights to certificates, which permits principals to "prioritize" certificates they prefer. We have shown that the process of proving a principal's authorization turns out to be the problem of finding certificate trees. Finding the certificate tree with minimum height (or highest priority) is, however, a more difficult problem. We have given a connection between SPKI/SDSI and pushdown systems, and shown that the problem of computing minimum-height certificate trees reduces to the reachability problem of alternating pushdown systems. We have consequently proposed an algorithm for solving the reachability problem.

References

[BEM97] Ahmed Bouajjani, Javier Esparza, and Oded Maler. Reachability Analysis of Pushdown Automata: Application to Model-Checking. In *Proc. CONCUR*, 1997.

[CEE+01] Dwaine Clarke, Jean-Emile Elien, Carl Ellison, Matt Fredette, Alexander Morcos, and Ronald L. Rivest. Certificate Chain Discovery in SPKI/SDSI. *Journal of Computer Security*, 9:285–322, 2001.

[EFL+99] Carl M. Ellison, Bill Frantz, Butler Lampson, Ron Rivest, Brian Thomas, and Tatu Ylönen. *RFC 2693: SPKI Certificate Theory*. The Internet Society, 1999.

[EHRS00] Javier Esparza, David Hansel, Peter Rossmanith, and Stefan Schwoon. Efficient Algorithms for Model Checking Pushdown Systems. In *Proc. CAV*, LNCS 1855, pages 232–247, 2000.

[JR04] Somesh Jha and Thomas Reps. Model Checking SPKI/SDSI. *JCS*, 12(3–4):317–353, 2004.

[Sch02] Stefan Schwoon. *Model-Checking Pushdown Systems*. PhD thesis, Technische Universität München, 2002.

[SJRS03] Stefan Schwoon, Somesh Jha, Thomas Reps, and Stuart Stubblebine. On Generalized Authorization Problems. In *Proc. CSFW*, pages 202–218. IEEE, 2003.

[SSE06] Dejvuth Suwimonteerabuth, Stefan Schwoon, and Javier Esparza. Efficient Algorithms for Alternating Pushdown Systems with an Application to the Computation of Certificate Chains. In *Proc. ATVA*, LNCS 4218, pages 141–153, 2006.

[Suw09] Dejvuth Suwimonteerabuth. *Reachability in Pushdown Systems: Algorithms and Applications*. PhD thesis, Technische Universität München, 2009.

Some Remarks on Andrew Secure RPC

Sirapat Boonkrong

King Mongkut's University of Technology North Bangkok
1518 Pibulsongkram Road
Bangsue, Bangkok
10800
Thailand
sirapatb@kmutnb.ac.th

Abstract: We review the Andrew secure RPC protocol and reveal some unsoundness of it. Some modifications are made to the protocol. The changes made include the encryption in the first message, the expansion of the second and third messages as well as the elimination of the fourth message. Our GNY analysis shows that even though changes have been made, the outcomes of the protocol do not change. That is, both client and server hold the same new secret key shared between themselves.

1 Introduction

Although more than twenty years old, the Andrew secure RPC [Sat89] is still widely used as an example in the literature. That is why we feel that there is the need to make it as secure and efficient as possible. Since the original protocol, several attempts [BM03, Low96] have been made in order to make the protocol more secure. Even that, we have discovered that the Andrew RPC still leaves rooms for improvements. We, therefore, make several modifications to the protocol. That is, we add encryption to the first message to prevent the known-plaintext attacks. Another nonce is added to the second message as a challenge for authentication purposes. We add an identity of the sender in the third message in order to prevent session-hijacking. Moreover, we agree with [Low96] that the fourth message really contains no information, hence no uses for security, so we eliminate the fourth message. The modified protocol was then proved for correctness using the logic of Gong, Needham and Yahalom, also known as the GNY logic [GNY90, MSnN94]. The analysis of the newly modified protocol shows that the outcomes do not change from the original, which means both client and server will end up having a new shared secret.

The rest of the paper is organised as follows. The notations of the GNY Logic [GNY90] and the background of the Andrew secure RPC, including the original protocol, the modification made in [BAN90] and the adapted Andrew RPC [Low96], are mentioned in Section 2. Section 3 presents some remarks on the Andrew secure RPC. The modified protocol as well as the GNY analysis will be in Section 4. Section 5 concludes the paper.

2 Background

This section contains a short description and notations of the GNY logic [GNY90, MSnN94] as well as the background knowledge on the Andrew secure RPC. The background on the Andrew RPC includes the description of the original protocol, the protocol after BAN analysis and the Adapted Andrew RPC protocol.

2.1 GNY Logic

The GNY logic is a formal tool that allows us to analyse cryptographic protocols, step by step according to the rules provided (they can be found in [GNY90]).

Here we list the notations of the GNY logic in the hope that the readers, who are unfamiliar with the logic will understand the protocol description as well as the proof of correctness better. The notations are extracted from [GNY90].

Let P and Q be principals. The followings are the basic notations used in the GNY protocol.

- $P \lhd X$: P *is told* formula X. P receives X, possibly after performing some computation such as decryption. That is, a formula being told can be the message itself, as well as any computable content of that message.
- $P \ni X$: P *possesses*, or is capable of possessing, formula X. At a particular stage of a run, this includes all the formulae that P has been told, all the formulae he started the session with, and all the ones he has generated in that run. In addition P possesses, or is capable of possessing, everything that is computable from the formulae he already possesses.
- $P \mid\sim X$: P *once conveyed* formula X. X can be a message itself or some content computable from such a message, i.e. a formula can be conveyed implicitly.
- $P \mid\equiv \sharp(X)$: P *believes*, or is entitled to believe, that formula X is *fresh*. That is, X has not been used for the same purpose at any time before the current run of the protocol.
- $P \mid\equiv \phi(X)$: P *believes*, or is entitled to believe, that formula X is *recognisable*. That is, P would recognise X if P has certain expectations about the contents of X before actually receiving X. P may recognise a particular value (e.g. his own identifier), a particular structure (e.g. the format of a timestamp), or a particular form of redundancy.
- $P \mid\equiv P \overset{S}{\leftrightarrow} Q$: P *believes*, or is entitled to believe, that S is a suitable *secret* for P and Q. S will never be discovered by any principal except P, Q. This notation is symmetrical: $Q \overset{S}{\leftrightarrow} P$ and $P \overset{S}{\leftrightarrow} Q$ can be used interchangeably.

- $P \lhd *X$: P is told a formula which he did not convey previously in the current run. That is, X can be regarded as a *not-originated-here* formula.
- Let C be a statement. $P \mid\equiv C$: P believes, or P would be entitled to believe, that statement C holds.

2.2 Andrew Secure RPC

The Andrew secure RPC was introduced in [Sat89]. It allows two parties, A and B (usually a client and a server), who already share a key K_{ab}, to agree upon a new key K'_{ab}. The protocol also performs an authentication handshake. There are four messages in the protocol exchange. The first three, A and B perform a handshake using a shared secret K_{ab}. In the final message, B sends a new key K'_{ab} to A. The protocol can be summarised as follows. Note that nonce N_a is chosen by A and nonces N_b, N'_b are chosen by B.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : \{N_a\}_{K_{ab}} \\ \text{Message 2.} & B \rightarrow A : \{N_a + 1, N_b\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_b + 1\}_{K_{ab}} \\ \text{Message 4.} & B \rightarrow A : \{K'_{ab}, N'_b\}_{K_{ab}} \end{array}$$

Unfortunately, Burrows *et al.* [BAN90] have pointed out that there is a problem with the freshness of the new key K'_{ab}. That is, there is nothing that can guarantee that K'_{ab} is fresh. Another problem has been mentioned by Clark and Jacob [CJ95] that an intruder could record the second message and substitute it in place of the fourth. The result is that A would accept $N_a + 1$ as a new key. However, for this attack to be successful, it depends on the property of the nonce N_a, i.e., whether or not the nonce is predictable. Due to the problems stated, Burrows *et al.* revised the protocol.

2.3 Andrew Secure RPC after BAN

Burrows *et al.* carried out an analysis on Andrew secure RPC using their logic of authentication or BAN [BAN90]. The result of the analysis shows that the original Andrew secure RPC could suffer from a replay attack, as mentioned in the previous section. Therefore, the original protocol was revised and the resultant protocol is as follows.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : A, N_a \\ \text{Message 2.} & B \rightarrow A : \{N_a, K'_{ab}\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_a\}_{K'_{ab}} \\ \text{Message 4.} & B \rightarrow A : N'_b \end{array}$$

Lowe [Low96] exposed the weakness of this revised protocol by introducing an attack on it. Lowe's attack shows that an intruder could engage in two protocol runs in parallel.

In run number one, A tries to contact B but an intruder I intercepts the message, and masquerades as B. In run number two, the intruder initiates the session with A while impersonating B. The description of the attack can be seen in [Low96]. Bird *et al.* have also presented the similar attack on the protocol [BGH$^+$91, BBG$^+$93]. As a result, Lowe fixed the problem to make it less vulnerable to this kind of attack.

2.4 Adapted Andrew RPC

Lowe [Low96] addressed the problem, stated in the previous section, by changing message 2 to include an encrypted copy of the sender's identity. This can prevent the attack in that an intruder will not be able to replay the message anymore. The Adapted Andrew RPC is described as follows. Note that message 2 now carries the identity of B.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : A, N_a \\ \text{Message 2.} & B \rightarrow A : \{N_a, K'_{ab}, B\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_a\}_{K'_{ab}} \\ \text{Message 4.} & B \rightarrow A : N'_b \end{array}$$

Even though problems with the Andrew secure RPC have been found and addressed, we believe that there are still things that need to be mentioned. They include possibilities of an attack as well as the efficiency of the protocol.

3 Remarks on Andrew Secure RPC

As mentioned earlier, since Andrew secure RPC still appears a lot in literature, we believe that if possible, we should make an attempt to make it as secure and efficient as possible. This section presents some remarks that we have on the Andrew secure RPC.

3.1 Attacks

After having studied the Adapted Andrew RPC, the latest variation of the Andrew secure RPC, we reckon there are a couple of vulnerabilities to the protocol. The first is the *known-plaintext attack*. The second is *session hi-jacking*. We discuss each of them in turn.

3.1.1 Known-Plaintext Attack

By definition, a known-plaintext attack occurs when a cryptanalyst or an attacker has access to the plaintext and the ciphertext of one or more pieces of data, and is at liberty to make use of them to reveal secret information, such as the encryption key. Let us take a look at the first and third messages of the Adapted Andrew RPC.

$$\text{Message 1.} \quad A \rightarrow B : A, N_a$$
$$\text{Message 3.} \quad A \rightarrow B : \{N_a\}_{K'_{ab}}$$

It can easily be observed that in message 1, the nonce N_a is sent in clear. That means, an attacker could eavesdrop and record the nonce. A little later, message 3 is sent. This time the content of the message is the nonce N_a encrypted with the new key K'_{ab}. Again, the same attacker could eavesdrop the conversation and record the encrypted copy of the nonce N_a that he has recorded earlier. Now, the attacker holds the plaintext, N_a, and the ciphertext, $\{N_a\}_{K'_{ab}}$. By having the plaintext and ciphertext pair, the attacker could initiate a *known-plaintext attack*. We understand that having one pair of plaintext and ciphertext may not be enough to successfully attack the protocol this way, but we do think that it is worth pointing out this weakness.

3.1.2 Session-Hijacking

Session-hijacking occurs when an attacker takes over a conversation between two parties. Here, we explain that message 3 of the Adapted Andrew RPC could lead to "session hijacking". We put the words in quote, because we do not think that the attacker could steal the session per se. What he could do is as follows.

By looking at message 3 of the Adapted Andrew RPC,

$$\text{Message 3.} \quad A \rightarrow B : \{N_a\}_{K'_{ab}}$$

we see that it is sent from A to B in order to confirm that A has correctly received the new key K'_{ab}. Without his identity as part of the message, A sends *only* the nonce N_a encrypted with the new key K'_{ab}. The implication of this is that an attacker could intercept the message and forward it to B. B would think that this message comes from the attacker, not A. We acknowledge that this vulnerability on its own does not reveal any secret, but B could then send subsequent messages to the attacker instead of A.

3.2 Excessive Message

In the previous section, a couple vulnerabilities in the Adapted Andrew RPC are introduced. Here, we look at the efficiency of the protocol. By efficiency, we mean the number of messages used to complete the protocol.

Having studied the Adapted Andrew RPC, we agree with [Low96] that message 4 of the protocol does not contain any information. We would like to emphasise this claim here that Message 4 : $B \rightarrow A : N'_b$ is *not* necessary for the main purpose of the protocol. That is, no security information is transferred from A to B. We, therefore, claim that message 4 can be eliminated from the procedure. The next section will show that even if this message is removed, the procedure can still accomplish the same thing as before. That is, both A and B hold the new shared key K'_{ab}.

In this section, we have mentioned the two vulnerabilities that could potentially lead to an attack on the Adapted Andrew RPC. Next section, we make an attempt to modify the Adapted Andrew RPC in order to address the weaknesses. We then give the analysis of the protocol to show that after the changes both parties, A and B, still hold the same secret key K'_{ab}.

4 Modified Andrew Secure RPC

First, the two vulnerabilities stated in the previous section will be addressed. The modified protocol will then be proved for correctness using the logic of Gong, Needham and Yahalom [GNY90, MSnN94].

4.1 The Protocol

In order to address the potential known-plaintext attack, we recommend that the first message should be encrypted using the already known shared key K_{ab}. By encrypting the first message, we get rid of the known-plaintext attack in that an attacker cannot have any plaintext and ciphertext pair anymore. For the second weakness, session-hijacking, we suggest that the identity of the sender should be a part of the message. By adding the identity, the attacker could still intercept and forward the message. However, the recipient would know who created that message, hence subsequent messages would then be sent to the legitimate party. Moreover, the fourth message of the Adapted Andrew RPC is removed from the procedure to increase the efficiency. Last, but not least, we think that the sender of the second message should add a new nonce to the message. This new nonce would act as a *fresh* challenge for the response in message 3. The resultant protocol is as follows.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : \{A, N_a\}_{K_{ab}} \\ \text{Message 2.} & B \rightarrow A : \{N_a, N_b, K'_{ab}, B\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{A, N_b\}_{K'_{ab}} \end{array}$$

4.2 Protocol Analysis

The section presents the analysis of the modified protocol. The GNY logic is used for the analysis. Therefore, all the postulates can be seen in [GNY90, MSnN94].

First, the modified protocol is idealised into the logic of GNY as follows.

$$\begin{array}{ll} \text{Message 1.} & B \lhd *\{*A, *N_a\}_{K_{ab}} \\ \text{Message 2.} & A \lhd *\{N_a, *N_b, *K'_{ab}, *B\}_{K_{ab}} \rightsquigarrow B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \\ \text{Message 3.} & B \lhd *\{*A, N_b\}_{K'_{ab}} \end{array}$$

The followings are assumptions of the Andrew secure RPC made in [BAN90]. Note that we do *not* add any new assumptions to this modified protocol.

$$
\begin{array}{ll}
A \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B & B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B \\
A \mid\equiv B \mid\Longrightarrow A \overset{K'_{ab}}{\longleftrightarrow} B & B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \\
A \mid\equiv \sharp(N_a) & B \mid\equiv \sharp(N_b)
\end{array}
$$

We now carry out the GNY analysis on the protocol.

Message 1: Applying the postulates T1 and T3, we obtain $B \lhd A, N_a$. That is, B has received or has been told A and N_a. Then the postulate P1 is applied, and we obtain $B \ni A, N_a$. That is, B now possesses A's identity and nonce N_a.

Message 2: First, we note that the extension to the message, $B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$, is valid because it is evident from the initial assumption.

Applying the postulates T1, T3 and P1, we obtain $A \lhd N_a, N_b, K'_{ab}, B$. That is, A now possesses the nonces N_a and N_b, the new key K'_{ab} and B's identity.

Applying F1, we obtain $A \mid\equiv \sharp(N_a, N_b, K'_{ab}, B)$. That is, A believes that the message is fresh, i.e., not a replay.

Applying R1, we obtain $A \mid\equiv \phi(N_a, N_b, K'_{ab}, B)$. That is, A believes that the contents of the message is recognisable.

Applying I1, we obtain $A \mid\equiv B \mid\sim (N_a, N_b, K'_{ab}, B)$, $A \mid\equiv B \mid\sim \{N_a, N_b, K'_{ab}, B\}_{K_{ab}}$, $A \mid\equiv B \ni K_{ab}$. That is, A believes that the message is originated from B and A believes that B possesses the key K_{ab}.

Applying I6, we obtain $A \mid\equiv B \ni N_a, N_b, K'_{ab}, B$. That is, A believes that B possesses the nonces N_a and N_b, his own identity B, and the new key K'_{ab}.

Applying J2, we obtain $A \mid\equiv B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$. That is, A believes that B believes that K'_{ab} is a good key for A and B.

Applying J1, we obtain $A \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$. That is, A believes that K'_{ab} is a good key for A and B.

Therefore, at the end of the second message, A possesses the new shared key K'_{ab} and A also believes that K'_{ab} is a good key shared between A and B. Furthermore, A recognises his own nonce N_a, which means that B has decrypted the first message correctly. That, in turn, means that B possesses the same key K_{ab}, hence A has authenticated B.

Message 3: Applying the postulates T1, T3 and P1, we obtain $B \ni A, N_b$. That is, B possesses A's identity and nonce N_b.

Applying F1, we obtain $B \mid\equiv \sharp\{A, N_b\}_{K'_{ab}}$. That is, B believes that the message is fresh, i.e., not a replay.

Applying R2, we obtain $B \mid\equiv \phi\{A, N_b\}_{K'_{ab}}$. That is, B believes that the contents of the message is recognisable.

Applying I1, we obtain $B \mid\equiv A \mid\sim (A, N_b)$, $B \mid\equiv A \mid\sim \{A, N_b\}_{K'_{ab}}$, $B \mid\equiv A \ni K'_{ab}$.

That is, B believes that the message is originated from A and B believes that A now possesses the new key K'_{ab}.

Here, the third message alone shows that B recognises his own nonce N_b, which means that A has decrypted the second message correctly. That, in turn, means that A possesses the key K_{ab}, hence B has authenticated A. Furthermore, B now believes that A also holds the new secret ey K'_{ab}, which is the same as the one B is holding.

On the whole, at the end of the protocol run, we obtain:

$$A \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \quad \text{and} \quad B \mid\equiv A \ni K'_{ab}.$$

This means that both A and B are now holding the new secret key K'_{ab}. A and B both believe that the new key K'_{ab} is a good key for subsequent communications. Moreover, A and B know that the other party possesses K'_{ab} as well.

5 Conclusions

We have presented an overview of the three variations of the Andrew secure RPC. They include original Andrew secure RPC, the Andrew secure RPC after BAN analysis and the Adapted Andrew RPC. We have also shown that weaknesses have been discovered and exploited in the original Andrew secure RPC and the Andrew secure RPC after BAN.

In this paper, a couple of vulnerabilities have been found in the adapted Andrew RPC. Those vulnerabilities could potentially lead to a known-plaintext attack as well as a session-hijack. The problem of known-plaintext has been addressed by encrypting the first message. The problem of session hijacking in the third message of the Adapted Andrew RPC has been fixed by adding the identity of the sender as part of the message. Furthermore, we have recommended that the sender of the second message should add a newly generated nonce to the message in order to make the authentication challenge fresh. In addition to those weaknesses, the efficiency of the protocol has been considered. It has been mentioned in this paper that the fourth message of the Andrew secure RPC has no use in security at all. We have, therefore, suggested that it should be removed from the protocol.

Having designed a protocol to address all the vulnerabilities mentioned in this paper, a GNY analysis on the resultant protocol has been carried out. It has been pointed out that despite the modifications made, the outcomes of the protocol have not been altered. That is, the protocol achieves mutual authentication. Both parties involved in the protocol run end up holding the same new secret key. Finally, they both believe that the new key is good for encrypting and decrypting subsequent messages, and they both believe that the other party possesses the new secret key as well.

References

[BAN90] Michael Burrows, Mart Abadi und Roger Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8:18–36, 1990.

[BBG+93] R. Bird, R. Bird, I. Gopal, I. Gopal, A. Herzberg, A. Herzberg, P. Janson, P. Janson, S. Kutten, S. Kutten, R. Molva, R. Molva, M. Yung und M. Yung. Systematic Design of a Family of Attack-Resistant Authentication Protocols, 1993.

[BGH+91] Ray Bird, Inder Gopal, Amir Herzberg, Phil Janson, Shay Kutten, Refik Molva und Moti Yung. Systematic Design of Two-Party Authentication Protocols, 1991.

[BM03] Colin Boyd und Anish Mathuria. *Protocols for Authentication and Key Establishment*. Springer, Berlin; London, 2003.

[CJ95] John Clark und Jeremy Jacob. On the Security of Recent Protocols. *Information Processing Letters*, 56:151–155, 1995.

[GNY90] Li Gong, Roger Needham und Raphael Yahalom. Reasoning about Belief in Cryptographic Protocols. In *Proceedings 1990 IEEE Symposium on Research in Security and Privacy*, Seiten 234–248. IEEE Computer Society Press, 1990.

[Low96] Gavin Lowe. Some New Attacks upon Security Protocols. Seiten 162–169. Society Press, 1996.

[MSnN94] Anish Mathuria, Reihaneh Safavi-naini und Peter Nickolas. Some Remarks on the Logic of Gong, Needham and Yahalom, 1994.

[Sat89] M. Satyanarayanan. Integrating Security in a Large Distributed System. *ACM Transactions on Computer Systems*, 7:247–280, 1989.

Generalizing of a High Performance Parallel Strassen Implementation on Distributed Memory MIMD Architectures

Duc Kien Nguyen[1], Ivan Lavallee[2], Marc Bui[2]

[1]CHArt - Ecole Pratique des Hautes Etudes & Université Paris 8, France
Kien.Duc-Nguyen@univ-paris8.fr

[2]LaISC - Ecole Pratique des Hautes Etudes, France
Ivan.Lavallee@ephe.sorbonne.fr
Marc.Bui@ephe.sorbonne.fr

Abstract: Strassen's algorithm to multiply two $n \times n$ matrices reduces the asymptotic operation count from $O(n^3)$ of the traditional algorithm to $O(n^{2.81})$, thus designing efficient parallelizing for this algorithm becomes essential. In this paper, we present our generalizing of a parallel Strassen implementation which obtained a very nice performance on an Intel Paragon: faster 20% for $n \approx 1000$ and more than 100% for $n \approx 5000$ in comparison to the parallel traditional algorithms (as Fox, Cannon). Our method can be applied to all the matrix multiplication algorithms on distributed memory computers that use Strassen's algorithm at the system level, hence it gives us compatibility to find better parallel implementations of Strassen's algorithm.

1 Introduction

Matrix multiplication (MM) is one of the most fundamental operations in linear algebra and serves as the main building block in many different algorithms, including the solution of systems of linear equations, matrix inversion, evaluation of the matrix determinant and the transitive closure of a graph. In several cases the asymptotic complexities of these algorithms depend directly on the complexity of matrix multiplication - which motivates the study of possibilities to speed up matrix multiplication. Also, the inclusion of matrix multiplication in many benchmarks points at its role as a determining factor for the performance of high speed computations.

Strassen was the first to introduce a better algorithm [Str69] for MM with $O(N^{\log_2 7})$than the traditional one which needs $O(N^3)$ operations. Then Winograd variant [Win71] of Strassen's algorithm has the same exponent but a slightly lower constant as the number of additions/subtractions is reduced from 18 down to 15. The record of complexity owed to Coppersmith and Winograd is $O(N^{2.376})$, resulted from arithmetic aggregation [CW90]. However, only Winograd's algorithm and Strassen's algorithm offer better performance than traditional algorithm for matrices of practical sizes, say, less than 10^{20}[LPS92]. The

full potential of these algorithms can be realized only on large matrices, which require large machines such as parallel computers. Thus, designing efficient parallel implementations for theses algorithms becomes essential.

This research was started when a paper by Chung-Chiang Chou, Yuefan Deng, Gang Li, and Yuan Wang [CDLW95] on the Strassen parallelizing came to our attention. Their implementation already obtained a nice performance: in comparison to the parallel traditional algorithms (as Fox, Cannon) on an Intel Paragon, it's faster 20% for $n \approx 1000$ and more than 100% for $n \approx 5000$. The principle of this implementation is to parallelize the Strassen's algorithm at the system level - i.e. to stop on the recursion level r of execution tree - and the calculation of the products of sub matrices is locally performed by the processors. The most significant point here is to determine the sub matrices after having recursively executed r time the Strassen's formula (these sub matrices are corresponding to the nodes of level r in the execution tree of Strassen's algorithm) and then to find the result matrix from these sub matrices (corresponding to the process of backtracking the execution tree). It is simple to solve this problem for a sequential machine, but it's much harder for a parallel machine. With a definite value of r, we can manually do it like [CDLW95], [LD95], and [GSv96] made (r = 1, 2) but the solution for the general case has not been found.

In this paper, we present our method to determine all the nodes at the unspecified level r in the execution tree of Strassen's algorithm, and to show the expression representing the relation between the result matrix and the sub matrices at the level recursion r; this expression allows us to calculate directly the result matrix from the sub matrices calculated by parallel matrix multiplication algorithms at the bottom level. By combining this result with the matrix multiplication algorithms at the bottom level, we have a generalizing of the high performance parallel Strassen implementation in [CDLW95]. It can be applied to all the matrix multiplication algorithms on distributed memory computers that use Strassen's algorithm at the system level, besides the running time for these algorithms decreases when the recursion level increases hence this general solution gives us compatibility to find better implementations (which correspond with a definite value of the recursive level and a definite matrix multiplication algorithm at the bottom level).

2 Background

2.1 Strassen's Algorithm

We start by considering the formation of the matrix product $Q = XY$, where $Q \in \Re^{m\times n}$, $X \in \Re^{m\times k}$, and $Y \in \Re^{k\times n}$. We will assume that m, n, and k are all even integers. By partitioning

$$X = \begin{pmatrix} X_{00} & X_{01} \\ X_{10} & X_{11} \end{pmatrix}, Y = \begin{pmatrix} Y_{00} & Y_{01} \\ Y_{10} & Y_{11} \end{pmatrix}, Q = \begin{pmatrix} Q_{00} & Q_{01} \\ Q_{10} & Q_{11} \end{pmatrix}$$

where $Q_{ij} \in \Re^{\frac{m}{2} \times \frac{n}{2}}$, $X_{ij} \in \Re^{\frac{m}{2} \times \frac{k}{2}}$, and $Y_{ij} \in \Re^{\frac{k}{2} \times \frac{n}{2}}$, it can be shown [Win71, GL89] that the following computations compute $Q = XY$:

$$\begin{aligned}
M_0 &= (X_{00} + M_{11})(Y_{00} + Y11) \\
M_1 &= (X_{10} + X_{11})Y_{00} \\
M_2 &= X_{00}(Y_{01} - Y_{11}) \\
M_3 &= X_{11}(-Y_{00} + Y_{10}) \\
M_4 &= (X_{00} + X_{01})Y_{11} \\
M_5 &= (X_{10} - X_{00})(Y_{00} + Y_{01}) \\
M_6 &= (X_{01} - X_{11})(Y_{10} + Y_{11}) \\
Q_{00} &= M_0 + M_3 - M_4 + M_6 \\
Q_{01} &= M_1 + M_3 \\
Q_{10} &= M_2 + M_4 \\
Q_{11} &= M_0 + M_2 - M_1 + M_5
\end{aligned} \tag{1}$$

The Strassen's algorithm does the above computation recursively until one of the dimensions of the matrices is 1.

2.2 A High Performance Parallel Strassen Implementation

In this section, we will see the principle of the hight performance parallel Strassen implementation presented in [CDLW95], which is foundation for our generalizing.

First, decompose the matrix X into 2×2 blocks of sub matrices X_{ij} where $i, j = 0, 1$. Seconde, decompose further these four sub matrices into four 2×2 (i.e. 4×4) blocks of sub matrices x_{ij} where $i, j = 0, 1, 2, 3$.

$$X = \begin{pmatrix} X_{00} & X_{01} \\ X_{10} & X_{11} \end{pmatrix} = \begin{pmatrix} x_{00} & x_{01} & x_{02} & x_{03} \\ x_{10} & x_{11} & x_{12} & x_{13} \\ x_{20} & x_{21} & x_{22} & x_{23} \\ x_{30} & x_{31} & x_{32} & x_{33} \end{pmatrix}$$

Similarly, perform the same decomposition on matrix Y and get:

$$Y = \begin{pmatrix} Y_{00} & Y_{01} \\ Y_{10} & Y_{11} \end{pmatrix} = \begin{pmatrix} y_{00} & y_{01} & y_{02} & y_{03} \\ y_{10} & y_{11} & y_{12} & y_{13} \\ y_{20} & y_{21} & y_{22} & y_{23} \\ y_{30} & y_{31} & y_{32} & y_{33} \end{pmatrix}$$

Then, use the Strassen's formula to multiply the matrices X and Y, and get the following

seven matrix multiplication expressions:

$$\begin{cases} M_0 = (X_{00} + X_{11})(Y_{00} + Y_{11}) \\ M_1 = (X_{10} + X_{11})Y_{00} \\ M_2 = X_{00}(Y_{01} - Y_{11}) \\ M_3 = X_{11}(-Y_{00} + Y_{10}) \\ M_4 = (X_{00} + X_{01})Y_{11} \\ M_5 = (-X_{00} + X_{10})(Y_{00} + Y_{01}) \\ M_6 = (X_{01} - X_{11})(Y_{10} + Y_{11}) \end{cases}$$

Next, apply the Strassen's formula to these seven expressions to obtain 49 matrix multiplication expressions on sub matrices x and y. Taking M_0 as an example:

$$\begin{aligned} M_{00} &= (x_{00} + x_{22} + x_{11} + x_{33})\,(y_{00} + y_{22} + y_{11} + y_{33}) \\ M_{01} &= (x_{10} + x_{32} + x_{11} + x_{33})\,(y_{00} + y_{22}) \\ M_{02} &= (x_{00} + x_{22})\,(y_{01} + y_{23} - y_{11} - y_{33}) \\ M_{03} &= (x_{11} + x_{33})\,(y_{10} + y_{32} - y_{00} - y_{22}) \\ M_{04} &= (x_{00} + x_{22} + x_{01} + x_{23})\,(y_{11} + y_{33}) \\ M_{05} &= (x_{10} + x_{32} - x_{00} - x_{22})\,(y_{00} + y_{22} + y_{01} + y_{23}) \\ M_{06} &= (x_{01} + x_{23} - x_{11} - x_{33})\,(y_{10} + y_{32} + y_{11} + y_{33}) \end{aligned}$$

Similarly, each of the remaining six matrix multiplication expressions M_i for $i = 1, 2, \ldots, 6$ can also be expanded into six groups of matrix multiplications in terms of x and y.

$$\begin{aligned} M_{10} &= (x_{20} + x_{22} + x_{31} + x_{33})\,(y_{00} + y_{11}) \\ M_{11} &= (x_{30} + x_{32} + x_{31} + x_{33})\,y_{00} \\ M_{12} &= (x_{20} + x_{22})\,(y_{01} - y_{11}) \\ M_{13} &= (x_{31} + x_{33})\,(y_{10} - y_{00}) \\ M_{14} &= (x_{20} + x_{22} + x_{21} + x_{23})\,y_{11} \\ M_{15} &= (x_{30} + x_{32} - x_{20} - x_{22})\,(y_{00} + y_{01}) \\ M_{16} &= (x_{21} + x_{23} - x_{31} - x_{33})\,(y_{10} + y_{11}) \end{aligned}$$

$$\begin{aligned} M_{20} &= (x_{00} + x_{11})\,(y_{02} - y_{22} + y_{13} - y_{33}) \\ M_{21} &= (x_{10} + x_{11})\,(y_{02} - y_{22}) \\ M_{22} &= x_{00}\,(y_{03} - y_{23} - y_{13} + y_{33}) \\ M_{23} &= x_{11}\,(y_{12} - y_{32} - y_{02} + y_{22}) \\ M_{24} &= (x_{00} + x_{01})\,(y_{13} - y_{33}) \\ M_{25} &= (x_{10} - x_{00})\,(y_{02} - y_{22} + y_{03} - y_{23}) \\ M_{26} &= (x_{01} - x_{11})\,(y_{12} - y_{32} + y_{13} - y_{33}) \end{aligned}$$

$$\begin{aligned} M_{30} &= (x_{22} + x_{33})\,(y_{20} - y_{00} + y_{31} - y_{11}) \\ M_{31} &= (x_{32} + x_{33})\,(y_{20} - y_{00}) \\ M_{32} &= x_{22}\,(y_{21} - y_{01} - y_{31} + y_{11}) \\ M_{33} &= x_{33}\,(y_{30} - y_{10} - y_{20} + y_{00}) \\ M_{34} &= (x_{22} + x_{23})\,(y_{31} - y_{11}) \\ M_{35} &= (x_{32} - x_{22})\,(y_{20} - y_{00} + y_{21} - y_{01}) \\ M_{36} &= (x_{22} + x_{33})\,(y_{30} - y_{10} + y_{31} - y_{11}) \end{aligned}$$

$$
\begin{aligned}
M_{40} &= (x_{00} + x_{02} + x_{11} + x_{13})(y_{22} + y_{33}) \\
M_{41} &= (x_{10} + x_{12} + x_{11} + x_{13})\, y_{22} \\
M_{42} &= (x_{00} + x_{02})(y_{23} - y_{33}) \\
M_{43} &= (x_{11} + x_{13})(y_{32} - y_{22}) \\
M_{44} &= (x_{00} + x_{02} + x_{01} + x_{03})\, y_{33} \\
M_{45} &= (x_{10} + x_{13} - x_{00} - x_{02})(y_{22} + y_{23}) \\
M_{46} &= (x_{01} + x_{03} - x_{11} - x_{13})(y_{32} + y_{33})
\end{aligned}
$$

$$
\begin{aligned}
M_{50} &= (x_{20} - x_{00} + x_{31} - x_{11})(y_{00} + y_{02} + y_{11} + y_{13}) \\
M_{51} &= (x_{30} - x_{10} + x_{31} - x_{11})(y_{00} + y_{02}) \\
M_{52} &= (x_{20} - x_{00})(y_{01} + y_{03} - y_{11} - y_{13}) \\
M_{53} &= (x_{31} - x_{11})(y_{10} + y_{12} - y_{00} - y_{02}) \\
M_{54} &= (x_{20} - x_{00} + x_{21} - x_{01})(y_{11} + y_{13}) \\
M_{55} &= (x_{30} - x_{10} - x_{20} + x_{00})(y_{00} + y_{02} + y_{01} + y_{03}) \\
M_{56} &= (x_{21} - x_{01} - x_{31} + x_{11})(y_{10} + y_{12} + y_{11} + y_{13})
\end{aligned}
$$

$$
\begin{aligned}
M_{60} &= (x_{02} - x_{22} + x_{13} - x_{22})(y_{20} + y_{22} + y_{31} + y_{33}) \\
M_{61} &= (x_{12} - x_{32} + x_{13} - x_{33})(y_{20} + y_{22}) \\
M_{62} &= (x_{02} - x_{22})(y_{21} + y_{23} - y_{31} - y_{33}) \\
M_{63} &= (x_{13} - x_{33})(y_{30} + y_{32} - y_{20} - y_{22}) \\
M_{64} &= (x_{02} - x_{22} + x_{03} - x_{23})(y_{31} + y_{33}) \\
M_{65} &= (x_{12} - x_{32} - x_{02} + x_{22})(y_{20} + y_{22} + y_{21} + y_{23}) \\
M_{66} &= (x_{03} - x_{23} - x_{13} + x_{33})(y_{30} + y_{32} + y_{31} + y_{33})
\end{aligned}
$$

After finishing these 49 matrix multiplications, we need to combine the resulting M_{ij} where $i, j = 0, 1, \ldots, 6$ to form the final product matrix.

$$
Q = \begin{pmatrix} Q_{00} & Q_{01} \\ Q_{10} & Q_{11} \end{pmatrix} = \begin{pmatrix} q_{00} & q_{01} & q_{02} & q_{03} \\ q_{10} & q_{11} & q_{12} & q_{13} \\ q_{20} & q_{21} & q_{22} & q_{23} \\ q_{30} & q_{31} & q_{32} & q_{33} \end{pmatrix}
$$

First, define some variables $\delta_i = \begin{cases} -1, \text{ if } i = 4 \\ 1 \text{ otherwise} \end{cases}$ and $\gamma_i = \begin{cases} -1, \text{ if } i = 1 \\ 1 \text{ otherwise} \end{cases}$, the 4 x 4 blocks of sub matrices forming the product matrix Q can be written as:

$$
\begin{aligned}
q_{00} &= \textstyle\sum_{i \in S_1} \delta_i \left(M_{i0} + M_{i3} - M_{i4} + M_{i6}\right) \\
q_{01} &= \textstyle\sum_{i \in S_1} \delta_i \left(M_{i2} + M_{i4}\right) \\
q_{02} &= \textstyle\sum_{i \in S_3} M_{i0} + M_{i3} - M_{i4} + M_{i6} \\
q_{03} &= \textstyle\sum_{i \in S_3} M_{i2} + M_{i4}
\end{aligned}
$$

$$
\begin{aligned}
q_{10} &= \textstyle\sum_{i \in S_1} \delta_i \left(M_{i1} + M_{i3}\right) \\
q_{11} &= \textstyle\sum_{i \in S_1} \delta_i \left(M_{i0} + M_{i2} - M_{i1} + M_{i5}\right) \\
q_{12} &= \textstyle\sum_{i \in S_3} M_{i1} + M_{i3} \\
q_{13} &= \textstyle\sum_{i \in S_3} M_{i0} + M_{i2} - M_{i1} + M_{i5}
\end{aligned}
$$

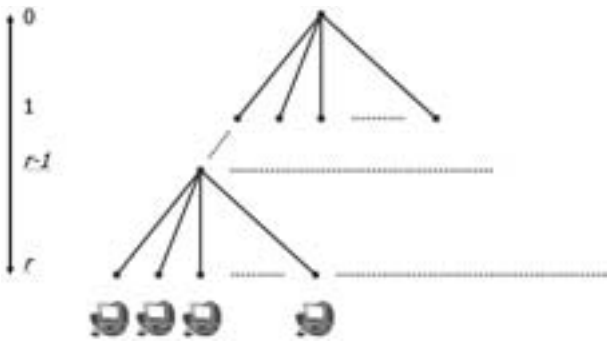


Figure 1: Principle of the Strassen parallelizing in [CDLW95].

$$
\begin{aligned}
q_{20} &= \textstyle\sum_{i \in S_2} M_{i0} + M_{i3} - M_{i4} + M_{i6} \\
q_{21} &= \textstyle\sum_{i \in S_2} M_{i2} + M_{i4} \\
q_{22} &= \textstyle\sum_{i \in S_4} \gamma_i \left(M_{i0} + M_{i3} - M_{i4} + M_{i6} \right) \\
q_{23} &= \textstyle\sum_{i \in S_4} \gamma_i \left(M_{i2} + M_{i4} \right)
\end{aligned}
$$

$$
\begin{aligned}
q_{30} &= \textstyle\sum_{i \in S_2} M_{i1} + M_{i3} \\
q_{31} &= \textstyle\sum_{i \in S_2} M_{i0} + M_{i2} - M_{i1} + M_{i5} \\
q_{32} &= \textstyle\sum_{i \in S_4} \gamma_i \left(M_{i1} + M_{i3} \right) \\
q_{33} &= \textstyle\sum_{i \in S_4} \gamma_i \left(M_{i0} + M_{i2} - M_{i1} + M_{i5} \right)
\end{aligned}
$$

As you saw above, it is not very simple although they have only 49 matrix multiplications. It become great complicated if we want to go further - when we have 343, 2401 or more matrix multiplications.

3 Generalizing of the Parallel Strassen Implementation

The principle of the method that has been presented is to parallelize the Strassen's algorithm at the system level - i.e. to stop on the recursion level r of execution tree - and the calculation of the products of sub matrices is locally performed by the processors. The most important point here is to determine the sub matrices after having applied r time the Strassen's formula, and to find the result matrix from the products of these sub matrices. In the preceding works, the solutions are given with fixed values of r (= 1, 2). But the solution for the general case has not been found.

Such are the problems with which we are confronted and the solution will be presented in this section.

3.1 Recursion Removal in Fast Matrix Multiplication

We represent the Strassen's formula:

$$\begin{aligned} &m_l = \sum_{i,j=0,1} x_{ij} SX(l,i,j) \times \sum_{i,j=0,1} y_{ij} SY(l,i,j) \\ &l = 0 \cdots 6 \\ &\text{and } q_{ij} = \sum_{l=0}^{6} m_l SQ(l,i,j) \end{aligned} \tag{2}$$

with

SX=

Pij	00	01	10	11
0	-1	0	1	1
1	1	0	0	0
2	0	1	0	0
3	1	0	-1	0
4	0	0	1	1
5	1	1	-1	-1
6	0	0	0	1

SY=

Pij	00	01	10	11
0	1	-1	0	1
1	1	0	0	0
2	0	0	1	0
3	0	-1	0	1
4	0	1	0	-1
5	0	0	0	1
6	1	-1	-1	1

SQ=

Pij	00	01	10	11
0	0	1	1	1
1	1	1	1	1
2	1	0	0	0
3	0	0	1	1
4	0	1	0	1
5	0	1	0	0
6	0	0	-1	0

Each of 7^k product can be represented as in the following:

$$\begin{aligned} &m_l = \sum_{i,j=0,n-1} x_{ij} SX_k(l,i,j) \times \sum_{i,j=0,n-1} y_{ij} SY_k(l,i,j) \\ &l = 0...7^k - 1 \\ &\text{and } q_{ij} = \sum_{l=0}^{7^k-1} m_l SQ_k(l,i,j) \end{aligned} \tag{3}$$

In fact, $SX = SX_1, SY = SY_1, SQ = SQ_1$. Now we have to determine values of matrices SX_k, SY_k, and SQ_k from SX_1, SY_1, and SQ_1. In order to obtain this, we extend the definition of tensor product in [KHJS90] for arrays of arbitrary dimensions as followed:

Definition. Let A and B are arrays of same dimension l and of size $m_1 \times m_2 \times \ldots \times m_l, n_1 \times n_2 \times \ldots \times n_l$ respectively. Then the tensor product (TP) is an array of same dimension and of size $m_1 n_1 \times m_2 n_2 \times \ldots \times m_l n_l$ defined by replacing each element of A with the product of the element and B.
$P = A \otimes B$ where $P[i_1, i_2, ..., i_l] = A[k_1, k_2, ..., k_l] B[h_1, h_2, ..., h_l], i_j = k_j n_j + h_j$ with $\forall 1 \leq j \leq l$;

Let $P = \bigotimes_{i=1}^{n} A_i = (...(A_1 \otimes A_2) \otimes A_3)... \otimes A_n)$ with A_i is array of dimension l and of size $m_{i1} \times m_{i2} \times \ldots \times m_{il}$. The following theorem allows computing directly elements of P

Theorem.

$$\begin{aligned} P[j_1, j_2, ..., j_l] &= \prod_{i=1}^{n} A_i[h_{i1}, h_{i2}, ..., h_{il}] \\ \text{where } j_k &= \sum_{s=1}^{n} \left(h_{sk} \prod_{r=s+1}^{n} m_{rk} \right). \end{aligned} \tag{4}$$

Proof. We prove the theorem by induction. With $n = 1$, the proof is trivial. With $n = 2$, it is true by the definition. Suppose it is true with $n - 1$. We show that it is true with n.

We have $P_{n-1}[t_1, t_2, ..., t_l] = \prod_{i=1}^{n-1} A_i[h_{i1}, h_{i2}, ..., h_{il}]$ where $t_k = \sum_{s=1}^{n-1} \left(h_{sk} \prod_{r=s+1}^{n-1} m_{rk} \right)$ with $\forall 1 \leq k \leq l$; and then $P_n = P_{n-1} \otimes A_n$.

By definition

$P_n[j_1, j_2, ..., j_l] = P_{n-1}[p_1, p_2, ..., p_l] A_n[h_{n1}, h_{n2}, ..., h_{nl}] = \prod_{i=1}^{n} A_i[h_{i1}, h_{i2}, ..., h_{il}]$

where $j_k = p_k m_{nk} + h_{nk} = m_{nk} \times \sum_{s=1}^{n-1} \left(h_{sk} \prod_{r=s+1}^{n-1} m_{rk} \right) + h_{nk}$

$= \sum_{s=1}^{n-1} \left(h_{sk} \prod_{r=s+1}^{n} m_{rk} \right) + h_{nk} = \sum_{s=1}^{n} \left(h_{sk} \prod_{r=s+1}^{n} m_{rk} \right)$

The theorem is proved. □

In particular, if all A_i have the same size $m_1 \times m_2 \times \ldots \times m_l$, we have $P[j_1, j_2, ..., j_l] = \prod_{i=1}^{n} A_i[h_{i1}, h_{i2}, ..., h_{il}]$ where $j_k = \sum_{s=1}^{n} \left(h_{sk} m_k^{n-s} \right)$.

Remark. $j_k = \sum_{s=1}^{n} \left(h_{sk} m_k^{n-s} \right)$ is a j_k's factorization in base m_k. We note $a = \overline{a_1 a_2 ... a_l}_{(b)}$ the a's factorization in base b hence $P[j_1, j_2, ..., j_l] = \prod_{i=1}^{n} A_i[h_{i1}, h_{i2}, ..., h_{il}]$ then $j_k = \overline{h_{i1} h_{i2} ... h_{in}}_{(m_k)}$.

Now we return to our algorithm. We have following theorem:

Theorem.

$$\begin{aligned} SX_k &= \bigotimes_{i=1}^{k} SX \\ SY_k &= \bigotimes_{i=1}^{k} SY \\ SQ_k &= \bigotimes_{i=1}^{k} SQ \end{aligned} \tag{5}$$

Proof. We prove the theorem by induction. Clearly it is true with $k = 1$. Suppose it is true with $k - 1$. The algorithm's execution tree is balanced with depth k and degree 7. Thanks to (3), we have at the level $k - 1$ of the tree:

$$\begin{aligned} &M_l = \\ &\left(\sum_{0 \leq i,j \leq 2^{k-1}-1} X_{k-1,ij} SX_{k-1}(l, i, j) \right) \times \left(\sum_{0 \leq i,j \leq 2^{k-1}-1} Y_{k-1,ij} SY_{k-1}(l, i, j) \right) \\ &0 \leq l \leq 7^{k-1} - 1 \end{aligned}$$

Then thanks to (2) at the level k we have

$$
\begin{array}{l}
M_l[l'] = \\
\sum\limits_{0\le i',j'\le 1}\left(\left(\sum\limits_{0\le i,j\le 2^{k-1}-1} X_{k-1,ij}[i',j']SX_{k-1}\left(l,i,j\right)\right)SX(l',i',j')\right)\times \\
\sum\limits_{0\le i',j'\le 1}\left(\left(\sum\limits_{0\le i,j\le 2^{k-1}-1} Y_{k-1,ij}[i',j']SY_{k-1}\left(l,i,j\right)\right)SY(l',i',j')\right) \\
0\le l\le 7^{k-1}-1 \\
0\le l'\le 6 \\
= \\
\sum\limits_{0\le i',j'\le 1}\left(\sum\limits_{0\le i,j\le 2^{k-1}-1} X_{k-1,ij}[i',j']SX_{k-1}\left(l,i,j\right)SX(l',i',j')\right)\times \\
\sum\limits_{0\le i',j'\le 1}\left(\sum\limits_{0\le i,j\le 2^{k-1}-1} Y_{k-1,ij}[i',j']SY_{k-1}\left(l,i,j\right)SY(l',i',j')\right) \\
0\le l\le 7^{k-1}-1 \\
0\le l'\le 6
\end{array}
\tag{6}
$$

where $X_{k-1,ij}[i',j'], Y_{k-1,ij}[i',j']$ are $2^k\times 2^k$ matrices obtained by division $X_{k-1,ij}, Y_{k-1,ij}$ in 4 sub matrices (i',j' indicate the sub matrix's quarter).

We present l,l' in the base 7, and i,j,i',j' in the base 2 and remark that $X_{k-1,ij}\left[i',j'\right] = X_k\left[\overline{ii'_2},\overline{jj'_2}\right]$. Then (6) becomes

$$
\begin{array}{l}
M[\overline{ll'}_{(7)}] = \\
\left(\sum\limits_{0\le \overline{ii'}_{(2)},\overline{jj'}_{(2)}\le 2^{k-1}-1} X_k[\overline{ii'}_{(2)},\overline{jj'}_{(2)}]SX_{k-1}\left(l,i,j\right)SX(l',i',j')\right)\times \\
\left(\sum\limits_{0\le \overline{ii'}_{(2)},\overline{jj'}_{(2)}\le 2^{k-1}-1} Y_k[\overline{ii'}_{(2)},\overline{jj'}_{(2)}]SY_{k-1}\left(l,i,j\right)SY(l',i',j')\right) \\
0\le \overline{ll'}_{(7)}\le 7^{k-1}-1
\end{array}
\tag{7}
$$

In addition, we have directly from (3):

$$
\begin{array}{l}
M[\overline{ll'}_{(7)}] = \\
\left(\sum\limits_{0\le \overline{ii'}_{(2)},\overline{jj'}_{(2)}\le 2^{k-1}-1} X_k[\overline{ii'}_{(2)},\overline{jj'}_{(2)}]SX_k\left(\overline{ll'}_{(7)},\overline{ii'}_{(2)},\overline{jj'}_{(2)}\right)\right)\times \\
\left(\sum\limits_{0\le \overline{ii'}_{(2)},\overline{jj'}_{(2)}\le 2^{k-1}-1} Y_k[\overline{ii'}_{(2)},\overline{jj'}_{(2)}]SY_k\left(\overline{ll'}_{(7)},\overline{ii'}_{(2)},\overline{jj'}_{(2)}\right)\right) \\
0\le \overline{ll'}_{(7)}\le 7^{k-1}-1
\end{array}
\tag{8}
$$

Compare (7) and (8) we have

$$
\begin{array}{l}
SX_k\left(\overline{ll'_7},\overline{ii'_2},\overline{jj'_2}\right) = SX_{k-1}\left(l,i,j\right)SX\left(l',i',j'\right) \\
SY_k\left(\overline{ll'_7},\overline{ii'_2},\overline{jj'_2}\right) = SY_{k-1}\left(l,i,j\right)SY\left(l',i',j'\right)
\end{array}
$$

By definition, we have

$$SX_k = SX_{k-1} \otimes SX = \overset{k}{\underset{i=1}{\otimes}} SX$$
$$SY_k = SY_{k-1} \otimes SY = \overset{k}{\underset{i=1}{\otimes}} SY$$

Similarly

$$SQ_k = SQ_{k-1} \otimes SQ = \overset{k}{\underset{i=1}{\otimes}} SQ$$

The theorem is proved. □

Thanks to Theorem 3.1 and Remark 3.1 we have

$$\begin{aligned} SX_k(l,i,j) &= \prod_{r=1}^{k} SX(l_r, i_r, j_r) \\ SY_k(l,i,j) &= \prod_{r=1}^{k} SY(l_r, i_r, j_r) \\ SQ_k(l,i,j) &= \prod_{r=1}^{k} SQ(l_r, i_r, j_r) \end{aligned} \tag{9}$$

Apply (9) in (3) we have nodes leafs m_l and all the elements of result matrix.

3.2 Generalizing

Now we known how to parallelize Strassen's algorithm in general case: firstwe stop at therecursion level r, thanks to the expressions (9) and (3),we have the entire corresponding sub matrices:

$$\begin{aligned} M_l = & \sum_{i,j=0,2^r-1} X_{ij} \left(\prod_{t=1}^{r} SX(l_t, i_t, j_t) \right) \\ & \times \\ & \sum_{\substack{i,j=0,2^r-1 \\ l=0...7^r-1}} Y_{ij} \left(\prod_{t=1}^{r} SY(l_t, i_t, j_t) \right) \end{aligned} \tag{10}$$

with

$$X_{ij} = \begin{pmatrix} x_{i*2^{k-r},j*2^{k-r}} & \dots & x_{i*2^{k-r},j*2^{k-r}+2^{k-r}-1} \\ \dots & \dots & \dots \\ x_{i*2^{k-r}+2^{k-r}-1,j*2^{k-r}} & \dots & x_{i*2^{k-r}+2^{k-r}-1,j*2^{k-r}+2^{k-r}-1} \end{pmatrix}$$
$$Y_{ij} = \begin{pmatrix} y_{i*2^{k-r},j*2^{k-r}} & \dots & y_{i*2^{k-r},j*2^{k-r}+2^{k-r}-1} \\ \dots & \dots & \dots \\ y_{i*2^{k-r}+2^{k-r}-1,j*2^{k-r}} & \dots & y_{i*2^{k-r}+2^{k-r}-1,j*2^{k-r}+2^{k-r}-1} \end{pmatrix}$$
$$i = 0, 2^r - 1, j = 0, 2^r - 1$$

The product M_l of the sub matrices $\left(\sum_{\substack{i=0,2^r-1 \\ j=0,2^r-1}} X_{ij} \left(\prod_{t=1}^{r} SX\left(l_t, i_t, j_t\right) \right) \right)$ and $\left(\sum_{\substack{i=0,2^r-1 \\ j=0,2^r-1}} Y_{ij} \left(\prod_{t=1}^{r} SY\left(l_t, i_t, j_t\right) \right) \right)$ is locally calculated on each processor by the sequential matrix multiplication algorithms.

Finally, thanks to (9) & (3) we have directly sub matrix elements of result matrix by applying matrix additions instead of backtracking manually the recursive tree to calculate the root in [LD95], [CDLW95], and [GSv96]:

$$\begin{aligned} Q_{ij} &= \sum_{l=0}^{7^r-1} M_l SQ_r(l,i,j) \\ &= \sum_{l=0}^{7^r-1} M_l \left(\prod_{t=1}^{r} SQ\left(l_t, i_t, j_t\right) \right) \end{aligned} \tag{11}$$

4 Conclusion

We have presented a general scalable parallelization for all the matrix multiplication algorithms on distributed memory computers that use Strassen's algorithm at inter-processor level. The running time for these algorithms decreases when the recursion level increases hence this general solution gives us compatibility to find better algorithms (which correspond with a definite value of the recursive level and a definite matrix multiplication algorithm at the bottom level). And from a different view, we have generalized the Strassen's formula for the case where the matrices are divided into 2^k parts (the case $k = 2$ gives usoriginal formulas) thus we have a whole new direction to parallelize the Strassen's algorithm. In addition, we are applying these ideas to all the fast matrix multiplication algorithms.

References

[CDLW95] Chung-Chiang Chou, Yuefan Deng, Gang Li, and Yuan Wang. Parallelizing Strassen's Method for Matrix Multiplication on Distributed Memory MIMD architectures. *Computers and Math. with Applications*, 30(2):4–9, 1995.

[CW90] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9(3):251–280, 1990.

[GL89] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 2nd edition, 1989.

[GSv96] Brian Grayson, Ajay Shah, and Robert van de Geijn. A High Performance Parallel Strassen Implementation. *Parallel Processing Letters*, 6(1):3–12, 1996.

[KHJS90] B. Kumar, Chua-Huang Huang, Rodney W. Johnson, and P. Sadayappan. A tensor product formulation of Strassen's matrix multiplication algorithm. *Applied Mathematics Letters*, 3(3):67–71, 1990.

[LD95] Qingshan Luo and John B. Drake. A scalable parallel Strassen's matrix multiplication algorithm for distributed memory computers. In *Proceedings of the 1995 ACM symposium on Applied computing*, pages 221–226, Nashville, Tennessee, United States, 1995. ACM Press.

[LPS92] J. Laderman, V. Y. Pan, and H. X. Sha. On Practical Algorithms for Accelerated Matrix Multiplication. *Linear Adgebra and Its Applications*, 162:557–588, 1992.

[Str69] Volker Strassen. Gaussian Elimination is not Optimal. *Numer. Math.*, 13:354–356, 1969.

[Win71] Shmuel Winograd. On multiplication of 2 x 2 matrices. *Linear Algebra and its Applications*, 4:381–388, 1971.

Using Digital Images to spread Executable Code on Internet

Roberto Gómez , Gabriel Ramírez

ITESM-CEM, Depto. Ciencias Computacionales, Km 3.5 Lago Guadalupe, 51296, Atizapan Zaragoza, Edo México, Mexico
{rogomez, A00472181}@itesm.mx

Abstract. . Steganography is defined as covered writing and it has been used to achieve confidentiality. Most of research in this domain has focused in just sending plain data. In this paper we use steganography as a way to spread executable code instead of just communicating a secret message. The application of our proposal may contrast, could be using for spreading malicious code or for protecting software copyright like a water marking alive. We present our implementation of this scheme with a developed tool and hence look in a direction that has not looked at detail so far.

1 Introduction

Steganography has evolved during history from handcrafted ways to very complex techniques, giving to the humans a way to conceal communication. However our purpose is to use steganography as a mechanism to hide executable code, using the power of steganography not to conceal communication, but to conceal the transportation of executable code.

Internet was a major breakthrough in the way we communicate with each other bringing a universe of information. This universe has a proliferation of digital images, being the most interchangeable kind of file. Multimedia data presents a highly redundant representation, which usually allows the hide of significantly large amounts of data. Due to this, image files are the ideal objects to hide information, especially executable code, besides other kind of information.

Most of the remote code execution schemes in Internet are based on a client/server model. A server listens on specific port, and the client sends a request to that port. Once the request arrives, and depending in the type of request, the server performs a specific action. In our model, a server waits for the arrival of an image, once it arrives, the server looks for the code embedded into the image file. If any code is found in the image, our system extracts the code, load it and execute it. If not code is detected it continues searching for other image files.

As far as we know there is not enough research about using steganography to propagate executable code. We propose a protocol to propagate code using the principles of steganography.

This paper is organized as follows: in section 2 we introduce ht main features of steganography. Section 3 presents BMP image file format. In section 4 we explain related work of steganography with executable code and the differences with our own approach. In section 5 we explain our model. In section 6 we show our experiments and results. In section 7 we give our conclusions, and the future work.

2 Steganography

The term steganography is derived from the Greek words steganos that means, "covered" and graphia that means "writing", i.e. covered writing. Steganography refers to the art and science of concealing a communication; unlike cryptography, where conceals the message but the communication is often known [CH03]. In fact the ideal would be a combination of these areas to accomplish a framework more robust to assure secure communications. Steganography is considered as an art because it is related with creativity and it is also considered as a science due to the areas involved. These two approaches combined give a powerful area in computer science. Classical examples of steganography are invisible ink and microdot.

The steganographic process involves the following elements: a carrier, a secret message, a key(s), an embedding algorithm and a protocol. In digital steganography, a carrier may be any kind of digital file for example a picture, movie, audio, etc; as well as the secret message. The key or keys could be any combination of steganography and cryptography keys and they may feed the embedding algorithm in order to generate certain seed to incrust the secret message based on that input. Obviously like in every kind of communication is needed a protocol, in this case is very similar what is managed in private and public key cryptography or could be without the interchange of any key, what is known as pure steganography. All these elements are used in order to generate an object, which is known as a steganogram, the carrier with the secret message embedded. The perceptual content of the cover object must be indistinguishable from the steganogram in order to consider the steganographic process effective and efficient.

Basically there are three ways to hide information into a digital object: adding the secret message, generating a steganogram from scratch (cover generation), and substituting data from the carrier. In [JDJ00] we found a good survey of these techniques.

The adding techniques incrust the secret message into a carrier without modified data at all. The technique appends the secret information in certain parts that do not affect the quality of the carrier itself. The disadvantage is that the size of the resultant steganogram files increments as much as the size of the secret message, looking suspicious if the size of the secret message is significant. Nevertheless the technique does not generate any degradation on the carrier. Examples of these techniques consist in hide information in not used fields of TCP/IP headers [CGW05], or to hide it after the end of a word file.

The cover generation techniques encode information in the way a cover is generated. They have the advantage that the generated steganogram presents certain properties to defend it against statistical attacks. By the other hand the resultant steganogram might not be common and look suspicious. As an example we can mention the Auto-

mated Generation of English Text, which uses a large dictionary of words categorised by different types, and a style source, which describes how words of different types can be used to form a meaningful sentence. It transforms the message bits into sentences by selecting words out of the dictionary, which conforms to a sentence structure given in the style source. Another example is a fractal image.

The substitution techniques are the most used by the steganographic tools. They substitute redundant parts of a carrier with a secret message. One of the simplest substitution methods is LSB (Least Significant Bits); it chooses a subset of cover elements and substitute least significant bit(s) of each element by message bit(s). LSB take advantage that the human eye is not capable to distinguish slight modifications in the carrier. The receptor of the message could only extract the message if he knows the exact positions where it was incrusted. This technique has been used in image, audio and video files, taking advantage of redundancy. It can be used en grey- scale and colour images with one, two, three or even four least significant bits. The disadvantage of this technique is that the information hidden can be altered or erased if the carrier is compressed or filtered by any method. An excellent study of LSB technique can be found in [CM01].

A pseudorandom number generator may be used to spread the secret message over the cover in a random manner. It must be noted that, according to [Ca98], that classical definition of steganography is statistical and not perceptual. In his paper Cachin defines a steganography technique to be E-secure if the relative entropy of the probability distribution of cover objects and steganogram is less than or equal to E. He calls a steganography technique to be perfectly secured if E is zero. He then demonstrates such steganographic techniques do exist and they are perfectly secure (however, the technique described by him is impractical).

A different approach in substitution is image downgrading. It differs from the previous one in the quality of the final image. An image is said to have been downgraded when its sensitivity label has been changed to the one strictly dominated by its previous value, [CPL95] For example a representation of 32 bits is reduced to 16 bits, with a quality degradation.

According to [WP00] there are two kinds of steganographics attacks, visual and statical attacks. More reliable attacks have been published for sequential (or not) LSB embedding and can be found in [FGD02].

Westfeld and Pfitzmann establishes that the idea of visual attacks is to remove all parts of the image covering the message, so the human eye can distinguish whether there is a potential message or still image content. The filtering process depends on the presumed steganographic method used. So the attacker needs to known the used technique.

By the other hand the idea of the statistical attack is to compare the theoretically expected frequency distribution in steganograms with some sample distribution observed in the possibly changed carrier medium. The attacker needs to compare the original carrier with the steganogram in order to detect this. In order to do this the attacker needs the original carrier, and this does not occur in most of the cases.

Both techniques do not allow the attacker to obtain the hiding data, but it gives enough information to detect that something is embedded in the carrier.

Another substitution technique that defends to statistical attacks is pseudorandom permutations; it takes random bits and bytes from carrier and replaces them with the

message performing some mathematical computation. Using this technique provides more robustness against statistical analysis.

There are more robust substitution techniques, known as transformation techniques [KP00]. These techniques incrust the secret message in significant data of the carrier based on a mathematical process that take the signal from one representation to another one, surviving only the significant data. The disadvantage of this method is the difficulty to find places where to hide the secret message, because redundancy is reduced at minimum.

Our proposal is based in substitution techniques and specifically LSB due to its simplicity, and efficient in implementation. The size of the resultant steganogram does not change at all, unless the carrier cannot afford space enough for transporting the executable code. Besides the degradation of the steganogram does not look suspicious and it cannot being used to detect that the carrier hides some information.

3 The BMP file format

Digital image formats are ways to represent images in digital form. The image is represented as a finite set of digital values, called picture elements or pixels. There are many different digital image formats in use today, but certainly the most used today are: BMP, GIF and JPEG.

The basic elements of a image file format are: fields, labels and blocks. Mastering these components is crucial to know where and how hide data into them.

GIF images are used extensively on the web. Supports animated images. The format supports only 255 colors per frame, so it losses quantization in full-color photos. It is based on a color table with a maximum of 256 colors. This format is popular for its reduced size based on LZW compression and because it allows small animations and interlacing.

JPEG divides an image in blocks of 8x8 pixels and applies a mathematical transformation to keep only the significant data and get rid of the data that is not important to visualize the image. By doing this, the image size is smaller. This image format is ideal to store pictures because of its powerful compression based on a DCT coefficients.

It is important to remark that JPEG itself specifies only how an image is transformed into a stream of bytes, but not how those bytes are encapsulated in any particular storage medium. A further standard, created by the Independent JPEG Group, called JFIF (JPEG File Interchange Format) specifies how to produce a file suitable for computer storage and transmission (such as over the Internet) from a JPEG stream. In common usage, when one speaks of a "JPEG file" one generally means a JFIF file.

Microsoft Windows programs, and the Windows operating system itself commonly use BMP image files. The BMP format consists basically on the following structures: a file header, an image header, a color palette and the data (pixels). According to the MSDN library, all the structures specify three main elements. The first element is the header that describes the resolution of the device on which the rectangle of pixels was created, the dimensions of the rectangle, the size of the array of bits, and so on. The

second element is a logical palette. And the third element consists of an array of bits that defines the relationship between pixels in the bitmapped image and entries in the logical palette.

The file header contains the first 14 bytes of the file, and it contains information about the bitmap data found elsewhere in the file. Next to the file header there is the image header. It has a length of 40 bytes, and it holds the fields that describe the image itself. Bitmaps are usually organized, either physically or logically, into lines of pixels. Following the image header we found the number of elements in the color table. This number is equal to the number of colors in the image. It is not common using a color table. In the 24-bit uncompressed RGB color mode, there is not a color table, each pixel consists of three 8-bit values, a blue byte, a green byte, and a red byte. The combination of the three values represents the pixel's color, and they represents all the image pixels lineal, from left to right upwards line-by-line starting at the lower left. This format conformation gives a high degree of redundancy and is possible to substitute the least significant bits of the carrier.

In addition, it could be used a fourth reserved byte of every pixel representation in the case if it exists a color table.

We choose to work with BMP format due to its plain structure and the extremely redundancy used to represent every single color of one pixel in digital images. An observer will not be able to distinguish between a carrier and a steganogram in a BMP file due to eye capacity to distinguish that kind of slight changes.

Implementing LSB in GIF and JPEG formats would imply decoding data flow first according to the compression process and then applied LSB to the data to finally decode for a second time with the compression process again.

4 Related work

Our idea is based in the techniques used in steganography to hide information, but instead of hiding data, we propose to hide executable code. This code will be extracted and executed once it arrives to a computer. As far as we know there is no substantial research in this kind of steganography. We found some empirical work of this approach

In the master degree thesis presented in [Co00] the author exposes the possibility of an attack in computer warfare. The author installs a steganographic parser in a computer target using an executable wrapper in form of video game, like a Trojan horse, which needs a previous intervention of the computer affected user. The author embeds executable code into a different format files like BMP, GIF, JPEG, TXT, and HTML, employing public steganographic tools available like S-Tools. Once the parser steganographic is installed every steganographic file may be used by this one to extract and execute code.

Another similar approach is presented in [Si02]. The authors employ an ActiveX installer in a Web page to install the steganographic parser. This approach works with the comment label of the JFIF format, a popular implementation of the JPEG standard. The authors embed a keylogger, which is a kind of spyware in the system that records every single action in the keyboard. The authors propose that, after certain

time, the keylogger send the logs (with valuable information) to the perpetrator of this attack.

The use of an AVI file to hide executable code is proposed in [Ro02]. The authors take advantage that this kind of file launches an event, so it is not necessary a previous installation of a steganographic parser. They propose to force the event in such a way that it runs out of a web browser, executing the hiding code without the need of the user authorization.

A slight different approach that takes advantage of a specific vulnerability in Microsoft software is presented in [Mi04]. It consists of a buffer overflow, "activated" when an image JFIF format file is created with a comment block with one or zero bytes of size. Knowing that any block in JFIF format consists at minimum of two bytes because includes the two bytes that indicate the size of the bock, Microsoft libraries does not know how to deal with this exception and might produce buffer overflow in the system.

In [Al02] the authors give a proof of concept that demonstrates propagating a computational virus using steganography. The author appends the executable code after the EOF label in a JFIF picture. It needs a previous intervention of the user to install the steganographic parser and infecting the first JFIF image file in the system. This approach spreads the executable code in a picture for every image infected being visualized in the same path.

5 Our proposal

The main objective of our system is to spread executable code embedded in BMP images. In order to accomplish our objective the system must fulfill some properties. The steganogram must not look suspicious about something hidden in it. The capacity of the carrier of embedding must be enough to transport the secret message without being perceptible. The algorithm employed must be efficient and robust in order to not consume computational resources and it must take into account the intentions of detecting and destroying steganographic content.

Our approach is divided in six phases (see figure 1): embedding executable code in a carrier (BMP image), installing a steganographic parser at the system target, delivering a steganographic image to the target, detecting the steganogram at its arrival, extracting executable code from image and finally executing the information hidden (executable code).

In the embedding executable code process in a BMP image we implement a substitution algorithm that substitutes 1, 2, 3 or 4 LSB depending on the users' choice.

In order to extract and run the executable code, the steganographic parser must have been previously installed; this is the hard part of our proposal. This can be achieved in several ways. We can send to the system's users a trojan program and wait that some user executes it. Another solution is to design and use a remote exploit to install the parser.

Once the parser installed in the system, it must keep hidden and not raise any suspicions. The behavior of the system must be as normal as if nothing new has occurred. By the other hand the system needs a way to let the steganographic parser analyze

every picture visualized in the system. The system modifies some keys in the registry of system configuration, to accomplish the previous requirements.

Delivering the steganographic image is the easy part because it could be in a web page or in an e-mail, etc.; the images are the kind of files that circulates in computer systems.

Detecting the steganographic material is accomplished by the steganographic parser installed previously. It looks for a specific mark inside the file that distinguishes it from a normal image.

Extracting information from BMP file is folding out LSB used in the embedding process.

Executing hiding information could be in a different fashion way, creating a new thread, or process, it depends on the purpose of the hidden executable code.

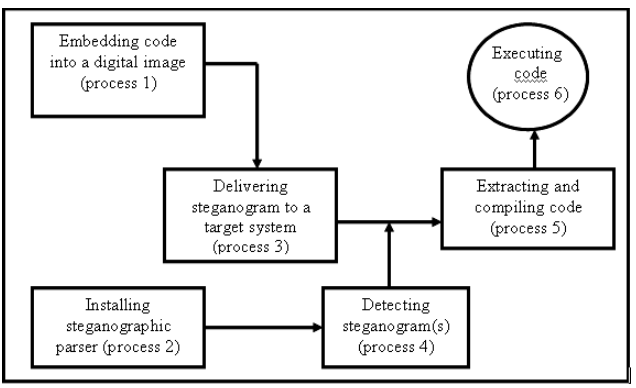


Fig. 1. Steganography executing code mode kernel

6 Experiments and results

We designed and implemented an application that is capable to hide executable a BMP file. The application is able to install the parser in the local host. We focus on BMP format because it has a plain structure and facilitates its implementation. Steganography experts recommend using shades of grey. Grey scale images are preferred because the shades change very gradually between its elements. This increases the images ability to hide information. However we decided color images in order to being more innocuous due to the fact that most users prefer color images.

We used substitution LSB steganographic technique in order that carrier size does not change that much, using data pixels. The information than can be hidden in an image file depends on how many bits per byte were used to incrust in the carrier.

All our experiments were done over Windows XP Professional. The steganographic parser is installed in the path c:\windows\system with the name system32b.exe for being innocuous. The added key to install the parser was My_PC\HKEY_CLASSES_ROOT\Applications\system32b.exe\shell\open\command

with value C:\WINDOWS\system\system32b.exe "%1". The modified key to register the parser was My_PC\HKEY_CURRENT_USER\Software\Microsoft\Windows\ CurrentVersion\Explorer\FileExts\.bmp. With the sub key Application with value system32b.exe.

The embedded executable code used in every experiment was a typical Windows calculator application with a size of 115 200 bytes.

In figure 2 we show a screenshot of the application. The first option gives the possibility to incrust in 1, 2, 3 or 4 least significant bits on a BMP image. The choice depends on how much degradation is allowed. The next options allow selecting the original carrier, the file containing the executable code and the desired name for the steganogram. The interface counts with an option to activate/deactivate the steganographic parser. This last option allows us to test the system's behavior with and without a steganographic parser.

The application was developed in C language because its power in order to manipulate bits of the BMP image file. Firstly the application incrusts a mark (for example "GR") to distinguish a steganogram from a typical image in the first bytes of data pixels, and then incrust the size of the executable code. If the executable code is bigger than the data pixel image it expands the steganogram in order to fit the executable code. Only in this case the steganogram grow more in size than the carrier, in all the others cases the size remains the same. At first the application assures that the steganographic parser exists and has been properly installed in the host system if it is not, proceeds to install this one.

Fig. 2. Application developed interface

In order to test our system we select different images. The image carrier was selected for containing a variety of colors with a size of 342 294 bytes. Using 1 LSB the code needs 921 600 bytes (115 200 * 8) for being carried. The result steganogram can be seen in figure 3.

Fig. 3. Image with 1 LSB modified

Using 2 LSB the code needs 460 800 (115 200 * 8)/2 bytes for being carried. The result steganogram can be seen in figure 4.

Fig. 4. Image with 2 LSB modified

Using 3 LSB the code needs 307 200 (115 200 * 8)/3 bytes for being carried. The result steganogram can be seen in figure 5.

Fig. 5. Image with 3 LSB modified

Using 4 LSB the code needs 230 400 (115 200 * 8)/4 bytes for being carried. The result steganogram can be seen in figure 6.

Fig. 6. Image with 4 LSB modified

The results indicate that using 1 and 2 LSB results in a steganogram almost imperceptible to notice differences between original carrier and steganogram. However, using 3 and 4 LSB results a suspicion steganogram because it decreases the quality of the images.

Execution of executable code was successful in each case, as we can see in figure 7.

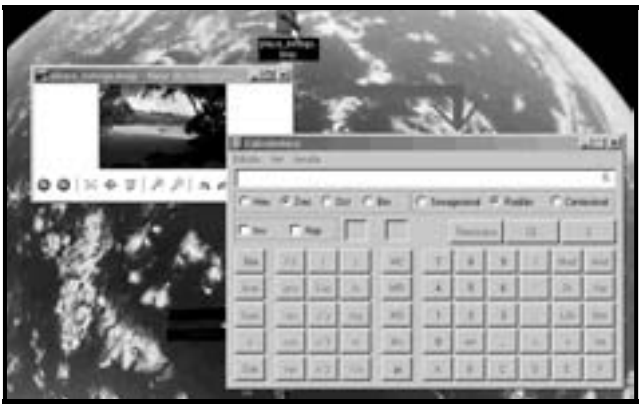


Fig. 7. Execution of calc.exe at the time visualizing 4 LSB steganographic image

The experiments were performed in other kind of images. For example images that present obscure colors like the one in the figure 8, present not degradation at all, even with its 4 LSB modified as in the case of figure 9. It is evident that with this kind of images is possible to conceal more information without raise suspicious unlike images with clear colors.

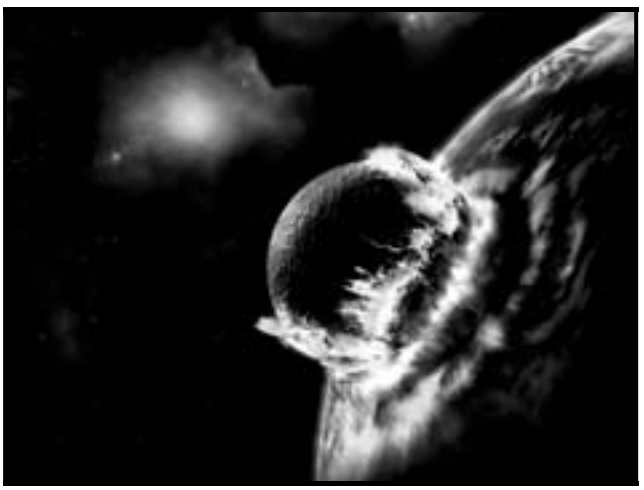


Fig. 8. BMP image with obscure colors

The results indicate that these images not alter its size and quality and hence they are the most appreciable candidates to be used in order to hide information.

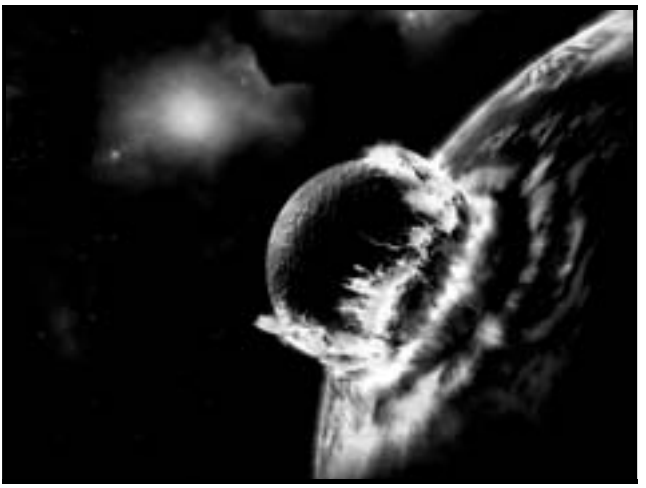


Fig. 9. BMP image with image with obscure colors modified in 4 LSB

7 Conclusions and future work

In this paper, we have presented a mechanism to use steganography in order to propagate executable code. We have developed an application to demonstrate that this is possible. The application allows embedding and extracting executable code from pictures in BMP format.

The BMP format is ideal to incrust data because of its excessive redundancy. The embedding capacity results indicate that pictures with obscure colors are the ideal medium to transport executable code without making them look abnormal in anyway. They present more capacity than clear images.

We showed a practical example hiding the code of windows calculator in different images. We proved that this code can be extract when the user selects the image carrying the code.

The major breakthrough in our approach is that this schema could be used to different kind of applications not necessarily to spread malicious code but also every kind of code. For example could be used as a way to improve copyright, executable code might be scanning, looking for anomalies in the way we use commercial software or even used as a way for authentication from both sides client and server applications o media.

Steganography of executable code is a vast field waiting to explore. One of our future works involves apply our system to assure that the code was not altered in its journey to a target system. Another area of exploitation is how can be install steganographic parsers without the intervention of a user in the system affected.

It is necessary looking for ways of compression and cipher the executable code before its embedding in the image file.

BMP format is really accepted in the traffic in Internet but because of its size is needed to analyze other image format files that can be spread more easily than BMP like GIF and JPEG, achieving this could reach a major degree of propagation than only with users of the BMP format.

Parallel to our work of hiding executable code in image file formats, is necessary to design and implement schemes of detection. The schemes must be able to detect executable code in mages to avoid the dangerous that this technology could bring at the computer community.

References

[CH03] Cole, E., Hiding in Plain Sight: "Steganography and the Art of Covert Communication", Canada: 2003 Wiliey Publishing.

[JDJ00] Johnson N.F.; Duric Z.; Jajodia S.: "Information hiding: Steganography and watermarking - attacks and countermeasures," Kluwer Academic Publishers, 2000.

[CGW05] Cauich E.; Gómez R.; Watanabe R.: "Data hiding in identification and offset IP fields", LNCS Springer Verlag, Fifth IEEE International Symposium and School on Advance Distributed Systems, ISSADS 2005, January 24-28 Guadalajara, Jalisco, México.

[CM01] Chandramouli, R.; Memon, N.:"Analysis of LSB based image steganography techniques Proceedings Image" Processing, vol 3, Greece, 2001. pp 1019-1022.

[Ca98] Cachin, C.: "An information-theoretic model for steganography," Proc. 2nd Information Hiding Workshop, vol. 1525, pp. 306-318, 1998.

[CPL95] Cha, S. D.; Park, G. H.; Lee, H. K.: "Solution to the On-Line Image Downgrading Problem", Annual Computer Security Applications Conference '95, Dec.13-15 1995. pp. 108-112.

[WP00] Westfeld A.; Pfitzmann, A.: "Attack on Steganographic Systems", Lectures Notes in Computer Science, vol. 1768, Springer-Verlag, Berlin, 2000, pp. 61-75.

[FGD02] Fridrich, J.; Goljan, M.; Du, R.: "Reliable Detection of LSB Steganography in Color and Grayscale Images", 2002.

[KP00] Katzenbeisser, S.; Petitcolas, F.: "Information Hiding Techniques for Steganography and Digital Watermarking", 2000 Artech House, Inc.

[Co00] Cochran, J.: "Steganographic Computer Warfare", Thesis, USAF, Air Force Institute of Technology, E.E.U.U., 2000.

[Si02] Singhal P. N.: A Possibility of Steganographic Trojan Installer. Network Security, IIT- Kanpur, Techkriti. 2002.

[Ro02] Rogers, M.: Steganographic Trojans DEF-CON X 2002.

[Mi04] Microsoft Security Bulletin MS04-028, Buffer Overrun in JPEG Processing (GDI+) Could Allow Code Execution. <http://www.microsoft.com/technet/security/bulletin/MS04-028.mspx> 2004.

[Al02] Alcopaul, EBCVG #2 magazine 2002.

Characterization and Generation of Synthetic Data Traces for IP Traffic Modeling

Marco Aurelio Turrubiartes Reynaga[1], Orlando Ezequiel Rincón Ferrera[2], Leopoldo Estrada Vargas[2], Deni Torres Román[2], David Muñoz Rodríguez[3] , Marlenne Angulo Bernal[1], Luis Rizo Domínguez[2]

[1]Universidad Autónoma de Baja California, México
[2] Cinvestav Research Center, Guadalajara, Jalisco, México
[3]ITESM Monterrey,N.L., México
{mturrubiartes, mangulo}@uabc.mx
{lestrada, orincon, lrizo, dtorres}@gdl.cinvestav.mx
dmunoz@itesm.mx

Abstract. fgN and fARIMA (0,d,0) are two of the models frequently employed in Internet traffic. However, there is not a set of data traces available. In this work, a set of 187 fGN data traces each one of 64KPoints were generated with three tools: S-Plus, SelQoS; and a fGN-MatlabScript; and a cumulative analysis with 1144 files was done. From this study, it is concluded that the best algorithms respect to the Minimum Mean Squared Error (MMSE) for Hurst index estimates were Modified Allan Variance in time domain far followed for R/S statistic, while in frequency domain Periodogram presented the MMSE followed very close for Local Whittle. A set of 53 "well-behaved" data traces with different theoretical values H_T were obtained and will be publicly available. The fGN data traces generated by our tool SelQoS compared with those generated with S-Plus, fGN, fARIMA (0, d, 0), and fGN-MatlabScript showed the best performance relative to MSE. For all data generation methods, the higher H is, the bigger MSE is.

Keywords: fractional Gaussian Noise fGN, fractional Auto-Regressive Integrated Moving Average fARIMA, Self-similarity, data traces, Hurst index

1 Introduction

Since the seminal work of Leland, Taqqu, Willinger, and Wilson [LTW94] in communication networks, an explosion of work has ensued investigating the multifaceted nature of this phenomenon, see for example Crovella and Bestavros [CB97]. Then, concepts such as self-similarity, long-range dependence (LRD), and heavy-tailed distributions (e.g. paretian- and α-stable distributions) are becoming more and more frequent in the telecommunications area. Consequently, these and others related concepts, such as autocorrelations or power spectral density have emerged as powerful tools for modeling the behavior of these real processes and

systems. Paxson and Floyd [PF95] explained "the Failure of Poisson Modeling" for packets networks where Poisson models for some telecommunication processes are not the most appropriate, e.g. network traffic or they don't capture well some behaviors. Abry and Veitch have made important contributions to the LRD in Internet's traffic see for example [AV98].

As time series or data with LRD appear quite frequently in many different areas of science and engineering, e.g., hydrology, physics, biology, telecommunications, many basic methods have been proposed for the estimation of the Hurst parameter *H*. However, some major issues have not been satisfactorily resolved:

- It is not clear for many cases under what circumstances the methods mentioned above yield consistent results.
- The mathematical theory of long-range dependence concerns the *behavior of the correlation of the time series* at large lags; however a measured time series is finite, and relatively short. This practical fact limits the study of: 1) the aggregated series, because theirs lengths decrease with the aggregation level; and 2) the cumulative behaviors.
- The set of well accepted software tools for data analysis is relative low, e.g. estimators of H, and the amount of data is huge; consequently, replication of results showed in the literature is a very time consuming task, see [Ka95]. Moreover, if the data trace source is not public or available, the task is unfeasible.

When the literature is studied, in the papers with more impact, besides of good algorithms, we find the following elements:

1. Sets of "good" synthetic data traces and/or traces obtained from measurements, and overall *publicly available*, sometimes as libraries.
2. A relative *high volume of measurements*.
3. Proprietary and public software tools and code.

When working with packet networks, engineers are interested in the estimations of several performance metrics and theirs behaviors respect to time, frequency and/or aggregation levels. Examples of these metrics are: volume of traffic per time unit, delays, delay variation or jitter, packet loss, inter-arrivals times, behaviors of some of these metrics per protocol, Internet routes and theirs anomalies, and Hurst parameter.

On the other hand, traffic generators are very important for good modeling. Therefore, in this paper we propose to adapt the colored noise generator model of Kasdin [Ka95], a digital model for accurate generation of digital sequences of noise with power law shapes $1/f^{\beta}$, in the frequency domain, to satisfy requirements of long range dependence and self-similar network traffic.

The generated noise traces are characterized in the frequency and time domain with the power spectrum and self-similarity estimation tools such as Whittle, periodogram, modified Allan variance method (MAVar). All the generated traces accomplish the randomness and the self-similarity index *H*.

1.1 Motivations

Although hundreds of papers has been written in this area, only a few of them show that the algorithms (estimators of H) should be calibrated respect to a set of "well-behaved" data traces with *H* theoretically known, that means data traces satisfying one or more

specific criteria. The major motivation is: *although fractional Gaussian Noise and fractional Auto Regressive Integrated Moving Average, fGN and fARIMA for short, are some of the traffic models frequently employed in Internet, there is not a set of "well-behaved" traces publicly available as reference to calibrate our implementations*.

1.2 Contributions

The main contributions of this work are: 1) A study of a set of 187 fGN and fARIMA (0, d, 0) data traces each one of 64KPoints (1KPoint = 1024 points) generated with three tools: S-Plus SelQoS, and fGN-MatlabScript; including a cumulative analysis for 1144 files (derived from the 187 datasets). 2) A study of the accuracy of algorithms for LRD based on the MMSE criterion is presented. 3) A set of 53 "well-behaved" data traces with different theoretical values H_T were obtained and will be publicly available.

2 Basic Concepts

2.1 Self-Similarity

Self-similarity describes the phenomenon where certain properties are preserved irrespective of scaling in space or time. It can be defined as follows:

Definition 1. A real valued continuous time stochastic process $\{Y(t), -\infty < t < \infty\}$ is said to be self-similar if for any constant $a > 0$, there exists $H > 0$, called index of self-similarity, such that

$$\{ Y(at),\ t \in \Re \} \overset{d}{=} \{ a^H Y(t),\ t \in \Re \} \qquad \forall a > 0 \in \Re \tag{1}$$

where $\overset{d}{=}$ means equality in the sense of finite- dimensional distributions.

A stochastic process $\{Y(t)\}$ is said to have stationary increments if the distributions of $\{Y(h + t) - Y(h)\}$ are independent of t. When $\{Y(t), t > 0\}$ is self-similar with stationary increments and its exponent is H, then it can be called *H-sssi*, for short. For a detailed study of second-order self-similar processes see Tsybakov and Georganas [8], where definitions of *exactly* and *asymptotically second-order self-similar processes* and some properties and relationships are discussed. Therefore, we are interested in *asymptotic second order self-similarity processes,* since exact second order *self-similar processes* require *invariability of second order statistics in all time scales*.

Second Order Self-Similar Discrete Time Series. When considering discrete stochastic time series the definition of self-similarity is given in terms of the aggregated processes. Let *{X(t), t = 0, 1, ..., N}* be a discrete time series derived from a process *H-sssi*, then others series can be obtained by aggregation. That means that new aggregated time series is a sequence given by the following eq. (2).

$$X^{(m)} = (X_k^{(m)} : k = 1, 2, 3, \ldots) \tag{2}$$

where each term $X_k^{(m)}$ is defined as:

$$X_k^{(m)} = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} X_i \qquad k = 1,2,3... \tag{3}$$

and where *m* represents the level of aggregation; that is, each new time series is obtained by partitioning the original time series into non-overlapping blocks of size *m* and then averaging each block to obtain the *k* values of the new series.

Definition 2. Let {X(t), t ε N} be a covariance stationary discrete time series with mean $\mu = 0$, variance σ^2 and autocovariance function $\gamma(k)$, and $X^{(m)}$ its aggregated series. Then it is said that X is H-ss, if the following eq. (4) is hold, i.e.

$$X^{(m)} \overset{d}{=} m^{H-1} X \tag{4}$$

and asymptotically *H-ss,* if the following eq.(5) is hold.

$$X^{(m)} \rightarrow_d m^{H-1} X \qquad m = 1,2,3,... \tag{5}$$

where $\rightarrow_d$ means convergence in distribution.

Definition 3. A discrete time series *X(t)* is exactly second order self-similar with Hurst parameter H ε (1/2, 1) if

$$\gamma_X^m(k) = \frac{\sigma^2}{2}\left((k+1)^{2H} - 2k^{2H} + (k-1)^{2H}\right) \tag{6}$$

for k > 0, and asymptotically second order self-similar if

$$\lim_{m\to\infty} \gamma^m(k) = \frac{\sigma^2}{2}\left((k+1)^{2H} - 2k^{2H} + (k-1)^{2H}\right) \tag{7}$$

where $\gamma^m(k)$ is the autocovariance function of the aggregated process of order *m*, *k* represents the lag, σ^2 the variance, and *H* represents the *Hurst* parameter.

2.2 Long-Range Dependence

Long-range dependence or long memory means that the autocorrelation function of a stochastic time series decays slower than an exponential or hyperbolically in time. This

slowly decaying behavior gives rise to the presence of large values of the time series *X(t)* with non-negligible probability.

Definition 4. Let *X(t)* be a stationary time series, it is said to be long-range dependent if its autocorrelation function, ACF, ρ(k) has the following asymptotic behavior.

$$\rho_X(k) \sim L_C(k)k^{-\beta} \qquad k \to \infty \qquad 0 < \beta < 1 \tag{8}$$

where $\beta = 2\text{-}2H$, *or* $H = 1 - \beta/2$, and $L_C(k)$ is a slowly-varying function, i.e. $\lim_{x\to\infty} L(tx)/L(x) = 1$. In *long-range dependence*, LRD, the sum of the autocorrelation function is infinity, i.e. $\sum_{k=0}^{\infty} \rho(k) = \infty$

2.3 Fractional Gaussian Noise fGn

Definition 5. *Fractional noises* are obtained by taking the increments of Fractional Brownian motion processes $\boldsymbol{B_H(j)}$, i.e. fractional Gaussian Noise fGn $\boldsymbol{Y_j}$ is defined as

$$Y_j = B_H(j+1) - B_H(j), \quad j = ...,-1,0,1,... \tag{9}$$

2.4. Method of generation for fractional noise based on Kasdin's algorithm

In this section a summary of the method developed by Kasdin [Ka95] is given. Simulation of a stochastic process is not an easy task, because most of processes are non stationary. However, many of the process' models used for simulation are asymptotically stationary. The output process, *x(t),* of a linear system driven by white noise is represented by the convolution integral:

$$x(t) = \int_0^t h(\theta)w(t-\theta)d\theta \tag{10}$$

Where *h(t)* is the causal impulse response function of the system and *w(t)* is Gaussian white noise with autocorrelation function $Q\delta(\tau)$ and the integration is defined in the sense of Ito [ST94]. The spectral estimation, even of a non-stationary process, is in most of cases, a legitimate basis for formulating linear systems models for the stochastic processes [Ka95].

In the case of a discrete linear system, the convolution integral in (10) can be replaced by the discrete convolution sum:

$$x_k = \sum_{a=0}^{k-1} h_a w_{k-a} \tag{11}$$

Where w_a is the iid white noise sequence.

We can generate the sequence x_k of the eq (11) by performing the discrete convolution. Another way to perform this is multiplying in the frequency domain their components in

the spectrum, using the FFT. Brownian motion is given as the integral of white noise. Its impulse response function is thus the unit step function and its transfer function is:

$$H(z) = \frac{1}{1 - z^{-1}}, \qquad z > 1 \tag{12}$$

A generalization of (12) proposed as the digital model in[6]:

$$H(z) = \frac{1}{(1 - z^{-1})^{D/2}}, \qquad z > 1 \tag{13}$$

Recalling that D = 2H-1 (13) can be expressed in terms of H instead of D.

$$H(z) = \frac{1}{(1 - z^{-1})^{H - \frac{1}{2}}}, \qquad z > 1 \tag{14}$$

Applying the inverse Z transform to (14) as shown in [Ka95] and computing it using a recursive algorithm:

$$ho = 1 \tag{15}$$

$$h_k = \frac{h_{k-1}}{k}\left(H + k - \frac{3}{2}\right)$$

Where, H is the Hurst parameter.
Consequently, at this point we can generate different fractional noises, as fGn, controlling the self-similarity index H. This algorithm was implemented in Matlab, it is called in this work as fGN-MatlabScript.

2.5 Estimators of Self-Similarity Index for Discrete Time Series

Hurst index as a single value is well-defined mathematically, but its measuring is problematic. The data must be measured at high lags/low frequencies where fewer readings are available. All estimators are vulnerable to trends or periodicity in the data, high-frequency oscillations, non-stationarities and other sources of corruption. In order to estimate the index of self-similarity, several different *estimators* for H should be implemented. Hurst estimators, $\hat{H}$, can be classified in three general groups: those operating in the *time domain*, those in the *frequency, and the third in wavelet domain*. Due to space constraints we can't give a complete description of all available estimators, but an overview appears elsewhere [TT98]. The most frequent are: R/S statistic, Absolute moment, Variance, Variance of Residuals, Periodogram, Local Whittle and Abry-Veitch. A less common estimator is based on the Modified Allan Variance and it was developed by Bregni and Primerano [BP04].

3 Behavior of H Estimators for Synthetic Data Traces fGN and fARIMA (0, d, 0) with H known

Many aspects of H-behavior are being studied. The present work focuses on the H-behavior respect to one or more algorithms in order to get a set of "well-behaved" data traces, which are analyzed in the following way: *global values or global sense*, *series of local values or local sense*, and *series of cumulative values or cumulative sense*.

Let $\mathbf{\Omega^H}$ be a finite set of data traces, where each element is represented by $\mathbf{\Omega_i^H} = \{D^i_j, N, M, H_T\}$, and *where* D^i_j $j = 1, 2, 3, \ldots, N$, are de data, N is the length of the series, M is the method for data generation, and H_T theoretical value of the Hurst parameter. H can be studied in several ways, e.g. in a *global sens*e, i.e., one single value; in a *local sense*, i.e., the time-interval is partitioned into K non-overlapping blocks and for each one a H-value is estimated; obtaining k values H_i, $i = 1, 2, \ldots, K$; in a *cumulative sense*, i.e., how H change for blocks of size jN/K for $j = 1, 2, \ldots, K$; and *where each block include its predecessor*, in this case we obtain again k values H_i, $i = 1, 2, \ldots, K$. As there are different H estimators, different values of H and different behaviors of H can be found. Different criteria for "well-behaved" traces can be done, in this work we employs the following:

C1. Criterion for the global behavior of H respect to an estimator. Given a finite set of traces $\mathbf{\Omega^H}$ with "equal" H_T, and a specific estimator of H, $\hat{H}_E$, the trace $\mathbf{\Omega_i^H}$ is *well-behaved in a global sense respect to* $\hat{H}_E$, if its estimated H-value is the closest to the theoretical or reference value H_T.

C2. Criterion for the local behavior of H respect to an estimator. Given a finite set of traces $\mathbf{\Omega^H}$ and a specific estimator of H, $\hat{H}_E$, the trace $\mathbf{\Omega_i^H}$ is *"well-behaved" in a local sense respect to* $\hat{H}_E$, if one of the following condition is hold: its expected value, $E(\hat{H}_E)$, *is the closest to the global H-value; or the minimum mean squared error MMSE(* $\hat{H}_E$*)* is obtained.

Mean squared error of the estimator $\hat{H}$ (MSE ($\hat{H}$)) is given by

$$\text{MSE}(\hat{H}) = \text{Var}(\hat{H}) + [\text{E}(\hat{H}) - \text{H}_\text{T}]^2 \qquad \textbf{(16)}$$

where H_T is the reference value.

C3. Criterion for the global behavior of H respect to a set of estimators. Given a finite set of traces $\mathbf{\Omega^H}$ with "equal" H_T and a set of algorithms estimators $\hat{H} = \{\hat{H}_{E1}, \hat{H}_{E2}, \ldots, \hat{H}_{Ek}\}$, the trace $\mathbf{\Omega_i^H}$ is *"well-behaved" in a global sense for a set of estimators* if it has the best estimated H-value for all estimators.

C4. Criterion for the local behavior of H respect to a set of estimators. Given a finite set of traces $\mathbf{\Omega^H}$ with "equal" H and a set of algorithms estimators $\hat{H} = \{\hat{H}_{E1}, \hat{H}_{E2}, \ldots, \hat{H}_{EK}\}$, the trace $\mathbf{\Omega_i^H}$ is *"well-behaved" in a local sense for a set of estimators* if one of the estimators $\hat{H}_{Em}$ satisfies the *criterion for the local behavior of H respect to an estimator, C2*.

C5. Criterion for the cumulative behavior of H respect to a set of estimators. Given a finite set of traces $\mathbf{\Omega^H}$ with "equal" H and a set of algorithms estimators $\hat{H} = \{\hat{H}_{E1}, \hat{H}_{E2}, \ldots, \hat{H}_{EK}\}$, the trace $\mathbf{\Omega_i^H}$ is *"well-behaved" in a cumulative sense for a set of estimators* if one of the estimators $\hat{H}_{Em}$ satisfies a *specific criterion of convergence involving accuracy and/or precision, for example MMSE*.

4 Generation, Selection and Analysis of Data Traces

In order to find a set of "well-behaved" traces of fGN and fARIMA (0, d, 0) sample paths, several traces should be generated and processed, therefore a tool is necessary. *The results obtained in this work were obtained using a software tool, called by us SelQoS, which is being developed at CINVESTAV-Guadalajara. This tool allows global, local and cumulative analyses of Hurst parameter and some basic statistics, among others features,* and a version will be publicly available and free in three months. After the analysis of many data traces, we observe that.

1. The *criteria for the global behavior of H respect to one or more estimators* are not strong because you don't see the convergence of the H-value, neither the behavior of H-for different blocks. Karagiannis in [KFM04] shows a global behavior of some estimators of H.
2. The *criterion for the local behavior of H respect to a set of estimators,* is better than the above mentioned, *and* allows to know the H-behavior for different blocks, but not its convergence.
3. The algorithm of Variance of residuals showed a high variance for local and cumulative analyses, and then it was not taken into account.

Consequently, we used the *criterion for the cumulative behavior of H respect to a set of estimators* to find the set of *"well-behaved"* data traces of fGN. *The estimators* $\hat{H}_E$ *employed were:* R/S statistic, Absolute moment, Variance, Periodogram, and Local Whittle, because they are implemented in our tool. Local and cumulative analyses of self-similar traces can be found in [RT06].

In order to obtain a set of "well-behaved" synthetic data traces, the following steps were done:

1. Generation of *5* synthetic data traces fGN for each Hurst values between 0.5 and 1 (using a step of 0.05), and for different lengths: 16, 32 and 64 KPoints.
2. For the data generation were we used S-Plus for fGN and fARIMA (0, d, 0), and SelQoS for a third set of fGN traces.
3. Selection of the data traces of 64KPoints according to the criterion C5.

Remarks about location of data traces. The set of all generated and selected data traces used in the present work can be downloaded from http://orion.gdl.cinvestav.mx/jcinv/minisitios/dtorres/.

4.1 Selection

Criterion selection and classification: Based on cumulative analysis, selected data traces can be classified into three groups: *golden traces* are those with MMSE($\hat{H}$) for all methods; *silver traces* are those with 3 or 4 MMSE($\hat{H}$) at least one of them in frequency domain, and *bronze traces* are those with at least two MMSE($\hat{H}$).

4.2 Analysis

A total of 187 files or data traces were studied, the *cumulative analysis of the Hurst parameter* was realized with a block size of 1KPoints. After that, MSE($\hat{H}$) versus theoretical value of Hurst index H_T was computed for all data traces. Finally, the Cumulative MSE (CMSE) was calculated in two ways: the first one for each estimator $\hat{H}_{Ek}$, the same H_T, and for a group of 5 data traces; and the second one for each data trace and all estimators.

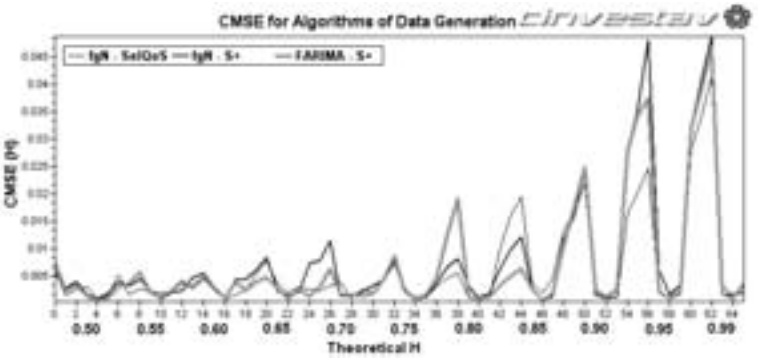


Figure 1: CMSE for the three algorithms of data generation

In Fig. 1 the behaviors of MSE for the different methods are depicted, the higher H is, the bigger MSE is. *From the Criterion for selection and classification*, the following results were obtained: *one golden trace* generated by SelQoS with $H_T = 0.85$, filename "fgN – SelQoS – 65536 – 0.85 – B.txt"; *15 silver traces*, 5 of each data generator; and *37 bronze traces*, among them 10 from fgN - SelQoS, 16 from fgN – S Plus and 11 from fARIMA (0, d, 0) – S Plus. Therefore, a set of 53 "well-behaved" traces with different theoretical values H_T were obtained.

Compared with the traces generated by SelQoS and S-Plus, the data traces generated by fGN-MatlabScript don't have the best performance, relative to MSE. But in many cases, this MSE was sufficiently low (in the order of 10^{-5}). An analysis done using alpha-stable distributions showed that parameters "alpha", "beta" and "gamma" of these fGN-MatlabScript traces were very similar to those generated by the other methods, but the localization parameter "delta" related with the mean was relatively high (not zero), and it grows as the Hurst parameter does.

4.3 Global behaviors: Basic Statistics, Probability Density Function and Outliers

Due to space constraints, only the golden trace analysis is presented in this section. The first step to study a time series is a first look in time to it, see Fig.2 Y(t) is a time-series with $\mu = 0$, $\sigma^2 = 255$, $\beta = 0.0080$, and kurtosis = 3.01; its Gaussian PDF is depicted too. Table 1 shows H estimated for different estimators.

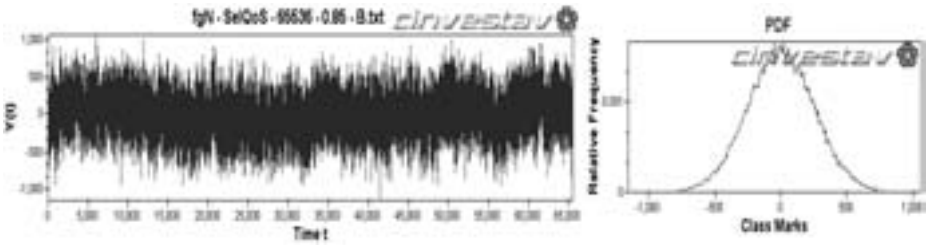


Figure 2: A first look to the golden trace "fgN – SelQoS – 65536 – 0.85 – B.txt" and its PDF

RS	AM	Var	VoR	MAV	Per	L.Whi
0.8666	0.8636	0.8568	0.8878	0.8587	0.8436	0.8525

Table 1: Global Hurst of the golden trace

4.4 Local Behaviors of H

A more precise study of the time series is achieved when it is divided into small blocks and local and cumulative analyses are done. Our tool SelQoS allowed the study of basic statistics and estimators $\hat{H}$ in a more detailed form. In this case, we used block size of 1024 Points (64 blocks) for MSE calculations. Excepting some blocks, good behavior of almost all time and frequency-domain estimators is observed respect to the theoretical value of H.

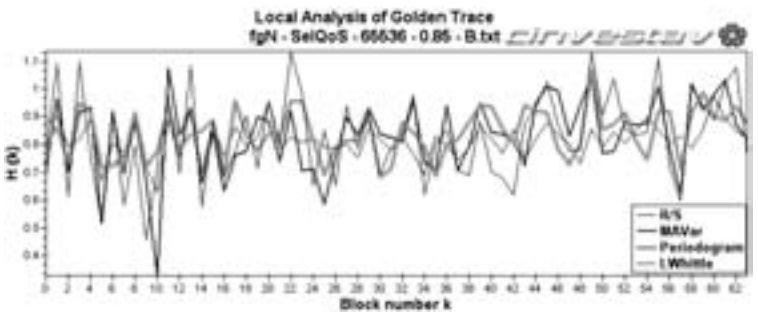


Figure 3: Local behaviors of H using some time and frequency-domain estimation algorithms

4.5 Cumulative Behaviors of H

The cumulative analysis proved that some estimators present a better convergence than others. Fig. 4 shows the behaviors of the best, *fGN – SelQoS – 65536 - 0.85 - B.t*xt, and worst data traces, *fgN - S+ - 65536 - 0.70 - D.txt*, both respect to MSE. From block 18, the convergence of the best data trace is significant better than the second one; this fact is emphasized from block number 44 until the end.

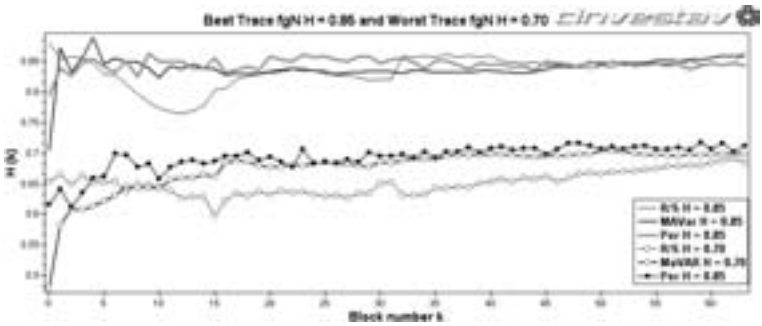


Figure 4: The best and the worst data trace

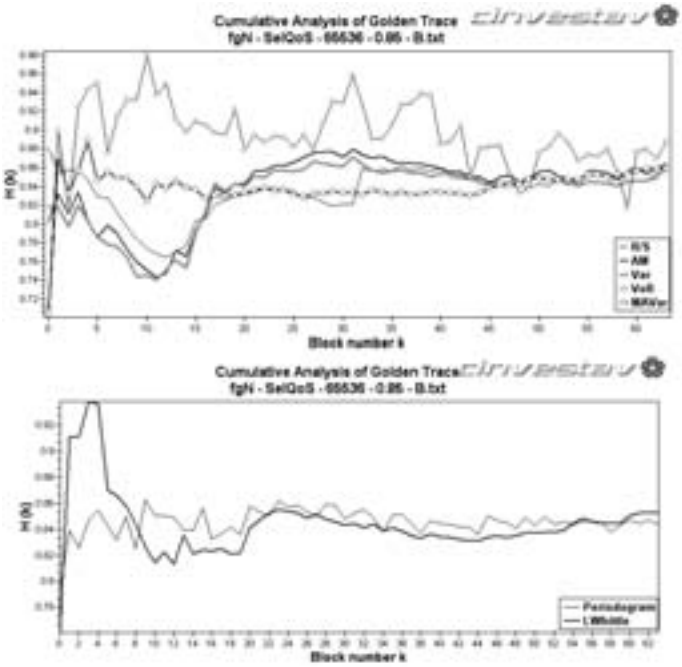


Figure 5: Cumulative behaviors of H using time and frequency-domain algorithms

The behavior of global estimated H for different estimators of the data trace fGN-SelQoS-65536-0.85-B.txt, H = 0.85 and aggregation levels m = 8, 16 and 32 was studied and, as trend, H decreases as the aggregation level increases. H estimates of this series using different methods took values greater than 0.78 and less than 0.91.

5 Conclusions

From a source of 187 fGN data traces, each one of 64KPoints, a cumulative analysis with 1144 files was realized and the following was concluded: in the time domain the algorithm with MMSE was *Modified Allan Variance*, far followed for *R/S statistic*; while

in frequency domain was the *Periodogram* presented the MMSE followed very close for *Local Whittle*.

Based on cumulative analysis from the selected data traces we obtained *1 golden data traces, 15 silver traces* and 37 *bronze traces*. The golden fGN data trace was generated by SelQoS with a length of 64 KPoints and H = 0.85. *A set of 53 "well-behaved" data traces with different theoretical values H_T were obtained*.

The fGN data traces, generated by our tool SelQoS[1], compared with those generated with S-Plus, fGN, fARIMA (0, d, 0) and fGN-MatlabScript (based on Kasdin's algorithm) showed a better performance relative to MSE. In all data generation methods, the higher H is, the bigger MSE is. A set of data traces of fGN will be publicly available at http://orion.gdl.cinvestav.mx/jcinv/minisitios/dtorres/.

References

[LTW94] Leland, W. E., Taqqu, M. S., Willinger, W., Wilson, D. V.:,On the Self-Similar Nature of Ethernet Traffic. IEEE/ACM Transactions on Networking (1994), pp. 1-15

[CB97] Crovella, M. E., Bestavros, A.: Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. IEEE/ACM Trans. Networking, Vol. 5, No. 6(1997), pp. 835-846.

[PF95] Paxson, V., Floyd, S.: Wide Area Traffic: The Failure of Poisson Modeling. IEEE/ACM Trans. Networking. Vol. 3, No. 3 (1995), pp.226-244

[AV98] Abry, P., Veitch, D.: Wavelet analysis of long range dependent traffic. IEEE Transactions on Information Theory, Vol. 44, No. 1 (1998, pp. 2-15

[Ka95] Kasdin, J.: Discrete Simulation of Colored noise and Stochastic Processes and 1/fα Power Law Noise Generation. Proceeding of the IEEE Vol. 83 No. 5 May 1995

[ST94] Samorodnitsky G. and M. S. Taqqu: Stable non Gaussian Random Process: stochastic models with infinite variance, Chapman and Hall, 1994. – Chapters 1, 2 and 7

[KFM03] Karagiannis, T., Faloutsos, M., Molle, M.: A User-Friendly Self-Similarity Analysis Tool. Special Section on Tools and Technologies for Networking Research and Education, ACM SIGCOMM Computer Communication Review (2003)

[TG98] Tsybakov, B., Georganas, N.: Self-similar Processes in Communication Networks. IEEE Transactions on Information Theory 44, Vol. 5 (1998), pp. 1713-1725

[TT98] Taqqu, M. S., Teverovsky, V.: On estimating the intensity of long-range dependence in finite and infinite variance series. In: Adler, R., Feldman, R., Taqqu, M. S. (eds.): A Practical Guide to Heavy Tails: Statistical Techniques and Applications. Birkhauser, Boston (1998), pp. 177-217

[BP04] Bregni, S., Primerano, L.: The Modified Allan Variance as Time-Domain Analysis Tool for Estimating the Hurst Parameter of Long-Range Dependent Traffic. Globecom (2004)

[KFM04] Karagiannis, T., Faloutsos, M., Molle, M.: Long Range Dependence: Ten years of Internet Traffic Modeling. IEEE Internet Computing (2004), pp. 57-64

[RT06] Ramírez, J. C., Torres, D.: Local and cumulative analysis of self-similar Traffic Traces. Proceedings of the 16th IEEE International Conference on Electronics, Communications and Computers CONIELECOMP (2006)

[1] The authors gratefully acknowledge the CONACYT's grants for the Master Program at CINVESTAV-Guadalajara, and the people that have collaborated in the SelQoS development process.

Chapter 7: Reliability and Availability

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Gert Pfeifer, Christof Fetzer, Martin Steuer
Rearchitecting DNS

Fares Saad Khorchef, Ismail Berrada, Antoine Rollet, Richard Castanet
Automated Robustness Testing for Reactive Systems: Application to Communicating Protocols

Harry Gros-Desormeaux, Hacène Fouchal, Philippe Hunel
A Distributed Cache Management for Test Derivation

Rearchitecting DNS

Gert Pfeifer Christof Fetzer
Martin Steuer

Dresden University of Technology, 01062 Dresden, Germany

{firstname.lastname}@inf.tu-dresden.de

Abstract: The Domain Name System (DNS) has been the naming service of the Internet for more than 20 years. It is the foundation of virtually all other distributed service. The deficiencies of DNS with respect to performance, availability, and reliability are well known. We want to systematically improve DNS by (1) measuring DNS performance, (2) evaluating past proposals, and (3) try to come up with a new architecture for DNS. SEDNS is a system in which we investigate various design ideas to speed up DNS requests and improve availability, reliability, and security.

1 Introduction

The Domain Name System (DNS) is a hierarchical distributed database that can map names to some data. In most cases it is used to map a name to an IP address. The system is mature and very successful. There is hardly any distributed Internet-based application that does not use it. Unfortunately DNS is not always sufficiently dependable for companies that must depend on it. Almost always some DNS servers are not reachable. Sometimes none of the DNS servers might be available (e.g., when DNS is misconfigured or under attack). DNS responses are not always correct, which may have different reasons, e.g., cache poising. However, the most alarming problem is that DNS can be used to attack other services or a downtime of DNS can make services unusable. Many systems depend on DNS without having a fallback. So a good way to attack a distributed application is, first to make sure that itself or its clients cannot resolve names anymore.

We are designing a system, SEDNS, to cope with all these problems. SEDNS is not a replacement for DNS. Replacing DNS would be difficult since it is mature, quite well understood, popular, and such a shotgun approach often fails. So we will continue to use DNS and, just in case that it does not work according to our expectations, we provide a combination of local caches together with a peer-to-peer based fallback system. We show how to optimize the performance using local caches similar to Supercaches as introduced by [AMT03].

The rest of the article is organized as follows: Sections 2.1 and 2.2 discuss the different sources for unavailability of DNS. Section 3 discusses DNSSEC in this context. Section 4 introduces our SEDNS approach and discusses how to optimize a Peer-to-Peer overlay for

this use. Section 5 concludes the paper.

2 DNS Dependability Issues

DNS is inherently unreliable, due to the fact that it is using plain-text communication via UDP as the default setting. Hence, modifications of the content can not be detected and in the case of packet loss, retries include the resolvers time-out as an extra delay penalty. TCP, which would solve this problem, is only used when the amount of data is too large to be contained in a single UDP packet. Of course, TCP introduces some overhead so that UDP seems to be the better choice if performance is the main concern.

The consequence of using UDP is that applications have to deal with requests not being answered or replies being dropped. To optimize yield, i.e., the ratio of answered requests out of the number of requests sent, we need to investigate the factors that impact the yield. We say that a failure occurred if the time-out of the resolver has expired, i.e., 5 seconds in most implementations, or that an error is returned by the server. Failures can be classified in client-side and server-side failures, like proposed by Park et al. [PPPW04].

Client-side failures are failures caused by the client's infrastructure, i.e., the stub resolver, the recursive resolver, or the LAN connection. Server-side failures are failures caused by the DNS servers involved in the processing of the request, including WAN communication failures. The following subsections describe these problems in more details.

2.1 Server-side problems

When a recursive resolver tries to find a certain resource record, it iteratively traverses the DNS name space. Starting from the top of the tree, i.e., a root server, it tries to find a DNS server that is responsible for the top level domain (TLD). From this server the search goes on from name server to name server until the request can be answered.

Since a successful DNS lookup depends on the cooperation of many remote components it is worth looking how reliable this process is. Jung et al. [JSBM01] tried to examine the effectiveness of DNS caching. In addition, their work shows the reliability of the DNS server infrastructure. They captured DNS packets at two locations, i.e., two Internet gateways. The traces show that over a third of all lookups where not successfully answered. The number of query packets is larger than the number of lookups because if a DNS server does not receive an answer for a query, it retransmits the query. This is a robustness mechanism that deals with UDP packet loss, leading to the fact that query packets for unanswered lookups account for more than 50% of the DNS query packets in the trace.

Another interesting behavior is that between 3.1% and 4.9% of the unanswered lookups formed a querying loop between two or more DNS servers. This is obviously a configuration problem. Erroneous configurations also cause many negative answers. Negative answers account for 10% to 42% of the answers in the trace. Often they are caused by

querying a name that does not exist. Of cause, this might be a result of user input, such as ”ww.att.com” in the address field of a browser, but the largest cause are inverse lookups.

Usually an administrator configures at least two mappings for a name. One mapping points from the name to the IP address, another one points from the IP address to the name. This is very useful and many services are using this feature, e.g., ssh. Obviously many administrators omit configuring these inverse mapping.

Other significant causes are NS or MX records that point to names that do not exist. This might be a problem of inconsistency in a DNS zone, where old entries have not been removed or new entries have not been checked for typos.

Apart from these configuration issues there is also the problem that servers might fail completely. Reasons are software and hardware failures as well as occasional denial-of-service (DOS) attacks.

DNS is a compelling target for DOS attacks since DNS servers are well known and easy to find, a successful attack to a single server or pair of servers can cut of a whole branch of the DNS name space, and DNS outages often cause outages of other services as well. Such an attack (in this case ping flooding) hit the 13 root servers on October 23, 2002, causing some slowdown in the Internet [Nar02]. Only four root servers were able to continue service during the attack. In January 2001, Microsoft suffered from such an attack resulting in downtime for its Web presence [Thu01]. As a result, Microsoft hired Akamai Technology to ensure that such attacks do not succeed in the future. Akamai maintains geographically separated DNS servers, while all of Microsoft's DNS servers were in the same location and thus easy to attack.

2.2 Client-side problems

Many DNS outages are not caused by global network outages or server failures, but by problems with local components, like recursive resolvers or stub resolvers. Stub resolvers are using local DNS servers (LDNS) as recursive resolvers. These servers are responsible for resolving names iteratively and also provide a cache for DNS resource records. Hence, they do the main part of resolving a name. This is necessary since stub resolvers are too limited for this task.

If these LDNS servers suffer from network problems, more or less frequent overload situations, or hardware and software failures, the DNS is unavailable. Now this seems to be a problem that can be solved by the DNS administrators. As discussed by Park et al. in [PPPW04], there might be reasons that are deterring administrators from solving this problem: Most of them optimize cost-to-benefit ratio expressed by CPU utilization. DNS is not CPU bound and it typically leaves enough CPU power to run other services on the same machine. Installing more services improves utilization but introduces a competition between services for resources. Waiting for memory or CPU, DNS might fail from time to time, i.e., let the resolvers time-out expire.

Park et al. propose to use an insurance-like service (CoDNS) where many partners share

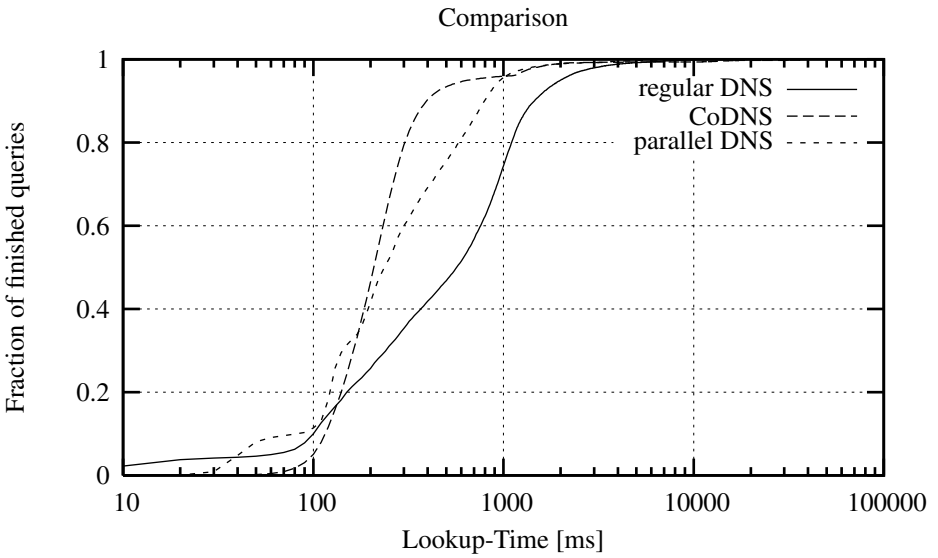


Figure 1: Comparison of average answer time without caching

their recursive resolvers. DNS requests can be forwarded, using a peer-to-peer overlay network, to other recursive resolvers. The performance of the system depends on a good selection of time-out values and knowledge about the health of other peer's LDNS servers. According to benchmarks in [PPPW04] CoDNS outperforms standard DNS by 27-82 % (latency) and increases availability by an extra 9. This can be reached by exchanging information about the health of the LDNS server, i.e., its performance and availability, with other peers in an overlay network. If the LDNS server fails, the DNS request is forwarded to a nearby overlay peer with a good LDNS server. CoDNS is deployed in Planet-Lab [Ros05], where we are using it to compare it with the front-end for SEDNS. It is configured using one extra peer for lookups and 200 ms time-out for the initial DNS request to the LDNS server. The overhead that is needed is, to monitor 10 other peers and keep track of their LDNS server's health.

SEDNS uses a modified stub resolver to query a super cache first. After a short time-out, other LDNS servers are queried in parallel requests. Using several LDNS servers masks client-side problems. In the case that this still does not obtain an answer, another local resolver is queried. It is using a peer-to-peer approach to find DNS data even in the presence of server-side problems.

Two benchmarks are presented here: (1) We measure our front-end without caching, i.e., we try to eliminate caching as good as it gets[1], and (2) with warmed-up caches. The first setting shows that the extra effort of almost doubled number of DNS messages is sensible and that performance is good in the case that the SEDNS super-cache does not produce

[1]We did not have exclusive access to the DNS servers. So our measurements have been influenced by each other and other resolvers.

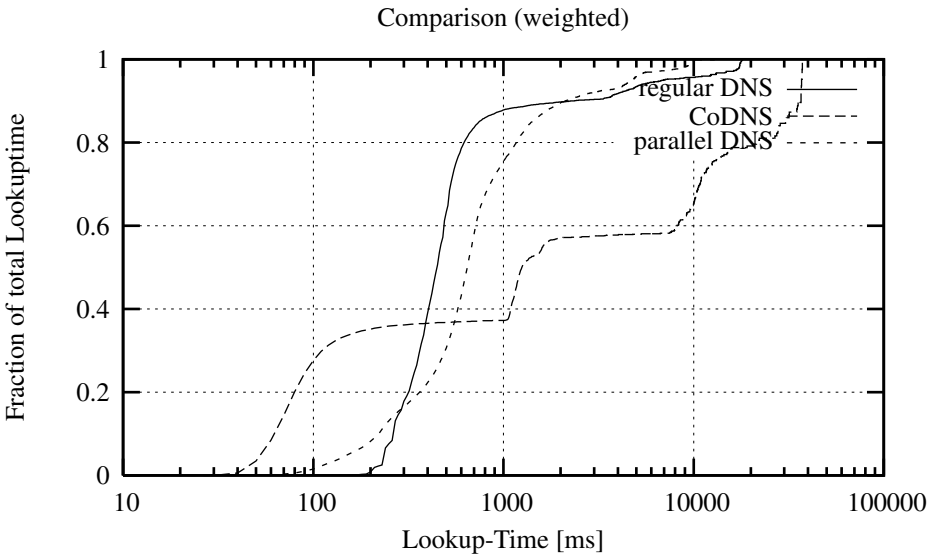


Figure 2: Comparison of weighted average answer time without caching

a hit. The second one shows the influence of LDNS and network failures on the request delay. The result is that our approach is most effective in masking client-side failures.

For using SEDNS the user must install a glibc patch for stub resolvers and the local administrator has to set up additional resolvers and a super cache in the LAN. SEDNS is usable even in the case that DNS is globally unavailable, which is not possible with CoDNS since it relies on DNS.

In comparison to CoDNS, we do not require a peer-to-peer component to be installed on each client machine, even though it would be possible, of course. Neither do we monitor the health of DNS servers.

Our approach separates different techniques to reach different goals as discussed in Sec. 4. We are still early in the design and development process, but we can already present some promising measurements.

So far we have implemented a glibc extension that uses parallel requests to the primary and secondary name server. Of course this doubles the DNS traffic in the case that both of them are doing fine, but we don't need any monitoring traffic and we keep the caches of both servers up to date. If the added traffic is of concern, one can delay the second request by a short time-out sufficient to get fast replies from the first server. In this way, one can balance the network load vs. the response time. Given the fact that most requests will be answered from cache, there is almost no additional load on the CPU, and LAN traffic is cheap, so this seems to be a practical approach.

Figure 1 shows a comparison between our resolver (parallel DNS), CoDNS and DNS. We used a list of 23771 DNS names and calculated the average of our measurements from 490

Method	lookups	total time	avg. time/lookup
DNS	23771	18074599ms	760ms
CoDNS	23771	9735546ms	409ms
parallel DNS	23771	9176794ms	386ms

Table 1: Results without caching

Planet-lab nodes[2]. We found out, that measuring the performance without caching effects is difficult. The graph shows that parallel DNS has gotten more responses within 40 and 200 ms than any of the other approaches. CoDNS has advantages for answers between 200 and 1000 ms. This is due to the ability to choose the best performing peer out of 10 neighbors while we just use our LDNS servers. However, we did not succeed in getting rid of caching effects between the different measurements, which makes evaluating the benchmark difficult. This is also caused by the fact that we had not the exclusive access to the used DNS servers. Without these effects our parallel resolver should always be at least as good as DNS.

For better comparison we provide a weighted CDF[3] in Fig.2. It shows the distribution of answer times weighted by the answer time itself. This is interesting because the extremely slow answers dominate the time, that a resolver spends for resolving queries. One can see that, e.g., our approach uses 21% of the overall lookup time for requests taking longer than 1 second. DNS spends 60% and CoDNS spends almost 50% of the overall lookup time for this fraction. This result differs from the benchmarks in [PPPW04], because these are our own measurements, where we used the CoDNS version that is available and deployed in Planet-Lab. Altogether our benchmark results show that parallel DNS outperforms the other approaches as shown in Table 1. The table shows that the additional number of DNS messages is justified by roughly 50% improvement of answer time.

Since DNS gains most of its performance from caching, we performed a measurement in which we repeated some queries to exploit caching. The main benefit of this benchmark is to isolate the delay penalty of LDNS failures. A fair measurement with caching is difficult to achieve, since the chronological order of benchmarks has an influence on the result. KyoungSoo Park proposed to show the better performance of an approach by running it first. All other approaches would find better caching behavior since the first run warmed caches. This might be unfair because the result would hardly be comparable with other benchmarks. Repeating small junks of queries solved this problem, since the 2nd junk always finds a warm cache[4]. Figure 3 and Table 2 show the results. Parallel requests always reach the best results. The reason is that LDNS or network failures are the reasons for higher delay in this case. Parallel requests perfectly hide these failures while the other approaches trigger timeouts and need retransmissions.

We also learned that asynchronous socket I/O is much better than multithreading. To simplify matters, we first used threads that parallelized calls to the glibc function *send_dg*

[2]Not all of them were available all the time. Reasons can be network problems in the Internet, nodes being re-booted after OS updates, node failures, and other problems. Usually we have been able to use half of them.

[3]as used by Park et al. [PPPW04]

[4]Of course this does not work for names that are not cacheable due to TTLs of just a few seconds. But this makes the method even more realistic.

Method	lookups	total time	avg. time/lookup
DNS	23771	5164570ms	217ms
CoDNS	23771	4169531ms	175ms
parallel DNS	23771	984413ms	41ms

Table 2: Results with caching

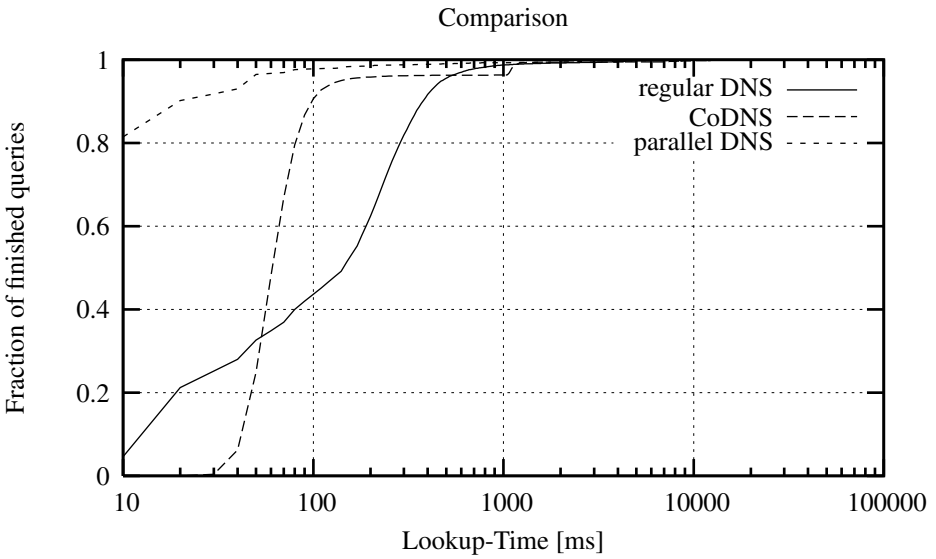


Figure 3: Comparison of average answer time with caching

that sends and receives UDP packets containing the DNS messages. When we tried to use a thread pool the performance dropped under the DNS performance in the area up to 16 ms.

CoDNS tries to pick the same peer for the same name to get some cache locality, but in the Planet-Lab configuration it uses just one peer, so this strategy might not even be used. However, CoDNS can also outperform DNS in the area between 40 and 500 ms.

Another important point for comparison is trustworthiness. Using CoDNS, the user agrees to forward DNS requests to other peers. Of course, this leaks information about the DNS names she resolves, i.e. the web sites she reads or the e-commerce services she uses. Furthermore, information about the internal structure of a company's network may also be disclosed. Using SEDNS, users can prevent this by carefully configuring their LDNS servers.

3 DNS Security Issues

Many security attacks on DNS are possible. DNS messages could be intercepted and modified on their way. This would not only be a source of confusion but would be a serious threat for many systems.

DNS Security Extensions (DNSSEC) have been designed to cope with this problem. A public key infrastructure is woven into DNS to allow using signed resource records.

With DNSSEC, the manipulation of DNS resource records is getting more difficult. An attacker can not just intercept a DNS message and change its contents. Sending a faked DNS answer directly to the stub resolver should also not be possible, since a security aware resolver, according to RFC 2535, must not trust the AD flag unless there is a secure communication channel between itself and the server, e.g., a VPN, which would prevent such attacks.

Hence, the introduction of a public key infrastructure solved many trustworthiness problems - but at the same time new problems are introduced. We have seen in Sec. 2.1 that configuration issues cause many failures. With DNSSEC the configuration of a DNS server gets even more difficult.

DNSSEC uses a hierarchical trust model that makes DNS even more fragile in the presence of configuration errors. Hence, if we want to improve the availability, DNSSEC does not seem to be the right choice. A solution that could reduce the configuration effort is a self-organizing completely decentralized peer-to-peer system. But it is already well understood, that these overlays without massive use of caching are too slow to satisfy the answer time expectations of a naming service [CMM02]. The complexity of the DNSSEC trust model can be reduced by using SSL-Certificates like described in [FPJ05]. This solutions helps to reduce DNS server dependencies, since the certification chain can be much shorter.

4 SEDNS

SEDNS tries to cope with all the dependability problems mentioned so far. We dedicate a part of our systems design to each of the problems: a local cache to improve QoS by reducing DNS latency, parallel requests to LDNS servers to mask client-side dependability problems, and a peer-to-peer approach to mask server-side dependability problems including misconfigurations and network problems. Figure 4 shows how an SEDNS resolver works in pseudocode.

Step 1 The resolver sends a query to a nearby caching-only name server. (Line 9 in Fig. 4) These servers are easy to install and the maintenance overhead is negligible. The main advantage is that the stub-resolver can learn, how long it takes to resolve a name from cache (Lines 21, 22). The time-out for this operation is dynamically adapted.

Step 2 In the case of a cache-miss, the resolver does not wait for the caching-only server to resolve the name. It proceeds to issuing parallel requests to the LDNS Servers. The earliest successful answer is delivered to the application. The number of needed DNS messages can be reduced by introducing small time-outs between transmitting the requests (Lines 10, 11). In this case the first name server should be selected randomly or using the rotate scheme in *resolv.conf* to have a better load distribution (Not shown in Fig. 4).

Step 3 In the case that no answer is obtained there must be a client-side failure or a server-side problem. The resolver now makes use of a peer-to-peer application. This service can be found using the *resolv.conf* as well[5] (Line 12, We assume that the last resolver is the peer-to-peer application). It implements the DNS protocol and is used as a gateway to the overlay network. The peer-to-peer network locates the DNS resource record and delivers it to the application, even during global DNS outages.

There have been some attempts to implement a DNS compatible service by using peer-to-peer overlay structures. Cox et al. [CMM02] simulated a Chord [SMK+01] based overlay implementing a DNS compatible server using the DNS traces of Jung et al. [JSBM01]. The results show that even for a small network of 1000 Nodes the additional delay of the overlay routing leads to rather slow answers. The estimation of answer times shows that less than 30% of the DNS queries can be answered within 200ms.

Ramasubramanian et al. [RS04b] showed that proactive caching and replication can mask this problem. An rather good performance was reached using Beehive [RS04a], a framework for proactive caching. More than 50% of all DNS requests did not incur any network traffic at all. They have been served from the local cache.

Now this approach seems to be great for long-TTL entries of DNS but for entries with short TTL it would suffer from the same problems as the solution of Cox et al. [CMM02].

The common problem is that in structured peer-to-peer networks data is located using distributed hash tables (DHT). The hash sum of objects or their names is calculated and the object is located at the peer with a node ID closest to this hash sum. If an object shall be retrieved, the distance to this node ID is reduced in each routing step. This does not mean to get closer to the target, since these IDs are not linked to the location in the network. So peers having almost the same ID can still be far away from each other in terms of round trip time.

There are some approaches to optimize the routing behavior in these overlays, e.g. Castro et al. [CDHR03]. The goal to use network layer knowledge for overlay routing decisions was achieved, but the core problem remains: The location of an object is determined by a DHT. All locality information is destroyed. This might be alright as long as the object or its name does not contain location related data, but in the case of DNS this would be a wrong assumption.

[5] It has to listen at the standard DNS port to receive the query.

```
01 resolve(request){
02
03  query=read(request);
04
05  cache=readln("/etc/resolv.conf");
06  for(i=0; i<resolvers_count; i++)
07   resolvers[i]=readln("/etc/resolv.conf");
08
09  scheduleQuery(cache, query, now);
10  for(i=0; i<resolvers_count - 1; i++)
11   scheduleQuery(resolver[0], query, cache_timeout + i*offset);
12  scheduleQuery(resolver[resolvers_count - 1], query, cache_timeout + p2p_offset);
13
14  pending = resolvers_count+1;
15  while(!valid && --pending > 0){
16   if ((error = select(inBuf, timeout))!=0) return error;
17   response = inBuf.read();
18   valid = !response.isError();
19  }
20
21  if(response.source()==cache) adapt(cache_timeout, LESS);
22  else adapt(cache_timeout, MORE)
23  return response.getAnswer();
24 }
```

Figure 4: Pseudocode of the SEDNS stub resolver

4.1 P2P overlay design

In the case, that the DNS infrastructure fails, we want to use a overlay network that efficiently locates DNS data. For this purpose we need a hash function that does not destroy any locality information.

DNS requests show locality, otherwise caching would not be as successful as it is. But the distance between two object's names or two URLs is often not determined by the round trip time between the hosts they are stored on but by the language of their content. This information is in many cases already included in the name in the shape of the country code.

While other approaches tried to exploit common keywords like Lu et al. [LHL05], we exploit the hierarchical structure of DNS. If some user looks up *www.harry-potter.uk*, it is more probable that the next lookup is *www.books.uk* instead of *www.harry-potter.fr*. So the design of our hash scheme must map this hierarchy into the hash. Therefore, we are using a prefix routing scheme, like used by Pastry [RD01] or Tapestry [HHWX01]. DNS names offer postfixes that indicate their location in the DNS name space. We directly map these postfixes to prefixes in the hash sum.

To improve the routing algorithm we exploit heterogeneity to select "Superpeers", i.e., nodes with more resources usable for routing. These Superpeers use group membership information to implement 1-hop routing.

5 Conclusion

We propose a multi-staged resolution scheme for DNS which can improve answer times by using an adaptive time-out for querying a caching-only server. Furthermore is uses different techniques to mask client-side failures and to reduce answer times at the same time. A peer-to-peer backend is only used in the case that DNS is unavailable due to client-side or server-side failures or attacks. The peer-to-peer network will be self-organizing and incrementally deployable.

The changes to the stub resolver will be implemented as glibc patch. No other software on the clients machine is necessary to install. The performance improvements will also be usable without participating in the overlay network.

References

[AMT03] Yair Amir, Daniel Massey, and Ciprian Tutu. A New Look at the Old Domain Name System, 2003.

[CDHR03] Miguel Castro, Peter Druschel, Y. Charlie Hu, and Antony Rowstron. Proximity neighbor selection in tree-based structured peer-to-peer overlays. In *Microsoft Technical report MSR-TR-2003-52*, 2003.

[CMM02] Russ Cox, Athicha Muthitacharoen, and Robert Morris. Serving DNS Using a Peer-to-Peer Lookup Service. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 155–165, London, UK, 2002. Springer-Verlag.

[FPJ05] Christof Fetzer, Gert Pfeifer, and Trevor Jim. Enhancing DNS Security using the SSL Trust Infrastructure. In *Proceeding of the IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS 2005)*, 2005.

[HHWX01] Fox Harrell, Yuanfang Hu, Guilian Wang, and Huaxia Xia. Survey of Locating & Routing in Peer-to-Peer Systems, December 2001.

[JSBM01] Jaeyeon Jung, Emil Sit, Hari Balakrishnan, and Robert Morris. DNS performance and the effectiveness of caching. In *IMW '01: Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pages 153–167, New York, NY, USA, 2001. ACM Press.

[LHL05] Yu-En Lu, Steven Hand, and Pietro Lio. Keyword Searching in Hypercubic Manifolds. In *P2P '05: Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing (P2P'05)*, pages 150–151, Washington, DC, USA, 2005. IEEE Computer Society.

[Nar02] Ryan Naraine. Massive DDoS Attack Hit DNS Root Servers. http://www.internetnews.com/dev-news/article.php/1486981, October 2002.

[PPPW04] KyoungSoo Park, Vivek S. Pai, Larry L. Peterson, and Zhe Wang. CoDNS: Improving DNS Performance and Reliability via Cooperative Lookups. In *OSDI*, pages 199–214, 2004.

[RD01] Antony Rowstron and Peter Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages 329–350, November 2001.

[Ros05] Timothy Roscoe. *The PlanetLab Platform*, chapter 33. The PlanetLab Platform, pages 567 – 581. Lecture Notes in Computer Science Springer-Verlag GmbH, 2005.

[RS04a] Venugopalan Ramasubramanian and Emin Gün Sirer. Beehive: O(1) Lookup Performance for Power-Law Query Distributions in Peer-to-Peer Overlays. In *NSDI*, pages 99–112. USENIX, 2004.

[RS04b] Venugopalan Ramasubramanian and Emin Gün Sirer. The design and implementation of a next generation name service for the internet. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 331–342, New York, NY, USA, 2004. ACM Press.

[SMK+01] Ion Stoica, Robert Morris, David Karger, Frans Kaashoek, and Hari Balakrishnan. Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In *ACM SIGCOMM*, 2001.

[Thu01] Paul Thurrott. Microsoft Suffers Another DoS Attack. http://www.windowsitpro.com/Articles/Index.cfm?ArticleID=19770&Displ, January 2001.

Automated Robustness Testing for Reactive Systems: Application to Communicating Protocols

Fares Saad Khorchef Ismail Berrada Antoine Rollet
Richard Castanet

LaBRI - CNRS - UMR 5800
33405 Talence cedex, France
{saad-kho, berrada, rollet, castanet}@labri.fr

Abstract: In the telecommunications field, protocols have to be seriously validated before their startup. Thus, it is necessary to test the conformance of a protocol, but it is also important to test its robustness in presence of unexpected events. This paper proposes a framework to test the robustness of a system. Firstly, we explain how to increase the nominal specification in order to take into account the hazards. Then, we show how to generate test sequences from the increased specification. Finally, we propose a case study on the SSL protocol, using the TGSE tool.

1 Introduction

Protocol specifications are used to develop products and services. To ensure correctness of such products, testing is the one of the used validation techniques. It consists of checking that the behaviors of a real implementation of a system (IUT for Implementation Under Test) is correct with respect to a specification.

With the exponential growth of Internet and with the growth of other services, protocol testing has become more difficult. While testing to ensure that requirements are met is necessary (i.e. conformance testing), tests aimed at ensuring that the system handles errors and failures appropriately are often neglected (i.e. robustness testing).

Although a precise definition of robustness is somewhat elusive, functionally the meaning is clear : the ability of a system to function in an acceptable way in presence of faults or stressful environmental conditions [CW03]. The term "hazards" will be used to gather faults and stressful conditions.

The aim of this paper is to provide a formal framework for robustness testing for Internet protocols. In order to decide the robustness of an IUT, a clear criterion is needed, taking into account the system behaviors in the presence of hazards. The contributions of this paper are :

(1) A framework for robustness testing including a formal definition of robustness and a test generation method. Our approach consists in enriching the *nominal specification* (i.e. protocol standard specification) with some representable hazards in order to get an

increased specification.

(2) A case study on the SSL protocol [Hic95]. We show how to integrate hazards in the specification of the handshake protocol in order to generate robustness test cases using the TGSE tool [BF05].

The remainder of the paper is organized as follows : Section 2 introduces models and notations used in the paper. Section 3 gives a definition and a classification of hazards. Section 4 presents a formalization and a method to test the robustness. The case study is given in Section 5. The related work is presented in Section 6. Section 7 concludes and draws some perspectives.

2 Basic Concepts

A reactive system is a software component which reacts to stimuli of its environment. I/O labelled transition systems ($IOLTS$) are used to describe the behaviors of such systems. This section introduces the $IOLTS$ model and some notations used throughout this paper.

2.1 Input Output Labelled Transition System

Definition 2.1 *An $IOLTS$ [TRE96] is an quadruplet $S = (Q, A, \rightarrow_S, q_0)$ such that: Q is a nonempty finite set of states, q_0 is the initial state, A is the alphabet of actions and, $\rightarrow_S \subseteq Q \times A \times Q$ is the transition relation.*

The alphabet A is partitioned into three sets $A = A_O \cup A_I \cup I$, where A_O is the output alphabet (an output is denoted by $!a$), A_I is the input alphabet (an input is denoted by $?a$) and I is the alphabet of internal actions (an internal action is denoted by τ). Usual notations are:

Notation	Meaning	Notation	Meaning
$q \stackrel{\varepsilon}{\Rightarrow} q'$	$q = q'$ or $q \stackrel{\tau_1 \cdots \tau_n}{\rightarrow} q'$	$Trace(q)$	$\{\sigma \in A^* \mid q \stackrel{\sigma}{\Rightarrow}\}$
$q \stackrel{a}{\Rightarrow} q'$	$\exists\, q_1, q_2 \mid q \stackrel{\varepsilon}{\Rightarrow} q_1 \stackrel{a}{\rightarrow} q_2 \stackrel{\varepsilon}{\Rightarrow} q'$	$Trace(S)$	$Trace(q_0)$
$q \stackrel{a_1 \cdots a_n}{\Rightarrow} q'$	$\exists\, q_0 ... q_n \mid q = q_0 \stackrel{a_1}{\Rightarrow} q_1 \stackrel{a_2}{\Rightarrow} ... \stackrel{a_n}{\Rightarrow} q_n = q'$	$Out(q)$	$\{a \in A_O \mid q \stackrel{a}{\Rightarrow}\}$
$q\ after\ \sigma$	$q' \in Q \mid q \stackrel{\sigma}{\Rightarrow} q'\}$	$Out(S, \sigma)$	$Out(S\ after\ \sigma)$
$S\ after\ \sigma$	$q_0\ after\ \sigma$	$ref(q)$	$\{a \in A_I \mid a \stackrel{q}{\not\rightarrow})\}$

The observable behaviors is described by $\Rightarrow$. $q\ after\ \sigma$ is the set of reachable states from q by σ. $Trace(q)$ is the set of observable sequences starting from q. $Out(q)$ is the set of all possible outputs of q. Finally, $ref(q)$ is the set of inputs which can not start a transition from q.

The $IOLTS$ S is called *deterministic* if no state accepts more than one successor with an observable action. It is called *observable* if no transition is labeled by τ. S is called *input-complete* if each state accepts all inputs of the alphabet.

2.2 Suspension Graph

In practice, the tester observes events from a system, but also the absence of events (quiescence). Several kinds of quiescence may happen in a state $q \in Q$: *outputlock* quiescence if the system is blocked on standby input of the environment ($Out(q) = \emptyset$), *deadlock* quiescence if there is no more evolution of the system ($\forall a \in A | q \stackrel{a}{\not\rightarrow}$) or *livelock* quiescence if $q \stackrel{\varepsilon}{\Rightarrow} q$.

Definition 2.2 *The suspension graph [JER03] of $S = (Q, A, \rightarrow, q_0)$ is an IOLTS $S^\delta = (Q, A^\delta, \rightarrow_\delta, q_0)$ such that: $A^\delta = A \cup \{\delta\}$ with $\delta \in A_O^\delta$ (δ is considered as an output). $\rightarrow_\delta$ is obtained from $\rightarrow$ by adding loops $q \stackrel{\delta}{\rightarrow} q$ for all quiescence states*

2.3 Meta-Graph

In order to model the behaviors of a system $S = (Q, A, \rightarrow, q_0)$ in the presence of hazards, we use the concept of the *meta-graph* associated to S. A meta-graph G is a graph such that each state of G corresponds to a set of states of S having the same behaviors in the presence of the same hazards.

Definition 2.3 *A meta-graph associated to S is a triplet $G = (V, E, L)$ such as:*

- *$V = V_d \cup V_m$ is a set of states. $V_m \subseteq 2^Q$ is called the set of meta-states and V_d is called the set of degraded states such that $V_d \cap Q = \emptyset$.*
- *L is an alphabet of actions,*
- *$E \subseteq V \times L \times V$ is a set of edges*

Definition 2.4 (Composition $IOLTS \oplus G$)
Let $S = (Q, q_0, A, \rightarrow_S)$ be an $IOLTS$ and $G = (V, E, L)$ a meta-graph associated to S. The composition of S and G, noted $S \oplus G$, is the $IOLTS$ $(Q^{S\oplus G}, q_0^{S\oplus G}, A^{S\oplus G}, \rightarrow_{S\oplus G})$ defined by: $Q^{S\oplus G} = Q \cup V_d$, $q_0^{S\oplus G} = q_0$, $A^{S\oplus G} = A \cup L$ and,

1. $q \stackrel{a}{\rightarrow} q' \quad \Rightarrow \quad q \stackrel{a}{\rightarrow}_{S\oplus G} q'$
2. $(v, a, v') \in E$ *and* $v, v' \in V_d \quad \Rightarrow \quad v \stackrel{a}{\rightarrow}_{S\oplus G} v'$.
3. $(v, a, v') \in E$, $v \in V_m$ *and* $v' \in V_d \quad \Rightarrow \quad q \stackrel{a}{\rightarrow}_{S\oplus G} v'$ *for all* $q \in v$.
4. $(v, a, v') \in E$, $v \in V_d$ *and* $v' \in V_m \quad \Rightarrow \quad v \stackrel{a}{\rightarrow}_{S\oplus G} q$ *for all* $q \in v'$.
5. $(v, a, v') \in E$ *and* $v, v' \in V_m \Rightarrow q \stackrel{a}{\rightarrow}_{S\oplus G} q'$ *for all* $q \in v$ *and* $q' \in v'$

This composition consists in adding in S the set of transitions and states of meta-graph G. Actually, for a state q of S member of a meta-state (i.e. a set of states) v of G, we add in S the set of transitions and/or states starting from v. In the following, this composition is used to integrate hazards modeled as meta-graph(s) in the nominal specification. Figure 3 illustrates this composition.

3 Hazards

In robustness testing, a *hazard* denotes any event not expected in the nominal specification of the system. In this section, we propose to extend the hazards classification given in [CW03]. Our classification considers :

Position of hazards relative to the system boundaries. We distinguish the internal (e.g. memory overflow, processor failure, etc...), external (e.g. intrusion, stressful conditions, etc...) and beyond the system boundaries hazards.

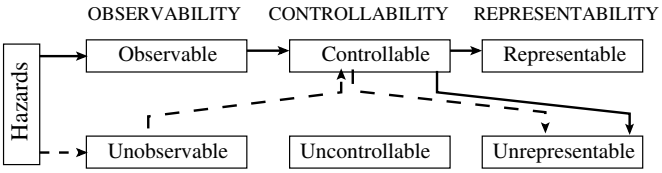


Figure 1: Classification of hazards

Position of hazards relative to the tester controllability. We regroup the hazards basing on their observability, representability and controllability by the tester. Figure 1 gives the different possible combinations :

1. Observable, controllable and representable hazards. They regroup *invalid inputs* (e.g. modified inputs, erroneous inputs or lost inputs) or *inopportune inputs* (e.g. messages in advance or late). These terms are defined below.
2. Observable, controllable and unrepresentable hazards. They are composed by some *external failures* whose influence on the inputs are not very clear or difficult to represent (e.g. pressure, radiation, temperature).
3. Unobservable, controllable and unrepresentable hazards. They are composed by some *complex internal failures* which we can not describe with classic models (e.g, memory overflow, processor bugs).

The other possible combinations of figure 1 are not considered because we will not able to test something not controllable. For observable and representable events, we identify three kinds of hazards :

Invalid Inputs describing some erroneous, specified inputs (e.g. incorrect values, errors of initialization, temporization faults).

Inopportune Inputs corresponding to actions which exist in the alphabet of the specification, but not expected in the given state. $ref(q)$ (see standard notations of $IOLTS$) denotes the inopportune inputs in a state $q \in Q$.

Unexpected outputs. Taking into account the hazards can lead the system, in some cases, to send some unexpected outputs. Sometimes, such outputs may be considered as acceptable. For example, restarting a session, resetting or closing a connection may be acceptable behaviors. As a consequence, all acceptable outputs must be added to the specification.

4 Proposed framework

In this part, we propose a formal approach to generate robustness test cases. Firstly, we show how to integrate the representable hazards in the initial model. The obtained model is called *increased specification*. Secondly, we formalize the robustness of an implementation compared to the increased specification. Basing on the previous relation, we explain how to produce robustness test cases using a test purpose. The approach diagram is given in figure 2.

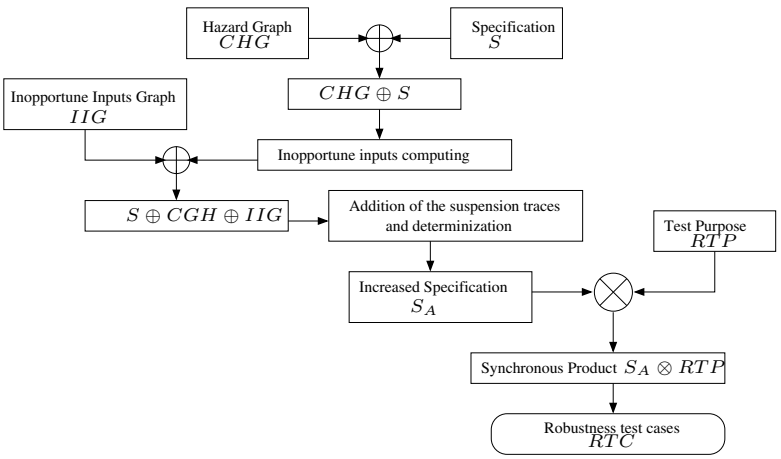


Figure 2: Approach diagram

4.1 Increase of the specification

The aim of the increased specification is to formally describe the acceptable behaviors in presence of controllable and representable hazards. In our approach, we describe the behavior of the system in presence of one or several hazard(s) modeled by meta-graph(s). Figures 3 (a) and (b) illustrate this concept. Assuming that S (figure 3 (a)) is in its initial state, if it receives the hazard $?a'$, it has to move to the degraded state $d2$ according to the meta-graph of figure 3 (b). Besides, if it sends the acceptable output $!x'$, it has to move to the degraded state $d1$, which permits the system to come back to a nominal behavior (here the initial state) in case of reception of $?a$.

In the following, we suppose that hazards (invalid inputs and acceptable outputs) are modelled by one or more meta-graph(s) CHG (*Controllable Hazard's Graph*). Then, inopportune inputs can be automatically computed and are represented by meta-graph(s) IIG (*Inopportune Input Graph*). The increased specification (see figure 3 (f)) is obtained as follows. For a nominal specification S and a set of hazards modeled by a meta-graph CHG, we firstly compose them to obtain $CGH \oplus S$ (see figure 3 (c)). Secondly, we compute the inopportune inputs of $CGH \oplus S$ to construct IIG (figure 3 (d)). Then, we build $CGH \oplus S \oplus IIG$ (figure 3 (e)). Finally, we add suspension traces and we proceed to determinization of $CGH \oplus S \oplus IIG$ (figure 3 (f)) in order to obtain the increased specification.

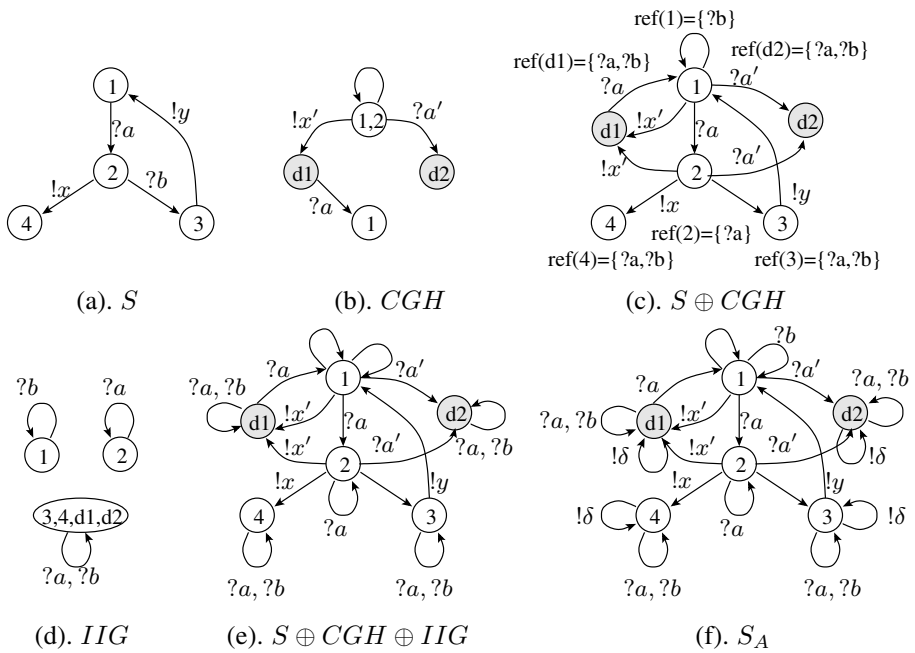


Figure 3: Increase of specification

4.2 Robustness relation

In order to describe formally the notion of robustness, the following hypothesis are needed :

Increased specification. We suppose that the nominal specification is modeled by an IOLTS $S = (Q, A, \rightarrow, q_0)$. The increased specification of S is modeled by a deterministic, observable and input-complete IOLTS $S_A = (Q^{S_A}, A^{S_A}, \rightarrow_{S_A}, q_0^{S_A})$ (the construction of S_A is the one given in the previous sub-section).

Implementation. The real implementation under test (IUT) is unknown, but to be able to reason formally on the robustness of an implementation I with respect to a specification S, we assume that :

1. I is modelled by an IOLTS,
2. I conforms to S,
3. I is input-complete on the alphabet A^{S_A}.

Robustness relation. Let I be an implementation of a specification S and S_A its increased specification. The robustness relation **Robust** is defined by :

$$I \textbf{ Robust } S_A \equiv_{def} \forall \sigma \in Trace(S_A) \backslash Trace(S^\delta) \Rightarrow Out(I^\delta, \sigma) \subseteq Out(S_A, \sigma).$$

Only the increased behaviors (added) are useful for robustness testing because the nominal behaviors (including valid quiescence) already passed the conformance testing.

Example 1 *Let us consider figure 4.*

- I_1 ***Robust*** S_A *because all traces in* I_1 *are included in* S_A
- *not(*I_2 ***Robust*** S_A*) because* I_2 *after* $?a'$ *sends* $!y$ *but* S_A *after* $?a'$ *sends* $!x'$.
- *not(*I_3 ***Robust*** S_A*):* I_3 *after* $?a'$ *reaches another state not specified in* S_A.
- *not(*I_4 ***Robust*** S_A*):* I_4 *after* $?a'$ *sends* $!y$ *and reaches another state not specified in* S_A.

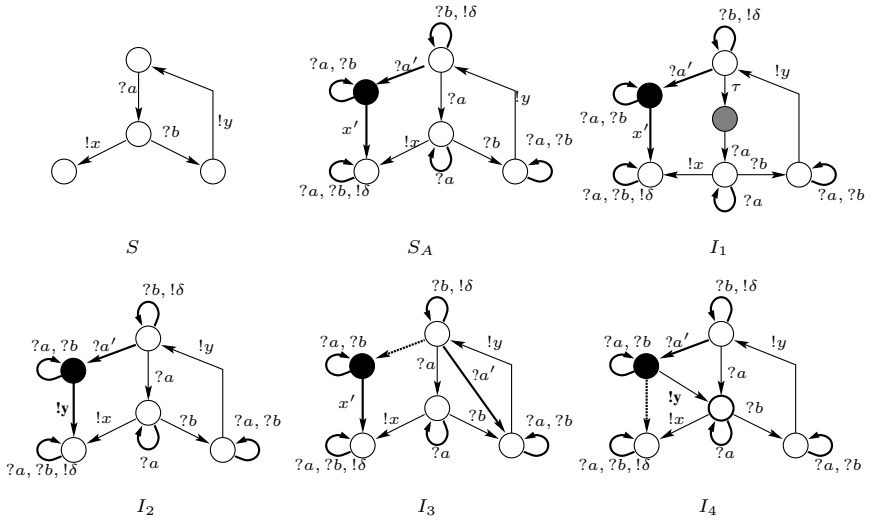


Figure 4: Robustness relation

4.3 Test generation

After the integration of hazards (invalid inputs, inopportune inputs and acceptable outputs), the size of the increased specification becomes very large. In order to reduce test costs, we propose to select the tests by using *test purposes*. This method was used in some conformance testing works [BF05, JER03]. The principle of this method is founded on the synchronization between the specification and the test purpose modelled by two $IOLTS$.

The robustness test generation may be summarized as follows. 1) Choice of robustness test purpose. 2) Synchronization between the specification and the test purpose in order to deduce the behaviors which satisfied the test purpose. 3) Mirror image (i.e. inputs become outputs and outputs become inputs) of the *synchronous product*. 4) Extraction of *robustness test cases*.
The *Robustness Test Purpose (RTP)* is used to check some robustness properties in the IUT. In the proposed case study, we use TGSE tool [BF05] in order to generate test cases.

5 Case study: The SSL Protocol

Netscape describes SSL as follows [Hic95] : "The SSL protocol is designed to provide privacy between two communicating applications (a client and a server). Second, the protocol is designed to authenticate the server, and optionally the client". SSL is standardized by the IETF (Internet Engineering Task Force). The full specification of the SSL proto-

col is written in the RFC 2246. There exists several implementations of the SSL protocol (Open SSL, SSLeay, BSAFF 3.0, SSL Plus, SSL Ref 3.0).

The SSL Protocol contains four under-protocols: *Handshake protocol*, *SSL Changes Cipher Spec protocol*, *SSL Alert protocol* and *SSL Record protocol*. The Handshake protocol is composed of two phases. First step deals with the selection of a cipher, the exchange of a master key and the authentication of the server. Second step handles client authentication, if requested and finishes the handshaking. After the handshake stage is complete, the data transfer between client and server begins. All messages during handshaking and after, are sent over the SSL Record protocol Layer.

Here, we deal only with the specification of the handshake protocol which describes three scenarios of communication as shown in the following table :

Case	**Sequences**
No Session Identifier, No Client authentication	!Client-Hello, ?Server-Hello, ?Client-Master-Key, ?Client-Finished, !Server-Verify, !Server-Finished
Session Identifier Used, No Client authentication	!Client-Hello, ?Server-Hello, ?Client-Finished, !Server-Verify, !Server-Finished
Session Identifier Used, Client authentication	!Client-Hello, ?Server-Hello, ?Client-Master-Key, ?Client-Finished, !Server-Verify, ?Request-Certificate, !Client-Certificate !Server-Finished

Table 1: Nominal scenarios of the Handshake protocol

The standard specification (RFC2246) defines the following errors :

- NO-CIPHER-ERROR. This error is returned by the client to the server when it can not find a cipher or key size. This error is not recoverable.
- NO-CERTIFICATE-ERROR. When a REQUEST-CERTIFICATE message is sent, this error may be returned if the client has no certificate to reply with. This error is recoverable (for client authentication only).
- BAD-CERTIFICATE-ERROR. This error is returned when a certificate is deemed bad by the receiving party. Bad means that either the signature of the certificate was bad or that the values in the certificate were inappropriate (e.g. a name in the certificate did not match the expected name). This error is recoverable (for client authentication only).
- UNSUPPORTED-CERTIFICATE-TYPE-ERROR. This error is returned when a client/server receives a certificate type that it can not support. This error is recoverable (for client authentication only).

Two error messages have been omitted from the specification document (this problem is noticed in [BD95]). The first, an UNSUPPORTED AUTHENTICATION TYPE ERROR message, is a mistake which would prevent the protocol using different methods of authentication of a client. The second, an UNEXPECTED-MESSAGE-ERROR would allow an implementation to close the connection cleanly if an implementation sent an out-off-order message.

In order to verify the robustness of the Handshake protocol, we increase the nominal specification by integrating hazards (*invalid inputs* and *inopportune inputs*). Besides, to model the previous hazards, we consider the following hypothesis. 1) if the implementation receives an invalid input then it closes the connection and, 2) if it receives an inopportune input then it loops in the same state. Formally, the previous hypothesis may be modelled by meta-graphs. The increased specification is given in figure 5 (Annexe), it is composed of 20 states and 176 transitions. In figure 5, the inopportune inputs are represented by $ref(q)$ for any state q but in the experimentations, they are automatically computed.

5.1 Robustness test generation with TGSE tool

In order to generate robustness test cases, we have defined a set of robustness test purposes aiming at checking the behavior of an implementation in presence of the certificate failures (No-Certificate-Error, Bad-Certificate-Error, Unsupported-Certificate-Type-Error), the cipher failures (No-Cipher-Error) and two other failures (Unexpected-Message-Error et Unsupported-Authentification-Type-Error) :

- **RTP1**: Closing the connection between the client and the server after detection of a certificate failure.
- **RTP2**: Closing the connection after detection of a cipher failure.
- **RTP3** : Closing the connection after detection of unexpected error message.

In addition, we mention that both the inopportune inputs and quiescence are automatically generated by the TGSE tool[BF05]. The following table summarizes the different results obtained by a TGSE implementation under Linux Fedora 3 station (Intel Pentium 4 CPU 1.80GHz, 128Mo of memory). "RTC size" represents the average size of the robustness test cases and "CPU Time" represents the average time needed to extract the RTC. We notice that Robustness Test Cases are significantly longer than in usual conformance testing methods. The reason is that the integration of hazards in the specification increases the number of possible transitions.

Robustness test purposes	RTC size	CPU Time(ms)
RTP1	53	0.9618
RTP2	10	0.3599
RTP3	20	0.15797

Table 2: Results obtained by TGSE tool

6 Related work

Many research have been done in the domain of protocol testing . The majority of these works deals with conformance testing [IEE04] (an overview may be found in [JER03]). In this section, we focus particularly on robustness testing works.

[CW03] proposes a study on robustness testing, focusing on hazard classification and some possible directions to handle the problem. Authors define the robustness notion as "the ability of a system to function acceptably in the presence of faults or stressful environmental conditions" and provide a state of the contributions in this domain.

The PROTOS project [RLT02] proposes to describe the system with a high level of abstraction and then to simulate abnormal inputs in the specification. It is mainly focused on the detection of vulnerabilities of a network software system. In this case, robustness is restricted to the notion of network security.

Some approaches are based on software fault injection :
The FIAT tool [BCSS90] modifies a processus binary image in memory, [Reg05] applies randomly interruptions in the IUT, whereas the BALLISTA tool works on data unexpected modifications. These approaches are based on integration of faults directly in the software implementation of the system, but do not care about interpretation of different behaviours.

Another approach consists in using model-based test generation. The main difficulty of such technics is to describe the hazards in the model. Many works consider such approach : [SKD05, FMP05, Rol03].

[SKD05] proposes a first approach based on a refusal graph used to model hazards. Contrary to our method, it only deals with inopportune inputs, but not with invalid inputs. Moreover, our approach distinguishes between inputs and outputs in the model.

[FMP05] uses a formal fault model in order to build a "mutant" specification. They use a fault model in order to add "fault" transitions in the specification. They define a robustness relation based on a robustness property. Contrary to our approach, they do not permit to integrate unexpected inputs in the model.

[Rol03] uses a degraded specification to model the behavior in case of critical situation, and integrates the hazards directly in the test sequences. A major difference between [Rol03] and this work is in the concept of robustness : we consider here that robustness implies conformance; [Rol03] does not.

7 Conclusion

In this paper, we have presented a formal framework and a generation technic to test the robustness of a protocol modeled as IOLTS. We proposed to integrate representable hazards in the specification after suspension traces addition and determinization. Then we used this increased specification in order to generate robustness test cases using a test purpose. Secondly, we proposed a case study on the SSL protocol by describing how to increase

the specification, and by generating test sequences with the TGSE tool. This case study permits to show the complementary aspect of conformance and robustness testing.

Currently, we are working on tools to help the designer of a system to describe the behavior of the system in case of unexpected events. Besides, we are studying a way to handle time constraints in robustness testing.

References

[BCSS90] J. H. Barton, E. W. Czeck, Z. Z. Segall, and D. P. Siewiorek. Fault Injection Experiments Using FIAT. *IEEE Trans. Comput.*, 39(4):575–582, 1990.

[BD95] J. Bradley and N. Davies. Analysis of the SSL Protocol. Technical Report CSTR-95-021, Department of Computer Science, University of Bristol, June 1995.

[BF05] I. BERRADA and P. FELIX. TGSE: Un outil générique pour le test. In *CFIP'2005: Ingénérie des Protocoles*, pages 67–84, 29 Mars 2005.

[CW03] R. CASTANET and H. WAESELYNK. Techniques avancées de test de systèmes complexes: Test de robustesse. Technical report, Action spécifique 23 du CNRS, 11 2003.

[FMP05] J-C. FERNANDEZ, L. MOUNIER, and C. PACHON. A Model-Based Approach for Robustness Testing. In LNCS, editor, *Testing of Communication Systems*, volume 3502, pages 333–348. ifip, may/june 2005.

[Hic95] Kipp Hickman. The SSL Protocol. Technical report, Netscape Communications Corp., Feb 9 1995.

[IEE04] IEEE. *International Organization for Standardization, Conformance testing methodology and framework - part 2: abstract test suite specification*, 2004.

[JER03] T. JERON. Génération de tests pour les systèmes réactifs. Un survol des théories et techniques. In IRIT, editor, *ETR2003. Systèmes, Réseaux et Applications*, pages 105–122. IRIT, Septembre 2003.

[Reg05] John Regehr. Random testing of interrupt-driven software. In *EMSOFT '05: Proceedings of the 5th ACM international conference on Embedded software*, pages 290–298, New York, NY, USA, 2005. ACM Press.

[RLT02] J. Röning, M. Laakso, and A. Takanen. PROTOS - systematic approach to eliminate software vulnerabilities. http://www.ee.oulu.fi/research/ouspg, May 2002. 2002.

[Rol03] A. Rollet. Testing robustness of real-time embedded systems. *In Proceedings of Workshop On Testing Real-Time and Embedded Systems (WTRTES), Satellite Workshop of Formal Methods (FM) 2003 Symposium, Pisa, Italy*, September 13 2003.

[SKD05] F. SAAD-KHORCHEF and X. DELORD. Une méthode pour le test de robustesse adaptée aux protocoles de communication. 29 mars 2005.

[TRE96] J. TRETMANS. Conformance Testing with Labelled Transition Systems: Implementation Relations and Test Generation. *Computer Networks and ISDN Systems*, 29:49–79, 1996.

Annexe

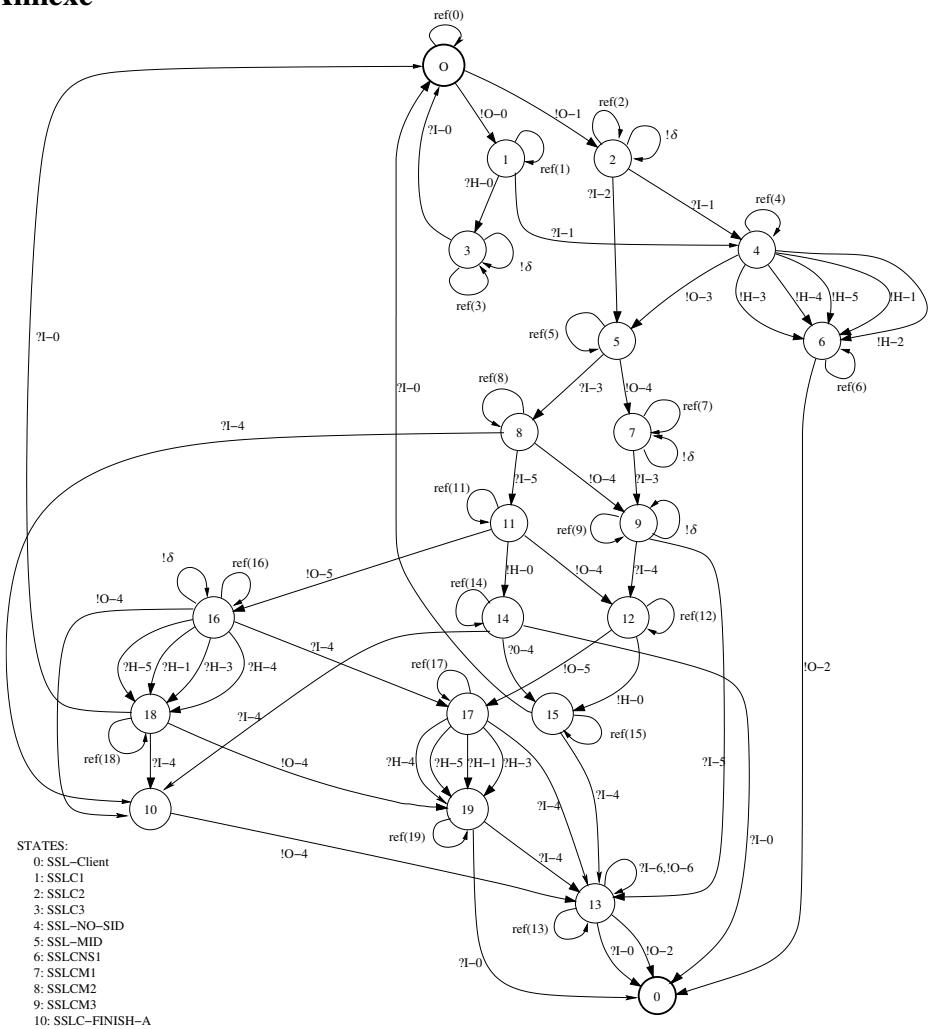


Figure 5: Increased specification of the Handshake protocol

A Distributed Cache Management for Test Derivation

Harry Gros-Desormeaux Hacène Fouchal
Philippe Hunel

GRIMAAG
Université des Antilles et de la Guyane, France
{harry.gros-desormeaux, hfouchal, phunel}@univ-ag.fr

Abstract: Complex systems need to be validated before industrial development. The last step in the validation process is testing. This part has to be considered with care in order to avoid troubles. This step takes a long time and requires a lot of resources.

In this paper, a complex system is described a Timed Labeled Transition System (TLTS). In such description, we focus on the specification of the event ordering respecting time constraints.

Since the TLTS may be very large (million of states for industrial systems), we present a solution to reduce the test derivation complexity. We aim to decompose the derivation process among some hosts participating to the generation algorithm. Each host will deal with a part of the system independently ; each host will derive test sequences for some fixed states from the system.

Some computations are redundant. In order to reduce them, on each host we use the *Bloom filters* concept used to manage a local cache containing computed sequences.

Then, we show how to compute the results given by all hosts in order give a set of test sequences for the whole system.

We suggest an implementation of this technique on the JXTA environment deployed on some hosts. We analyze a large number of experiments on different TLTS. Finally, we have shown that the use of Bloom filters make increases the test derivation performances.

Key-words : Distributed Environment, P2P Computing, Conformance Testing, Protocol verification.

1 Introduction

Complex systems are being used everywhere. However, the success of their deployment depends on low development cost and reduced time to market. In order to achieve this goal, validation steps should be handled with care. Among these steps, conformance testing is highly needed to avoid dramatic errors and to tackle the industrial development of the product with a high confidence.

Peer-to-peer frameworks revealed great potential for scientific applications which require a lot of resources. Recently, projects like SETI@home[1] has started to bring attention to massive parallel computing and now, it is common to tackle long running time computations with peer-to-peer environments. The term peer-to-peer refers to machines which interact in a distributed fashion to reach the same goal.

In this study, we model a complex system by the Timed Labeled Transition System (TLTS) model. It is defined as an automaton where each transition can bear either an input action (an event from the environment) or an output action (a reaction of the system). On each transition, we may have timing constraints expressed as equations on defined clocks. Finally, on transitions, we may have clock resets. Each state represents a stable state of the system. It is widely used for the description of timed systems [AD94]. Then, we present a test sequence generation technique which derives a specific test sequence -from the specification -for each controllable state (represents situations where the system waits for stimuli from the environment) of a specification. The purpose is to check if every controllable state (of the specification) will be correctly implemented on any implementation supposed to be conform to the system specification.

Some industrial systems are modeled by huge automata (containing millions of states). An exhaustive test sequence generation is not possible in a sequential environment. For this reason, we present a distributed algorithm which considers the test derivation issue. Each host involved in the computation will receive the whole automaton (representing the system) as well as the list of states for which it has to generate test sequences. Network diversity leads us to design two algorithms which generate tests sequence : one which hold for homogeneous networks and another version which takes account of network heterogeneity. Each host will handle a local cache to keep the recent computed test sequences. The management of such caches is done by using the Bloom filters concept. Algorithms have been implemented on the JXTA environment with a PC cluster.

This paper is structured as follows: Section 2 gives an overview on studies done on testing of timed systems and peer-to-peer computing. Section 3 describes the test sequence generation algorithm. Section 4 details the decomposition issue and gives an insight to our distributed algorithm. Section 5 is devoted to the improvement of the testing process by using local caches based on Bloom filters. Section 6 gives a conclusion and some ideas about future work in particular on different evolutions of the distributed algorithm.

2 Related work

In this section, we present briefly the main related work in testing real-time systems as well as the most important approaches in peer-to-peer computing.

2.1 Timed system testing

[SVD01] gives a general outline and a theoretical framework for timed testing. They proved that exhaustive testing of deterministic timed automaton with a dense interpretation is theoretically possible but is still difficult in practice since the number of test sequences is very high.

[ENDKE98] differs from the previous one by using discretization step size depending only on the number of clocks which reduces the timing precision of the action execution.

The resulting model has to be translated into a kind of Input/Output Finite State Machine, finally they extract test cases by using the Wp-method [FBK+91].

[NS01] suggests a selection technique of timed tests from a restricted class of dense timed automaton specifications. It is based on the well known testing theory proposed by Hennessy in [DNH84].

In [CL97], the authors derive test cases from specifications described in the form of a constraint graph. They only consider the minimum and the maximum allowable delays between input/output events.

[CO02] presents a method for networks of deterministic timed automata extended with integer data variables where only a part of the system can be visible.

[RNHW98] gives a particular method for the derivation of the more relevant inputs of the systems.

[PF99] suggests a technique for translating a region graph into a graph where timing constraints are expressed by specific labels using clock zones.

[HNTC01] derives test cases from Timed Input Output automaton extended with data. Automata are transformed into a kind of Input Output Finite State Machine in order to apply classical test generation technique.

In [KT04], suggests a framework for a black-box conformance testing of real-time systems, where specifications are modeled as nondeterministic and partially-observable timed automata. This work is an extension of a previous work on the same issue [AT02].

All of these studies focus on reducing the specification formalism in order to be able to derive test cases feasible in practice. In contrast to these studies, we use the timed automaton model without neither translation nor transformation of labels on transitions. In order to reduce the generation execution time, we suggest a distributed technique to perform the test generation.

2.2 Peer-to-Peer Computing

Peer-to-peer frameworks revealed great potential for scientific applications which require a lot of resources. Recently, projects like SETI@home has started to bring attention to massive parallel computing and now, it is common to tackle long running time computations with peer-to-peer environments (Distributed.net [2], Distributed Folding Project[3], Grub[4], Evolution@Home[5], etc). The term peer-to-peer refers to machines which interact in a distributed fashion to reach the same purpose. These systems can be :

- *centralized :* A single machine, the server, manages the peer-to-peer layer. This configuration exhibits an important drawback : if the server disappears, the peer-to-peer environment collapses.
- *decentralized :* Each machine can play the same role in the network. They can manage the peer-to-peer layer as they can share their resources as workstations.

- *mixed :* The peer-to-peer network can be divided in sub-networks where some peers are servers and the others, workstations. This configuration tends to provide more tolerance to faults when a server leaves since peer-to-peer systems are usually build over volatile nodes which can appear or disappear at "will".

Generally, we tend to find two types of peer-to-peer applications even though they exhibit only a fragment of the peer-to-peer philosophy. On one hand, we find file-share based applications like Napster[6], Overnet[7], Kazaa[8], Bittorent[9] which are well-known today due to the several Majors' lawsuits brought recently. On the other hand, CPU cycle stealing projects like the ones cited above as peer-to-peer computing are very popular due to their altruistic interests. Nevertheless, other projects exist, like OceanStore which offers global persistent data store designed to scale to billions of users. Even VoIP applications (cf. Skype[10], Gizmo Project[11],...) leverage peer-to-peer concepts for constructing their "small world". Peer-to-peer philosophy is so attractive that it spawned a revolution in the manner to tackle some type of problems.

By gaining more and more attention, peer-to-peer model drove Sun Microsystems to design the JXTA technology which provides interoperability, independent platform and ubiquity. These advantages offer simplicity to anyone who wants to make a portable peer-to-peer application by hiding the complexity of the physical network as well as the knowledge of all operating systems core of the different hosts participating in the network. Thereby JXTA appeared as a suitable platform to implement our application.

JXTA provides several predefined protocols for developing large-scale peer-to-peer applications. In the JXTA environment, nodes are groups that propose services, and synergize in order to reach a common goal. Each node which connects to the peer-to-peer layer, a loosely-consistent DHT* network, joins the principal peer group before belonging to one or more dedicated groups. In fact, groups advertise services and can be only reached according to their policies. Another strength of JXTA is to provide *relay points* in order to bypass firewalls. So, nodes can fetch data to these relays when they are forbidden to receive network data packets. The *pipes* feature, a JXTA communication concept, abstract routes between two endpoints in the peer-to-peer network. *Pipes* are published through *advertisements* on the network and allow nodes with similar pipe advertisements to communicate between them. So we can bypass the physical details of the network and focus only on communications. For example, from the node point of view, we do not need to know which node will receive our packet. We only send data through a pipe if binded, JXTA services will ensure consistency and routing in the peer-to-peer layer.

JXTA provides simplicity and easiness for who wants to implement a peer-to-peer application today. Although it has shown some drawbacks in the past as a peer-to-peer computing platform — due to bad bandwidth management — it seems that it has been improved and now is able to reach optimal efficiency for WANs [AHJN05]. We use JXTA as our peer-to-peer platform to develop our distributed application.

*Distributed Hash Table

3 Test sequence generation algorithm

The purpose is to find for any controllable state a sequence in order to identify the state.A controllable state is a state where outgoing transitions are labeled by input actions. That means when the system reaches such states, it can only wait for external events. We will extracts for each controllable state a sequence of action starting with an input action and ends with an output one. If this sequence is not recognized by any other state, then this state is considered as identified. We will operate in a similar way for all other states with sequences containing an arbitrary number of input actions. This number – that we will call from now *depth sequence* — is fixed in advance.

In fact, the algorithm is a little bit similar to the UIO technique [SD88] used for untimed system testing described as IOFSM (Input Output Finite State Machine).

Algorithm 1: Test generation

Data: An automaton
Result: Derived sequence for each controllable state

```
We initialize depth sequence to 1
d=1
foreach controllable state e of the automaton do
    We initialize the data structure which will store the sequences
    SeqUniq[e] = ∅
threshold is the maximum depth sequence
while d ≠ threshold do
    foreach controllable state e of the automaton do
        if SeqUniq[e] = ∅ then
            RSS= set of all sequences of depth d starting from state e
            foreach Sequence s ∈ RSS do
                if s is not accepted by other controllable states than e then
                    SeqUniq[e] = s
    d = d + 1
```

Comments

The aim of this algorithm is to derive for each controllable state a test sequence. A test sequence starts with a input event (to ensure controllability from the user during the test execution) and ends with a output event. Each derived sequence for a state should be checked on all other states and should not be applicable on these states. It should be unique.

Theorem 3.1. *Let's n, d, t respectively be the number of states of an automaton A, the maximum depth of a test sequence and the maximum number of transitions per state in an automaton. Algorithm 1 found all test sequences of depth d in time $\mathcal{O}(k \cdot n^2)$ with $k = dt^d$.*

Proof. A state accept a test sequence of depth d in $\mathcal{O}(d)$ steps. So, we know if a sequence is unique in $\mathcal{O}(nd)$ and testing all sequences of a rss set is done in $\mathcal{O}(t^d nd)$, then finding a unique sequence of depth d for each state costs $\mathcal{O}(dt^d n^2)$ steps. □

Example:

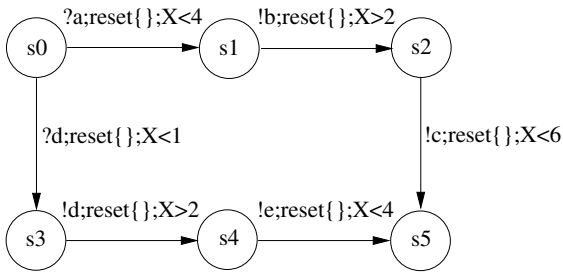


Figure 1: An example of specification

On Figure 1, state s_0 is controllable. Then, the following sequences are extracted:

- $(?a, X < 4), (!b, X > 2)$ is recognized by s_0,
- $(?d, X < 1), (!d, X > 2), (!e, X < 4)$ is recognized by s_0,
- $(?b, X > 1), (!c, X > 6)$ is not recognized by s_0.

The next section presents a peer-to-peer algorithm which computes the algorithm described below.

4 A distributed approach

We design a MPMD[†] algorithm based on the *master-slave* paradigm which implements the test generation algorithm. As we said in section 3, we can find almost all test sequences with a suitable depth. For large automata (with millions states), this depth turn out to be too high for sequential applications which take too much time to complete. Our distributed algorithm takes as input an automaton and outputs all minimal test sequences for each state.

[†] Multiple Program Multiple Data

4.1 A general framework

We describe briefly our distributed algorithm composed of the master and the slaves:

- A master gives jobs (computation of some controllable states) to slaves and then get the results (derived sequences) when jobs end.
- Slaves fetch jobs from the Master, compute sequences and send them to the Master.

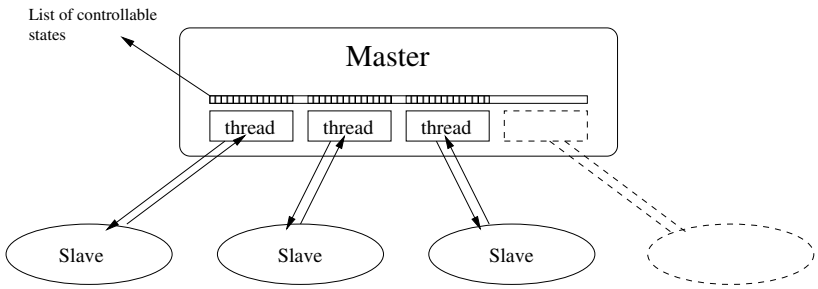


Figure 2: The Master-Slave approach

4.1.1 The master

The role of the master is to coordinate the global work. It distribute *jobs* to the working slaves and gives jobs to additional nodes which join the peer-to-peer network. Our algorithm takes as input an automaton and gives a singular test sequence for each controllable state. The set of all controllable states of the automaton is divided in sublists. A job is defined as a computation of test sequences for controllable states contained in a sublist. That means that each job will handle a sublist of states. Whenever a slave joins the peer-to-peer network, it fetches a job from the master. Test sequence results are saved for subsequent return to the master.

From an implementation point of view, we start a thread process for each slave which connects to the master and send to it a job taken from the *job list* as well as the automaton. We send the automaton only once, the slave stores it for subsequent work if any.

4.1.2 The slaves

Slaves are nodes which help to the global work by running the generation algorithm on a subset of controllable states. Each slave starts a *Message Event Listener* thread to capture messages sent by the master and asks for jobs. When receiving their first job, they get in the same time the automaton and store it for later use. Each job is recognized by a list of integers which represent the states that it has to compute. Then the generation algorithm is run on each of these states and result sequences are saved in a *results* list. Subsequently

Algorithm 2: Master (sublist fixed size version)

```
Master()
begin
    Results R = ∅
    We generate the automaton
    A = Automaton(nbStates,minNbTrans,maxNbTrans,nbClocks)
    Compute JobList J = {J_1, ..., J_n}
    while J ! = ∅ do
        We wait that a slave connect
        Wait for connection
        We send job and automaton
        Start Thread SendAutomatonAndJob()
    Wait for all Threads
    Exit(0)
end

SendAutomatonAndJobs()
begin
    if Slave has not received the automaton then
        Send(A)
    Send(J_i)
    We wait that the slave finished and send its results
    Wait for Results
    Add Results to R
    Close connection
end
```

the *results* are sent to the Master. Slaves may be added freely to our platform even though it is not always an efficient way to solve the problem.

4.2 Job list decomposition

We have first designed an algorithm which partitions the list of controllable states in equal parts before scatters them between slaves. Suitable for homogeneous networks, this scheme has exhibited some insufficiency whenever machines and communications links were manifold. Consequently, we develop a scheme which copes with network heterogeneity. Indeed, we can choose between two strategies when dealing with jobs size. Dynamic approaches gracefully adapt jobs size whenever new nodes join the network whereas static ones fix *a priori* this size once for all. In these two cases, job size limit is an important parameter. If a job size is too high, we may loose performance and scalability since some joining machines could not find any job. A contrario, if jobs are too small, we communicate too often and performances collapse. Furthermore, too many processors imply too much communications. Indeed, performance degrades whenever communication

Algorithm 3: Slave

```
Slave()
Automaton A = ∅
begin
    Connect to the Master
    Start Thread MessageListenerEvent()
end

MessageSlaveListenerEvent()
begin
    if A = ∅ then
        Ask for Automaton to Master
        Store automaton in A
    Ask for Job
    Compute step one for each state of the received job
    Send Results to Master
end
```

time overlaps processing time.

In the next subsections, we present two schemes for distributing jobs to slave nodes. On one hand, a simple static strategy which partitions the sublist size in fixed parts and on the other hand, a simple dynamic job distribution which varies the received job size according to its *resource capabilities*.

4.2.1 The fixed approach

Here, we decompose the list of all controllable states in subsets of fixed size which are distributed to nodes joining the network. These jobs are processed and results are stored in the master. This scheme, though efficient for homogeneous networks, does not really suit for their heterogeneous counterparts.

4.2.2 A simple dynamic distribution

Our following technique is more adapted to cope more gracefully with heterogeneous networks. The major drawback one can notice when dealing with heterogeneous networks is the inadequate size of the job provided to a node which is not tuned w.r.t its power and its communication link bandwidth to the master. In our adaptive algorithms, a node supplies its *characteristics* to the master in order to get a *fitting job* when it joins the peer-to-peer network.

A straightforward scheme

The simplest manner to distribute jobs to processors is to give *fitting job* according to the node characteristics. Indeed, from now on, a joining node gives its characteristics when

joining the network. In our dynamic scheme, we give to it a job having adequate size to its capability. Although this *modus operandi* seems simple, it is likely to be efficient.

5 Speeding up the test generation process

The test generation process seen in the previous section can introduce *redundant computations* which slow down our application. Indeed, several controllable states can share one or more sequences since they are spawn from all possible sequences found along a "trail of consecutive states". Since our process does not have any storage mechanism, each non-singular processed sequence is lost whenever a new state has to be identified. Thus, keeping in memory sequences already processed by the use of a cache seems to be a natural way to avoid this "amnesia". We present in this section some cache mechanisms based on *Bloom Filters* — an elegant alternative to lookup tables — which helps our algorithm to avoid the same computation.

Bloom Filters are generally used when the domain size or the set which needs to check membership is too large to be kept in memory. Bloom filters were used in early UNIX spell-checker [McI82] as well as in database computations for speeding up semi-joins operations [LR95]. In [Goh03], Goh proposes a bloom filter based scheme to search keywords in encrypted documents in constant time. Recently, Broder [BM02] has given a very comprehensive and historical survey of the use of Bloom filters for network applications. Indeed, Bloom Filters are extensively used in some Web cache sharing mechanisms [RW98], [FCAB00] to reduce the shared cache size. In fact, this probabilistic structure can be used whenever space complexity has to be considered with allowable membership error. In the early seventies, Bloom [Blo70] has shown that great storage reduction can be gained at the expense of introduced false positives when multiple hash functions are used to determine fewer element membership among lots of one. We leverage this mechanism to store the non-singular processed sequence in a Bloom filter which help us to avoid to re-process sequences already "seen". Unfortunately, the algorithm can "miss" possible singular sequence due to false sequences spawn by this technique. Therefore we can still find longer singular sequences since minimal singular sequences prefix longer ones. So, we are allowed to miss some singular sequences at the expense of their minimality.

5.1 Bloom Filters

Let's $S = s = s_1, s_2, \ldots, s_n$, a set of n elements which has to be to stored and A, an array of m bits initially set to 0 which will store the set S. Bloom Filters are d independent hash functions h_i which maps pseudo-randomly each element of the universe in a series of random numbers over the range $\{0, \ldots, m-1\}$ such as $A[h_i(s_k)] = 1$. An element membership is tested as follows : the element is hashed through the d hash functions and each resulted location in the array is tested. If one hash function maps to a bit at 0 in the

array, the element does not belong to the set S. This technique can drastically save storage space whenever the number of elements to store is low compared to the universe space. The drawback with this method is that it introduces some false positives which probability to occur is

$$(1 - (1 - \frac{1}{m})^{dn})^d \simeq (1 - e^{-dn/m})^d \quad (1)$$

There exists a trade-off to find between the size of the array of bits, the number of hash

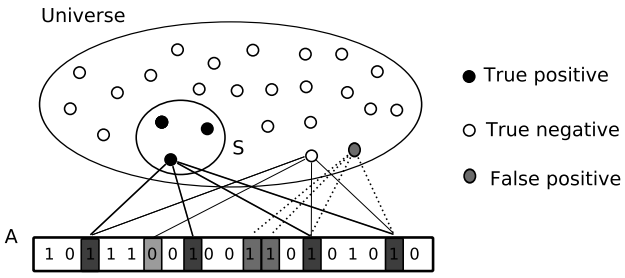


Figure 3: Bloom Filters with 4 hash functions

functions and the probability to find false positives. Nevertheless, it becomes relatively easy to find an optimal parameter by fixing the two others.

5.2 Experimental Results

Using Bloom Filters seems to speedup a lot the process whenever initial sequence depth does not suffice to find all possible unique sequences as shown on Figure 4. This figure details the time execution and the gain of the algorithm over a set of automata of 1000, 2000,..., 9000 states having the same number of labels(16). The average transitions per state is 5 and the average number of clocks is 5. Four types of experiments have been done : a sequential algorithm without Bloom (line : seq wo Bloom Filters), a sequential algorithm with Bloom (line : seq w Bloom Filters), a distributed algorithm without Bloom (line : 2 proc wo Bloom Filters) and a distributed algorithm with Bloom (line : 2 proc w Bloom Filters). In fact, it is shown that our algorithm becomes more efficient whenever the number of states rises. This can be explained by the fact that the more "the population" grows, the less you can find different "individuals" spawned from the same "strain" : the set of the all possible sequences does not vary from a certain number of states.

However, some cares must be taken with the choice of the depth. For very large deterministic automata with many transitions at each state, the non-singular processed sequence set can be large too and so must be the cache size. Fortunately, the larger the depth sequence is, the more the probability to find a unique sequence is if it exists. This explains why Bloom Filters does not really help our process from certain depth threshold. Indeed, our first possible sequence spawn at this depth threshold is unique with very high probability and cache mechanism does not trigger.

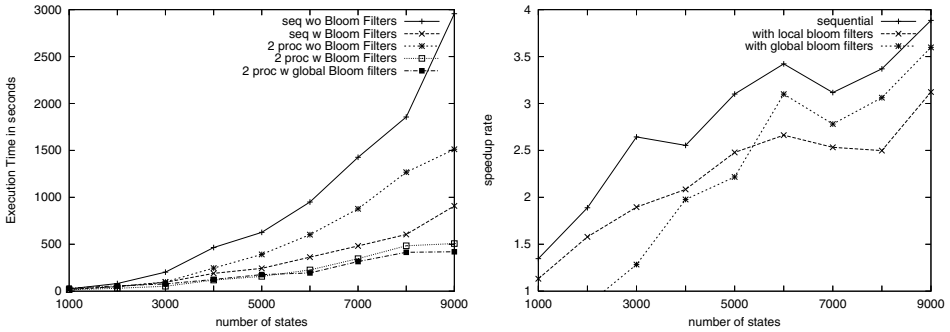


Figure 4: Time execution and Gain algorithm

The use of Bloom Filters were also conducted on a straightforward parallel version of our algorithm where Bloom Filters were used locally at each process running on each slave. As we could expect, results are less efficient whenever the number of slaves participating in the computation grows. To a great extent, if the master distributes *jobs* of one state to slaves as numerous as jobs, the Bloom filter is totally inefficient since it cannot help the others slaves. Indeed, slaves do not know their counterparts. As in some uses of Bloom Filters in network, they have to be shared using gossiping strategy for example.

A shared Bloom filter version has also been implemented. Its behavior is not so different from the ones used in shared web cache mechanisms. Each slave hold a local bloom filter which updates the master's one if need be. In this case, the filter is sent with the sequence results and is merged with the master's filter. The master then updates other local bloom filters when slaves fetch new jobs. So the master is responsible for sharing the updated global filter in our distributed environment.

As we could expect, results with global bloom filter have shown some improvements for large automata. However, we can notice that before some threshold (here, the number of states of the automaton), the bloom filter is inefficient. This is due insofar to costly communications which are used to share the filter between the slaves. Fortunately, they become negligible as the computation time for identifying a controllable state rises.

6 Conclusion and future work

We have suggested in this paper a distributed algorithm which is able to find test sequences for large timed systems (a suitable model to describe real-time protocols, embedded systems, web services). Large-scale parallelism opens the way to test sequence generation. As far as we know, this study is one of the first attempt to merge large-scale peer-to-peer parallelism and test generation. In order to have better results, we suggest to use a local cache on each slave which is able to keep some calculated sequences which may be needed later. This cache uses the Bloom filters concepts (an elegant technique to lookup tables). The use of these filters have shown better results (for the distributed algorithm) even if

Bloom filters offer a better gain in a sequential algorithm.

This study needs to be extended in order to handle fault tolerance. In the present solution, each slave has to reply to the master with its set of test sequences. In case of troubles, the algorithm is not able to take care of them.

The use of this technique on industrial system (multimedia protocol, real-time system, ..) will be undertaken very soon in order to show the efficiency of the technique.

References

[AD94] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[AHJN05] Gabriel Antoniu, Phil Hatcher, Mathieu Jan, and David A. Noblet. Performance Evaluation of JXTA Communication Layers (extended version). Technical Report PI-1686, IRISA, January 2005.

[AT02] Karine Altisen and Stavros Tripakis. Tools for Controller Synthesis of Timed Systems. July 03 2002.

[Blo70] Burton H. Bloom. Space/Time Trade-offs in Hash Coding with Allowable Errors. *Communications of the ACM*, 13(7):422–426, 1970.

[BM02] A. Broder and M. Mitzenmacher. Network Applications of Bloom Filters: A Survey, 2002.

[CL97] Duncan Clarke and Insup Lee. Automatic Generation of Tests for Timing Constraints from Requirements. In *Proceedings of the Third International Workshop on Object-Oriented Real-Time Dependable Systems,* Newport Beach, California, February 1997.

[CO02] Rachel Cardell-Oliver. Conformance Test Experiments for Distributed Real-Time Systems. In *International Symposium on Software Testing and Analysis (ISSTA?02), ACM Press, July 2002*, July 2002.

[DNH84] R. De Nicola and M. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.

[ENDKE98] A. En-Nouaary, R. Dssouli, F. Khendek, and A. Elqortobi. Timed Test Cases Generation Based On State Characterization Technique. In *19th IEEE Real Time Systems Symposium (RTSS'98)* Madrid, Spain, 1998.

[FBK+91] S. Fujiwara, G. Bochmann, F. Khendek, M. Amalou, and A. Ghedamsi. Test Selection Based on Finite-State Models. *IEEE Transactions on Software Engineering*, 17(6):591–603, June 1991.

[FCAB00] Li Fan, Pei Cao, Jussara Almeida, and Andrei Z. Broder. Summary cache: a scalable wide-area Web cache sharing protocol. *IEEE/ACM Transactions on Networking*, 8(3):281–293, 2000.

[Goh03] E. Goh. Secure Indexes, March 16 2003.

[HNTC01] Teruo Hogashino, Akio Nakata, Kenichi Taniguchi, and Ana R. Cavalli. Generating Test Cases for a Timed I/O Automaton Model. October 2001.

[KT04] Krichen and Tripakis. Real-Time Testing with Timed Automata Testers and Coverage Criteria. 2004.

[LR95] Zhe Li and Kenneth A. Ross. PERF Join: An Alternative to Two-way Semijoin and Bloomjoin. In *CIKM*, pages 137–144, 1995.

[McI82] M. Douglas McIlroy. Development of a spelling list. *IEEE Trans. Communications*, COM-30:91–99, 1982.

[NS01] Brian Nielsen and Arne Skou. Automated Test Generation from Timed Automata. In T. Margaria and W. Yi, editors, *Proceedings of the Workshop on Tools and Algorithms for the Construction and Analysis of Systems,* Genova, Italy, volume 2031 of *Lecture Notes in Computer Science*, pages 343–357. Springer-Verlag, April 2001.

[PF99] E. Petitjean and H. Fouchal. From Timed Automata to Testable Untimeed Automata. In *24th IFAC/IFIP International Workshop on Real-Time Programming, Schloss Dagstuhl, Germany*, 1999.

[RNHW98] P. Raymond, X. Nicollin, N. Halbwatchs, and D. Waber. Automatic testing of reactive systems, Madrid, Spain. In *Proceedings of the 1998 IEEE Real-Time Systems Symposium, RTSS'98*, pages 200–209. IEEE Computer Society Press, December 1998.

[RW98] Alex Rousskov and Duane Wessels. Cache digests. *Computer Networks and ISDN Systems*, 30(22–23):2155–2168, 1998.

[SD88] K. Sabnani and A. Dahbura. A Protocol Test Generation Procedure. *Computer Networks and ISDN Systems*, 15:285–297, 1988.

[SVD01] J. Springintveld, F.W. Vaandrager, and P. R. D'Argenio. Timed Testing Automata. *Theoretical Computer Science*, 254(254):225–257, 2001.

Chapter 8: Overlays and Ubiquitous Computing

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Raphael Chand, Luigi Liquori, Michel Cosnard
Resource Discovery in the Arigatoni Model

Christophe Guéret, Nicolas Monmarché, Mohamed Slimane
Self-Organizing Ant-based Information Gossiping Algorithm for P2P Networks

Hyosook Jung, Jinhyun Ahn, Seongbin Park
A JXTA-based System for Adaptive and Collaborative Learning

Sergio Maffiolett, Simon Schubiger, Michèle Courant, Béat Hirsbrunner
A Homogeneous Service Framework for Pervasive Computing Environments

Resource Discovery in the Arigatoni Overlay Network

Raphael Chand Luigi Liquori
Michel Cosnard

INRIA, France
{Raphael.Chand,Luigi.Liquori,Michel.Cosnard}@inria.fr

Abstract: Arigatoni is a lightweight Overlay Network for dynamic and generic *Resource Discovery*. Entities in Arigatoni are organized in *Colonies*. A colony is a simple virtual organization composed by exactly one leader, offering some broker-like services, and some set of *Individuals*. Individuals are subcolonies of individuals, or basic units called *Global Computers*. Global computers communicate by first registering to the colony and then by mutually asking and offering services. The leader, called *Global Broker*, has the job to analyze service requests/responses coming from its own colony or arriving from a surrounding colony, and to route requests/responses to other individuals. After this discovery phase, individuals get in touch with each others without any further intervention from the system, typically in a P2P fashion. Communications over the behavioral units of the overlay network are performed by a simple *Global Internet Protocol*. Arigatoni provides fully decentralized, asynchronous and scalable resource discovery, that can be used for various purposes from P2P applications to more sophisticated Grid applications.

The main focus of this paper is to present the resource discovery algorithm used in Arigatoni, that is reminiscent to some algorithms employed in the publish/subscribe paradigm. We show some simulations that show that resource discovery in Arigatoni is efficient and scalable.

1 Introduction

Motivations. The *Global Computing Communication Paradigm*, *i.e.* computation via a seamless, geographically distributed, open-ended network of bounded resources owned by agents acting with partial knowledge and no central coordination is one of the most interesting challenges for the next decade. Aggregating many global computers sharing similar or different resources leads to a *Virtual Organization*. Moreover, organizing many overlay computers, using, *e.g.* tree- or graph-based topology leads to an *Overlay Network*.

The main challenge in this new field of research is how single resources, offered by the global/overlay computers are discovered. The process is called *Resource Discovery*: it requires an *up-to-date* information about widely-distributed resources. This is a challenging problem for very large distributed systems particularly when taking into account the continuously changing state of resources offered by global/overlay computers and the possibility of tolerating intermittent participation and dynamically changing status and

availability.

The first presentation of the Arigatoni overlay network was given in [BCLV06]. Reciprocity and hierarchical organization of the virtual organization in *Colonies*, governed by a clear leader (called *Global Broker*) are the main achievements of Arigatoni. Global computers belong to only one colony; requests for services and resources located in the same/another colony traverse a broker-2-broker negotiation which security is guaranteed via standard PKI mechanisms. Once the resource offered by a global computer has been found, the real resource exchange is performed out of the Arigatoni itself, *e.g.* in a P2P fashion.

In this paper, we explain how Arigatoni offers decentralized, asynchronous, and generic *resource discovery*. Once a global computer has issued a request for some services, Arigatoni finds some individuals that can offer the resources needed, and communicates their identities to the (client) global computer as soon as they are found.

The fact that Arigatoni only deals with resource discovery has one important advantage: the complete generality and independence of any given requested resource. Therefore, Arigatoni can fit with various scenarios in the global computing arena, from classical P2P applications, like file sharing, or band-sharing, to more sophisticated Grid applications, like remote and distributed big (and small) computations, until possible, futuristic *migration computations*, *i.e.* transfer of a non completed local run in another GCU, the latter scenario being useful in case of catastrophic scenarios, like fire, terrorist attack, earthquake etc.

Arigatoni extends the pub/sub paradigm for resource discovery. Arigatoni takes inspiration by the *Publish/Subscribe* paradigm [EFGK03]; several pub/sub have been developed recently, such as XNet [CF04, CF03], Siena [CRW01] or IBM Gryphon [BCM+99]. In [Hei01], the authors propose to adapt the Siena publish/subscribe system to achieve Gnutella-like resource discovery, by publishing queries to the notification service. In contrast, Arigatoni implements its own resource discovery algorithm, especially designed for generic and scalable resource lookup.

In Arigatoni, resource discovery works by asynchronously disseminating request messages in the system until some individuals have been found. More precisely, when global computers log in the system (a colony), they declare the list of services that they can offer. When a global computer asks for some services, it issues a service request message to its leader, without addressing it to any particular receiver. The system disseminates the message according to the services included in it *and* according to the services that the other global computers have declared. As a consequence, the communication model underlying Arigatoni extends conservatively pub/sub. Indeed, in the pub/sub paradigm, consumers subscribe to the system (typically called the *Notification Service*) to specify the type of information that they are interested in receiving. Producers publish data to the system. The notification service disseminates the data to all (if possible) the consumers that are interested in receiving it, according to the *content* of the data *and* the interests declared by the consumers. In Arigatoni, global computers "subscribe" to the system by declaring the services that they offer to serve. The same global computers also "publish" data in the system when they issue service requests. Arigatoni disseminates the data according to

the services included in the requests and the services that the other global computers have declared.

The pub/sub like communication form used in Arigatoni for resource discovery has several advantages. First, it allows Arigatoni to realize a full decoupling, in time, space, and synchronization, between the global computers. Second, due to its asynchronous nature, Arigatoni is, potentially, more scalable and can work in "disconnected" mode (*e.g.*, for mobile users and wireless devices). Third, indirect addressing makes it possible for the infrastructure to implement reliability, load balancing, fault-tolerance, persistence, or transactional semantics. More practically, since Arigatoni has a tree-like topology, we can use the pub/sub subscription mechanisms described in existing tree-based pub/sub systems such as XNet [CF03, CF04, Cha05] or Siena [CRW01], for subscription management, *i.e.*, for the construction and the update of *consistent* routing tables in the system. In addition, we can use the reliability mechanisms described in [CF04] to allow Arigatoni to be fault-tolerant or to adapt to dynamic topology changes.

However, one major difference between Arigatoni and classic pub/sub systems lies in their *functionality*. Indeed, the classic pub/sub paradigm deals with the publication of messages whereas Arigatoni focuses on *pure* resource discovery. More precisely, classic pub/sub systems aim at disseminating published messages to *all* interested consumers. In contrast, in Arigatoni, when a service request is issued, the goal is to find one (or maybe some) individuals able to provide the services included in the request, but not *all* the potential individuals. As a consequence, a much smaller fraction of the system is traversed. Besides, the routing strategy of the colony leader consists in always trying to find potential reseources in its own colony first. If it fails, it then delegates the request to its leader. This strategy is reminiscent of the *dynamic method lookup* employed in all Object-Oriented languages, and increases *resource encapsulation* inside colonies, another concept strongly related to Object Orientation.

Another major difference lies in the nature of the published events in classic pub/sub systems and the nature of service requests in Arigatoni. Indeed, in classic pub/sub systems, subscriptions are constraints on the set of all possible events. In contrast, in Arigatoni, service requests are also expressed as constraints. This latter point will be explained in more details in Section 3.

2 System units

Two different kinds of units compose the Arigatoni system: *Global Computer Units* (GCU), and *Global Broker Units* (GBU).

• A GCU is the basic peer of the global computing paradigm. It is typically a small device, like a PDA, or a PC, connected via IP.

• A GBU is the basic unit devoted to register and unregister GCUs, to receive service queries from client GCUs, to contact potential servant GCUs, to negotiate with the latter the given services, to trust clients and servers and to send all the information necessary to allow the client GCU, and the servants GCUs to communicate. Every GCU can register

to only one GBU, so that every GBU controls a colony of collaborating global computers. Hence, communication intra-colony is initiated via only one GBU, while communication inter-colonies is initiated through a chain of GBU-2-GBU message exchanges. In both cases, when a client GCU receives an acknowledgment for a requested service (with trust certificate) from the proper GBU, then the client will enjoy the service directly from the servant(s) GCU, *i.e*. without a further mediation of the GBU itself.

• A *Colony* is a simple virtual organization composed by exactly one leader and some set (possibly empty) of individuals. Colonies are organized in a tree structure where the root of a colony is its *leader*. Individuals are global computers (think it as an *Amoeba*), or subcolonies (think it as a *Protozoa*). An individual can be a GCU or a GBU (representing the leader of a a subcolony). GCUs cannot have children in the hierarchy. As such, GBUs can have both GBUs and GCUs as their children. As such, a colony has *exactly* one leader GBU and has at least one individual (the GBU itself), and may contains individuals (GCU's, or colonies).

• A *Community* is a raw set of colonies and global computers (think it as a *soup* of colonies and GCU without a leader). Starting from a community, the Arigatoni protocol allows individuals to dynamically aggregate in colonies. This topic has been addressed and formalized in [CLC06].

The possibility for individuals to log/delog from a colony, or the possibility for a colony's leader to delog some "lazy" individuals makes *de facto* the network topology *dynamic*. This dynamicity implies that if GBUs hold routing tables about the services provided by their colony, particular care must be taken to maintain consistency when individuals log/delog. Moreover, due to the fact that individuals are not *slaves* but global computers with their own proper activity, a service request may lead to run-time failures. This happens when an individual gets busy by a local request, or when it suddenly delogs from the colony during the routing of the service request, or worst, when it gets hardware failures.

3 Resource discovery

Let $\mathcal{R}$ be the set of all possible resources (maybe infinite). GCUs provide resources by registering services to the system. A service S is a constraint on the set of resources. Let $match(S) \in \mathcal{R}$ be the set of resources that satisfy S. A GCU X that registers S announces that it can provide the set of resources $match(S)$. A GCU Y that issues a service request for service S' is looking for a resource that satisfies constraint S', *i.e*., a resource in $match(S')$. If $match(S) \cap match(S') \neq \emptyset$, then there exists a resource that satisfies both S and S', and X can provide a resource to Y. We say that S and S' *overlap* iff $match(S) \cap match(S') \neq \emptyset$. For example $S = [\mathsf{Type} = \mathsf{CPU}] \wedge [\mathsf{Time}{<}\mathsf{10s}]$ and $S' = [\mathsf{Type} = \mathsf{CPU}] \wedge [\mathsf{Time}{>}\mathsf{5s}]$ overlap, since any resource with attribute Time between 5s and 10s matches.

The principle of resource discovery in Arigatoni is as follows. When a GCU sends a request for a set of services $S_1 \cdots S_n$, it builds a "ServiceRequest" message containing the set of services and sends it to its leader GBU. The message is then recursively processed by the GBUs in the system so as to find some individuals able to serve the services included in the

Algorithm 1 The resource discovery algorithm in the Arigatoni GIP protocol

```
1:  case Message is
    SREQ :
2:      ReturnPath{Message.Id} ← Message.Sender
3:      SendList ← SelectPeers(Message.Services, search_mode)
4:      for each (P, Serv(P)) ∈ SendList do
5:          Send ServiceRequest(Serv(P)) to P
6:      end for
7:      for each S∈Message.Services such that ∄(P, Serv(P))∈SendList, S∈Serv(P) do
8:          Append S to RejectList
9:      end for
10:     Send ServiceResponse({},RejectList) to ReturnPath[Id]
11: SRESP :
12:     for each S∈Message.AcceptedServices do
13:         if (S was not already accepted ) ∨ (EXHAUSTIVE_REPLY is set) then
14:             Append S to AcceptList
15:         end if
16:     end for
17:     SendList ← SelectPeers(Message.RejectedServices, intra_Colony_mode)
18:     for each (P, Serv(P))∈SendList do
19:         Send ServiceRequest(Serv(P)) to P
20:     end for
21:     for each S∈Message.RejectedServices such that ∄(P, S(P))∈SendList, S∈Serv(P) do
22:         Append S to RejectList
23:     end for
24:     Send ServiceResponse(AcceptList,RejectList) to ReturnPath[Id]
25: end case
```

request. The main basic principle of the protocol is that every GBU that receives a request always searches its own colony first to find the potential individuals able to serve the services included in the request. If no individuals are found, then the request is delegated to its leader GBU, and the process proceeds recursively. In addition, if the GBUs maintain some information about the services provided by their children, then they can transform a received request into sub-requests, so as to only ask a given child for the services that it (or its colony) provides.

The process eventually leads to some GCUs receiving a request. When one such GCU receives a request for some services, it chooses the services that it accepts to serve and the ones that it refuses to serve. It then sends a "ServiceResponse" message containing the list of accepted services and the list of rejected services, and sends it to its leader GBU. The response messages are then propagated recursively in the system, following the reverse path.

The resource discovery algorithm is the core of the GIP protocol; it is described in pseudo-code in Algorithm 1 and explained as follows. We only focused on the case of GBUs. The resource discovery algorithm in the case of GCUs is similar and has been voluntarily omitted (see [BCLV06] for details). Indeed, the involvement of GCUs in the process of resource discovery is limited to directly replying to request messages. Arigatoni only deals with the discovery of resources, while the real resource exchange is done in a P2P fashion. Let GBU N receive a message from a neighbor.

Case of Service Request (SREQ)**.** We first consider the case of request messages. A request message received by GBU N means that N is asked to find some individuals to provide the services included in the request. For that purpose, N first maps the "*Id*" of the request included in the message to the sender of the message (line 2), so as to allow reply messages to follow the reverse path of the request.

Line 3**: Various intra colony search modes.** The leader N then calls function "SelectPeers", taking as input the list of services, $Message.Services$, included in the request message, (line 3). SelectPeers returns a list of pairs $\{(\mathsf{P}, Serv(\mathsf{P}))\}$, called *SendList*, where the first element P of a pair is the *Id* of a neighbor, and the second element $Serv(\mathsf{P})$ is a list of services, subset of $Message.Services$, that contains the list of services to ask to neighbor P. The *search_mode* determines the way function SelectPeers determines the *SendList*. The *search_mode* depends itself on whether P maintains some information about the services provided by its colony, *i.e*. a routing table. Currently, the following search mode are allowed: *broadcast* and *selective*, where the latter is itself sub-divided into three sub-modes: *exhaustive*, *greedy random*, and *greedy ordered*. If P does not maintain a routing table, then it has no other choice than to ask all its children for all the services included in the message, *i.e*., to *broadcast* the request message. We will refer to this search mode as the *broadcast* mode. Now if P maintains a routing table that indicates which child leads to a potential individual able to serve a given service, then P can *selectively* send *customized* requests to its children. More formally, P only asks a child for a service that *overlaps* a service that it advertised, *i.e*. there exists a resource that satisfies both the requested service and the advertised service. We will refer to this mode as the *selective* mode. Consequently, P can choose some children and send them a request for the services that overlap the ones that they advertised. The selective search mode can then be refined as follows. Consider a particular service S included in the request message.

• In the *exhaustive* mode, P sends a request for service S to all the children that can serve it (*i.e*., that contain potential individuals in their colony).

• In the *greedy random* mode, P sends a request for S to only one child that can serve the request, chosen uniformly at random.

• In the *greedy ordered* mode, P sends the request to only one child, chosen according to some predefined or *ad hoc* criteria (*e.g*., depending on network factors, or according to the quantity of services that were accepted by each child, *à la* tit-for-tat).

In addition, we can refine even more the greedy modes, by introducing a parameter n, that defines the number of children to whom the request is sent. We could then define the *n-greedy random* or the *n-greedy ordered* modes. It is important to mention that the *SendList* variable can contain N's leader, call it L. That is, it may contain a pair $(\mathsf{L}, Serv(\mathsf{L}))$. For a particular service $S \in Serv(\mathsf{L})$, this happen when *no* child advertised some services that overlap S, *i.e*., there are no potential individuals able to serve service S in N's colony. GBU N then *delegates* service S to its leader GBU. To prevent routing loops, the sender of the request message is never considered as a service provider.

Lines $4-6$**: Forwarding service request messages.** Consequently, for each pair denoted $(\mathsf{P}, Serv(\mathsf{P}))$ in the *SendList*, N sends to neighbor P a service request message for services $Serv(\mathsf{P})$ (lines $4-6$).

Lines $7-9$**: Services rejection.** Finally, each service S included in the request message,

and such that no potential individual was found amongst N's neighbors, is reported as rejected by N, to the original issuer of the request message (lines $7-10$). Since N may only maintain information about its own colony (apart the *Id* of the leader), this may only happen if N is the root of the topology or if the request message originated from N's leader.

Case of Service Response (SRESP). We now consider the case of reply messages. As previously explained, the process of propagating SREQ messages eventually leads to a certain number of GCUs receiving a request. Each GCU sends a reply message to its leader, with the list of accepted and the list of rejected services, along with its *Id*. Consequently, a given GBU N that participated in the propagation of the SREQ message eventually receives a certain number of SRESP messages from each of its children that was sent an instance of the (maybe transformed) SREQ message. Consider now an SRESP message sent to GBU N by a neighbor Q.

Lines $12-16$ and 24: Reporting accepted services. For each accepted service S, there are two different possibilities: either Q is the first child that accepted to serve the service, or the service was already accepted by some child other than Q. In the first case, N sends the reply back to the original sender or the request, reporting that service S has been accepted (lines 14 and 29). Otherwise, some neighbor other than Q already accepted to serve service S (*i.e.*, an individual in its colony). Then, if the EXHAUSTIVE_REPLY parameter flag is set (either in the GBU or included in the original request message), N also reports the reply back. Consequently, in the EXHAUSTIVE_REPLY mode, every GCU that accepted to serve a given service will be reported back to the GCU that issued the request. Otherwise, for each service asked in the request, only one single servant GCU will be reported. Furthermore, it is easy to add more flexibility by including a threshold $T_r>1$ on the number of replies. For example each GBU would report back T_r replies for the same service(s).

Lines 17: Finding other individuals for rejected services. We now consider the case of rejected services S. This means that in Q's colony, no potential individuals serving service S could be found, or no individuals accepted to serve it. Then, N has to find other neighbors that might contain individuals for service S. Consequently, N calls again function SelectPeers, with the list of rejected services as input (line 17). The function works as previously explained, except that it does not consider the peers (including Q) that were already sent a particular service. Also, logically enough, the services that were previously accepted are ignored. Finally, the original sender of the request is not considered (*i.e.*, $ReturnPathId$). Note that in the case where the *exhaustive* search mode is used, then the list $SendList$ returned by function SelectPeers may only contain a single pair $(\mathsf{L}, Serv(\mathsf{L}))$ (L is N's leader). Indeed, in the *exhaustive* search mode, all possible children in N's colony have already been asked for all the services included in the request message, that they can serve. Hence, rejected services are directly delegated to the leader L, if possible (*i.e.* if the latter was not the original sender of the request). The variable *SendList* contains a list of pairs $(\mathsf{P}, Serv(\mathsf{P}))$, where neighbor P is an individual that can potentially serve the services in $Serv(\mathsf{P})$, *and* has not been sent a request for any of them yet.

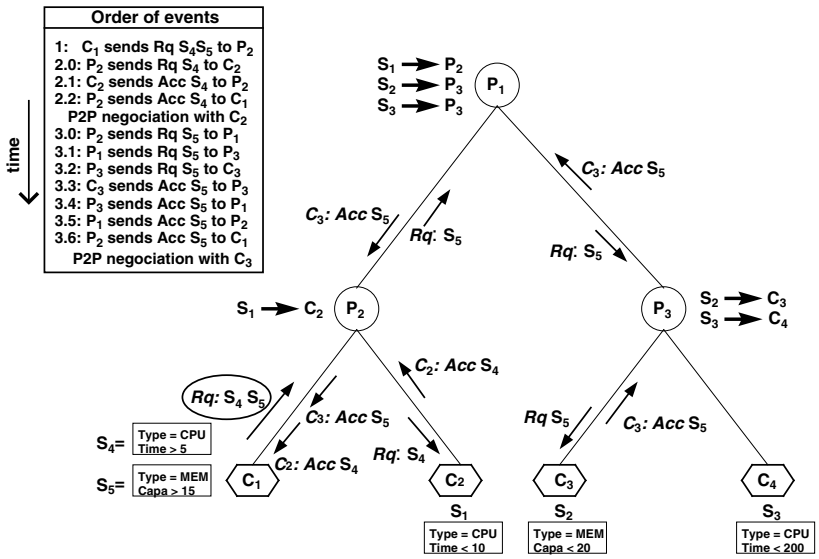


Figure 1: Resource discovery scenario.

Lines $18-20$**: Forwarding request messages for rejected services.** Consequently, for each pair $(\mathsf{P}, Serv(\mathsf{P}))$ included in *SendList*, N sends to neighbor P a service request message for services $Serv(\mathsf{P})$ (lines $18-20$).

Lines $21-23$ **and** 24**: Service rejection.** Finally, each service S included in the list of rejected services, and such that no additional potential individual could be found amongst N's neighbors, is reported as rejected by N, to the original issuer of the request message (lines $21-24$).

Example. Consider the example illustrated in Figure 1. Three GBUs are represented, namely $\mathsf{P}_1 \cdots \mathsf{P}_3$, and 4 GCUs, namely $\mathsf{C}_1 \cdots \mathsf{C}_4$. GCUs C_1 and C_2 (resp. C_3 and C_4) have P_2 (resp. P_3) as their leader, while P_1 is the leader of GBUs P_2 and P_3. GCUs C_2, C_3 and C_4 have registered services S_1, S_2 and S_3, respectively, and the routing tables of the upstream GBUs have been updated accordingly. In the example, resources are expressed as conjunctions of attribute/value pairs, and services are conjunctions of constraints on those attributes. We suppose that the *search mode* is set to *selective*, and we consider the scenario where GCU C_1 issues a service request for services S_4 and S_5, to its leader P_2. Since S_4 and S_1 overlap (any resource with $5<\mathsf{Time}<10$ satisfies both S_1 and S_4), GBU P_2 forwards a service request for service S_1 to GCU C_2. Note that given that S_5 and S_1 do not overlap, S_5 is not included in the request. Since P_2 does not find any GCU potentially able to serve S_5 (*i.e.*, no services in its routing table overlap with S_5), it delegates it to its leader GBU P_1. When C_2 accepts to serve S_4, it sends a reply message with its *Id* and the accepted service S_4, back to GBU P_2, which, in turn, forwards it back to C_1. Then

C_1 can directly negotiate the resource with C_2. When P_1 receives the service request for S_5, it forwards it to P_3 (since S_2 and S_5 overlap), which in turn forwards it to GCU C_3. When C_3 accepts to serve S_5, the same process then repeats as for GCU C_2. Eventually, C_1 receives a reply message with the *Id* of GCU C_3 and the accepted service, namely S_5. We have an illustration of the asynchronous communication (C_1 received the reply messages independently of each others) and the encapsulation of resources in Arigatoni (GBU P_2 only searched for service S_4 in its own colony, *i.e.* GCU C_2).

Discussions, load balancing, scalability. We mainly focused on the resource discovery mechanism used in Arigatoni. Total decoupling between GCUs in space (GCUs do not know each others), time (GCUs do not participate in the interaction at the same time), and synchronization (GCUs can issue service requests and do something else, or may be doing something else when being asked for services) is a major feature of Arigatoni. Another important property is the encapsulation of resources in colonies. Those properties play a major role in the scalability of resource discovery in Arigatoni.

As stated before, the subscription mechanisms of classical tree-based pub/sub systems [CF03, CF04, Cha05, CRW01] can be used for the maintenance and update of consistent routing tables. Furthermore, as for the reliability of subscription advertisement, we can adapt the reliability mechanisms described in [CF04] to allow Arigatoni to be fault-tolerant or to adapt to dynamic topology changes.

The reliability of the resource discovery mechanism itself, although desirable, is of lesser importance, given the fact that service provision is not guaranteed at all in Arigatoni. In other words, when a GCU issues a service request, it is possible that no individual is found for some of the services included in the request. This happens, for example, if those services were not declared by any GCUs in the system, or if all the GCUs that declared themselves as potential individual refuse to serve them. However, at the cost of memory and bandwidth requirements, it is still possible (future work) to implement reliable resource discovery by using a reliable transmission protocol (TCP), an acknowledgment scheme in combination with a retransmission buffer, and persistent data storage.

As defined above, GBUs are organized as a dynamic tree structure. Each GBU is a node of the tree, leader of its own subcolony and root of a subtree corresponding to the GBUs of its colony. It is then natural to address scalability issues that arise from that tree structure. In [CCL06], we show that, under reasonable assumptions, the Arigatoni model is scalable. However, a complete performance evaluation is out of the scope of this paper and will rather be studied in a future work.

4 Protocol evaluation

To assess the effectiveness and the scalability of our resource discovery protocol, we have conducted simulations using large numbers of units and service requests.

Simulation setup. We have generated a network topology using the transit-stub model of the Georgia Tech Internetwork Topology Models package [ZCB96], on top of which we added the Arigatoni overlay network. The resulting network topology, contains 103 GBUs. GCUs were not directly simulated in the network topology. Instead, to simulate the population of GCUs, we added a GCU *agent* to each GBU in the system. The GCU agent of a GBU represents the local colony of GCUs that are attached to that GBU as their leader.

We considered a finite set of resources $R_1 \cdots R_r$ of variable size r, and represented a service by a direct mapping to a resource. In other words, a service expresses the conditional presence of a single resource. We have a set of r services $\{S_1 \cdots S_r\}$, where service S_i expresses the conditional presence of resource R_i. A GCU declaring service S_i means that it can provide resource R_i. This simple model is still generic and sufficient for the main purpose of our experiments, which is to study the scalability of resource discovery in our system.

To simulate GCU load, we then randomly added each service with probability ρ at each GCU agent, and had it registered via the registration service of Arigatoni. The routing tables of the GBUs were updated starting at the initial GBU and ending at the root of the topology. In other words, it is as if each GBU has a probability ρ of having a GCU which registered service S_i, for any S_i. Thus, the parameter ρ can be seen as either the global availability of services, or as the density of population of GCUs (since the more the number of GCUs, the more likely it is that a given service is provided).

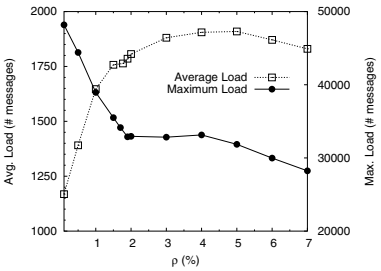
We then issued n service requests at GCU agents chosen uniformly at random. Each request contained one service (requests with k services can be seen as k service requests with one service), also chosen uniformly at random. Each service request was then handled by the resource discovery mechanism of Arigatoni described in Section 3. We used a service acceptation probability of $\alpha = 75\%$, which corresponds to the probability that a GCU that receives a service request *and* that declared itself as a potential individual for that service (*i.e.* that registered it), accepts to serve it.

The resource discovery algorithm was implemented in C++ and compiled using GNU C++ version 2.95.3. Experiments were conducted on a 3.0 Ghz Intel Pentium machine with 2 GB of main memory running Linux 2.4.28. The different experimental parameters are summarized in Figure 2. Upon completion of the n requests, we measured for each GBU its load as the number of requests (messages) that it received. We then computed the average load as the average value over the population of GBUs in the system. We also computed the maximum load as the maximum value of the load over all the GBUs in the system. Similarly, we computed the average and maximum load fractions as the average and max loads divided by the number of requests. The average load represents the average load of a GBU due to the completion of the n requests. The average load fraction represents the fraction of requests that a GBU served, in average. The maximum fraction represents the maximum fraction of the requests that a GBU served. Note that since a GBU receives at most one request message corresponding to a given service request, the average load fraction can be seen as the fraction of GBUs in the system involved in a service request, in average.

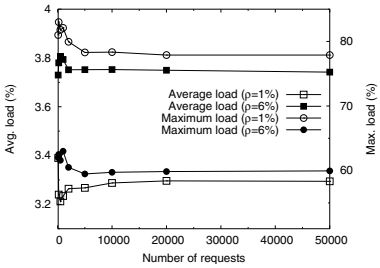
Finally, we computed the average service acceptation ratio as follows. For each GCU

Vars	Description	Value
K	Number of GBUs	103
r	Size of services pool	128
ρ	Service availability	0.1% to 7%
α	Service accept. prob.	75%
n	Number of SREQ issued	100 to 50000

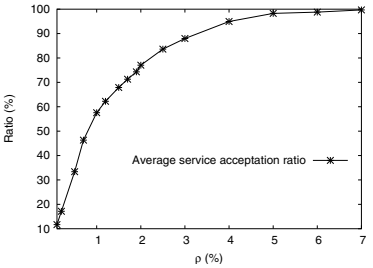
(a)



(b)



(c)



(d)

Figure 2: (a) Parameters of the experiments. (b) Average and maximum load w.r.t. service availability ρ. (c) Average and maximum load fraction w.r.t. the number of requests issued. (d) Average service acceptation ratio w.r.t. service availability ρ.

agent, we computed the local acceptation ratio as the number of service requests that yielded a positive response (*i.e.* the system found at least one individual), over the number of service requests issued at that GCU agent. We then computed the average acceptation ratio as the average value over the number of GCU agents (that issued at least one service request).

We repeated the experiments for different values of ρ and n. Results are illustrated in Figure 2. Figure 2(b) and (d) were obtained with a fixed value of n of 50000 service requests.

Results and interpretations. Figure 2(b) shows the evolution of the average and maximum load when varying the service availability ρ. The maximum load was obtained for GBUs at the top of the leader hierarchy in the tree topology. It appears that the maximum load decreases with the service availability, while the average load increases. In other words, the load is more evenly distributed amongst the GBUs in the system. This is due to the strategy of our resource discovery mechanism which consists in always searching for individuals in its own colony first before delegating to its leader. Indeed, as the service availability increases, GBUs have a higher chance to find individuals in their own colony. Hence, the root leader (say GBU Top) of the topology participate less in the process of resource discovery, and the direct subleaders (say GBU SubTop) participate more. In other

words, the resource discovery mechanism used in Arigatoni does not overload superleaders in the tree topology.

We also observe in Figure 2(b), for values of $2\% \leq \rho \leq 4\%$, a *"plateau"* in the curve of the maximum load, followed by a decreasing phase ($\rho > 4\%$), but with a much lower slope than before ($\rho < 2\%$). This is due to the fact that for $\rho < 2\%$, the root leader Top of the whole topology has the maximum load in the system. For $\rho > 2\%$, however, the immediate direct subleader Subtop takes over. This transition can be explained by the fact that for higher values of ρ, less messages are delegated to Top. At some point ($\rho \simeq 2\%$), the load of Top becomes less important than that of SubTop, due to the high number of colonies that the latter manages. The constantness observed in the curve around that value is probably due to the fact that a transition phase is necessary for SubTop to be sensitive again to the increase of ρ. The following decreasing period with a lower slope corresponds to the fact that SubTop is less sensitive to an increase of ρ (indeed, SubTop is mostly concerned with the availability of services in its own colonies).

Finally, we observe that the average load stabilizes, which shows that the system scales to large number of GCUs (since as previously mentioned, the service availability ρ can be assimilated to the number of GCUs in the system).

Figure 2(c) shows the average and maximum load fractions w.r.t. the number of service requests. It appears clearly that Arigatoni scales to large numbers of requests. In fact, the average number of requests received by a GBU increases linearly with the total number of requests, at a rate of $\sim 3.5\%$. In other words, in average, a GBU only receives $\sim 3.5\%$ of the total number of requests. Equivalently, only 3.5% of the overall population of GBUs in the system participate in the process of discovering a particular resource, in average. Figure 2(c) also shows that low level GBUs in the topology are not particularly overloaded (the most overloaded GBU manages 60% of the overall load for $\rho = 6\%$). Finally, it corroborates the assertion that higher values of ρ favor the maximum load over the average load, *i.e.*, load balancing gets more effective.

Figure 2(d) shows that, unsurprisingly, the average service acceptation ratio increases exponentially with the availability of services. This shows that Arigatoni is efficient in searching individuals for requested services. Indeed, a service availability of 4% enables the system to achieve an acceptation rate of 90%. In other words, the more the number of GCUs in the system, the more chances to find an individual for a service request.

5 Conclusion

In this paper, we presented the Arigatoni lightweight overlay network. We exposed in details the mechanisms that allow Arigatoni to offer dynamic and generic resource discovery. The main achievements are the complete decoupling between the different units in the system, and the encapsulation of resources in local colonies, which enable Arigatoni to be potentially scalable to very large and heterogeneous populations. We are currently improving Arigatoni with several new features, such as the possibility to ask a certain number of instances of a service (*i.e.*, the system should find the specified number of GCUs capable

of providing that service), or the possibility to embed services in conjunctions (*i.e.*, the services in a conjunction should be provided by the same GCU). We are also working on the implementation of a real prototype and the subsequent deployment on the PlanetLab experimental platform, and/or on GRID5000, the experimental platform available at the INRIA. As part of our ongoing research, we are also working on a more complete statistical study of our system, based on more elaborate statistical models and realistic assumptions.

Acknowledgment. The authors would like to thank Philippe Nain for its invaluable comments and interactions on the Arigatoni performance model. This work is supported by Aeolus FP6-2004-IST-FET Proactive.

References

[BCLV06] D. Benza, M. Cosnard, L. Liquori, and M. Vesin. Arigatoni: A Simple Programmable Overlay Network. In *Proc. of John Vincent Atanasoff International Symposium on Modern Computing*. IEEE, 2006. To appear. Also as INRIA RR 5805.

[BCM+99] G. Banavar, T. Chandra, B. Mukherjee, J. Nagarajarao, R.E. Strom, and D.C. Sturman. An efficient Multicast Protocol for Content-Based Publish-Subscribe Systems. In *Proc. of ICDCS*, 1999.

[CCL06] R. Chand, M. Cosnard, and L. Liquori. Resource Discovery in the Arigatoni Model. Technical Report 5924, INRIA, 2006.

[CF03] R. Chand and P. Felber. A Scalable Protocol for Content-Based Routing in Overlay Networks. In *Proc. of NCA*, 2003.

[CF04] R. Chand and P. Felber. XNet: A Reliable Content-Based Publish/Subscribe System. In *SRDS 2004, 23rd Symposium on Reliable Distributed Systems*, 2004.

[Cha05] R. Chand. *Large scale diffusion of information in Publish/Subscribe systems*. PhD thesis, University of Nice-Sophia Antipolis and Institut Eurecom, 2005.

[CLC06] M. Cosnard, L. Liquori, and R. Chand. Virtual Organizations in Arigatoni. *DCM: International Workshop on Developpment in Computational Models. Electr. Notes Theor. Comput. Sci.*, 2006. To appear.

[CRW01] A. Carzaniga, D.S. Rosenblum, and A.L. Wolf. Design and Evaluation of a Wide-Area Event Notification Service. *ACM TOCS*, 19(3), 2001.

[EFGK03] P. Th. Eugster, P. Felber, R. Guerraoui, and A.M. Kermarrec. The many faces of publish/subscribe. *Computing Survey*, 35(2):114–131, 2003.

[Hei01] D. Heimbigner. Adapting publish/subscribe middleware to achieve Gnutella-like functionality. In *SAC '01: Proc. of SAC*, pages 176–181, 2001.

[ZCB96] E.W. Zegura, K. Calvert, and S. Bhattacharjee. How to Model an Internetwork. In *Proc. of INFOCOM*, 1996.

A Self-Organizing Ant-based Information Gossiping Algorithm for P2P Networks

Christophe Guéret, Nicolas Monmarché, Mohamed Slimane

Université François Rabelais Tours, Laboratoire d'Informatique,
Polytech'Tours, 64 avenue Jean Portalis - 37200 Tours, France
{gueret,monmarche,slimane}@univ-tours.fr

Abstract: They appeared in our life only few years ago and now they are everywhere: computers have become ubiquitous and, almost, irreplaceable. Classical ways of creating, managing and exchanging information have been progressively replaced by electronic means. Everyday, information diffusion tools like the World Wide Web, E-mails, Forums and other Blog software are now commonly used. However, in spite of this plebiscite, computer based information managing still suffers some weaknesses. Mainly, software aimed to do CSCW (Computer Supported Collaborative Work) can be blamed for requiring the user to do an effort to use them. In this paper we present an algorithm aimed at perform autonomous selective dissemination of messages within a network. It constitutes the communication layer of our framework called PIAF ("Personal Intelligent Agent Framework") which is intended to help users transparently share information. This algorithm works in a fully decentralized way, using epidemic diffusion mechanism and artificial ants paradigm to achieve self-organization and information flows management.

1 Introduction

Computer based technology occupy an important place in our daily life and now are considered to be ubiquitous [WGB99]. During the last decades, using computers has modified users' habits. Electronic documents have changed the way to write, archive and diffuse content while Internet has changed the way we collaborate. Now, it is possible to work on a same project, exchanging documents or chatting regardless the physicals positions of the co-workers.

People doing collaborative work may have to share two types of content: data and knowledge. Data can be any electronic container such as text files, video or web links. Emails and file-sharing software are among the most used form of data sharing. Knowledge is related to the user's mind and may not be represented as a data file on a computer. For instance, a recommendation about a good restaurant or an advise concerning a research topic are both examples of knowledge sharing. Supposing it is possible to represent knowledge as a data file to share, from now on, we will use the generic term of "resource" to design both shared data and knowledge.

Depending on the user, we can distinguish between two kinds of resource sharing: implicit

or explicit. Sending an email is an explicit act while using a software to share idle CPU time is implicit for the user. Explicit sharing is the most challenging task for the user. Let us suppose a user finds an interesting website and wants to have all other, potentially interested, peers know about it. The strategy may be either 1) within the set of known peers, inform the subset of peers more likely to be interested by the website 2) inform all peers and let them decide if they are interested or not. In the first case, the risk for the sender is to omit some interested peers while in the second, the risk is to spam (*ie*: sending unwanted messages to some peers). This problem is one of the difficulties related to the usage of Computer Supported Collaborative Work (CSCW) software [Gru88]:

- Lack of mutual awareness: sharing content efficiently implies a global knowledge of the peers' needs. In order not to bother every single one, every user should know what his/her peers are interested in. But users may randomly appear and disappear in the network. Also they may be interested in different domains or make spurious searches from time to time. Thus, maintaining such knowledge is difficult.
- Users might not be motivated enough in using a software helping them sharing resources. Such software may involve, for instance, sending emails to people inside the network or using dedicated tool to tell them about what they have found. In both cases the user has to make an effort. Users generally do not like to change their habits and such solutions may weaken their motivation and dissuade them from using the software.
- Users can not define precisely what they are interested in: if we take the example of web browsing, users are most likely to jump from page to page looking for interesting links rather than follow a precise and predetermined path.

We believe that a resource sharing system based on implicit sharing could cope with those problems. We have designed the PIAF software in order to follow this idea. PIAF stands for Personal Intelligent Agent Framework. This framework is divided in two main layers: communication and dialog. The communication layer takes in charge information flows within the network. The dialog layer is the interface between the user and the network. In this paper we focus on the communication layer.

Our algorithm uses an ant paradigm, an idea which as already been explored in the context of content-based searches in unstructured P2P networks [BMM02]. Many ant species are known to use chemical trails to perform navigation and food exploitation. In this case, the involved volatile substances, called pheromones, are a kind of indirect communication mean between workers of the colony. These biological principles have been translated to many combinatorial problems modeled with a graph [BDT99] and even when a network routing problem are considered (see for instance the case of mobile ad hoc networks [DDG05]). The good behavior of this class of algorithm, especially within distributed environment, has led us to use similar principles.

The reminder of this paper is organized as follows. In section 2 we discuss some existing solutions for message circulation and dynamic topology management in P2P networks and state on the originality of our proposition. The following section 3 presents the algorithms

we have developed. Finally, we present experimental results in sections 4 and 5 before concluding in section 6.

2 Information flows and rewiring schemes

Message flows in a network are generated by exchanges between a server having a shared resource and a client asking for it. Actually, in a network of collaborators each user may act as a client (looking for a resource) or a server (informing users about what it shares). We are then in front of a so-called P2P network. Because of this duality of roles, finding a given resource is not easy. The tricky task for a client being mainly to find a relevant provider. In centralized networks such as Napster [Nap03], a server knowing which resources are shared and who is sharing them in the network is used to find them. However, those solutions have proved not to scale efficiently and fully decentralized P2P architectures are preferred. In this context, two main strategies can be considered for resources exchanges: they correspond to the "push" and "pull" strategies for information exchange as introduced by [FZ98]:

- *Query processing*: A users sends a query and the system returns a list of peers to contact. In a structured overlay, the location of peers and data depends on their respective identifiers. Usually, the search space is partitioned among peers with a Distributed Hash Table [IMK+01] algorithm or a hierarchical ordering of identifiers [PRR97]. A query is routed from peer to peer until it reaches a peer having a particular identifier. On the opposite, an unstructured overlay does not rely on a predefined architecture. A query is blindly (*e.g.* without information on the underlying topology) spread in the network until a result is obtained.

- *Selective dissemination*: The dual scenario of query processing is when users do not send query at all. Instead, an event based system is considered: clients subscribe to the event service by submitting a profile while servers publish events which will be dispatched to attended recipients. The publish/subscribe (pub-sub for short) scheme may be topic-based or content-based depending if the profile defines constraints concerning the topic of the event or its content. If the pub-sub is built over a structured overlay, events are routed. Whereas over an unstructured overlay, probabilistic dissemination (e.g. Gossip) is a more suitable strategy. Gossip protocols [DGH+88] consist in sending a message to a subset of the connected peers, according to a given probability.

The communication algorithm of PIAF is aimed at automatically disseminating news using a gossip-based diffusion scheme. Hence it falls under the category of selective dissemination algorithms. The peers will have to be informed of the existence and location of shared resources. Following the idea of Newscast [VJvS03], we will refer to this information as a "News". The objective is not to have a reliable multicast, that is which ensures that all peers recover all existing news [DGH+88, CAPMN03, VJvS03], but, instead, which performs a directed and focused diffusion [LM00]. A given information is gossiped to peers

more likely to be interested in it. Unlike other selective information dissemination systems for P2P network [KTID03], we do not consider the user has to define a profile to get useful news. Our approach is similar to the concept of autonomous gossiping introduced by Datta *et al.* [DQA04] although, unlike them, we do not use individual profiles to defines users interests nor we associate categories to news.

Each peer is connected to a limited amount of other peers. This provides them with a partial view of the whole network. Dynamic topology is used to adapt this set of connections according to a given criteria. It has been observed that a social network of collaborator exhibits small world properties: the network is made of many dense groups loosely connected to each other [WS98]. Those groups appear when individuals congregate as they found themselves having shared center of interest. Dynamically adjusting the topology of the P2P overlay network in order to make it similar to the underlying small world can improve sharing efficiency. A criteria is used to decide if two peers have similar interests or not. Depending on it, a given connection may be dropped or kept. To compute this criteria, it is necessary to have a model of peer's interests. This model, usually referenced to as a profile, may commonly be published by the peers [HS04] or exchanged on demand [Sch04]. We propose a third new strategy inspired by the ideas of overhearing [BSSZ01] and use of information trails [Pay98] in a network. We consider that whenever a peer sends a message over the network, he gives an hint about what he is interested into. Hence, instead of inquiring about the expertises of one's peers, we guess them from traffic they generate over the network.

3 PIAF Communication layer

One can view the P2P network as a directed graph where each node $n_i \in \mathcal{N}$ is a peer. An edge $(i,j) \in \mathcal{C}(t)$, represents a connection from a peer n_i to a peer n_j present at an instant t. Introducing the set of possible arc in the graph $\mathcal{E}$, we have $\mathcal{C}(t) \subset \mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$. For a given peer n_i, we define the neighborhood $V_i(t) = \{n_j \in \mathcal{N} \mid (i,j) \in \mathcal{C}(t)\}$ has the subset of peers it is connected to. This model defines a social network, that is a network in which edges define relations between nodes. In our case, the relations reflect shared centers of interest. As stated earlier, groups tend to appear in social networks. The density of those groups is measured by a clustering coefficient $\gamma_i(t)$. For a peer n_i, it quantifies how dense its neighborhood is. Therefore, if $\gamma_i(t) \simeq 1$, then n_i is considered to be part of a dense group.

$$\gamma_i(t) = \frac{\text{total edges in } V_i(t)}{\text{total possible edges in } V_i(t)} \tag{1}$$

Artificial pheromones are defined on a vector space $\mathbb{R}^n$. Each news item I has an associated pheromone vector $\tau(I)$. Pheromone vectors $\tau_{i \to j}(t)$ are also associated to connections (i,j). We suppose the existence of a similarity s defined on this vector space $s : \mathbb{R}^n \times \mathbb{R}^n \mapsto [0;1]$. Thus, it is possible to evaluate the similarity between two connections, as well as the similarity of a connection related to a news item. On a connection, pheromones are used as a memory for news received from other peers. Therefore, a given

peer n_i will store incoming pheromone vectors $\tau_{i \leftarrow j}(t), j \in V_i(t)$ and update them every time he receives a news from one of its neighbors.

3.1 Ant's gossiping activity

Ants work as follows when disseminating a news item I. Every T_g unit of time, the ant will try to push I from its nest n_i to another nest n_j randomly chosen in $V_i(t)$. This activity consists in first choosing a destination and then update pheromones. Ants stop the diffusion of I once it has decided to stay at the nest k consecutive times. This parameter denotes its patience.

3.1.1 Choose a destination

A stochastic algorithm is used to select a destination n_j within $V_i(t)$. According to a similarity threshold s_{min}, neighbors are first sorted in two groups depending if they are likely to be interested by I or not. This similarity estimates if I is similar to other news items previously sent to a given neighbor. Those groups are respectively defined has $V_i(I,t)$ and $\overline{V_i(I,t)}$.

$$V_i(I,t) = \{n_j \in V_i(t) \mid s(\tau_{i \leftarrow j}(t), \tau(I)) \geq s_{min}\} \quad (2)$$

$$\overline{V_i(I,t)} = \{n_j \in V_i(t) \mid n_j \notin V_i(I,t)\} \quad (3)$$

This classification is also used to update two counters $PV_{i \rightarrow j}(t)$ and $\overline{PV_{i \rightarrow j}(t)}$ used to record how many positive valuations (PV) a given connection received. For a neighbor n_j, $PV_{i \rightarrow j}(t)$ is incremented if $n_j \in V_i(I,t)$ whereas $\overline{PV_{i \rightarrow j}(t)}$ is incremented if $n_j \in \overline{V_i(I,t)}$. Until it has finished diffusing it, an ant is not allowed to send a news item twice to a same peer. Thus a subset of $V_i(I,t)$ and $\overline{V_i(I,t)}$ is defined where visited peers $seen(I,t)$ are excluded.

$$V_i^{unseen}(I,t) = V_i(I,t) \setminus seen(I,t) \quad (4)$$

$$\overline{V_i^{unseen}(I,t)} = \overline{V_i(I,t)} \setminus seen(I,t) \quad (5)$$

The probability $P_{i \rightarrow j}(I,t)$ of a peer to be elected as a destination by the ant depends of the group it was assigned to (see equation (6)). For an interested peer, this probability is proportional to its relative similarity with I. Meanwhile, non interested peers may be equiproportionally chosen. Sending to either an interested or not interested peer is a mater of exploitation versus exploration considering the problem of finding an optimal messages flow. Performing only exploitation can help reaching this optimum but, on the other hand, exploration is needed to find news peers to connect to. Therefore, a trade-off must be found to allow trying to send news items to some other neighbors even if they does not seem to be interested. To achieve this, ants are granted with a notion of "freewill". According to a

probability η, an ant may choose to select a destination likely to be interested or not. Also, it has n^+ times more chances to stay at the nest rather than sending I to a non interested peer.

$$P_{i\to j}(I,t) = \begin{cases} \dfrac{\eta \cdot \delta(|V_i(I,t)| > 0) \cdot s(\tau_{i\leftarrow j}(t), \tau(I))}{\sum\limits_{z \in V_i(I,t)} s(\tau_{i\leftarrow z}(t), \tau(I))} & \text{if } n_j \in V_i^{unseen}(I,t), \\ \dfrac{1 - \eta \cdot \delta(|V_i(I,t)| > 0)}{|\overline{V_i^{unseen}(I,t)}| + n^+} & \text{if } n_j \in \overline{V_i^{unseen}(I,t)} \\ \dfrac{n^+ \cdot (1 - \eta \cdot \delta(|V_i(I,t)| > 0))}{|\overline{V_i^{unseen}(I,t)}| + n^+} & \text{if } i = j \end{cases} \quad (6)$$

3.1.2 Adjust pheromones

This step on the algorithm only occurs when the ant decides not to stay in the nest. On its way toward the peer n_j it has chosen, the ant will lay down pheromones. The amount of pheromones is defined by a factor $\rho(I, \Delta t)$, both used for evaporation and deposit of pheromones.

$$\tau_{j\leftarrow i}(t + \Delta t) = (1 - \rho(I, \Delta t)) \cdot \tau_{j\leftarrow i}(t) + \rho(I, \Delta t) \cdot \tau(I) \quad (7)$$

with Δt the time elapsed since a message was last transferred through this connection. $\rho(I, \Delta t)$ depends on two factors: the activity on the link and the source of I. The more messages are transferred through a connection, the more pheromones deposit will be important. Also, pheromones deposit should decrease as the news item is farther away from its origin. We have chosen to use a Gaussian for each factor and defined a maximum amount of deposit ρ_{max} (see equation 8).

$$\rho(I, \Delta t) = \rho_{max} \exp^{-\alpha r(I)} \exp^{-(\frac{\Delta t}{\sigma})^2} = \rho_{max} \exp^{-(\frac{\Delta t}{\sigma})^2 - \alpha r(I)} \quad (8)$$

α and σ are two regulation factors used to adjust the trade-off wanted between reactivity and memory of the system. $r(I)$ is the round count, that is, the number of peers this news item has crossed by since its creation.

3.2 Nest mobility

Moving the nest consists in modifying its neighborhood by adding or removing some links. A peer may have a maximum of V_{max} opened connections. To move the nest, the first step is to try fetching a contact from a directory of peers n_i knows. In case of success, this peer is contacted, otherwise, n_i ask one of his neighbor for help.

3.2.1 Connect to a new peer

Supposing such a contact is found, and before connecting to him, the peer still has to verify if $|V_i(t)| < V_{max}$. If not, the less efficient connection would be dropped.

1. Efficiency is defined as a ratio between the number of time a peer was estimated to be interested and the total number of estimations performed by ants.

$$\forall n_j \in V_i(t)\ ,\ U_j(t) = \frac{PV_{i\to j}(t)}{PV_{i\to j}(t) + \overline{PV_{i\to j}(t)}} \tag{9}$$

2. If $U(n_j)$ falls under a given threshold β, the connection is not considered not to be efficient enough. Hence it has a probability $P_{i\to j}^{drop}(t)$ to be dropped. The lower $U_j(t)$, the higher this probability is.

$$\forall n_j \in V_i^{'}(t)\ ,\ P_{i\to j}^{drop}(t) = \frac{\beta - U_j(t)}{\sum_{z\in V_i^{'}(t)} \beta - U_z(t)} \tag{10}$$

   with $V_i^{'}(t) = \{n_j \in V_i(t) | U(n_j) < \beta\}$ the subset of inefficient neighbors.

3. If $|V_i(t)| < V_{max}$ a connection is established with the peer previously picked up from the directory. Otherwise, all connections were useful and no one was dropped.

3.2.2 Ask for a suggestion

If n_i was not able to find a peer in is directory, it asks one of is neighbor to send him a suggestion. It sends to a peer n_j a message with a copy of is directory. n_j then browses its own directory and answers to n_i sending him back the address of the peer most likely to be useful for him. The peer n_j is itself picked from $V_i(t)$ with a rank based selection based on similarities.

4 Simulation environment

A discrete event simulator was used in order to implement and test our information dissemination and topology management algorithms. We have chosen to use the OmnetPP discrete event simulator [Var02].

We make not assumption concerning the nature of metadata represented by the pheromones. During all the algorithms steps, only the similarity between two vectors is considered. Hence, we choose to generate an artificial dataset that suits to our needs. It is made of 4 categories of news items C_1, C_2, C_3 and C_4, each populated with 100 vectors of dimension $n = 100$. For the simulations, the similarity used is the standard cosine (equation

11). The definition of the topics is not directly used in the algorithms, it is only used to compute the values of performance criteria.

$$\forall A, B \in \mathbb{R}^n \ , \ s(A,B) = \frac{\sum_{i=1}^{n} a_i * b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{11}$$

The average similarity between two elements from two distinct classes is, at maximum, of 0.25 and, at minimum, of 0.08. The average similarity for two news items from the same class is 0.74. Ant's λ parameter is set to 0.7 in order to have high probability to correctly recognize elements of a same class.

Each peer is only allowed to connect to 4 other peers. Initially, no connection is established and in their directory peers have the address of a unique randomly chosen peer. To simulate the presence of a user, each peer as an agent periodically sends news items related to the peer's center of interest. News items are grabbed from a global repository ensuring no same news is send twice by two different peers. For the simulations, the generation period was set to $\mathcal{U}[100, 1500]$ units of time and the amount of news produced was limited to 10 items.

The performance of the diffusion algorithm is evaluated through 3 estimators: the network clustering coefficient γ and averaged values for completeness and efficiency factors. Network clustering coefficient γ is the average of clustering coefficient for each peer as defined in equation 1. Completeness defines the number of interesting news item a peer gathered compared to the total amount of interesting news items available in the network. Precision is the ratio between the number of interesting news divided by the total number of news fetched. Those definitions are similar to classical recall and precision but, in the our context, does not have the same meaning.

5 Results

Because of space constraints, the results presented here deal with the adjustment of β. The objective is to test when it is worth considering a peer being inefficient.

5.1 First test

In this first test, the network is made of 20 peers interested by one of the four subject previously defined. Five peers are assigned to each of the topics of the dataset. They are supposed to be only interested in the topic they were assigned to. Figure 1 shows the evolution of clustering coefficient for different values of β.

The best result is obtained with $\beta = 0.2$. Higher values of β lead to drop many connections, therefore almost no cluster can be formed. On the other hand, if β is set to lower values, fewer connections may be dropped and the network becomes static. The figure 2 shows how information dissemination evolves during the simulation. Worsts results are for

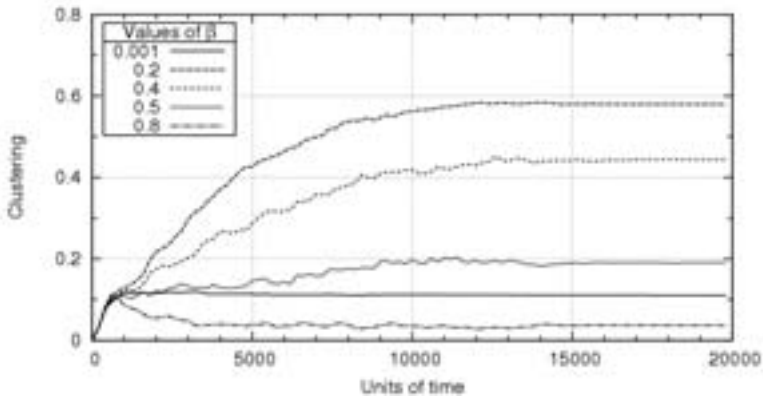
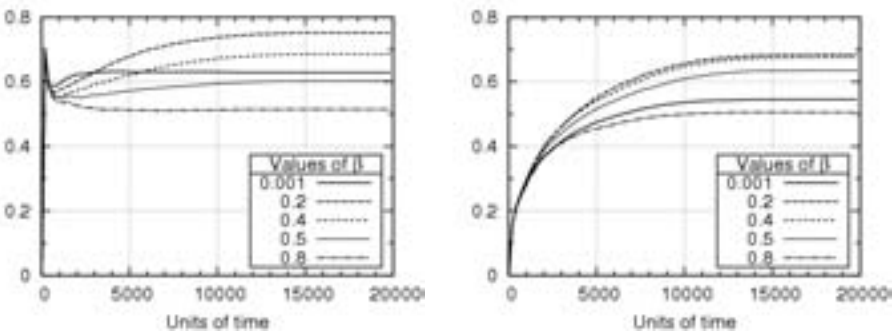


Figure 1: Evolution of clustering coefficient for various values of β

$\beta = 0.8$, when the network can not be clustered. For 0.2 and 0.4, completeness is similar but precision differs. This shows that a peer will get the same amount of interesting news in the two cases but if $\beta = 0.4$ it will also get more less interesting news items.



(a) Precision (b) Completeness

Figure 2: Evolution of completeness and precision for various values of β

Considering the small amount of news produced and the frequency of publication, the initialization phase does not last longer than 15000 units of time. Hence, the figures depict the stabilization of the system when no more news item are injected. The system is tested with 2 ants by peer, each having a patience k of 2. Best results are achieved when $\overline{PV_{i\to j}(t)} > 4 \cdot PV_{i\to j}(t)$, that is when the two ants had evaluated twice a connection has being useless. This tends to prove that the influence β is related to k and the number of ants.

5.2 Stability of result

In order to check the stability of this result, we have performed other tests using a different dataset and networks of different sizes. This second test involve 4 networks of 20, 25, 40 and 100 peers interested only in one subject among 4 or 5 available topics. Figure 3 shows the final, stabilized, value of clustering coefficient. For each network, this final value has been averaged over 50 runs of the algorithm.

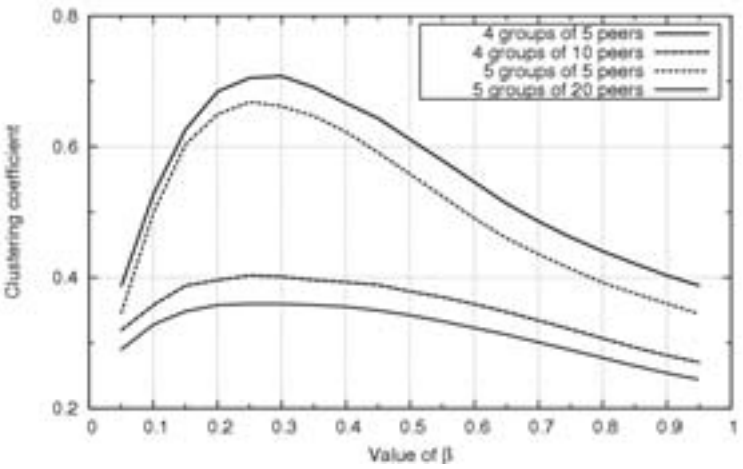


Figure 3: Evolution of clustering coefficient for various values of β

The first interesting result is that, independently from the size of the network, a maximum value for the clustering coefficient is reached when $\beta = 0.25$. That confirms the conclusion of the first presented test. An other constant result is the profile of the curves drawn which proves that the behavior of the rewiring algorithm is not influenced by the size of the network.

6 Conclusion and future directions

In this paper, we have presented an algorithm using artificial ants to diffuse information in a P2P network. The information diffusion is proactive and transparent for the user. Our main contribution is the use of estimated profiles in order to perform probabilistic broadcast. We use the artificial ants paradigm where artificial pheromones defines a memory for information exchanged. Firsts tests has led to an earlier version of the algorithm and proved the interest of using estimated profiles [GMS05, GMS06]. In this paper, we highlight a relation between the number of ants, their patience and the tolerance in estimating the utility of neighbors. During future development, tests on larger datasets along with a theoretical study will be performed in order to confirm this tendency. Also, we consider developing third party applications mandatory to replace the agent creating news by a real

user and perform real life tests.

References

[BDT99] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, 1999.

[BMM02] Ozalp Babaoglu, Hein Meling, and Alberto Montresor. Anthill: A Framework for the Development of Agent-Based Peer-to-Peer Systems. In *Proceedings of the 22th International Conference on Distributed Computing Systems (ICDCS '02)*, Vienna, Austria, July 2002.

[BSSZ01] P. Busetta, L. Serafini, D. Singh, and F. Zini. Extending Multi-Agent Cooperation by Overhearing. Technical Report 0101-01, Istituo Trentino di Cultura, January 2001.

[CAPMN03] F. M. Cuenca-Acuna, C. Peery, R. P. Martin, and T. D. Nguyen. PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. In *Proceedings of the 12th International Symposium on High Performance Distributed Computing (HPDC)*, June 2003.

[DDG05] Frederick Ducatelle, Gianni Di Caro, and Luca Maria Gambardella. Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks. *International Journal of Computational Intelligence and Applications (IJCIA)*, 5(2):169–184, 2005. Special Issue on Nature-Inspired Approaches to Networks and Telecommunications.

[DGH+88] Alan J. Demers, Daniel H. Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard E. Sturgis, Daniel C. Swinehart, and Douglas B. Terry. Epidemic Algorithms for Replicated Database Maintenance. *Operating Systems Review*, 22(1):8–32, 1988.

[DQA04] Anwitaman Datta, Silvia Quarteroni, and Karl Aberer. Autonomous Gossiping: A Self-Organizing Epidemic Algorithm for Selective Information Dissemination in Wireless Mobile Ad-Hoc Networks. *LNCS*, 3226:126–143, 2004.

[FZ98] M.J. Franklin and S.B. Zdonik. "Data in Your Face" : Push Technology in Perspective. In *Proceedings ACM SIGMOD International Conference on Management of Data*, pages 516–519, 1998.

[GMS05] C. Guéret, N. Monmarché, and M. Slimane. Aide à la navigation sur Internet : utilisation de fourmis artificielles pour l'échange d'informations dans un réseau P2P. In *Actes du congré de la ROADEF*, 2005.

[GMS06] Christophe Guéret, Nicolas Monmarché, and Mohamed Slimane. Sharing Resources with Artificial Ants. In *Proceedings of the 9th International Workshop on Nature Inspired Distributed Computing (NIDISC'06)*, Rhodes Island, Greece, April, 25-29 2006. 8 pages CD-ROM.

[Gru88] Jonathan Grudin. Why CSCW applications fail: problems in the design and evaluation of organization of organizational interfaces. In *Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 85–93, Portland, Oregon, United States, 1988.

[HS04] Peter Haase and Ronny Siebes. Peer selection in peer-to-peer networks with semantic topologies. In *Proceedings of the 13th International World Wide Web Conference*, New York City, NY, USA, 2004.

[IMK+01] IonStoica, Robert Morris, David Karger, M. Frans Kaashoe, and Hari Balakrishnan. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. In *SIGCOMM'01*, August 27-31 2001.

[KTID03] Manolis Koubarakis, Christos Tryfonopoulos, Stratos Idreos, and Yannis Drougas. Selective Information Dissemination in P2P Networks: Problems and Solutions. *SIGMOD Record, Special Issue on Peer-to-Peer Data Management*, 32(3):71–76, 2003.

[LM00] Meng-Jang Lin and Keith Marzullo. Directional Gossip: Gossip in a Wide Area Network. In *Proceedings of European Dependable Computing Conference*, 2000.

[Nap03] Napster. Napster file sharing. `http://www.napster.com`, 2003.

[Pay98] David W. Payton. Discovering Collaborators by Analyzing Trails Through an Information Space. In *AAAI Fall Symposium on Artificial Intelligence and Link Analysis*, October 23-25 1998.

[PRR97] C. Greg Plaxton, Rajmohan Rajaraman, and Andrea W. Richa. Accessing Nearby Copies of Replicated Objects in a Distributed Environment. In *Proceedings of ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, June 1997.

[Sch04] Christoph Schmitz. Self-Organization of a Small World by Topic. In *First International Workshop on Peer-to-Peer Knowledge Management (P2PKM)*, August 2004.

[Var02] András Varga. OMNeT++ discrete event simulation environment. `www.omnetpp.org`, 2002.

[VJvS03] Spyros Voulgaris, Mark Jelasity, and Maarten van Steen. A Robust and Scalable Peer-to-Peer Gossiping Protocol. In *Proceedings of the 2nd International Workshop on Agents and Peer-to-Peer Computing (AP2PC03)*, Melbourne, Australia, 2003.

[WGB99] M. Weiser, R. Gold, and J. S. Brown. The origins of ubiquitous computing research at PARC since the late 1980s. *IBM Systems Journal*, 38(4):693–696, 1999.

[WS98] Duncan J. Watt and H. Strogatz Steve. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 4 June 1998.

A JXTA-based System for Adaptive and Collaborative Learning

Hyosook Jung, Jinhyun Ahn, Seongbin Park*

Department of Computer Science Education
Korea University, Seoul, Korea
{est0718, budongsim, psb}@comedu.korea.ac.kr

Abstract: In this paper, we present an adaptive and collaborative learning system that is based on JXTA technology [jx205, Gon01]. JXTA enables any connected devices in a network to participate in communication and collaboration as peers and our system supports additional functionalities such as adaptive navigation, adaptive group formation, and instant messaging. We tested our system against undergraduate students who took a data structure class. The experimental results indicate that our JXTA-based learning system helped students learn effectively.

1 Introduction

In this paper, we present a JXTA-based learning system which supports various types of adaptive features by utilizing user models that reflect user behaviors during learning. The system also provides collaborative functionalities such as instant messaging that allow users to communicate and collaborate with their peers in real time [Tra05, Mah04]. In order to complement the weak points of collaborative learning in P2P networks such as a free rider [BV03] and differences between learners, it also supports adaptive functionalities such as adaptive navigation and dynamic group formation depending on the *learning context* of a student, where a learning context refers to various properties about the student such as whether the student has sufficient knowledge on the problems that are to be learned or not.

This paper is organized as follows. In section 2, related works are described. In section 3, the adaptive and collaborative learning system is explained. In section 4, we describe the experimental results in detail and the paper concludes in section 5.

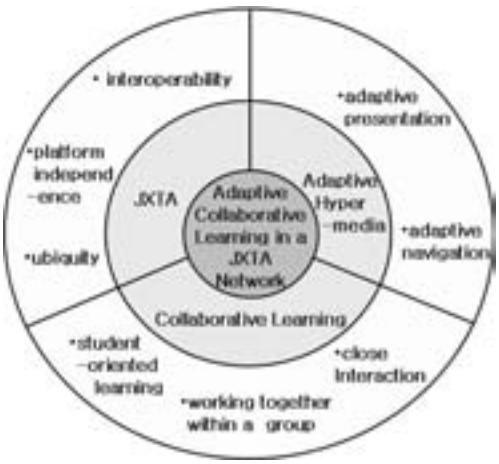


Figure 1: The relationships between our work and related areas

2 Related works

There are three areas that are related to our work and figure 1 depicts the relationship between our work and related fields.

Collaborative learning is an education method that motivates students, promotes critical thinking, and develops social behavior for working together. In order to produce good outcomes, the learning method must be well prepared [Bis05].

JXTA technology allows any connected device on a network, ranging from cell phones and wireless PDAs to PCs and servers, to communicate and collaborate. JXTA peers create and join a virtual network where they can interact with their peers and resources directly [jx205, Gon01].

While general P2P systems such as Napster and Gnutella can be used for collaborative learning, these do not have enough tools for small groups to interact strongly [EL04]. EDUCOSM [MNF+03] focuses on the collaborative learning in Web-based courses and supports a shared document pool to collect Web resources, collaborative annotation of the documents, and publication of the student's own work. COMTELLA [Vas04] is a P2P system that can be used for sharing files, mainly papers, and retrieving files among those that are shared by their colleagues. EDUTELLA [NWQ+02] is an educational P2P service for exchanging educational resources in the W3C metadata standard RDF. GROOVE [EL04] is a P2P software designed to facilitate collaboration and communication among small groups. It focuses on the shared workspace that users create and provides many tools that can be used in the workspaces such as calendar, discussion, file sharing, outliner, pictures, notepad, sketchpad, Web browser, etc.

*To whom correspondence should be addressed.

In a complex hyperspace, learners may experience the problems of disorientation and cognitive overload [Con87]. For overcoming these problems, adaptive hypermedia presents personalized contents and link structures based on a user model. The adaptation is described as adaptive presentation and adaptive navigation. Adaptive presentation provides information on a topic in different methods according to the user's knowledge, goals, preferences and so on. Adaptive navigation changes the link structure so that a user is guided toward interesting or relevant information [BHW99]. AHA! is one of adaptive hypermedia systems which supports adaptive presentation by the conditional inclusion of fragments and adaptive navigation by the link annotation or hiding [BAB+03].

3 JXTA-based learning system

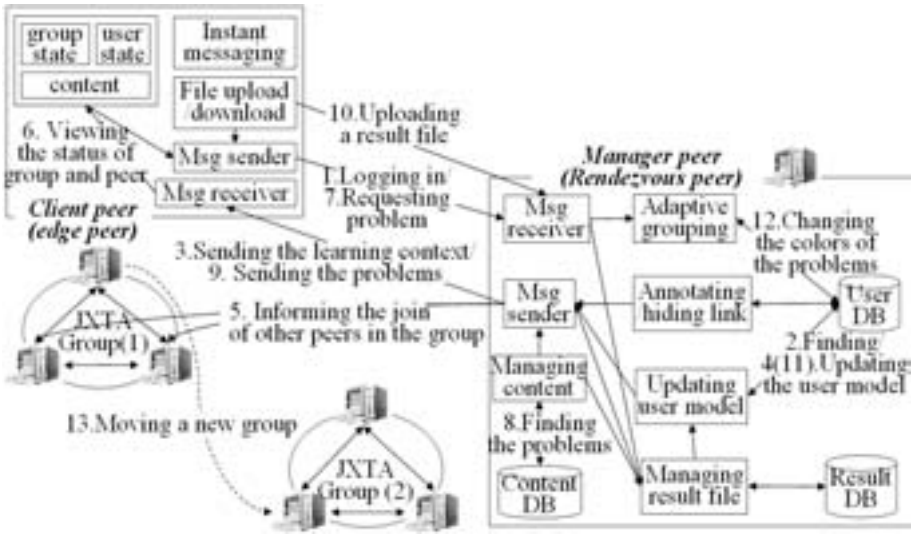


Figure 2: The architecture of the JXTA-based adaptive learning system

Figure 2 shows the architecture of our system and a typical usage scenario is as follows:

1. A user runs a student program at a client peer, joins in a JXTA network, and logs in the manager peer.
2. The updating module in the manager peer finds user's attributes and their values in the user database.
3. The manager peer sends both user's learning context (i.e., low-level or high-level knowledge) and the information about user's group (i.e., basic or superior group) to the client peer.

4. The updating module updates the user model of the client peer whenever the student solves a problem.
5. The manager peer informs all peers in a peer group that a new client peer participates in the group.
6. The client peer can see connected members in its own group, their learning contexts, and a list of learning problems.
7. The client peer requests a problem by clicking a link anchor in order to learn.
8. The managing content module in the manager peer finds the requested problem in the content database.
9. The manager peer sends the requested problem to the client peer.
10. The client peer uploads a result file to the manager peer after the user solves the problem.
11. The updating module in the manger peer updates the knowledge of the student and the level of the group.
12. The annotating/hiding link module changes the colors of link anchors of the problems depending on the updated user model.
13. The adaptive grouping module in the manager peer switches the existing group to a new group.

Table 1 shows the types of adaptative functions that our system provides. The attributes and adaptation rules are defined according to how often values of attributes are updated. The more dynamic attributes are updated frequently while the less dynamic attributes are updated infrequently.

Figure 3 shows a student program. A student logs in the manager program with ID and password. The student can see a list of problems that are selected based on the user model of the student. The learning context is updated dynamically according to the knowledge of the student that is changed whenever the student submits a result file. The titles of the problems (i.e. link anchor) are colored based on the learning context of the student such as not enough (black), available (blue), or completed (purple). The colors guide the student to navigate the content adaptively (i.e., link annotation).

The student can also see the information of the assigned group such as the number of connected peers, their status (i.e., name, major, and programming skill), or shared files. The student can exchange messages and share files with the members in the group. (See figure 4.) These behaviors are informed to the manager program to update the user model and can be used to support dynamic adaptation such as rearranging the groups and giving appropriate problems.

Figure 5 shows a manager program. It stores the user models of all students in a user database and the contents to be learned in a content database. It also stores the submitted

	attribute	adaptation rule
more dynamic	knowledge level of a student	System increases the knowledge level when a student submits a solution
	group index for a student	System changes a current group into a new one which is appropriate for the new knowledge level when the knowledge level increases
	the number of file uploadings	Students should upload their files at least once to download other files
	the color of each problem	System presents the titles of the problems in different colors depending on whether the problem is solved or not
less dynamic	difficulty level of a problem	Author defines the difficulty level of all problems
	personal features (name, password, grade, major, programming skill, etc.)	When a students updates the values of attributes, the values are changed
	a list of problems for a group	The difficulty level of the problem is the same as the level of the group

Table 1: The type of adaptation

result file in a result database. The user model is a file that contains the attributes and their values such as the latest login date, the number of login times, the score of each problem, the time to solve a problem, the type of the assigned group as well as the personal information (i.e., ID, password, programming skill, major). The attributes are used to determine which problems to provide for the student.

Each student can belong to exactly one group and the manager program assigns a student to a certain group that is suited to the knowledge of the student. It always monitors the behaviors of the student and updates the user model of the student. It rearranges the student to another group and provides appropriate problems based on the updated user model.

4 Experimental results

We experimented our system against thirty eight undergraduate students at the department of computer science education of Korea university. The students were skilled in C language and had been taking a Java language class that consisted of lectures in a classroom and practices in a laboratory. In our experiment, a professor gave a lecture about the concept of recursion and students solved three simple problems about recursion as a pretest.

We divided the students into four groups as follows.

- Group A : it is a control group and students learn individually.

Figure 3: A screenshot of a student program

- Group B : students learn individually and use an adaptive navigation function which guides learning process using three colors such as black (not enough), blue (available), and purple (completed). It represents student's current learning context.
- Group C : students learn collaboratively and use collaborative functions such as instant messaging and file sharing in a static group which does not change.
- Group D : students learn collaboratively in a group which changes depending on their current learning contexts. (i.e., dynamic grouping). In addition, it supports a ban of free-riding (i.e., students must upload their files at least once in order to share the results with others) as well as adaptive navigation and collaborative functions.

	adaptive function	collaborative function	learning type
Group A	X	X	individual learning
Group B	adaptive navigation	X	individual learning
Group C	X	instant messaging, file sharing in a static group	collaborative learning
Group D	adaptive navigation learning and grouping	instant messaging, file sharing, and a ban of free-riding in a dynamic group	collaborative learning

Table 2: The differences of function and learning style between four groups

Table 2 outlines the differences between four groups defined by three properties such as adaptive and collaborative function and learning style for the evaluation of our system.

Figure 4: A student solves problem 3 by collaborating with another student in the same group.

Table 3 outlines the differences of learning activity between three subject groups defined by three properties for the evaluation of our system. Students in the control group (i.e., group A) read an examination paper, learned individually, and wrote the solutions of the problems on the paper. Students in the subject groups used their own systems; they joined a JXTA network and solved the problems individually or collaboratively, using their JXTA applications. Students solved three intermediate level problems and two difficult level problems as a posttest.

To evaluate the efficiency of each system, we measured average scores of the pretest and posttest for each group after grading the final results. Table 4 shows the results for the pretest and posttest of each group. There was score improvement at the subject groups (Group B, C, D), but not at the control group (Group A).

From the experimental results, we conclude that the adaptive navigation and collaborative

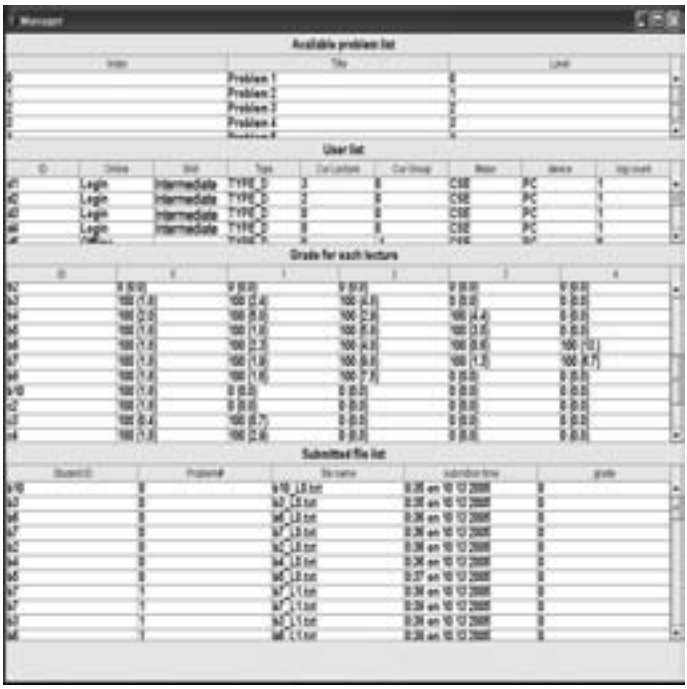


Figure 5: A screenshot of the manager program.

functions could have a good effect on learning. However, unlike our prediction that the students in group D would achieve better improvement of score than other groups because both adaptive navigation and collaborative functions were provided, group C achieved better scores than group D where only collaborative functions were supported. Our interpretation of this result is as follows. Thirty eight students took part in the experiment and they were randomly assigned to one of four groups. Twelve students in group D were too small to form different level groups by dynamic grouping. Therefore, it was difficult to communicate between the students actively during relatively short period of time. On the other hand, students in group C could communicate with their group members steadily since the group did not change. So with only collaborative functions, group C would be able to obtain better achievement than group D.

5 Conclusions and Future Works

In this paper, we presented a JXTA-based adaptive learning system. The system provides functionalities such as dynamic group formation, adaptive navigation, and instant messaging. We tested our system against undergraduate students for a data structure class and the experimental results were promising. We are currently extending the system so

Step	Group B	Group C	Group D
1	registration	registration	registration
2		join a static group which does not change	join a dynamic group which changes according to student's knowledge level
3	use an adaptive navigation function which changes the title color of a problem according to student's knowledge level	upload the result of the problem	use an adaptive navigation function which changes the title color of a problem according to student's knowledge level, upload the result of the problem
4		download member's results of the group, discuss if the solutions are appropriate for the problems	use adaptive grouping according to their knowledge level, download member's results of the group when uploading their own results, discuss if the solutions are appropriate for the problems
5	submission of the results	submission of the results	submission of the results

Table 3: The differences of learning activity between three subject groups.

	pretest	posttest
Group A	95	90
Group B	80	85
Group C	83	93
Group D	83	90

Table 4: The average scores for the pretest and posttest of each group

that learners can use handheld devices such as PDAs as well as desktop computers in the JXTA network. We also plan to implement more adaptive functionalities (either, *finer*, or *coarser*, or both) than are provided currently.

References

[BAB+03] P. De Bra, A. Aerts, B. Berden, B. De Lange, B. Rousseau, and T. Santic. AHA! The Adaptive Hypermedia Architecture. In *Proceedings of the ACM Hypertext Conference*, UK, August 2003.

[BHW99] P. De Bra, G.J. Houben, and H. Wu. AHAM: a Dexter-based reference model for adaptive hypermedia. In *HYPERTEXT '99: Proceedings of the 10th ACM Conference on Hypertext and hypermedia ACM*, pages 147–156, Darmstadt, Germany, 1999.

[Bis05] J. Bistrom. Peer to Peer Networks as Collaborative Learning Environments. HUT T-110.551 Seminar on Internetworking, April 2005.

[BV03] H. Bretzke and J. Vassileva. Motivating Cooperation on Peer to Peer Networks. *Lecture Notes in Computer Science, Springer-Verlag GmbH*, 2702:218–227, August 2003.

[Con87] J. Conklin. A Survey of Hypertext. *IEEE Computer*, 20(9):17–41, September 1987.

[EL04] C. Eikemeier and U. Lechner. Peer-to-Peer and Group Collaboration - Do They Always Match? In *Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE04)*, Modena Italy, June 2004. IEEE Computer Society Press.

[Gon01] L. Gong. JXTA : A Network Programming Environment. *IEEE Internet Computing Industry Report*, pages 88–95, 2001.

[jx205] JXTA v2.3.x: JavaTM Programmer's Guide. Sun Microsystems, 2005.

[Mah04] R. Mahler. Instant Messaging/Presence: Next-Generation Implementations and Applications. JavaOne Online Technical Sessions, TS-2369, Sun Microsystems, Inc., 2004.

[MNF+03] M. Miettinen, P. Nokelainen, P. Floreen, H. Tirri, and J. Kurhila. EDUCOSM - Personalized Writable Web for Learning Communities. In *International Conference on Information Technology: Computers and Communications*, Las Vegas, California, USA, April 2003. IEEE Computer Society Press.

[NWQ+02] W. Nejdl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmjorn, and T. Risch. EDUTELLA: a P2P networking infrastructure based on RDF. In *Proceedings of the 11th international conference on World Wide Web, SESSION: Query Language for Semantic Web*, pages 604–615, Honolulu Hawaii, May 2002.

[Tra05] B. Traversat. JXTA Technology Beyond File Sharing: P2P Grows Up. JavaOne Online Technical Sessions, TS-7208, Sun Microsystems, Inc., 2005.

[Vas04] J. Vassileva. Harnessing P2P Power in the Classroom. *Lecture Notes in Computer Science, Springer-Verlag GmbH*, 3220:305–314, August 2004.

A Homogeneous Service Framework for Pervasive Computing Environments

Sergio Maffioletti, Simon Schubiger, Michèle Courant, Béat Hirsbrunner

Department of Informatics
University of Fribourg
Chemin du Musée 3
1700 Fribourg, Switzerland

Abstract: This paper introduces a model addressing the heterogeneity problem of Pervasive Computing systems from a resource, service and context management viewpoint. The presented system architecture, called UBIDEV is a collection of abstraction services for semantic-driven management of the physical environment that provide a design methodology that allow a description of the behavior of the whole system in terms of coordinated homogeneous services. Most of the existing approaches try to hide heterogeneity using a single uniform abstraction layer like the Java VM; these models do not fit the requirements of pervasive computing systems where the dynamism and the heterogeneity if the environment need to be taken into account even at the design level. The presented model faces heterogeneity of pervasive computing systems allowing applications to be described in terms of services provided rather than their low level instantiation details. The main contribution of UBIDEV is in the holistic approach in the management of the environment from the resources, services and context viewpoint. The resulting coordination model allows applications to be described in terms of their functionality while maintaining the degree of dependence they have with the physical environment. At the application level, the provided abstractions allow to build applications that were previously seen as difficult to build: context-awareness that scale along several dimensions, resource and service management that copes with heterogeneity using an agreed semantic, holistic coordination of resources in a service-oriented abstraction model. An example application scenario is then described to underline the approach and the added value of such architecture in terms of system design and resource and service management.

1 Introduction

Heterogeneity in computing systems is not meant to disappear in the future, but instead will increase as the range of computing devices increase. Requirements for a Pervasive Computing infrastructure are centered on a high-level conceptual model consisting of resources, users, context, services, coordination models and applications level interfaces [Nor99].

Resource is one of the key aspects in adaptability because they represent the endpoint used by the application to provide its functionality. Role of the infrastructure is to provide to the application an easy way to describe the adaptation patterns without having to deal

directly with resource management. That also means take into account the role and even the intention that some resources may have in a given environment: take the example of an autonomous mobile robot; when it enters a new environment it should be able to negotiate with the application the services to use as well as provide information about its goal. That implies the infrastructure should be able to manage and describe such resource to the application in a way that is compliant with the application's knowledge.

Context represents the main shift from classical distributes systems because in Pervasive Computing the surrounding physical environment is explicitly taken into account by both the infrastructure and the application to adapt the behavior of the whole system to the caching that occur during the lifecycle of the system.

Services vary greatly as well: from home/office printer access, to local driving directions, to global services such as search engines and web access. Services tend to relay on a given hardware configuration for their execution; they have resource requirements that should be met to ensure their correct execution. Most of the time the coupling between software components and the hardware involved is so thigh that the notion of a service embodies the two. The infrastructure should allow to describe the fundamental interrelation between software components and resources while keeping their coupling in terms of functional dependencies. That should also be reflected at application design level.

This paper presents UBIDEV, a service framework aimed to tackle these three levels of heterogeneity that characterize Pervasive Computing systems. UBIDEV relays on classification and encapsulation techniques for semantic-driven resource, service and context management providing at application level a homogeneous coordination space where interactive entities represents running services in a service-oriented approach.

The rest of the paper is organized as follow: Section 1 introduces the heterogeneity problem in Pervasive Computing systems from a resource, service and context perspective; it also describes the requirements from an infrastructure to cope with this heterogeneity. Section 2 presents the UBIDEV model underlying the role of the classification and encapsulation techniques as well as their related modules such as application ontology, classifiers, capsule, resources, service and context managers describing how the provided abstractions contributes to present at application level an homogeneous coordination space. Section 3 discusses some related projects and their different approaches in facing similar problems. Finally some concluding remarks are given in section 4.

2 Heterogeneity in Pervasive Computing

2.1 Resources

Resource heterogeneity implies differences in shape, capabilities, power and usability; an infrastructure must be able to recognize such diversities in order to adapt the services it provides and the services that it controls on behalf of an application. Heterogeneous devices are required to interact seamlessly, despite wide differences in hardware and software capabilities; this requires an infrastructure that maintains knowledge of device character-

istics and manages the integration of devices into a coherent system that enables arbitrary device interactions (for example, between a mobile phone and a desktop workstation).

There are three elements that could be associate with resource heterogeneity:

- **Physical**: For a given cost and level of technology, weight, power, size and ergonomics represent a limitation with respect of computational resources such as processor speed, memory size, and disk capacity.
- **Communication**: Some buildings may offer reliable, high-bandwidth wireless connectivity while others may only offer low-bandwidth connectivity. Over time, the synchronous model implicit in the use of RPC will become inadequate. What is required is a reliable transport layer that works with legacy servers, while hiding the effects of wireless losses and asymmetry that typically ruin TCP performance. Eventually, very wide-area distributed systems will have to be structured around an asynchronous model.
- **Power**: While battery and energy production technology will undoubtedly improve over time, the need to be sensitive to power consumption will not diminish. Concern for power consumption must span many levels of hardware and software to be fully effective.

Resource heterogeneity increases the tension between autonomy and interdependence that is characteristic of all distributed systems. To cope with this tension means to introduce a certain level of adaptability into the system. At one extreme, adaptation is entirely the responsibility of individual applications. While this approach avoids the need for system support, it lacks a central arbitrator to resolve incompatible resource demands of different applications and to enforce limits on resource usage.

The other extreme of application-transparent adaptation places entire responsibility for adaptation on the system. This approach is attractive because it is backward compatible with existing applications. The system provides the focal point for resource arbitration and control. The drawback of this approach is that there may be situations where the adaptation performed by the system is inadequate or even counterproductive because it operates without taking into account the application's perspective.

2.2 Services

Service discovery allows clients to locate services in the face of mobility and heterogeneity. Existing service discovery architectures have a few limitations making them unsuitable for wide deployment in the Pervasive Computing domain.

Some of these protocols like SLP [Gut00] and Salutation [Inc99] are deployed primarily within the enterprise or office environment; others like UPnP [Cor05] and Bluetooth [AJF02] were conceived for a more informal, casually connected environment, which could include networked vehicles and small offices as well as home networks. A networking solution should be able to accommodate heterogeneity, both in terms of underlying

connectivity, and in terms of the discovery infrastructure that is supported. The infrastructure for pervasive computing must support diverse types of software component. It should be able to integrate software components, which may reside in fundamentally different environments (such as home or office computing environments), into compositions that can successfully interact and cooperate to achieve common tasks.

Services are heterogeneous in nature; they should be defined in terms of the their functionality and capabilities. The existing service discovery infrastructures lack expressive languages, representations and tools that are good at representing a broad range of service descriptions and are good for reasoning about the functionality and the capabilities of the services.

Services need to interact with clients and other services across environments. Service descriptions and information need to be understood and agreed among various parties. In other words, well-defined common ontology must be present before any effective service discovery process can take place. Common ontology infrastructures are often either missing from or are not well represented in the existing service discovery architectures. Architectures like Service Location Protocol, Jini [Jin03] and Salutation do provide some sort of mechanisms to capture ontology among services. However, these mechanisms like Java class interfaces and ad-hoc data structures are difficult to be widely adapted by the industries to become standards.

2.3 Context

Invisibility of applications is accomplished by reducing input from users and replacing it with knowledge of context. "A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user task" [Dey00]. Context-aware software components exploit information such as the activities in which the user is engaged, proximity to other devices and services, location, time of day and weather conditions. Knowledge of context also is required to enable adaptation to changing environmental conditions, such as changing bandwidth and input and output devices, which can be brought about by mobility.

The infrastructure for pervasive computing should support context awareness by facilitating the gathering of information from sources such as sensors and resource monitors; performing interpretation of data; carrying out dissemination of contextual information to interested parties in a scalable and timely fashion; and providing models for programming context-aware applications. A very challenging aspect is interpretation, which involves steps such as integration of data from different sources (for example, combining height and horizontal position into a three dimensional position); inference (for example, "Bob is in the meeting room and Alice is in the meeting room, therefore a meeting between Bob and Alice is taking place"); prediction based on context history; resolution of inconsistencies between context data from different sources; and provision of estimates of the accuracy of contextual information.

The importance of appropriate abstractions for gathering and reasoning about the context

information has led an ontology-based shift in the research focus of the context awareness community [KBM+02], [Coe98], [RHC+02]. Ontology-based approach may improve over classical context modeling approaches by providing improved support for interoperation and sophisticated type of reasoning.

2.4 The Role of an Infrastructure for Pervasive Computing Systems

A Pervasive Computing infrastructure should be highly available, cost effective, and sufficiently scalable to support millions of users. In general, computation, storage and complexity should be moved from the resources into the infrastructure, thus enabling powerful services, better overall cost performance, and small, light-weight, low-power, inexpensive mobile devices to increase functionality [BKA+98].

As a result, a Pervasive Computing Systems will be composed of different services and resources interacting with each other. A coordination model is required to formalize such interactions as well as the dependencies between the coordinated entities.

Adaptation:

is required in order to overcome the intrinsically dynamic nature of pervasive computing. Mobility of users, devices and software components can occur, leading to changes in the physical and virtual environments of these entities. Moreover, applications can be highly dynamic, with users requiring support for novel tasks and demanding the ability to change requirements on the fly. It should be the role of the infrastructure for pervasive computing to facilitate adaptation, which may involve adapting individual software components and/or reconfiguring bindings of components by adding, removing or substituting components. Adaptation may be done in an application-aware or application-transparent manner [Nob00] [GMGN04].

There are three common strategies for adaptation in Pervasive Computing: first, a client can guide applications in changing their behavior so that they use less of a scarce resource. This change usually reduces the user-perceived quality, or fidelity, of an application as in [FS99], [NSN+97]. Second, a client can ask the environment to guarantee a certain level of a resource. This is the approach typically used by reservation-based QoS systems [NCN98]. Third, a client can suggest a corrective action to the user. Smart spaces such as [JF04] and [Rek98] are examples of environments capable of accepting resource reservations.

Metacomputing Abstraction:

Metacomputing environments [CS92] such as [MS99] and [KUB00] are component-based: the heterogeneous computing environment is aggregated and a concurrent programming platform emulated through a set of coordinated components. Through the composition and the coordination of such components the heterogeneous environment is aggregated within

a concurrent programming platform.

Metacomputers hides the existence of multiple computers and provides a single-system image to its users through the use of a Distributed Virtual . Differently from a Network Operating System approach [Tv02] where a user is fully aware of the machines on which his job is executed, metacomputer dynamically and automatically allocates jobs to the machines of the system. The key concept behind these features is "transparency". Metacomputing, if conceived as a Distributed Operating System, supports several forms of transparency to achieve the goal of providing an abstraction of networked machines as a metacomputer. Harness [BDF+99] is a good example of a system conceived to centralize the management of the underlying resources providing a uniform abstraction to the applications and users. In a Pervasive Computing Systems, however, applications can greatly benefit from knowing some relevant functional details of the computational environment. That could allow them to configure themselves and adapt to every heterogeneous and dynamic aspect of an environment.

2.5 Ontology in Pervasive Computing

Most of the existing infrastructures and solutions have made progress in various aspects of pervasive computing but are still weak in supporting knowledge sharing and reasoning. A significant source of this weakness is their lack a common ontology with explicit semantic representation.

A key requirement for realizing Pervasive Computing systems is to give computer systems the ability to understand their situational conditions. To achieve this, it requires contextual information to be represented in ways that are adequate for machine processing and reasoning. At the same time resources and services needs to be described ad managed following a common semantic that should, also, be reflected at coordination level. The goal is to abstract the description of the application behavior from the management of the different instances that compose the environment.

Semantic Web languages are well suited for this purpose for the following reasons:

- Ontologies provide a means for independently developed systems to share resources, services and context knowledge,
- RDF and OWL are knowledge representation languages with rich expressive power that are adequate for modeling various types of contextual information, e.g., information associated with people, events, devices, places, time, and space.
- These knowledge representation languages are well suited to, also, describe resources and services.
- Because ontologies have explicit representations of semantics, they can be reasoned by the available logic inference engines. Systems with the ability to reason about context can detect and resolve inconsistent knowledge.

- The Semantic Web languages can be used as meta-languages to define other special purpose languages such as communication languages for knowledge sharing, policy languages for privacy and security [KFJ03]. A key advantage of this approach is better interoperability. Tools for languages that share a common root of constructs can better interoperate than tools for languages that have diverse roots of constructs.

A common agreed standard ontology could be the "panacea" for most of the interoperability and openness issues raised in distributed systems and Pervasive Computing. But this is only an idea scenario, quite difficult to realize given the current situation where lot of systems have developed their own communication protocols, description scheme and ontologies. Interoperability at such level is a very challenging and still open issue. Most of the approaches implies human intervention to solve, for example, ontologies mapping.

Despite the difficulties that the use of ontologies rise when considering interoperability between systems, it is a very efficient and interesting technology for describing a system as an isolate entity. All resources, services, contextual information as well as interaction and dependencies, when described with an agreed common semantic, could be better managed by the infrastructure resulting in a more clear separation between the application pure functional level and its specific system instantiation. This has been the approach that has inspired the UBIDEV model.

3 UBIDEV: Classification and Encapsulation

UBIDEV is a lightweight infrastructure built around the reference model presented in [MKH04], [MH02] and [SMTH00]: Physical Entities, Resources, Services, Context, Coordination and Application.

UBIDEV can then be considered as a unified management model for resources, services and context information. It proposes a context centric management of the environment: it is the application context that determines the semantic of the resources, of the services involved and of the contextual information. Consequently resource configuration, service instantiation, description and composition, context model as well as user task solving are based on this semantic. This allows Pervasive Computing applications to automatically reconfigure themselves according to context changes. As a result UBIDEV presents at application level a homogeneous coordination space seen as an unified mechanism for dynamic interaction of services [CA94] [AC93] as depicted in figure 1.

A key element in realizing this architecture is the use of an application ontology that undergrids the communication between entities and the representation of the environment. A generic knowledge representation system uses various terms with different domain specific definitions in order to describe the knowledge model. Instead of introducing its own semantic, UBIDEV identifies the internal representation of semantic and the relation to the environment as relations to context and resources. That leads to small topic-oriented ontology used to classify the whole environment in terms of resources, services and contextual information. An application can be described according to the conceptual model of the ontology that is independent of the specificity of the underlying environment. Once an

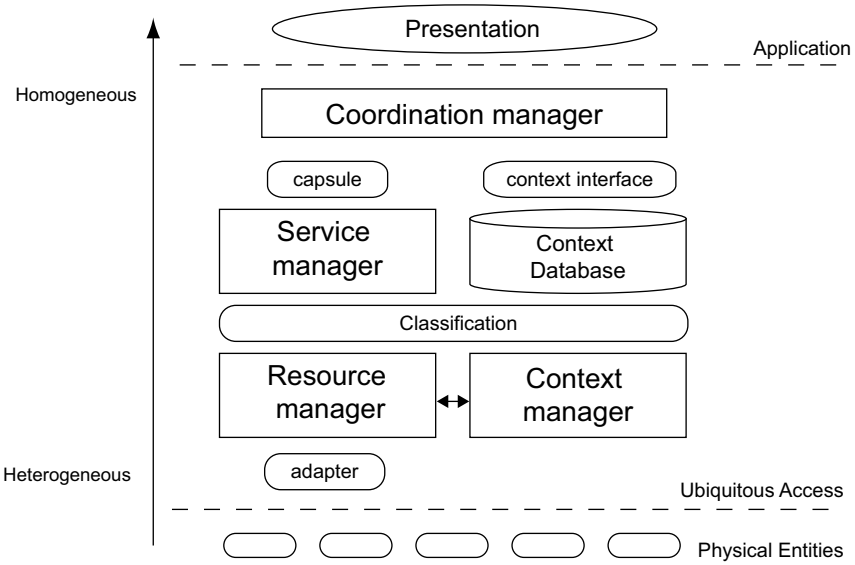


Figure 1: UBIDEV **implemented architecture.**

application is instantiated in a specific environment, UBIDEV can ensure that the resource, the contextual information and the services are described and managed according to this model, thus shielding the application to directly dealing with it.

Coordination Model:

Coordination manages the dependencies in the interaction process of the application-level user task solving. It manages the dependencies between services and contextual information when, at application level, a user task request is defined. These dependencies influence the service composition that is at the heart of the whole coordination process. To better describe these dependencies, we need to identify and separates the computation and the coordination aspects of a generic Pervasive Computing Systems.

Differently from classical implementations of coordination spaces, no explicit coordination is defined in UBIDEV; *coordination manager* reacts to the application level requests expressed in terms of user tasks. Such tasks are described as resource transformation. *coordination manager* solves user tasks by composing basic services according to the resource transformation request, the availability of basic services and according to the contextual and behavioral rules defined in the context model. The current implementation of the *coordination manager* uses a first order logic to represent the terminological knowledge of an application domain in a structured and formally well-understood way; more precisely it is used to describe behavioral and contextual rules as well as the application ontology. Such rules are expressed as Prolog facts that are evaluated (all or just a part of them) when solving the user task. First activity of the *coordination manager* is to deter-

mine, according to the configuration of the context and according to the user task, which are the rules that have to be evaluated. The result of the evaluation determines the run-time dependencies and constrains that *coordination manager* takes into account in the service composition process.

Thus, the resulting composite service that is in charge to accomplish the user task, is influenced by: the active part of the context where the request has been generated, the social rules defined for such context, the availability of resources and services, the social laws related to the context, services and resources. The core of *coordination manager* is based on XSB Prolog engine [SSW94]. *Coordination manager* can not be directly programmed but only instructed by a richer set of Prolog facts. By analyzing both contextual and behavioral rules, *coordination manager* has a full knowledge of the status of the environment and of its role.

The Coordination model allows defining rules for every context that may also restrain the algorithm of resource and service management. These rules are specified by the context itself; the structure of these rules are the key results of the XCM model [TCH04], to be integrated into the UBIDEV middleware.

3.1 Classification

The main problem related to the management of resources is the role that each resource may have in a given context. This is referred as the semantic and is reflected on all action that could be taken on a resource, typically description, discovery and access.

Classification is the operation of identifying the elementary symbols of the model by testing whether an entity belongs to a specific entity set or not. This approach that has already been considered in the COCA model and previously in the EMUDS [RCC98] project, is an attempt to face the symbol grounding problem [Har90] by defining instruments for identify the iconic representation of the real world in a specific environment.

UBIDEV relies on the COCA[Sch02] principle that is the application context that should determine the meaning of resources. In doing so it relies on the ontology stated by the application and by classifiers. Classifiers [Sch02] are services that given a resource and an ontology, output concepts of that ontology. This mechanism replaces the classical scheme where resources supply their description directly.

A classifier accesses resources through adapters, associates one or more concepts it knows with a resource in a given context, and tags the resource as an instance of that concept. Classifications of resources are stored and used as a cache when an instance of a concept is requested by services. The process of requesting an instance of a concept is called ”addressing by concept”, because the instance is referred by a concept instead of specific resource identification, such as memory address, or name. Thanks to the classification process, UBIDEV decouples the high-level concepts (abstractions) from the instances implemented by a context. The concept ”nearest printer” [Kam00] for instance may be used no matter how a context supplies the corresponding implementation. In such a way a user moving around different environments will not have to reconfigure her printing applica-

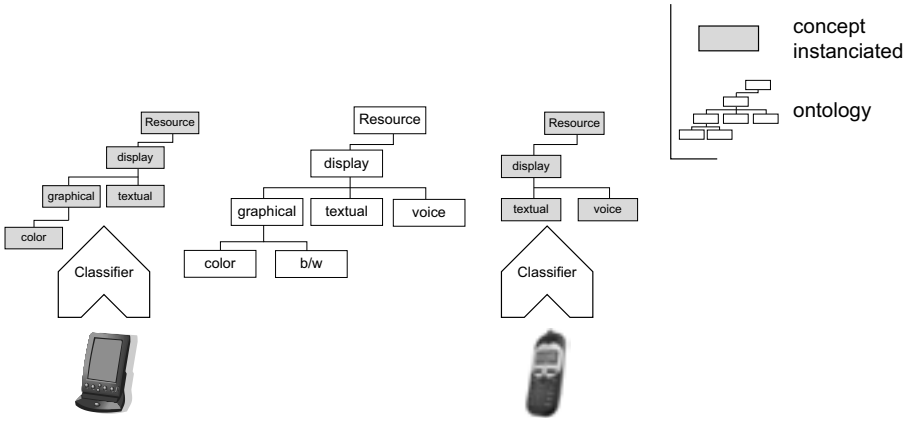


Figure 2: **UbiDev Classification Process**. *An example of classification of physical resources. Classifiers are seen as the knowledge required to identify the relation between the icons in the symbol system and the physical elements.*

tion. This means that an application may express its resource requirements in terms of concepts instead of addressing specific resources directly (i.e. by an URL).

Figure 2 shows an example of classification. According to the application ontology, a PDA and a portable phone are classified respectively as instances of concepts ”display”, ”textual”, ”graphical” and ”color” the former and ”display”, ”textual” and ”voice” the latter. A service that manipulates textual display will find these resources semantically equivalent; the service may access both of them transparently addressing the concept ”textual”. The related adapters will do the rest resulting for the PDA to receive the text as a new Memo note and, for the portable phone to receive the text as a SMS.

One way of defining a classifier for a resource is to wrap an existing classification scheme and embed it into the classifier class; in such a way the most part of resources could be classified without having to write any particular resource dependent analysis. For example it does not make much sense to redefine the semantics of the concept ”Wav document” when there are media players available that can easily decide if a file is in a wav format or something else. Thus building a classifier for the concept ”Wav document” means simply invoking a media player and looking at its exit code. This leads on the one hand to direct reuse of semantics and knowledge, encoded and available in computer programs today and on the other hand it helps constructing new semantic out of the existing pool of programs.

The modularity of the COCA specification allows deploying a pool of different classifiers that could be re-used in different applications. Similarly to the rest of the system, in fact, the level of standardization they are supposed to meet is at the ontology level. As an example, the classifiers used in the UMS prototype for document classification (ascii, pdf, ps, bmp, gif,), have been reused for another UBIDEV prototype called Document Classifiers [pai]; the adaptation of a classifier from one system to another one is on the mapping from the ontology of the former to the ontology of the latter. Another example of sharing ontology is brought by the need on inter-application communication; in both cases

the COCA model proposes three approaches [Sch02]:

- **Explicit declaration of concept equivalence**: Meta ontology may be used to explicitly state equivalence of concepts in different ontology. This meta-ontology will require careful maintenance, likely done by humans.
- **Proving concept equivalence**: If two concepts in different ontology are defined by the same classifier they are considered equivalent allowing inference of further properties through each ontology relations.
- **Inferring concept equivalence**: Concept equivalence may also be automatically inferred by observing classifiers by so-called meta-classifiers. That is, the automated version of building meta-ontology. Concepts may be considered equivalent as long as different classifiers consistently output the same concepts for a set of resources under observation.

3.2 Encapsulation

EXecution Environment:

We argue that operating systems and middleware for Pervasive Computing Systems must take into account the dependencies between software components as well as the dependencies between software and hardware components. Finding a suitable representation of such dependencies would allow implementing services that can configure themselves and adapt to every heterogeneity and dynamic environment.

To address the problem from the service management viewpoint, UBIDEV explores the service dependencies in terms of *EXecution Environment (EXE)*: requirements for loading a service into a runtime system. As long as UBIDEV knows the requirements for installing and running each service, it can automate the installation and configuration of new components. It can improve application performance by analyzing the dynamic state of system resources, analyzing the characteristics of each service module and configuring each of them in the most efficient and suitable way. It can also adapt the configuration policy to the contextual information the environment provides, resulting in a fully context-sensitive system. Requirements usually are expressed as dependencies on both persistent and dynamic resources.

Service's EXecution Environment (EXE) must specify any special requirement needed to load, configure and execute the service. It is included in the service description together with the input and output concepts. Even the EXE is expressed in terms of the application ontology since the semantic of resources, hence their capabilities, is captured in the classification phase. A service manager might use this information to determine where, how and when to execute the service.

The analysis of the inter-component dependencies, expressed in terms of relation among services and between services and the context, can help to automate and improve the

configuration process. UBIDEV can scan the EXecution Environment to ensure that all concepts required for the execution of a particular service are met before the service is instantiated.

This can also prevent many problems that are common in existing systems where detection of the lack of a particular resource happens only after a service is running.

Services and Capsules:

UBIDEV services are described using an XML-like proprietary description format. Further extension of the system will include the standardization of service description to WSDL [CCMW01]. That will also simplify the opening UBIDEV to web-services. Service description includes information about the service name, its interfaces in terms of input and output concepts and the prerequirements.

A UBIDEV service is encapsulated in a specific dedicated environment that fulfills its requirements according to the service's EXecution Environment. This environment contains all the resource access information the service requires for execution, expressed in the form of the UBIDEV access protocol [MKH04].

A run-time instance of a service inside its environment is represented by a homogeneous entity called *capsule*. A *capsule* is homogeneous in the sense that it hides to coordination all heterogeneous aspects related to the execution of the service it embodies. A *capsule* exports to the upper layer only the service interface in terms of input and output concepts. In this way a *capsule* represents a new organizational unit to encapsulate a service computing environment within the system architecture, just as a process is an organizational unit for the components of a running application [Tan97].

Service manager parses the service prerequirements in order to determine whether the current service is entitled to be instantiated or not. If so it issues a resource reservation request to *resource manager* that replies with the corresponding resource access reference. Resource access reference together with the service run-time module and an instance of the ubidev protocol handler are encapsulated into a new capsule.

UBIDEV services are able to act upon stateful resources providing access to, and manipulating a set of logical stateful resources based on messages sent and received. That means UBIDEV service executes against dynamic state, i.e., state for which the service is responsible between message exchanges with its requester. UBIDEV service is stateless because it delegates the control of its internal state as well as the state of the associated resources to the capsule. In this way service migration is made easier because the capsule allows dumping its complete state, and because the representation of the state does not change from service to service. However, the consistency of the use of the service state has to be ensured by the service implementation. Whenever a service prerequirement is no longer met, *service manager* stops the execution of the corresponding capsule and tries to reconfigure it. UBIDEV supports completely transparent capsule reconfiguration. After a reconfiguration process, a service inside a capsule continues to interact with its environment regardless of the changes. The capsule state is divided in two sub-states: the service state that can be migrated and the user state that is related to the model of the context and

may not be migrated. The service state contains the program code and a representation of the service data. The user state contains user dependent data that are organized according to the model of the context. For this reason it is not necessary for this state to migrate because the service can access it as a contextual information provided by its execution environment.

This approach is robust and efficient because the consistence of the mapping between concepts and resources is always granted by the infrastructure;; so services do not have to take into account resource specific information that are not classified according to the application ontology.

One way to obtain a service for the UBIDEV architecture is to wrap an existing application. For instance Emacs, Microsoft Word, PalmOS memo-pad and the QTopia text-editor can each be wrapped to become suppliers of a text editing service. Such wrappers map abstract service descriptions into application-specific settings.

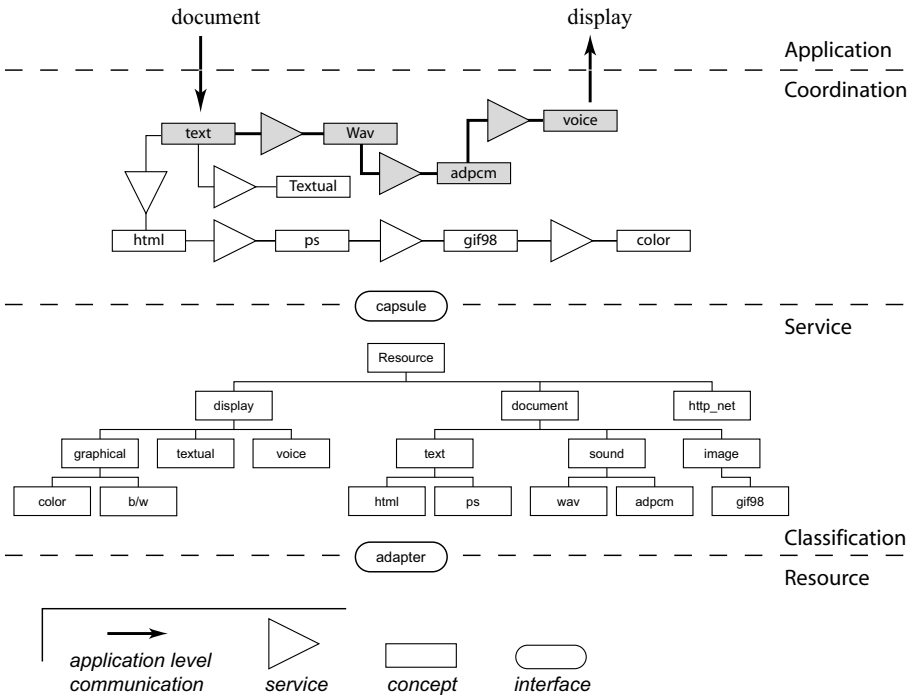


Figure 3: *A composite service can be realized by the coordination manager by composing existing services according to the classification of the underlying environment. Coordination manager solves user tasks by finding the path from the input resource generating the output resource. This path represents a service invocation chain. Service manager takes in charge the service execution process.*

The main advantage of facing the heterogeneity problem with the capsule abstraction is at coordination level, where the system is seen as an unified mechanism for dynamic communication, coordination, and sharing of homogeneous objects (in distributed systems, JavaSpace [Jav03] is a good representer of such a mechanism).

4 Related Works

4.1 Gaia

Gaia OS [RHC+02] is a meta-operating system that aims at supporting the development and execution of portable applications for active spaces. Gaia is a distributed middleware infrastructure that coordinates software entities and heterogeneous networked devices contained in a physical space. Gaia exports service to query and utilize existing resources, to access and use current context, and provide a framework to develop user-centric, resource-aware, multi-device, context-sensitive, and mobile applications. The system is built as a distributed object system. The Gaia Kernel, the Gaia Application Framework, and the Applications Active Space form the building blocks of the whole architecture. An Event Manager is a mechanism to expose changes of the environment through a publish subscriber model. The context infrastructure consists of a number of components, called context providers that provide information about the current context. Gaia implements a bootstrap protocol that interprets a configuration file (Lua script) and starts the kernel services. The configuration file contains information about the Gaia Kernel services, such as the name of the service, the name of the interface of the service, the Gaia node or nodes that will host the service, the service instantiation policy (i.e., instantiate the service in all Gaia nodes or only in the first available Gaia node), and start parameters. This approach works fine when the topology of the active space is well known a priori. Fixed categorization of resources implies that the relation between hardware and software level is, in a sort of way, pre-defined. That makes reuse of existing resources as well as the introduction of new ones a difficult task. Gaia makes use of ontology at different levels [MRMC03], as for the classification of the context and the semantic service discovery. UBIDEV shares the same philosophy in the use of ontology to augment the action and reaction of a Pervasive Computing system. Differently from Gaia, UBIDEV relies on the assumption that is the application that determines how a specific environment should be configured and managed in terms of resources, services and context.

4.2 Aura

The Aura project [SG02] is about distraction-free pervasive computing. It supports mobile users inside a computing environment by maximizing the use of available resources inside the environment and by minimizing the user distraction and focus on user attention. Its goal is to provide each user with an invisible halo of computing and information services that persists regardless the location. It defines the notion of personal aura that can be considered as a service proxy for the mobile user it represents. Aura proposes a programming model for task-based computing [WG00]. In this model, tasks are viewed as compositions of services. Both tasks and services have explicit representations. Services are described by virtual service types, which define functional, state and configuration interfaces and dependencies upon other services. Virtual service types can be related through inheritance, and can also be composed to form new virtual services. Tasks are toplevel compo-

sitions of services that are specified as flows that decompose tasks into steps of subtasks or primitives (actions carried out by services). Tasks are instantiated by a protocol that is responsible for gathering information about available services, selecting suitable services to carry out tasks and binding them together, and, finally, performing configuration and initialization of services. A coordination protocol manages the plugging and unplugging of services in response to resource changes. Tasks are also managed by a third protocol responsible for task migration, obtaining consistent snapshots of task state, and managing replication and consistency. Aura shares lot of similarities with the UBIDEV project; both of them, in fact, take a holistic approach in the model of the environment. The definition of abstract services and the idea of mapping them into concrete instances are similar to the service classification model in UBIDEV. Also the Task Manager, as the coordination manager in UBIDEV, acts as a control unit for resources, services and context. Differently from UBIDEV, abstract services are mapped into concrete instances by analyzing service description. In Aura the whole process relies on the assumption that suppliers of a given service type share a vocabulary of tags and the corresponding interpretation. Aura however lacks in defining at Prism level a proper model of the context. In UBIDEV the coordination manager has a complete and consistent model of the environment in terms of resources, services, context and their dependencies. This approach allows a more easy writing of context dependent rules that may drive the application behavior.

4.3 Jini and JavaSpace

Jini [Jin03] is a distributed system based on Java. It offers a service model based on three components: an infrastructure for federating services in a distributed environment, a programming model for distributed services, and a set of system services, including a lookup service used by clients to locate required services and dynamically access them through the use of Java RMI stubs. Jini does not address the management of component-based applications and inter-component dependence. It only provides static look-up (exact matching) of services and does not consider the run-time resource constraints for small clients. Also, the large memory requirements imposed by Jini makes it not viable for most mobile devices. In addition, Jini announces service using UDP multicast by default, which may be suitable only in LAN-based application, but may not be applicable for large-scale deployment such as the Internet. In the Jini architecture, service functionality and capabilities are described in terms of Java object interface types. Service capability matching is processed in the object-level and syntax-level only. For instance, the generic Jini Lookup and other discovery protocols allow a service client to find a printing service that supports color printing, but the protocols are not powerful enough to find a geographically closest printing service that has the shortest print queue. The protocols do exact semantic matching while finding a service. Thus they lack the power to give a ”close match” even if it was available. An important part of the Jini system is the coordination model for generative communication called JavaSpace. A JavaSpace is a Linda-like coordination system that stores tuples representing a typed set of references to Java objects. Multiple JavaSpaces may coexist in a single Jini system. Jini does provide a specialized look-up service that allows clients

to look up registered services using an attribute-based search facility. Each service has an associated service identifier, which is a globally unique 128-bit value generated by the lookup service. A service uses this identifier to register a service item at the lookup service. A client can provide a template tuple when looking for specific tuple instances. The lookup service will select only those tuples that match the template. Jini implicitly forces the use of ad-hoc pre-defined ontology by ensuring that client queries contain interfaces. The lack of a well-defined ontology for service descriptions could result in false matching. These protocols do not solve the problem of making service discovery more flexible and powerful

5 Conclusions

To face the heterogeneity problem in Pervasive Computing this paper has introduced an architecture for supporting the designing, building, execution of applications that relay of semantic-based and application-centered management of resources, services and context. Following this approach, system support allows a certain level of visibility of the heterogeneity of the underlying environment and, at the same time, allow applications to explicitly provide their own semantic of the environment. The overall abstraction is realized at coordination level where the system is described in terms of autonomous and uniform interacting services. That way, applications can see resources, services and contextual changes in a uniform way and then adapt to it instead of forcing users to constantly reconfigure their systems.

We have presented UBIDEV, a system architecture for pervasive computing, which embodies this approach to building pervasive applications. The architecture supports a simple design process for building applications, starting from the definition of the system ontology, the pool of classifiers that will tag resources and contextual information as instances of the concepts composing the ontology, simple rules to express the interrelation between concepts of the ontology that will be used by the coordination control unit to solve the user's tasks. That could be summarized with the following contributions:

- **Resources**: the notion of representer as a virtualization of a physical resource, allows a system to have a full representation of the physical environment in terms of involved resources. In such a perspective, resources are seen as interacting entities that can both produce and receive stimuli from the control systems and from other entities. Classification of resources according to the system ontology allows a uniform description and configuration as well as a simple access protocol based on "addressing by concepts".
- **Services**: UBIDEV introduces the notion of service prerequirements as a collection of system properties that have to be met for ensuring the executability of a given service. By matching such prerequirements with the current available resources, UBIDEV builds a proper execution environment for each service. Capsules are instantiated to frame the notion of a running service together with its execution environment. Capsules exports at coordination level only service interfaces.

- **Coordination**: is a control unit where all information from resources, context, services, users and application are orchestrated. Coordination module is in charge of solving user's tasks by composing existing services; in doing so it analyses contextual information to determine which services to invoke and under which conditions.
- **Holistic Management**: in UBIDEV there is the explicit notion of system as a physical environment, a service infrastructure and an application. Resources, services and context are managed as part of a common space where the same semantic is used. Users are described as resources belonging to the environment; they can interact directly with other resources and be virtualized within the system by their role, identity and intentions.

In section 3 we have presented different projects that tackle the same class of problems UBIDEV does with similar approaches. The main differences from such approaches are:

- In UBIDEV the semantic of the environment is left to application by a system reference ontology. Then the classification of environment is done according to such ontology. That means that at application level, resources, services and contextual information are described and configured in a uniform way that application expects.
- UBIDEV focuses on the holistic management of resources, services and contextual information rather than describing, configuring and addressing them separately.
- UBIDEV allows a system to be described in terms of interacting homogeneous entities while exposing at coordination level the properties of the environment that may influence the whole system.

References

[AC93] G. Agha and C. J. Callsen. ActorSpace: An Open Distributed Programming Paradigm. In *Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, San Diego, CA, May 1993. Also published as a Special Issue of SIGPLAN Notices vol. 28, No. 7, pp 23-32, July, 1993.

[AJF02] S. Avancha, A. Joshi, and T. Finin. Enhanced Service Discovery in Bluetooth. *IEEE Computer*, 35(6):96–99, June 2002.

[BDF+99] M. Beck, J. Dongarra, G. Fagg, G. Geist, P. Gray, J. Kohl, M. Migliardi, K. Moore, T. Moore, P. Papadopoulous, S. Scott, and V. Sunderam. HARNESS: A Next Generation Distributed Virtual Machine. *Future Generation Computer Systems*, 15(5-6):571–582, 1999.

[BKA+98] E. Brewer, R. H. Katz, E. Amir, H. Balakrishnan, Y. Chawathe, A. Fox, S. D. Gribble, T. Hodes, G. Nguyen, V. N. Padmanabhan, M. Stemm, S. Seshan, and T. Henderson. A Network Architecture for Heterogeneous Mobile Computing. *IEEE Personal Communications Magazine*, 5(5):8–24, October 1998.

[CA94] C. J. Callsen and G. A. Agha. Open Heterogeneous Computing in Actorspace. *Journal of Parallel and Distributed Computing*, 21(3):289–300, 1994.

[CCMW01] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. Web Services Description Language (WSDL) 1.1. `http://www.w3.org/TR/wsdl`, March 2001. W3C Note.

[Coe98] M. H. Coen. Design Principles for Intelligent Environments. In *Fifteenth National Conference on Artificial Intelligence*, pages 547–554, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.

[Cor05] Microsoft Corporation. Universal Plug and Play: Device Architecture Version 1.0. http://www.upnp.org/, May 2005.

[CS92] C. Catlett and L. Smarr. MetaComputing. *Communications of the ACM*, 35(6):44–52, 1992.

[Dey00] A. K. Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, Georgia Institute of Technology, 2000.

[FS99] J. Flinn and M. Satyanarayanan. Energy-aware Adaptation for Mobile Applications. In *Proceedings of the 17th ACM Symposium on Operating Systems and Principles*, Kiawah Island, December 1999.

[GMGN04] X. Gu, A. Messer, I. Greenberg, and D. Milojicic K. Nahrstedt. Adaptive Offloading for Pervasive Computing. *IEEE Pervasive Computing*, 3(3):66–73, July-September 2004.

[Gut00] E. Guttman. *Service Location Protocol Modifications for IPv6*, January 2000. IETF Internet Draft.

[Har90] S. Harnad. The Symbol Grounding Problem. *Physica D*, 42:335–346, 1990.

[Inc99] The Salutation Consortium Inc. Salutation Architecture Specification (Part 1). http://www.salutation.org, 1999. Version 2.1.

[Jav03] JavaSpaces Service Specification. `http://wwws.sun.com/softwarejini/specs/`, June 2003.

[JF04] B. Johanson and A. Fox. Extending Tuplespaces for Coordination in Interactive Workspaces. *Journal of Systems and Software archive*, 69(3):243–266, January 2004. Special issue: Ubiquitous Computing.

[Jin03] Jini Architecture Specification. `http://wwws.sun.com/software/jini/specs/`, June 2003.

[Kam00] A. Kaminsky. Jini Print Service Design. `http://print.jini.org/`, February 2000.

[KBM+02] T. Kindberg, J. Barton, J. Morgan, G. Becker, D. Caswell, P. Debaty, G. Gopal, M. Frid, V. Krishnan, H. Morris, J. Schettino, B. Serra, and M. Spasojevic. People, Places, Things: Web Presence for the Real World. *Mobile Networks and Applications*, 7(5):365–376, October 2002. Kluwer Academic Publishers.

[KFJ03] L. Kagal, T. Finin, and A. Joshi. A Policy Language for a Pervasive Computing Environment. In *POLICY '03: Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks*, pages 63–75, Washington, DC, USA, 2003. IEEE Computer Society.

[KUB00] P. Kropf, H. Unger, and G. Babin. WOS: an Internet Computing Environment. In *Workshop on Ubiquitous Computing, IEEE International Conference on Parallel Architecture and Compilation Techniques*, pages 14–22, Philadelphia, PA, October 2000.

[MH02] S. Maffioletti and B. Hirsbrunner. UbiDev: an Homogeneous Environment for Ubiquitous Interactive Devices. In *Short Paper in Pervasive 2002 - International Conference on Pervasive Computing*, Zurich, Switzerland, August 2002.

[MKH04] S. Maffioletti, S. Kouadri, and B. Hirsbrunner. Automatic Resource and Service Management for Ubiquitous Computing Environments. In *Middleware Support for Pervasive Computing Workshop (at PerCom '04),PerWare '04*, pages 219–223, Orlando, Florida (USA), 14 March 2004.

[MRMC03] R. E. McGrath, A. Ranganathan, M. D. Mickunas, and R. H. Campbell. Investigations of Semantic Interoperability in Ubiquitous Computing Environments. In *15th IASTED International Conference on Parallel And Distributed Computing And Systems (PDCS 2003)*, Seattle Marina del Rey, CA, USA, 3-5 November 2003.

[MS99] M. Migliardi and V. Sunderam. The Harness Metacomputing Framework. In *Proceedings of the Ninth SIAM Conference on Parallel Processing for Scientific Computing*, San Antonio (TX), USA, 22-24 March 1999.

[NCN98] K. Nahrsted, H. Chu, and S. Narayan. QoS-aware Resource Management for Distributed Multimedia Application. *Journal on High-Speed Networking*, 7(3), 1998.

[Nob00] B. Noble. System Support for Mobile, Adaptive Applications. *IEEE Personal Communications*, 7(1), February 2000.

[Nor99] D. A. Normann. *The Invisible Computer*. MIT Pres, 1999.

[NSN+97] B.D. Noble, M. Satyanarayanan, D. Narayanan, J.E. Tilton, J. Flinn, and K.R. Walker. Agile Application-Aware Adaptation for Mobility. In *In Proceedings of the 16th ACM Symposium on Operating Systems Principles*, Saint-Malo, France, October 1997.

[pai] Pervasive and Artificial Intelligence Research Group. `http://www.unifr.ch/diuf/pai/`.

[RCC98] A. Robert, F. Chantemargue, and M. Courant. Emuds: Grounding Agents in EMud Artificial Worlds. In *Proceedings of the First International Conference on Virtual Worlds, VW'98*, Paris, France, 1-3 July 1998.

[Rek98] J. Rekimoto. A Multiple Device Approach for Supporting Whiteboard-Based Interactions. In *Proceedings of ACM CHI 98 Conference on Human Factors in Computing Systems*, pages 344–351, Los Angeles, CA USA, April 1998.

[RHC+02] M. Roman, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt. Gaia: A Middleware Infrastructure to Enable Active Spaces. *IEEE Pervasive Computing*, pages 74–83, Oct-Dec 2002.

[Sch02] S. Schubiger. *Automatic Software Configuration*. PhD thesis, Department of Computer Science, University of Fribourg (CH), October 2002. No. 1393. A short version appeared in: S. Schubiger and B. Hirsbrunner. A Model for Software Configuration in Ubiquitous Computing Environments. In Pervasive 2002, International Conference on Pervasive Computing. 26-28 August 2002, Zurich.

[SG02] J. Pedro Sousa and D. Garlan. Aura: an Architectural Framework for User Mobility in Ubiquitous Computing Environments. In *Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture*, pages 29–43, 25-31 August 2002.

[SMTH00] S. Schubiger, S. Maffioletti, A. Tafat, and B. Hirbrunner. Providing Service in a Changing Ubiquitous Computing Environment. In *Workshop on Infrastructure for Smart Devices - How to Make Ubiquity an Actuality, HUC2K, Bristol, UK*, September 2000.

[SSW94] K. Sagonas, T. Swift, and D. S. Warren. XSB as an Efficient Deductive Database Engine. *ACM SIGMOD Record*, 23(2):442–453, June 1994.

[Tan97] A. S. Tanenbaum. *Operating Systems: Design and Implementation*. Prentice-Hall, Inc., Upper Saddle River, New Jersey, 1997. 2nd edition.

[TCH04] A. Tafat, M. Courant, and B. Hirsbrunner. A Generic Coordination Model for Pervasive Computing Based on Semantic Web Languages. In *9th International Conference on applications of natural languages to information Systems, ICANLIS'04*, Manchester, June 2004. Appeared in Lectures Notes in Computer Science, Springer, F. Meziane, E. Mtais, eds., 2004 vol. 3136. pp. 265-275.

[Tv02] A. S. Tanenbaum and M. van Steen. *Distributed Systems: Principles and Paradigms*. Prentince Hall, Upper Saddle River, New Jersey, 2002.

[WG00] Z. Wang and D. Garlan. Task Driven Computing. Technical Report CMU-CS-00-154, Carnegie Mellon University, May 2000.

Chapter 9: Information and Knowledge Management

Contributions to 6th I2CS 2006, Neuchâtel, Switzerland

Fabien Mathieu, Laurent Viennot
Local Aspects of the Global Ranking of Web Pages

Benoît Garbinato, Ian Rickebusch
Impossibility Results on Fair Exchange

Achim Dannecker, Ulrike Lechner
On the Demand for E-Services by Health Communities

Gilson Yukio Sato, Jean-Paul Barthès
A Multi Agent System Application to Support Communities of Practice: Preliminary Analysis

Jacques Savoy
Stemming Strategies for European Languages

Local Aspects of the Global Ranking of Web Pages

Fabien Mathieu
Orange Labs
38 rue du général Leclerc
92 130 Issy les Moulineaux France

Laurent Viennot
INRIA Rocquencourt
F-78153 Le Chesnay Cedex France

Abstract: Started in 1998, the search engine *Google* estimates page importance using several parameters. *PageRank* is one of those. Precisely, *PageRank* is a distribution of probability on the Web pages that depends on the Web graph. Our purpose is to show that the PageRank can be decomposed into two terms, internal and external PageRank. These two PageRanks allow a better comprehension of the PageRank signification inside and outside a site. A first application is a local algorithm to estimate the PageRank inside a site. We will also show quantitative results on the possibilities for a site to boost its own PageRank.

1 Introduction

PageRank [PBMW99] was a major algorithmic breakthrough for evaluating the importance of Web pages achieved by exploiting the topology of the Web induced by hyperlinks. Numerous works have then been devoted to better understand the relation between this Web graph structure and the quality of Web pages. Some authors have proposed alternative methods for ranking pages [Kle98, SB03] based on similar matrix computations. Other results propose different computations of an approximation of the PageRank either to obtain a faster algorithm [KHMG03a, BLMP04] or an incremental algorithm [APC03].

This paper tries to model how the PageRank decomposes with regards to the site partition of the Web. A site can be seen as the collection of pages on a given Web server or more generally as a set of pages tightly related. As noted by [MV03, MV02, RGM03], a block structure of the Web adjacency matrix can be observed from an url-induced ordering of the pages[1], showing how an intrinsic site partition could be defined. This paper assumes that a site partition is given, i.e. the Web is decomposed in a collection of sites.

Even if one can naturally state that the Web graph structure is tightly related to the site partition (most of the links are local), the Web graph has mainly been studied disregarding this property. This is the case for the PageRank computation. In [KHMG03a, BLMP04], a site partition is exploited to efficiently compute an approximated PageRank. On the other hand, this paper makes an exact decomposition of the PageRank computation, showing how the PageRank can be split into an internal PageRank (related to internal links of a

[1] This means clusters of pages naturally appear at different levels of the URL hierarchy, such as domain names, hosts, directories of Web servers...

site) and an external PageRank (related to inter-site links). In [BGS05, BGS02], the sum of the PageRanks of the pages of a site is decomposed according to internal, incoming links, outgoing links and sinks. The authors give basic hints on how the link structure of site can alter its PageRanks. A stability property of the overall PageRanks when a site changes its internal link structure is also shown. Our model of decomposed PageRank allows to push forward their analysis to better understand how a site can alter its own PageRanks.

Another contribution of our site decomposition model of PageRank is a framework for evaluating locally the global PageRank. This could be useful for a local search engine to rank the pages of a site according to a global importance knowing only locally the Web structure.

The paper is organized as follows. Section 2 defines more formally the PageRank. Section 3 introduces our model for decomposing the PageRank according to a site partition of the Web. Section 4 shows how to locally estimate the global PageRank of the pages of a site. Finally, Section 5 analyzes how a site administrator can alter the PageRank of its pages by modifying the links inside the site.

2 PageRank Definition

Let $G = (V, E)$ be an oriented aperiodic strongly connected graph, without self-loop, and $\mathcal{S} = (S_1, \ldots, S_k)$ a partition of G, with $k > 1$. G is supposed to be a Web graph, and $\mathcal{S}$ a site partition of G (elements of $\mathcal{S}$ are sites).

If $d^+(v)$ is the out degree of $v \in V$, we can define the following stochastic matrix $A : V \times V \to \mathbb{R}^+$

$$A = (a_{i,j})_{i,j \in V}, \text{ with } a_{(i,j)} = \begin{cases} \dfrac{1}{d^+(i)} \text{ if } i \text{ links to } j, \\ 0 \text{ otherwise.} \end{cases}$$

According to Markov processes theory [SC96], there is a unique probability P on V such that:

$$\forall v \in V, P(v) = \sum_{w \to v} \frac{P(w)}{d^+(w)} \tag{1}$$

The matrix version of this is:

$$P = A^t P, \tag{2}$$

where A^t is the transposed matrix of A.

The probability vector P defines the PageRank of the graph G. This concept of PageRank was introduced in 1998 [PBMW99] and used by the search engine *Google* [Goo98].

Good convergence properties and unicity of P are obtained if A is irreducible and aperiodic, that is if the underlying graph is strongly connected (and aperiodic). More insight on how A is altered to obtain such properties if G is not strongly connected is given in Section 5.2.

3 Internal PageRank, external PageRank

3.1 Notations

For $v \in V$, we call $S(v)$ the element of $\mathcal{S}$ such that $v \in S(v)$. We also define $\delta_{\mathcal{S}}$: $V \times V \rightarrow \{0, 1\}$ as follows:

$$\delta_{\mathcal{S}}(v, w) = \begin{cases} 1 \text{ if } S(v) = S(w), \\ 0 \text{ otherwise.} \end{cases}$$

Let A_{int} be the matrix of the projection of A on the internal elements according to $\mathcal{S}$: $A_{\text{int}} = (a_{v,w}\delta_{\mathcal{S}}(v, w))_{v,w \in V}$. A_{ext} is defined the same way so we have $A = A_{\text{int}} + A_{\text{ext}}$

We also need to define the internal degree d_i^+ (resp. the external degree d_e^+) of a vertex v as its out degree in the graph induced by $S(v)$ (resp. $\{v\} \cup (V \setminus S(v))$).

Lastly we can define the notions of internal and external PageRank, deduced from the PageRank P seen on formula (2).

- The incoming internal PageRank P_{ii} (resp. incoming external PageRank P_{ie}) of $v \in V$ is the probability to come in v from a page of $S(v)$ (resp. $V \setminus S(v)$), that is:

$$P_{ii} = A_{\text{int}}^t P, \tag{3}$$
$$P_{ie} = A_{\text{ext}}^t P = P - P_{ii}. \tag{4}$$

- The outgoing internal PageRank P_{oi} (resp. outgoing external PageRank P_{oe}) of $v \in V$ is the probability to go from v to a page of $S(v)$ (resp. $V \setminus S(v)$), that is:

$$P_{oi} = (A_{\text{int}}.\mathbf{1}).\times P, \tag{5}$$
$$P_{oe} = (A_{\text{ext}}.\mathbf{1}).\times P = P - P_{oi}, \tag{6}$$

where $.\times$ is the element by element product and $\mathbf{1}$ is a vector filled with ones.

3.2 Conservation laws

Using the definitions, we have the following equation:

$$P = P_{ie} + P_{ii} = P_{oe} + P_{oi} \tag{7}$$

We can now give the internal and external conservation laws. For each $S \in \mathcal{S}$, we see that

$$\sum_{v \in S} P(v) = \sum_{v \in S} \sum_{w \to v} \frac{P(w)}{d^+(w)} = \sum_{(w,v) \in E \cap (V \times S)} \frac{P(w)}{d^+(w)} \tag{8}$$

$$= \sum_{(w,v) \in E \cap S^2} \frac{P(w)}{d^+(w)} + \sum_{(w,v) \in E \cap (V \setminus S \times S)} \frac{P(w)}{d^+(w)} \tag{9}$$

$$= \sum_{w \in S} P_{oi}(w) + \sum_{v \in S} P_{ie}(v) \tag{10}$$

We can deduce from (7) and (10) the external conservation law:

$$\sum_{v \in S} P_{ie}(v) = \sum_{v \in S} P_{oe}(v) \tag{11}$$

We can also deduce from (7) and (11) the internal conservation law:

$$\sum_{v \in S} P_{ii}(v) = \sum_{v \in S} P_{oi}(v). \tag{12}$$

The relation (11) shows that a site gives as much PageRank (outgoing external) that it receives (incoming external). If PageRank is a random surfer flow, there is a conservation of the external PageRank flow on the graph $G/\mathcal{S}$. That remark will lead us to an intra-site and an inter-sites calculation of PageRank.

3.2.1 Remark:

If we formalize carefully the PageRank as a flow, we have another proof of (11): the PageRank is actually a stationary flow, so the flow on every subset S is stationary, therefore we have (11). We have preferred a matrix approach for proof because matrices will be widely used in this article.

4 Local computation of the global Ranking

4.1 Relation between external PageRank and PageRank

From (3) and (4) we can write $A_{\mathcal{S}}^t.P = P - P_{ie}$, and then $P_{ie} = (Id - A_{\text{int}}^t)P$, where Id is the identity matrix.

Lemma 1. *The matrix $(Id - A_{int}^t)$ is invertible.*

Proof. As G is strongly connected, there are links between sites. Thus we have $0 \leq A_{\text{int}} < A$. A_{int} is strictly inferior to an irreducible stochastic matrix, so its spectral radius is strictly inferior to 1. Therefore $(Id - A_{\text{int}}^t)^{-1}$ exists. □

Lemma 1 allows us to express P as a function of P_{ie}:

$$P = (Id - A_{\text{int}}^t)^{-1} P_{ie} \tag{13}$$

Knowing the incoming external PageRank P_{ie} of a site S, we can theoretically use the projection of Equation (13) on S to compute the PageRank of the pages of S with only the local graph G_S.

4.1.1 Remark

$(Id - A_{\text{int}}^t)^{-1} = \sum_{k=0}^{\infty} (A_{\text{int}}^t)^k$ is a diagonal by blocks matrix, that can be interpreted as the transition matrix of all the internal paths.

4.2 External PageRank matrix

We want to translate the intuition of Equation (11) in a conservation law with P_{ie} only. From (4) and (13), we have:

$$P_{ie} = (A - A_{\text{int}})^t P = (A - A_{\text{int}})^t (Id - A_{\text{int}}^t)^{-1} P_{ie} \tag{14}$$

We thus define the external PageRank transition matrix A_e:

$$A_e = (Id - A_{\text{int}})^{-1} A_{\text{ext}}$$

Lemma 2. *The matrix A_e is stochastic.*

Proof. We have to show that A_e is positive and that $A_e.\mathbf{1} = \mathbf{1}$. As $A_e = (\sum_{k=0}^{\infty} A_{\text{int}}^k) A_{\text{ext}}$, positivity is obvious. We conclude the proof by writing that

$$
\begin{aligned}
A_e.\mathbf{1} &= \left(\sum_{k=0}^{\infty} A_{\text{int}}^k\right)(A - A_{\text{int}}).\mathbf{1} \\
&= \left(A + \left(\sum_{k=1}^{\infty} A_{\text{int}}^k\right)(A - Id)\right).\mathbf{1} \\
&= A.\mathbf{1} + \left(\sum_{k=1}^{\infty} A_{\text{int}}^k\right)(A.\mathbf{1} - Id.\mathbf{1}) \\
&= \mathbf{1} + \left(\sum_{k=1}^{\infty} A_{\text{int}}^k\right)(\mathbf{1} - \mathbf{1}) = \mathbf{1}.
\end{aligned}
$$

This ensures that A_e is stochastic. □

4.3 Partially distributed PageRank algorithm

From (13) and (14) we can suggest a half-distributed algorithm for computing the PageRank:

- Each site S computes from its block of A_{int} a block of the matrix $(Id - A_{\text{int}}^t)^{-1}$.
- The coefficients of A_e are centralized.
- The external PageRank P'_e associated with A_e is centrally computed using $A_e^t P'_e = P'_e$.
- Each site S gets its own PageRank thanks to the relation $P' = (Id - A_{\text{int}}^t)^{-1} P'_e$ applied to its block.

Lemma 3. *The vector P' we obtain is, once normalized, the PageRank P of G.*

Proof. We have to show that P' is an eigenvector of A^t, and that its eigenvalue is 1:

$$
\begin{aligned}
A^t P' &= A^t (Id - A_{\text{int}}^t)^{-1} P'_e \\
&= A_{\text{ext}}^t (Id - A_{\text{int}}^t)^{-1} P'_e + A_{\text{int}}^t (Id - A_{\text{int}}^t)^{-1} P'_e \\
&= A_e^t P'_e + \left((Id - A_{\text{int}}^t)^{-1} - (Id - A_{\text{int}}^t)(Id - A_{\text{int}}^t)^{-1}\right) P'_e \\
&= P'_e + \left((Id - A_{\text{int}}^t)^{-1} - Id\right) P'_e \\
&= P'_e + P' - P'_e = P'
\end{aligned}
$$

As the principal eigenvalue of A, that is 1, is unique, P and P' are homothetic, so $P = P'$ (after normalization). □

4.3.1 Efficient computation of external PageRank:

The above algorithm could seem useless since the centralized step operates on A_e which is a $|V| \times |V|$ matrix. However, from $P'_e = A^t_e P'_e$, we can write $P'_e = A^t_{\text{ext}} P''$, where $P'' = (Id - A^t_{\text{int}})^{-1} P'_e$. That means that only vertices with incoming external links have a non null value in P'_e. Thus we can safely reduce A_e and P'_e to their projections on V_{ext}, where V_{ext} is the set of vertices that have at least one incoming external link. The new equation to compute the external PageRank is then:

$$P_{Reduced} = A^t_{Reduced} P_{Reduced},$$

where $P_{Reduced}$ is a vector of V_{ext} and $A_{Reduced}$ is a $V_{ext} \times V_{ext}$ matrix defined by $A_{Reduced}(i,j) = A_e(i,j)$ for each $i,j \in V_{ext}$. Thanks to the reduction to V_{ext}, the computational cost may be greatly reduced: for instance, analysis of Web logs and crawls shows that for the site `inria.fr` we have $|V_{ext}| \leq 0.1\,|V|$.

4.4 Link between our algorithm and BlockRank

Kamvar et al. [KHMG03a], and more recently *Broder et al.* [BLMP04], use a similar algorithm based on the local block structure of the Web [MV02]. They first compute a local PageRank, that with our notation is the PageRank of the A_{int} matrix, then use it to compute a BlockRank matrix B that is a substochastic matrix on the quotient graph $G/\mathcal{S}$ defined as transitions between sites. Their approximation of PageRank is the local PageRank weighted by the probability of being in a given block $S \in \mathcal{S}$, obtained using B (all details can be found in [KHMG03a, BLMP04]). Although they look similar, their algorithms and our have some differences that we would like to point out:

- The algorithm presented in Section 4.3 converges to the exact PageRank. Thus the problem of the weaker PageRank on root pages in BlockRank [KHMG03a, BLMP04] does not occur.
- The local PageRank matrix A_{int} and the BlockRank matrix B used in BlockRank correspond respectively to our internal transition matrix $(Id - A^t_{\text{int}})^{-1}$ and our external PageRank matrix A_e. This difference, which comes from our modeling of the PageRank as a flow, allows us to compute an exact PageRank, whereas BlockRank is just an estimation.

4.4.1 *À la BlockRank* algorithm:

We just saw that B is a $k \times k$ matrix, when $A_{Reduced}$ is a $V_{ext} \times V_{ext}$ matrix, with $k \leq |V_{ext}| \leq |V|$. This size difference is the price of the exact computation of PageRank. Indeed $A_{Reduced}$ can still be reduced up to a $k \times k$ matrix if we centralise all the external flow to one or a few chosen page(s) per site (the main page(s) for example). Reduction is

done by choosing a set of main pages V_{main}, with $m(S) \geq 1$ page(s) per site, and using the projection A_{main} of the matrix $A_e.R$ to V_{main}, where:

$$R(v,w) = \begin{cases} \dfrac{1}{m(S(w))} \text{ if } S(v) = S(w) \text{ and } w \in V_{main}, \\ 0 \text{ otherwise.} \end{cases}$$

The size of the external matrix becomes $|V_{main}| \times |V_{main}|$, with V_{main} fully configurable down to one page per site (thus reaching the $k \times k$ limit), but the corresponding ranking is now an approximation, which should lead to an overestimation of the actual PageRank of the main pages (when BlockRank underestimates them). Our formalism thus allows to obtain an approximated PageRank algorithm similar to BlockRank with additional tunability from fast computation time to exact PageRank computation.

4.5 Estimation of a site PageRank

A natural question is whether a site S can estimate the ranking of its pages only knowing local data. This can be very valuable for an internal search engine that wants to estimate the global ranking of its pages without crawling all the Web or asking an external search engine. From (13), all we need is an estimation of the incoming external PageRank.

According to [PBMW99], PageRank models the statistic behaviour of surfers crawling the Web. It seems then natural to estimate the PageRank of a page by the average hits it gets. More specifically, the incoming external PageRank should be proportional to the average hits from outside the site. So each site can get an estimation of the incoming external rank from analysing the logs files of its Web server. This gives webmasters a smart way for locally computing a structural ranking flavoured with real traffic statistics.

Abiteboul et al. [APC03] states that the incoming degree is a good estimation of the PageRank. Thus the number of external references for each page (obtained from the logs files) is another estimation, which is now purely structural, of P_{ie}.

Both estimation methods of the incoming external PageRank will be furthered studied in future works.

Once P_{ie} is estimated, the global PageRank can be obtained by $P = (Id - A_{\text{int}}^t)^{-1} P_{ie}$. In fact, $(Id - A_{\text{int}}^t)^{-1}$ does not have to be calculated explicitly. It is better to solve $P = A_{\text{int}}^t P + P_{ie}$ using iterative methods, for example by choosing P_0 and iterating

$$P_{n+1} = A_{\text{int}}^t P_n + P_{ie}.$$

This converges, because the spectral radius of A_{int} is strictly inferior to 1. Empirical results [PBMW99, KHMG03a] suggest a fast convergence of such algorithms applied to portions of the Web graph.

4.5.1 Remark

There are lot of methods to improve the convergence of that sort of iterative computation [AK98, KHMG03b]. As the purpose of this paper is not to optimize this part of the computation, they are not discussed here.

4.5.2 Interest of our method

One could wonder why not keeping the average hits per page as an estimation of the PageRank? We believe our method can give a better PageRank to pages newly created, that do not get a lot of hits yet but are well linked and will surely get known.

Another advantage is that the P_{ie} input can be very flexible. The webmaster could manually alter P_{ie} to promote some pages while keeping a minimum of ranking.

5 Locally altering the PageRank

Our decomposition of the PageRank explains some results about the ability that a site has to alter its own PageRank. A first approximation is to say that if a site can hardly alter the external PageRank, this is much easier for the internal PageRank.

5.1 Amplification factor

Let S be a site, $P(S) = \sum_{v \in S} P(v)$ and $P_{ie}(S) = \sum_{v \in S} P_{ie}(v)$. We can define the amplification factor of S by $\alpha(S) = \frac{P(S)}{P_{ie}(S)}$. This factor depends on both S and the distribution of the actual external PageRank[2], but knowing S we can estimate $\alpha(S)$.

Lemma 4. *The amplification factor can be estimated by:*

$$\frac{1}{1-\omega} \leq \alpha(S) \leq \frac{1}{1-\Omega} \tag{15}$$

with $\omega = \min_{v \in S} \frac{d_i^+(v)}{d^+(v)}$ *and* $\Omega = \max_{v \in S} \frac{d_i^+(v)}{d^+(v)}$.

Proof. We define A_S as the restriction of A_{int} to site S. If we see S as a $|S|$-dimensional vector space, for each base vector e_v, $v \in S$, we have

$\|A_S(e_v)\|_1 = \frac{d_i^+(v)}{d^+(v)}$, therefore $\omega \|X\|_1 \leq \|A_S X\|_1 \leq \Omega \|X\|_1$ for any vector $X > 0$ defined on S.

[2]Note that the external PageRank can more or less depend on the structure of site S, especially if S is close to pages pointing to it.

The first inequality of (15) is obtained as follows:

$$P(S) = \sum_{v \in S} P(v) = \left\| \sum_{k \in \mathbb{N}} (A_S^t)^k (P_{ie}) \right\|_1 \geq \sum_{k=0}^{\infty} \omega^k \left\| P_{ie} \right\|_1 = \frac{1}{1-\omega} P_{ie}(S)$$

The proof of the second inequality is similar. □

The consequences of this amplification system is that a site can arbitrarily increase its PageRank. In the limit case where the site has no external link[3], we have a short-circuit phenomenon. This is known as the sink hole phenomenon [PBMW99]: a set of pages with no outgoing link absorbs all PageRank.

Fortunately, we will see how the damping factor reduces this effect.

5.2 The damping factor

As said in Section 2, good convergence properties are obtained whenever G is a strongly connected graph. Otherwise, transient pages can exist (they obtain a zero PageRank), and A may be sub-stochastic (if some pages do not have out-link). As the Web graph is far from being strongly connected [BKM$^+$00], there are several techniques to overcome this, often by altering the transition matrix A. We focus on the damping factor[4], introduced by [BP98]. It is originally used by *Google* on a graph where leaves are non-recursively removed and reinjected after P converged. The principle of the damping factor is to replace A by $d.A + \frac{1-d}{|V|}\mathbf{1}.\mathbf{1}^t$, where d is the so-called damping factor. The new transition matrix represents a weighted strongly connected graph, and it is stochastic (we still suppose that the genuine matrix A is stochastic; see [PBMW99] for normalization issues about pages without outgoing link). We have then a superposition of classic transitions ($d.A$) and damping transitions ($\frac{1-d}{|V|}\mathbf{1}\mathbf{1}^t$). Damping transitions are supposed to model the action of moving anywhere in the Web without following any static link (user *Bookmarks*, search engines, keyboard input,...). Note that ($\frac{1}{|V|}\mathbf{1}\mathbf{1}^t$) corresponds to the uniform transition matrix from any page to any page.

Instead of splitting the damping flow into an external one and an internal one, we find more interesting to introduce the notions of induced PageRank P_{ind} and dissipated PageRank P_{dis}. We have now six different elementary PageRanks corresponding to three types of flows as shown in Figure 1:

[3] A real site does not have to respect the strong connectivity of A. In particular, many commercial sites do not have any external link [BKM$^+$00].

[4] For a non-exhaustive view of the other techniques:

- *Page et al.* [PBMW99] suggests to compensate the flow leak in A by normalizing P at each iteration.
- *Haveliwala et al.* [Hav99] turn A explicitly into a stochastic matrix by removing recursively pages without link.
- *Abiteboul et al.* [APC03] adds a virtual damping page that links to and is linked to every other page.

flow	incoming	outgoing
internal	$P_{ii} = dA_{\text{int}}^t P$	$P_{oi} = d(A_{\text{int}}\mathbf{1}).\times P$
external	$P_{ie} = d(A - A_{\text{int}})^t P$	$P_{oe} = d((A - A_{\text{int}})\mathbf{1}).\times P$
damping	$P_{ind} = \frac{1-d}{\|V\|}\mathbf{1}$	$P_{dis} = (1-d)P$

Figure 1: The different flows of PageRank in the damping factor case

We can rewrite the conservation laws considering the whole damping flow as external. Of course there are internal damping transitions, but we choose to tag them as external. Thus the internal flow conservation law does not change, but we have a new external flow conservation law:

$$\sum_{v \in S} \left(P_{ie}(v) + P_{ind}(v)\right) = \sum_{v \in S} \left(P_{oe}(v) + P_{dis}(v)\right),$$

that we will note

$$P_{ie}(S) + P_{ind}(S) = P_{oe}(S) + P_{dis}(S) \tag{16}$$

5.2.1 PageRank stability

The equation (16) shows the stability of the classic flow at the site level. From $P_{ind}(S) = (1-d)\frac{|S|}{|V|}$ and $P_{dis}(S) = (1-d)P(S)$, we can tell that for a site whose PageRank $P(S)$ is above (resp. below) the average PageRank (which is $\frac{|S|}{|V|}$ for a site of size $|S|$), the outgoing external PageRank $P_{oe}(S)$ is inferior (resp. superior) to the incoming external PageRank $P_{ie}(S)$. In other words, a *rich* site (in term of PageRank) will be greedy and will give less than it receives (damping excluded), and vice versa. The damping factor causes a retro-action that limits the phenomenon of over-amplification, as developed in next section.

5.3 Damping and amplification

The transition matrix is of the form $d.A + \frac{1-d}{|V|}\mathbf{1}\mathbf{1}^t$. We obtain results similar to Section 5.1 by replacing A by dA and P_{ie} by the total incoming external PageRank $P_{ie} + P_{ind}$.

Lemma 5. *The amplification factor* $\alpha'(S) = \frac{P(S)}{P_{ie}(S)+P_{ind}(S)}$ *verifies:*

$$\frac{1}{1 - d\omega} \leq \alpha'(S) \leq \frac{1}{1 - d\Omega}. \tag{17}$$

Proof. It is the same than for (15); we can write:

$$
\begin{aligned}
P(S) &= \sum_{v\in S} P(v) = \left\| \sum_{k\in\mathbb{N}} (dA_S^t)^k (P_{ie} + \frac{1-d}{|V|}\mathbf{1}) \right\|_1 \\
&\leq \sum_{k=0}^{\infty} (d\Omega)^k (\|P_{ie}\|_1 + (1-d)\frac{\|\mathbf{1}\|_1}{|V|}) \\
&\leq \frac{1}{1-d\Omega}\left(P_{ie}(S) + (1-d)\frac{|S|}{|V|} \right)
\end{aligned}
$$

The second inequality is obtained similarly.

□

5.3.1 Numerical Value

It is not impossible for a real site to have $\omega = \Omega = 0$ (site without internal link) or $\omega = \Omega = 1$ (site without external link). So the amplification factor can vary between 1 and $\frac{1}{1-d}$. The empirical value of d being 0.85, we deduce that with a fixed incoming external PageRank, the PageRank of a site can fluctuate up to a factor $\frac{20}{3}\ldots$

5.3.2 PageRank robustness

Bianchini et al. [BGS02] states that the effect that a site can produce onto the Web is bounded by the PageRank of this site. If only links from a site S have changed between two instants t and $t+1$, they show that:

$$\sum_{v\in V} |P_t(v) - P_{t+1}(v)| \leq \frac{2d}{1-d} \sum_{s\in S} P_t(s)$$

This result is a straightforward implication of Lemma 5: if the site S changes between t and $t+1$, the PageRank variation inside S is at most $\frac{d}{1-d}P(S)$, implying a variation up to another $\frac{d}{1-d}P(S)$ outside the site, since the total PageRank stays equal to 1.

5.4 Amplification of a given page

However, a search engine answers a lot of pages for most of the requests. This implies that a site administrator may be less interested by getting a large average PageRank than getting a few pages with hight PageRank or even a single one. We thus consider the following problem: let S be a site of $n+1$ pages and P_{ie} its incoming external PageRank; how can we maximize the PageRank of a given page $v_0 \in S$?

The answer is not difficult once we remark that the optimal link structure is when v_0 links to all other pages of S and all other pages of S link to v_0 and only v_0[5]. It is not hard then to give a tight upper bound for the ranking $P(v_0)$ of v_0:

$$P(v_0) \leq \frac{P_{ie}(S)}{1-d^2} + \frac{1+nd}{(1+d)\,|V|}, \tag{18}$$

with equality if and only if $P_{ie}(S) = P_{ie}(v_0)$.

This suggests some strategies to improve the PageRank of a page v_0[6]. For instance:

- If v_0 links to all other pages without backward links[7], adding the links to v_0 can increase the PageRank of v_0 up to $\frac{1}{1-d^2} \simeq 3,6$.
- The optimal strategy ensures for v_0 a minimal PageRank at least equal to the average PageRank $\frac{1}{|V|}$ even if P_{ie} is null.
- If $1 \ll n \leq |V|$ (for a large site dynamically generating pages linking to v_0), the ratio $\frac{P(v_0)}{P_{average}}$ is about $\frac{d}{1+d}n$.

6 Conclusion

We have proposed a decomposition of the PageRank flow in accordance with the notion of site, showing how to use it for estimating locally the global PageRanks inside a site. However, this relies on estimating the incoming PageRank either with real user hits or external referer counts. Further experiments are needed for fully validating this approach. Another interesting research direction includes distributed computation of the PageRank: assuming that several sites collaborate, how to compute the PageRank induced by their union? Our model is certainly the first step for that. It can also be useful for evaluating approaches that alter the PageRank computation based on a site decomposition as proposed by [KHMG03a, BLMP04] for speeding up the computation. Another related issue is the identification and the ranking of sites rather than pages.

At least, the flow decomposition has allowed to analyze some strategies that the webmasters could use if an unrefined version of PageRank was used by search engines. We have shown that the PageRank defined in [PBMW99] can be very versatile when subject to non-cooperative strategies. It also seems that P_{ie} can be more robust, assuming we are able to find a site partition $\mathcal{S}$ that reflects the reality.

[5] PageRank algorithms systematically remove self-loops, so a single page cannot amplify itself.

[6] In fact, it seems that *Google* is rather aware of these strategies, so they do not work as well as they should in theory...

[7] A typical situation when using *frames*.

References

[AK98] G. Allaire and S. M. Kaber. *Algèbre linéaire numérique*. Ellipses, 1998.

[APC03] Serge Abiteboul, Mihai Preda, and Gregory Cobena. Adaptive on-line page importance computation. In *Proc. 12th International World Wide Web Conference*, pages 280–290. ACM Press, 2003.

[BGS02] Monica Bianchini, Marco Gori, and Franco Scarselli. PageRank: A Circuital Analysis. In *Proc. 11th International Word Wide Web Conference*, 2002.

[BGS05] Monica Bianchini, Marco Gori, and Franco Scarselli. Inside PageRank. *ACM Transactions on Internet Technology*, 5:92–128, 2005.

[BKM+00] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, and A. Tomkins. Graph structure in the web. In *Proc. 9th International World Wide Web Conference*, pages 309–320, 2000.

[BLMP04] Andrei Z. Broder, Ronny Lempel, Farzin Maghoul, and Jan Pedersen. Efficient PageRank approximation via graph aggregation. In *Proc. 13th International World Wide Web conference on Alternate track papers & posters*, pages 484–485, 2004.

[BP98] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[Goo98] Google. http://www.google.com/, 1998.

[Hav99] T. Haveliwala. Efficient computation of PageRank. Technical Report 1999-31, Computer Science Department, Stanford University, 1999.

[KHMG03a] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub. Exploiting the block structure of the web for computing PageRank. Technical Report 2003–16, Stanford University, march 2003.

[KHMG03b] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub. Extrapolation methods for accelerating PageRank computations. In *Proc. 12th International World Wide Web Conference*, pages 261–270, 2003.

[Kle98] J.M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677, San Francisco, California, 25–27 January 1998.

[MV02] Fabien Mathieu and Laurent Viennot. Structure intrinsèque du web. Technical Report RR-4663, INRIA, December 2002.

[MV03] F. Mathieu and L. Viennot. Local Structure in the Web. In *Proc. 12th International World Wide Web Conference on Alternate track papers and posters*, 2003.

[PBMW99] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999–66, Computer Science Department, Stanford University, 1999.

[RGM03] S. Raghavan and H. Garcia-Molina. Representing web graphs, 2003.

[SB03] Pierre P. Senellart and Vincent D. Blondel. Automatic discovery of similar words. In Michael W. Berry, editor, *Survey of Text Mining*. Springer-Verlag, 2003.

[SC96] L. Saloff-Coste. Lectures on finite Markov Chains. In G.R. Grimmet E. Giné and L. Saloff-Coste, editors, *Lecture Notes on Probability Theory and Statistics*, number 1665 in LNM, pages 301–413. Springer Verlag, 1996.

Impossibility Results on Fair Exchange

Benoît Garbinato Ian Rickebusch

Université de Lausanne
CH-1015 Lausanne, Switzerland
{benoit.garbinato, ian.rickebusch}@unil.ch

Abstract: The contribution of this paper is threefold. First, we propose a novel specification of the fair exchange problem that clearly separates safety and liveness. This specification assumes a synchronous model where processes communicate by message passing and might behave maliciously. In this model, we prove a first impossibility related to the notion of trust, stating that no solution to fair exchange exists in the absence of an identified process that every process can trust a priori. Finally, we derive an enriched model where processes are divided into trusted and untrusted processes, and we show that an additional assumption is still necessary to solve fair exchange. Intuitively, this result expresses a condition on the connectivity of correct but untrusted processes with respect to trusted processes. We also revisit existing fair exchange solutions described in the literature, in the light of our enriched model, and show that our second impossibility result applies to them.

1 Introduction

In our modern daily lives, the notions of fair exchange and trust are ubiquitous: everyday, without even noticing, we participate in numerous commercial exchanges, which we expect to be fair (and most actually are). A key enabler to make all these exchanges occur is the notion of *trust*. In the physical world, this trust is supported by the identification and the implicit reputation of tangible exchange partners. In the digital world, on the contrary, fair exchange is a surprisingly difficult problem. This can be explained by the lack of trust that characterizes the digital realm. Yet, fair exchange is a fundamental problem that has lately regained interest [AGGV05]. This is partly due to the advent of m-business as a natural evolution of e-business, i.e., extending the possibilities of e-business through the use of mobile devices, e.g., cellular phones.. When it comes to solve fair exchange in such semi-open environments, i.e., where all parties are not necessarily identified a priori, carefully modeling and analyzing trust relationships between processes is a key issue.

Most successful e-business solutions today are following a classical client/server architecture (centralized). This implies that current e-business solutions somehow fail to take full advantage of the Internet's underlying protocols, which were designed to support fully decentralized approaches.[1] For example, current e-business solutions do not provide a

[1] The ARPANET project aimed at building a network with no single point of failure, in order to survive a nuclear strike.

favorable environment for electronic exchanges in the absence of some centralized and trusted server, i.e., they fail to support peer-to-peer only settings. On the other hand, the emergence of mobile devices and ad hoc networks, which often have to operate in a disconnected manner with respect to the Internet, is forcing us to reconsider the current e-business architectures. In that respect, in a companion paper [GR06], we advocate that fair exchange is a key building block when it comes to support peer-to-peer m-business interactions at the middleware level, and we propose such a middleware.

Research on Fair Exchange. Various specifications of the fair exchange problem have been proposed, with slightly different sets of properties [AGGV05, ASW00, Ate99, FR97, Mic03, PG99, RRN05]. Among these properties, fairness is the most difficult to capture and where these specifications tend to differ [RRN05, MGK02]. Most specifications are actually meaningful for exchanges involving only two processes, i.e., they are impossible to satisfy in models allowing more than one Byzantine process. In [AGGV05] for instance, fairness requires that if any correct process does not obtain its item, then no process obtains any items from any other process, which is clearly unsustainable in the presence of two or more Byzantine processes. Indeed, one cannot prevent two Byzantine processes from privately exchanging their items. Our specification of fair exchange, on the contrary, considers the general case where more than two processes might be involved.

Most solutions to fair exchange rely on some kind of Trusted Third Party (TTP). A TTP is a process directly accessible to all processes. Fairness is thus trivially ensured by having processes sending their items to the TTP, which then forwards the items, if the terms of the exchange are fulfilled [BP90]. A TTP brings synchronism and control over terms of the exchange in order to ensure fairness but constitutes a bottleneck and a single point of failure. For this reason, various *optimistic* algorithm have been proposed that only involve the TTP when something goes wrong [ASW00, Mic03, BP90, BWW00]. Other approaches aim at relaxing the assumption made on its trustworthiness. In [FR97] for instance, Franklin and Reiter propose a solution using a *semi-trusted* third party that can misbehave on its own but does not conspire with either of the two participant processes. Departing from the traditional TTP-based approach, some authors proposed solutions based on fully decentralized tamperproof modules [AGGV05, GR06]. Both these solutions assume a network topology where the modules are embedded in their respective processes and cannot communicate directly with one another, however they differ in the power given to the tamperproof modules.

Impossibility Results. Besides proposing a specification or a solution, some authors also discuss the difficulty of fair exchange and propose impossibility results in various models. Oddly enough, it is difficult to find published work on impossibility of fair exchange other than technical reports. Mostly, impossibility results on fair exchange have to be inferred from other impossibilities on problems somehow related to fair exchange, which is far from being straightforward. In [PG99], fair exchange is measured against consensus, and an impossibility result on fair exchange in asynchronous models is shown by comparison with the FLP impossibility [FLP85]. In [EY80], fair exchange is shown to be impossible to solve *deterministically* in an asynchronous system with no Trusted Third Party (TTP). In another feasibility study [KGM95], complex exchanges are broken into

sub-exchanges – each relying on a different TTP – and represented as a graph. Reduction rules are then applied to the graph in order to demonstrate the feasibility of the exchange. This method also makes it possible to illustrate how closely exchange feasibility relies on trust. Another attempt to study what is inherently possible and impossible in safe exchange is proposed in [SW02], using game-theoretic solution concepts. However, the basic exchange model considered in this study only involves two parties, plus a Trusted Third Party. In this paper, we present an impossibility result stating that fair exchange cannot be solved in a synchronous model in the absence of some identified process that every other process can trust *a priori*. We then present another impossibility result in the context of a model with trusted processes.

Contributions and Roadmap. In Sect. 2, we present a general distributed system model with Byzantine failures and we give a fine-grained specification of fair exchange. We also prove that there is no solution to the problem in that model without a notion of trust. In Sect. 3, we modify our model to obtain an enriched model, which can be used as a framework to describe and compare solutions to the fair exchange problem.[2] In this model, we derive necessary conditions for the solvability of the problem. Finally, Sect. 4 revisits existing solutions in the light of our enriched model and points out the common essence of these solutions, while Sect. 5 concludes this paper by giving hints about ongoing and future work.

2 Fair Exchange in the Absence of Trust

We consider a distributed system consisting of a set Π of n processes, $\Pi = \{p_1, \ldots, p_n\}$. Processes are interconnected by some communication network and communicate by message passing. The system is *synchronous*: it exhibits *synchronous computation* and *synchronous communication*, i.e, there exists upper bounds on processing and communication delays. To help our reasoning, we also assume the existence of some global real time clock, whose tick range, noted T, is the set of natural numbers.[3]

Regarding the network topology, we merely assume that processes of Π form a connected graph. Links are reliable bidirectional communication channels, i.e., if both the sender and the receiver are correct, any message inserted in the channel is *eventually* delivered by the receiver, i.e., after some finite amount of time (termination property). A reliable channel also ensures that no message is delivered if it was not previously sent (no creation property).

Executions and Failure Patterns. We define the *execution* of algorithm A as a sequence of steps executed by processes of Π. In each step, a process has the opportunity to atomically perform all three following actions: (1) send a message, (2) receive a message and

[2] As discussed in [GR06], our enriched model can be implemented in practice via dedicated tamperproof chips, already available on the market today.

[3] This global clock is virtual in the sense that processes do not have access to it.

(3) update its local state.[4] Based on this definition, a *Byzantine process* is one that deviates from A in any sort of way, so a Byzantine process is Byzantine against a specific algorithm A. It is a known result that Byzantine failures can only be defined with respect to some algorithm [DGG05]. A *Byzantine failure pattern* f is then defined as a function of T to 2^{Π} where $f(t)$ denotes a set of Byzantine processes that have deviated from A through time t. In a way, a failure pattern f can be seen as a projection of all process failures during some execution of A. Once a process starts misbehaving, it cannot return to being considered correct, i.e., $f(t) \subseteq f(t+1)$. We also define F as the set of all possible failure patterns of A, so $f \in F$.[5]

Let $\mathsf{Byz}(f) = \bigcup_{t \in T} f(t)$ and $\mathsf{Cor}(f) = \Pi - \mathsf{Byz}(f)$ denote respectively the set of Byzantine processes in f and the set of correct processes in f. We define the set F_b of all failure patterns where no more than b processes are Byzantine. More formally, F_b is the largest subset of F such that, for any failure pattern $f \in F_b$, $|\mathsf{Byz}(f)| \leq b$, with $0 \leq b \leq n$:

$$F_b = \{f \in F \ : \ |\mathsf{Byz}(f)| \leq b\} \ with \ 0 \leq b \leq n \ .$$

Note that b is bounded by n, the number of processes in Π. From this definition, we know that b is the maximum number of Byzantine processes in any failure pattern of F_b and that $F_n = F$. Finally, we define the set $F_f^{\sim}$ of all failure patterns producing the same set of Byzantine processes than f. More formally, given some failure pattern f, $F_f^{\sim}$ is the largest subset of F such that, for any failure pattern f' of $F_f^{\sim}$, $\mathsf{Byz}(f') = \mathsf{Byz}(f)$:

$$F_f^{\sim} = \{f' \in F \ : \ \mathsf{Byz}(f') = \mathsf{Byz}(f)\} \ .$$

2.1 The Fair Exchange Problem

The fair exchange problem consists in a group of processes trying to exchange digital items in a fair manner. The difficulty of the problem resides in achieving fairness. Intuitively fairness means that, if one process obtains the desired digital item, then all processes involved in the exchange should also obtain their desired digital item. The assumption is made that each process knows both the set Π of processes participating in the fair exchange and the terms of the exchange. The terms of the exchange are defined by a set D of item descriptions, $D = \{d_1, \ldots, d_n\}$, and a set Ω of pairs of processes (p_i, p_j). Description d_i is the description of the item expected by process p_i. Furthermore d_i is unique, so if $i \neq j$, then $d_i \neq d_j$. A pair (p_i, p_j) defines the receiver p_j of the item offered by p_i. Elements of Ω are defined such that p_j is the image of p_i through a bijective map (or permutation) of Π, with $i \neq j$. Finally, let M denote the set of digital items m_i actually offered by process p_i during an execution of fair exchange, $M = \{m_1, \ldots, m_n\}$. Note that, accordingly, for each description in D there does not necessarily exist a corresponding item in M, since M includes items that might have been offered by Byzantine processes.

[4] At each step, the process can of course choose to skip any of these actions, e.g., if it has nothing to send.

[5] This way of modeling executions and failures is fairly classical [DGG05, CT96].

Fair Exchange as Service. Fair exchange can be seen as a service allowing processes to exchange digital items in a fair manner. Each process offers an item in exchange for a counterpart of which it has the description. The exchange is concluded when every process releases the desired counterpart or the abort item φ, meaning that the exchange has aborted. To achieve this, the service offers the two primitives described below.

> **offer**(m_i, p_j) – Enables the process p_i to initiate its participation in the exchange with processes of Π by offering item m_i to p_j, in exchange for the item matching description d_i, with d_i and Π known a priori.
>
> **release**(x) – Informs the process that the exchange completed and works as a call-back. Process p_i receives item x, which is either the item matching d_i or the abort item φ.

Note that, at the end of an exchange, we say that p_i *releases* an item, meaning that the service calls back the *release* operation of p_i. This convention is similar to the one used for classical *deliver* primitives, e.g., in reliable broadcast [HT93].

Fair Exchange Properties. We now specify the formal properties of the fair exchange problem. While several other specifications exist in the literature [AGGV05, AV03, PG99], our specification differs in that it separates safety and liveness via *fine-grained properties*. Such elemental properties then allow us to better reason on the impossibility to solve fair exchange in various models.

Validity. If a correct process p_i releases an item x, then either $x \in M$ and x matches d_i, or x is the abort item φ.

Uniqueness. No correct process releases more than once.

Non-triviality. If all processes are correct, no process releases the abort item φ.

Termination. Every correct process *eventually* releases an item.

Integrity. No process p_j releases an item m_i, with process p_i correct, if m_i matches description d_k of some correct process p_k, with $p_k \neq p_j$.

Fairness. If any process p_i releases an item m_j matching description d_i, with p_i or p_j correct, then every correct process p_k releases an item matching description d_k.

Among these six properties, the last two, *integrity* and *fairness*, are specific to the problem of fair exchange and define precisely the possible outcomes of fair exchange algorithms. Other specifications of fair exchange usually rely on a single property to capture the notion of fairness [AGGV05, ASW00, PG99]. However we argue that if those specifications are suitable for cases where $n = 2$, they are impossible to satisfy in models allowing more than one Byzantine process. In [AGGV05], for example, the *fairness* property requires that if any correct process does not obtain its item, then no process obtains any items from any other process. This is clearly unsustainable in the presence of two or more Byzantine

processes because one cannot prevent two Byzantine processes from conspiring in order for one of them to obtain the item of the second one. A simple but flawed fix would be to modify the definition as follows: if any correct process does not obtain its item, then no process obtains any items from any *correct* process. If it first seems correct, this definition of *fairness* now allows a correct process to obtain the item of a Byzantine process, even if other correct processes do not obtain anything.

Coming back to our specification, *integrity* ensures that no process obtains an item offered by a correct process and matching the description of a correct process. Notice that this does not prevent a Byzantine process from illicitly obtaining the item destined to or offered by some other Byzantine process, since such a behavior cannot be prevented and does not prejudice any correct process. Then, *fairness* guarantees that if any process obtains its desired item offered by some other process, with at least one of them being correct, then every correct process also obtains its desired item. In other words this property prevents a Byzantine process from taking advantage of a correct process but does not protect other Byzantine processes. More trivially, it also ensures that no correct process takes advantage of any process.

2.2 Impossibility Result

In [EY80], fair exchange is proved to have no solution in an asynchronous model prone to Byzantine failures. In the following, we show that the exchange problem has no deterministic solution even in the context of a perfectly synchronous model, if no complementary trust hypothesis is made. This is the subject of Theorem 1 hereafter.

In order to prove this, we define the notion of *trusted process* as a process that is known to be correct a priori by all other processes. In our model, no such assumption is made about any process of Π, so each process is potentially correct or Byzantine. Now, for sake of simplicity and without loss of generality, we assume that an item is indivisible, i.e., it cannot be sent in pieces. Note that this assumption does not reduce the scope of the impossibility, since allowing an item to be broken into pieces, e.g., using techniques from [Sha79], would result in facing the same fair exchange problem for sending the last piece of item. For the same reason, we assume the item is not encrypted, since having to later exchange the keys in a fair manner in order for the processes to decipher the items would again let us face the same fair exchange problem.

Theorem 1 *In the context of a synchronous model with Byzantine failures, there is no* deterministic *solution to the fair exchange problem, if there is no trusted process, even in the presence of only a single Byzantine process.*

Proof. The proof is by contradiction.
Assume that some algorithm A solves fair exchange and that there are no trusted process. Consider an execution E of A in which all processes are correct. From the *non-triviality*, *termination* and *validity* properties of FE, in E, every process releases its desired item, and in particular, some process p_i releases item m_j, with m_j matching description d_i and

$(p_j, p_i) \in \Omega$, and some process p_k releases item m_i, with m_i matching description d_k and $(p_i, p_k) \in \Omega$. Now since no process can be trusted and Byzantine processes cannot be detected, in any execution, no process other than p_i and p_k may hold item m_i. So we know that in a previous step of E, p_k receives m_i from p_i. We now consider the two following cases: either (a) p_i sends m_i after receiving m_j or (b) p_i sends m_i before receiving m_j.

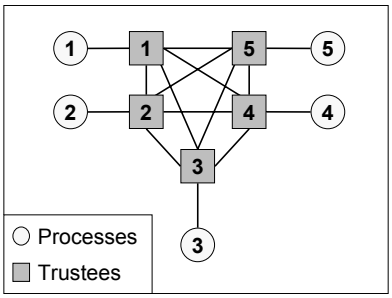
Case (a): Since there is no trusted process, if p_i sends m_i after receiving m_j, we can derive an execution E', similar to E, in which p_i is Byzantine and deviates from A after receiving m_j by omitting to send m_i and by releasing m_j. Since no process is trusted, in E', no process other than p_i holds m_i. So from the *no creation* property of reliable channels, p_k never receives and thus never releases m_i. To satisfy the *validity* and *termination* properties, in E', p_k releases φ but thus violates *fairness*. So, in E, p_i does not send m_i after receiving m_j. Furthermore, this is true for every process, so from the definition of Ω and by circular reasoning, in E, all items are sent at the same time. This now leaves us with case (b).

Case (b): We know that, in E, p_i sends m_i before receiving m_j and that all items are sent at the same time. Now, since there is no trusted process, we can derive an execution E'', similar to E, in which p_j is Byzantine and deviates from A by omitting to send m_j. Since all items are sent at the same time, p_j receives and releases some item m_x matching d_j. Since no process is trusted, in E'', no process other than p_j holds m_j. So from the *no creation* property of reliable channels, p_i never receives and thus never releases m_j. To satisfy the *validity* and *termination* properties, in E'', p_i releases φ but thus violates *fairness*. So finally, Algorithm A does not solve fair exchange. A contradiction. □

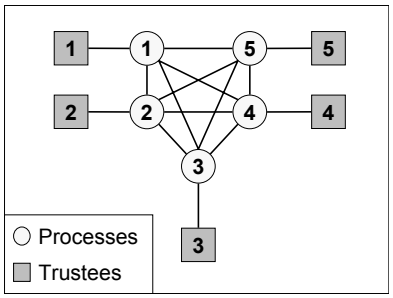
3 Fair Exchange in the Presence of Trust

As described in Sect. 2, we consider a distributed system consisting of a set Π of n *processes*, $\Pi = \{p_1, \ldots, p_n\}$. Processes of Π are also called *participants*. We complete our model with a set Π' of n trusted processes, $\Pi' = \{p'_1, \ldots, p'_n\}$, i.e., a *trusted process* is known a priori to be correct by all other processes. Processes of Π' are called *trustees* or *trusted* processes. Note that, hereafter, to avoid confusion, the term *process* will only be used to describe participants, unless clearly mentioned otherwise. Furthermore, each trustee p'_i is matched in a one-to-one relationship with the corresponding participant p_i and is directly connected to it. Π^+ is then the set of all $2n$ participants and trustees, i.e., $\Pi^+ = \Pi \cup \Pi'$. As illustrated in Fig. 1, processes and trustees are interconnected by a communication network and no additional assumption is made about the network topology, other than the fact that it is a connected graph. Participants are process actually taking part in the exchange by offering and demanding items, and they may exhibit Byzantine behaviors. Trustees on the contrary are *trusted processes* that have no direct interest in the exchange. Intuitively, the role of a trustee is to decide when it is appropriate to provide its matched participant with its expected item.

As we shall see in Sect. 4, this model allows to describe and compare various solutions



(a) TTP-based topology.



(b) Guardian angels topology.

Figure 1: Examples of valid topologies as defined in our model.

proposed in the literature. Simpler topologies, using less trustees and hence some of them matched with more than one participant, can easily be transformed to fit our model. Indeed, any trustee matched with several participants can be represented by a cluster of as many trustees, fully interconnected, and with each trustee matched with one participant. For example, in the case of the Trusted Third Party (TTP), the TTP can be transformed into a cluster of n fully interconnected trustees.

3.1 Impossibility Result in the New Model

In Sect. 2.2, we have shown that without some notion of trust, one cannot solve fair exchange. On the other hand, we know that a cluster of trustees acting as a TTP yields a solution. However, depending on the network topology, the presence of trustees is not sufficient. In the following, we show that in the context of the enriched model, a necessary condition to solve fair exchange is to have every correct participant reliably connected to a majority of trustees. This is the subject of Theorem 2 given hereafter and which relies on Lemma 1.

Beforehand, we need to define the notion of *reliable path* as follows. Let p_i and p_j be two correct processes of Π_+ (either processes or trustees). We say that p_i and p_j are connected through a *reliable path*, if there exists at least one path between p_i and p_j such that, either p_i and p_j are directly connected or p_i is directly connected to a *correct* process $p_k \in \Pi_+$ and there exists a *reliable path* between p_k and p_j. Furthermore, given a process $p_i \in \Pi$ (participants) and a failure pattern $f \in F$, we define $C^f_{p_i}$ as the largest subset of Π' (trustees) such that any trustee p'_j of $C^f_{p_i}$ is connected through a reliable path to p_i. Finally, we define the *reachable majority* condition as the condition under which, for any correct process $p_i \in \Pi$ and any failure pattern $f \in F_b$, $|C^f_{p_i}| > \frac{n}{2}$, even in the presence of up to b Byzantine processes. Intuitively, this means that, even in the worst case scenario, p_i is connected through a reliable path to a majority of trustees, i.e., $\lfloor \frac{n}{2} + 1 \rfloor$. Note that the *reachable majority* condition is a connectivity condition that translates into a maximum

number of Byzantine processes that a certain model of network topology is capable of sustaining. This point is further illustrated in Sect. 4.

Lemma 1 *If an algorithm A solves fair exchange with up to b Byzantine processes, then for any failure pattern $f \in F_b$, there exists an execution associated with a failure pattern $f' \in F_f^{\sim}$ such that every process of $\mathsf{Cor}(f')$ releases its expected item.*

Proof. Consider an execution E of A in which all processes are correct, from the *non-triviality, termination* and *validity* properties, in E every process releases its correct item. Now consider any failure pattern $f \in F_b$. From E, we derive an execution E' associated with a failure pattern $f' \in F_f^{\sim}$, i.e., $\mathsf{Byz}(f') = \mathsf{Byz}(f)$, such that, in E', every process of $\mathsf{Byz}(f')$ deviates from A just before releasing its item, e.g., by crashing. Since E' is indistinguishable from E for all correct processes, every process in $\mathsf{Cor}(f')$ releases its correct item.

Theorem 2 *In the context of a synchronous model with trustees and Byzantine failures, there is no deterministic solution to the fair exchange problem, if the reachable majority condition is not satisfied, even in the presence of a single Byzantine process, i.e., $b = 1$.*

Proof. The proof is by contradiction.
Assume that some algorithm A solves fair exchange and that the *reachable majority* condition is not satisfied, i.e., there is some correct process $p_i \in \Pi$ and some failure pattern $f \in F_b$ for which $|C_{p_i}^f| \leq \frac{n}{2}$, even for $b = 1$. From Lemma 1, we know that there exists an execution E' associated to a failure pattern $f' \in F_f^{\sim}$, such that every process in $\mathsf{Cor}(f')$ releases its expected item. Hence, in E', p_i receives its expected item, e.g., m_j, from its trustee p_i'. We now have to consider the two following cases: (a) the transmission of m_j from p_i' to p_i depends on the reception by p_i' of some message x sent by some trustee $p_j' \in \Pi' - C_{p_i}^f$, and (b) the transmission of m_j from p_i' to p_i is independent of the reception by p_i' of any message sent by any trustee $p_j' \in \Pi' - C_{p_i}^f$.[6]

Case (a): We can derive an execution E'', similar to E', where message x is blocked by some Byzantine process along the *unreliable* path between p_i' and p_j', as well as any following messages. Since E'' is indistinguishable from E' for any process unreliably connected to p_i', i.e., processes associated with trustees of $\Pi' - C_{p_i}^f$, these processes thus release their expected item in E''. However, in E'', p_i' never receives x. Since the transmission of m_j depends on the reception of x, p_i' never sends m_j to p_i. To satisfy the *validity* and *termination* properties, p_i releases φ but thus violates *fairness*. This leaves us with case (b).

Case (b): We can derive an execution E''', similar to E', in which some Byzantine process p_k fails to send the expected item to some trustee $p_j' \in \Pi' - C_{p_i}^f$, with p_j correct and $(p_k, p_j) \in \Omega$. Since the transmission of m_j from p_i' to p_i is independent from the reception of any message sent by any trustee of $\Pi' - C_{p_i}^f$ (included p_j'), for p_i' and

[6] Note that by definition, $C_{p_i}^f = C_{p_i}^{f'}$, for any failure f and f' such that $f' \in F_f^{\sim}$.

p_i, executions E''' and E' are indistinguishable. So, in E''', p_i releases its expected item. However, since p'_j never receives the expected item of p_j, neither does p_j. To satisfy the *validity* and *termination* properties, p_j eventually releases φ but thus violates *fairness*. So algorithm A does not solve fair exchange. A contradiction. □

4 Revisiting Existing Solutions

Trusted Third Party (TTP). Several algorithms described in the literature rely on the TTP paradigm. The simplest TTP-based algorithm consists in having processes send there items to a centralized trustee, the TTP. The TTP verifies that the terms of the exchange are respected and, if this is the case, forwards the items. These TTP-based solutions naturally fit in our enriched model. Our model uses n trustees compared to a unique one in TTP-based solutions. Mapping the TTP model to our model is done by having all n trustees jointly play the role of the TTP by forming a cluster of fully interconnected trustees. The network topology of this solution is such that each process is directly connected to one distinct trustee of the cluster. It is then fairly obvious that the TTP topology is so secure that the reachable majority condition is satisfied for any number of Byzantine processes. Figure 1(a) shows an example of the TTP topology with five processes.

Guardian Angels. In [AGGV05, AV03], guardian angels are defined as tamperproof security devices that are considered correct. Processes are fully interconnected by a communication network with bidirectional reliable channels. There are n guardian angels but each of them is only connected to one process. In other words, each process can directly communicate with its assigned security device but needs to go through some untrusted process to communicate with other security device. Intuitively, in order to solve fair exchange, each item is encrypted and sent to the security device of the corresponding process, i.e., the process expecting the item. Security devices then enter a synchronization protocol, which upon success enables the devices to send the items to the processes. The assumption is made that the security devices are able to check the validity of the items and to encrypt messages. In a model with no upper bound on the number of Byzantine processes, the solution given solves fair exchange with a certain probability. The authors of [AGGV05, AV03] also show that, even in a synchronous model with security devices, no deterministic algorithm solves fair exchange without an honest majority, i.e., without $b < \frac{n}{2}$.

The guardian angels approach fits our enriched model by having each of the n trustees represent one distinct security device. The network topology being symmetric, we can limit our reasoning to one process. Theorem 2 tells us that if any process p is not connected through a trusted path to a majority of trustees, there is no solution to the problem. Since each trustee is behind a distinct process, which is potentially Byzantine, there must be a majority of correct processes. From Theorem 2, we can then say that the guardian angels approach cannot deterministically solve fair exchange, if there is not a majority of correct processes. As one could expect, this result concurs with the result found in [AGGV05].

Nonetheless, a very interesting feature of the approach proposed in [AGGV05] lies in its ability to gracefully degrade its quality of service from deterministic fairness to probabilistic fairness. Figure 1(b) shows an example of the guardian angels topology with five processes.

Fair Exchange in the Pervaho Middleware. In [GR06], we propose a modular algorithm solving fair exchange in the context of the Pervaho middleware [EGH05]. This solution follows the same topology as with guardian angels. This algorithm relies on the use of two key building blocks: a tamperproof secure box module and a module solving the well-known Byzantine agreement problem. The secure boxes are not connected directly with each other and they are only needed in key steps of the algorithm, contrary to the guardian angels approach. Since the network topology is identical to the Guardian angels topology, our enriched model yields the same results in both cases.

5 Concluding Remarks

In this paper, we proposed a formal description of the fair exchange problem which clearly separates liveness and safety, thanks to fine-grained properties. We proved that fair exchange cannot be solved in a synchronous model with Byzantine failures without at least one identified correct process. Based on this result, and by enriching our previous model with identified correct processes (trustees), we defined a generic model for describing a wide range of solutions to the fair exchange problem. We then gave a necessary condition for solving fair exchange in this model. Intuitively, this condition states that each correct process must be reliably connected to a majority of trustees.

Acknowledgements This research is partly funded by the Swiss National Science Foundation, in the context of Project number 200021-104488.

References

[AGGV05] G. Avoine, F. Gärtner, R. Guerraoui, and M. Vukolic. Gracefully Degrading Fair Exchange with Security Modules (Extended Abstract). In *In Proceedings of the 5th European Dependable Computing Conference - EDCC 2005*, 2005.

[ASW00] N. Asokan, V. Shoup, and M. Waidner. Optimistic Fair Exchange of Digital Signatures. *IEEE Journal on Selected Area in Communications*, 18:593–610, 2000.

[Ate99] G. Ateniese. Efficient verifiable encryption (and fair exchange) of digital signatures. In *CCS '99: Proceedings of the 6th ACM conference on Computer and communications security*, pages 138–146, New York, NY, USA, 1999. ACM Press.

[AV03] G. Avoine and S. Vaudenay. Fair Exchange with Guardian Angels. Technical report, Swiss Federal Institute of Technology (EPFL), 2003.

[BP90] H. Bürk and A. Pfitzmann. Value Exchange Systems Enabling Security and Unobservability. *Computers & Security*, 9(9):715–721, 1990.

[BWW00] B. Baum-Waidner and M. Waidner. Round-Optimal and Abuse Free Optimistic Multiparty Contract Signing. In *Automata, Languages and Programming*, number 1853 in Lecture Notes in Computer Science (LNCS), pages 524–535. Springer, 2000.

[CT96] T. D. Chandra and S. Toueg. Unreliable failure detectors for reliable distributed systems. *Journal of the ACM*, 43(2):225–267, 1996.

[DGG05] A. Doudou, B. Garbinato, and R. Guerraoui. *Dependable Computing Systems: Paradigms, Performance Issues, and Applications*, chapter Tolerating Arbitrary Failures with State Machine Replication, pages 27–56. Wiley, 2005.

[EGH05] P. Eugster, B. Garbinato, and A. Holzer. Location-based Publish/Subscribe. In *Proceedings of the 4th IEEE International Symposium on Network Computing and Applications (IEEE NCA05)*, Cambridge (MA), July 2005.

[EY80] S. Even and Y. Yacobi. Relations Among Public Key Signature Systems. Technical report, Technion - Israel Institute of Technology, 1980.

[FLP85] M. Fischer, N. Lynch, and M. Paterson. Impossibility of Distributed Consensus with One Faulty Process. *J. ACM*, 32:374–382, April 1985.

[FR97] M.K. Franklin and M.K. Reiter. Fair exchange with a semi-trusted third party (extended abstract). In *CCS '97: Proceedings of the 4th ACM conference on Computer and communications security*, pages 1–5, New York, NY, USA, 1997. ACM Press.

[GR06] B. Garbinato and I. Rickebusch. A Modular Solution to Fair Exchange for Peer-to-Peer Middleware. Technical Report DOP-20060410, University of Lausanne, DOP Lab, 2006.

[HT93] V. Hadzilacos and S. Toueg. Fault-tolerant broadcasts and related problems. pages 97–145, 1993.

[KGM95] S. Ketchpel and H. García-Molina. Making Trust Explicit in Distributed Commerce Transactions. In *Proceedings of the International Conference on Distributed Computing Systems*, 1995.

[MGK02] O. Markowitch, D. Gollmann, and S. Kremer. On Fairness in Exchange Protocols. In *Proceedings of the 5th International Conference Information Security and Cryptology (ICISC 2002)*, volume 2587 of *Lecture Notes in Computer Science*, pages 451–464. Springer, November 2002.

[Mic03] S. Micali. Simple and fast optimistic protocols for fair electronic exchange. In *PODC '03: Proceedings of the twenty-second annual symposium on Principles of distributed computing*, pages 12–19, New York, NY, USA, 2003. ACM Press.

[PG99] H. Pagnia and F. Gärtner. On the Impossibility of Fair Exchange without a Trusted Third Party. Technical report, Swiss Federal Institute of Technology (EPFL), 1999.

[RRN05] I. Ray, I. Ray, and N. Natarajan. An anonymous and failure resilient fair-exchange e-commerce protocol. *Decision Support Systems*, 39(3):267–292, 2005.

[Sha79] Adi Shamir. How to Share a Secret. *Commun. ACM*, 22(11):612–613, 1979.

[SW02] T. Sandholm and X. Wang. (Im)possibility of safe exchange mechanism design. In *Eighteenth national conference on Artificial intelligence*, pages 338–344, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence.

On the Demand for E-Services by Health Communities

[1] Achim Dannecker, [2] Ulrike Lechner

[1] Fachhochschule Nordwestschweiz
achim.dannecker@fhnw.ch
[2] Universität der Bundeswehr München
ulrike.lechner@unibw.de

Abstract: Virtual communities of patients provide health-related information and mutual support for members. This paper presents an analysis of the demand of virtual communities of patients for novel electronic services. Results include the success factors of virtual communities of patients, the demand for electronic services for new and experienced community members and the demand for electronic services for short-term affected and long-term affected patients.

1 Introduction

Virtual communities of patients (also referred to as Virtual Communities in Health Care – VCHC) provide information and mutual support for their members. VCHC provide information concerning diseases, treatments or new research results in the area of interest. Information shared among members includes experience reports on how the disease was diagnosed, how it affects the daily life and how to cope with it. Experiences with medical institutions or treatments are sometimes topic in VCHC.

Our research on success factors and e-services for VCHC is motivated twofold. First, we think that strong VCHC benefit patients, as patients look for reliable information, need mutual support and orientation. Second, we think that novel e-services are able to strengthen VCHC by attracting more members, by motivating them to contribute more and by collecting more and different kinds of information.

In our research of VCHC we established a model to capture the success factors of VCHC and the mutual dependencies of these success factors [DL06c; DL06a]. In this paper, we present the results of an analysis of the success factors of subgroups of users of those communities. The objective of community management is to attract members of the target group and convert them to long-time and active users. We compare new and long-term members as well as newly affected and long-term affected members. Our goal is to identify success factors, i.e. design issues for e-services and communication management, and derive adequate e-services. Our indicator for success is active contribution by members. Our guiding hypothesis is that communities should attract members early and keep them for a long time as active members. A large number of active members with diverse profile benefit the community to perform better in providing information and mutual support.

The paper is organized as follows. We discuss the state of the art (Sect. 2), present the research approach (Sect. 3, 4), success factors (Sect. 5) and discuss then success factors for subgroups (Sect. 6, 7 and 8). A discussion concludes the paper (Sect. 9).

2 State of the art

The health care is in a process of reorganization and people use the internet to find health related information, manage their personal health record via the internet, get information about health care services and regulations that govern them [Go05; HT05; PUC06]. The integration of the information available via the internet with information people received by physicians is an important factor in health-related decisions [PUC06].

Self-help organizations and self-help groups are the "traditional", off-line form for people that are affected by a disease to exchange experiences. Self-help organizations typically inform members about all aspects of a disease and they act as representative (patient unions). Janke et al. postulate that patients in self-help organizations are better informed on their disease than patients not attached to self-help organizations [JKG05]. Borgaonkar et al. show that providing disease-related information only to patients worsens health-related quality of life (HRQOL) in inflammatory bowel disease [BTD02]. Interviews with operators of self help organizations confirmed that providing information only and constantly reminding on the patient's illness (e.g. through mailing, brochures or newsletters) is counterproductive as it leads to a decrease of HRQOL and, frequently, to the cancellation of the self-help organization membership. Kennedy et al. [KRN03] emphasizes "…patients given a patient-developed guidebook of self-management skills experienced significantly improved HRQOL". This indicates that e-services that provide information only are not sufficient to benefit patients. A study to the effectiveness of Australian Medical Portals (medical information only) shows that users find them useful[MF06]. This indicates demand for medical information.

Participants in self-help groups meet on a regular basis mainly to exchange information related to a disease. Self-help groups have two main goals: mutual support and exchange of information [Bo04]. Participants benefit from experiencing that they are not the only one affected by a disease or the only ones disease related problems in the daily life. Topics discussed in self-help groups include medics, clinical institutions, rehabilitation centres, treatments, medicaments, research and participation in clinical studies. Note that this variety of topics is not found in online self-help groups [DL04].

Online communities or at least forums are part of online-offerings of self-help groups. An interview partner (community manager) describes the typical situation and that little has changed in the past years – as the forum has the size of approximately 100 regular, but mostly not long time visitors. Similar "newbie" questions are being asked over and over again, with the same (eventually dangerous) theories about origin of the disease and possible cures being discussed in a not so profound way, with newly diagnosed people asking one or two urgent questions and leaving again. Only relatively few people stay and profound discussions take place only partly online in the forum. This self-help organization is proud of the collection of relevant medical information it provides.

3 Research method

The objective of our research is to find out what services virtual communities of patients help to perform better in providing information and mutual support.

A questionnaire was developed on the basis of a study of web communities [DL04], interview with self-help groups leaders, an empirical study of Leimeister et al. [Le04b], and a literature review. Two versions of the questionnaire were created: one for the members and one for the operators of the VCHC, i.e. for the persons that maintain a community and that provide the platform. Note that operators are typically also members of VCHC. Ten VCHC (we already with their operators) were contacted to send the community operators a first version of the questionnaire for a review. We identified VCHC in the German speaking context based on an Internet research done on Yahoo and Google. Cross linked sites in the context of VCHC were also taken into consideration. 250 VCHC in the German speaking context were identified. VCHC with less than 50 members and communities with the most recent contribution older than one year were discarded. This led to 117 VCHC from which 73 (63%) were chosen randomly. The ten VCHC to which the first version of the questionnaire was sent were added to the sample. The questionnaire was sent to the VCHC operators with the request to support the study, to provide a link to the questionnaire to VCHC members, and to fill out the operator questionnaire. The questionnaire was available for three weeks in June 2005. All empty and duplicate entries were eliminated and 295 entries by members and 21 entries by operators form the sample. For interpretation and validation of quantitative results, interviews with operators and members as well as presentations with the management of two self-help organizations have been done.

The information that we requested from study participants included the usual demographic information with age, gender, time online, number of VCHC the study participant is a member of, and for how long she is affected by the disease.

	VCHC	No participants	Ratio
1	rheuma-online.de (rheumatism)	50	11,74%
2	fibromyalgie-aktuell.de (pain patients)	35	8,22%
3	dccv.de (morbus crohn / colitis ulcerosa)	31	7,28%
4	croehnchen-klub.de (morbus crohn / colitis ulcerosa)	24	5,59%
5	sylvia.at (morbus crohn / colitis ulcerosa)	15	3,50%

Table 1: Top 5 of the VCHC according to the number of study participants

People participating in this study are active in a total of 145 different VCHC. The „Top Ten“ of the VCHC according to the number study participants account for about 50% of the participants, the “Top 5” for 38%. 16 communities account for two study participants and 100 communities for one participant.

More than 95% of the study participants are affected by a chronic disease. Most participants suffer from rheumatism (incl. fibromyalgie) (20%), followed by morbus crohn (17%), cancer (11%), diabetes (6%) and tinnitus (5%). Note that the majority of participants are affected by a chronic disease which allows an analysis on an homogenous sample.

The research sample (Nmembers = 295, Noperators = 21) consists of 69% female and 31% male participants.

Type	Selected Questions/ Statements
Social/Altruistic issues (S)	That people understand you with your problems Assistance for new members by experienced members The feeling to be in a place at home
Technical issues (T)	Stability of the website Fast reaction time of the website
Medical Content (MC)	Offering up-to-date information and information about relevant clinical trials Push of research within the field of your disease
Medical Quality Assurance (MQA)	Contributions of members/operators for members e.g. Active quality assurance of the content of the community done by members Statements of the community about medics Possibility of discussions about alternative methods of treatment Moderation of member contributions by the operator

Table 2: Types of success factors / Topics and Questions

Particular for an online study are an average age of above 40 years and a high percentage of women in the categories members and operators. The participants contribute more than once a week in average. In average, the members are affected by their illness for nearly 10 years with a VCHC membership of twenty-eight months. Note that the high percentage of woman is typical for health related online communities in general. Note that the illnesses of most patients inflict both men and woman and that this does not explain the high percentage of women in this study. The high average age of the study participants coincides with occurrence of chronic illnesses typically later in life.Let us give a brief impression of the questions in the study as summarized in Tab. 2.. We had questions concerning the social network of a community following the literature on virtual communities as discussed e.g. in [DL06c; DL06a], the technical aspects of services and infrastructure, the attitude towards medical content in communities and the support of medical research, the attitude towards quality management in the health care sector done by the community.

5 Success Factors

Let us analyze the success factors (i.e. what is relevant to members) of VCHC. The 7 most important success factors are ordered according to the member perspective in Fig. 1. The figure contains an abbreviated version of the original question and the arithmetic mean of the answers. The sample was tested due to normal distribution using an exact "Kolmogorov-Smirnov-Test". All results are significant with $p<.001$.

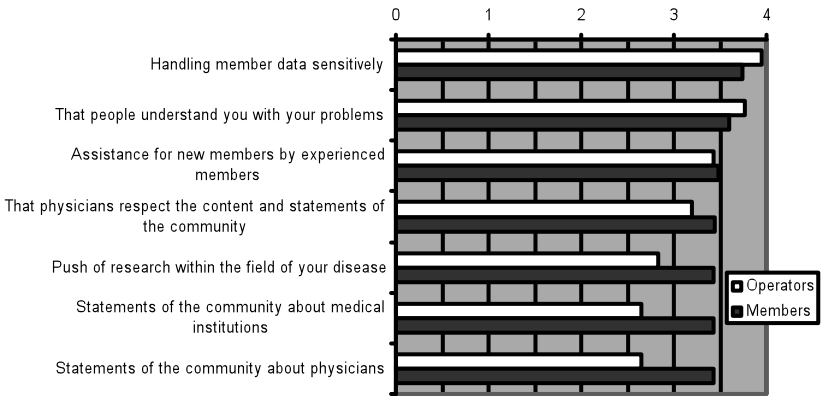


Fig. 1. Top 7 success factors ordered by the member views (highly agree 1 – highly disagree 5)

Handling member data is the most important factor for the members and it is considered even more important by operators. The feeling of being understood with problems within the VCHC is next, followed by the assistance of new members by experienced members. Note that success factors concerning medical issues and medical quality issues are important and of special interest to the community. These factors are statements of the community about medical institutions and physicians and support of medical research.

The difference between operators and members occurs in aspects concerning the medical information a community contributes (that physicians respect the content available in communities, statements of the community about medical institutions and physicians and offering up-to-date information about medical trials). This is reflected by the current situation as VCHC hardly provide e-services for members to contribute experiences with physicians or treatments. This difference is explained in interviews with operators: Operators are concerned whether such kinds of contributions would be feasible or whether members would appreciate such kind of e-services.

6 Target Group Specific Demand

We assume that VCHC can be effective in communicating authentic and important disease related information and providing mutual support. Therefore it is desirable to reach patients that just have been diagnosed and are affected for a short time by the disease (short-term affected) as well as patients that have been affected for a long time (long-term affected). Our guiding hypothesis is that it desirable to attract visitors and to keep members for a long time. We look into the differences in demand for e-services between community newbies and experienced community members and between newly affected and long-term affected patients to find out which e-services attract new members or newly diagnosed patients and which services are relevant for experienced members or long term afflicted patients. First, we analyze the relation between the time of membership in VCHC and the time the members are affected by their disease.

		Affected since (years)							
		< 1y	< 2y	< 3y	< 5y	< 10y	< 20y	>= 20y	Total
Member since (months)	< 3m	4			1	3	4	3	15
	< 6m	7	2	3	3	5	7	1	28
	< 9m	4	3	1		2	2		12
	< 12m	7	8	3	6	8	5	2	39
	< 18m	1	6	5		1	3	2	18
	< 24m		6	8	9	9	12	8	52
	< 36m			14	7	13	12	5	51
	>= 36m			2	12	20	11	11	56
Total		23	25	36	38	61	56	32	277

Table 3.Time of membership vs. time affected by disease

Note that our study indicates no significant correlation between the time the people are affected by a disease and the time the people are member of a community (bivariate correlation based on Pearson .043, significant at p=.480). Tab. 3 shows the distribution of the members regarding the time people are affected by their disease and the time of membership.

VCHC are a rather new phenomenon compared to the time people are possibly affected by a chronic disease. We distinguish between the timeframe regarding the time people are affected by their disease (less than 1 year…more than 20 years) and the time people are a member of the VCHC (less than 3 months…more than 36 months). We observe (Tab. 3) that 9 study participants were longer affected that 5 years but are member of a VCHC for less than three months. 28 study participants are less than 6 months member of a VCHC and 21 of those 28 members are affected for more than 5 years. In the category "members longer than 3 years" (36 months) almost all study participants are affected by the disease much longer. Note, that this lack of relation between length of community membership and time people are affected is an indicator that VCHC do not attract newly diagnosed patients very well. Let us discuss whether communities are able to bind members and transform visitors to members. A lot of study participants are member for either a relatively short time or for a rather long time (longer than 24 months). The increase of respondents from members <3 months to members <6 months is an indicator that new members stay for some time. The results for study participants with memberships longer than 6 months and less than 24 months are inconclusive. This is an indicator that new members stay for a while, but that some members leave. Long-term members seem to be quite loyal. This indicates that VCHC bind active members.

7 New vs. experienced members

We assume that the interests and needs of members that are new to a community typically differ from the ones of long-time members. Keeping new members, activating passive, non contributing members or keeping active members active are considered important goals in community management [HA97; Pr00].

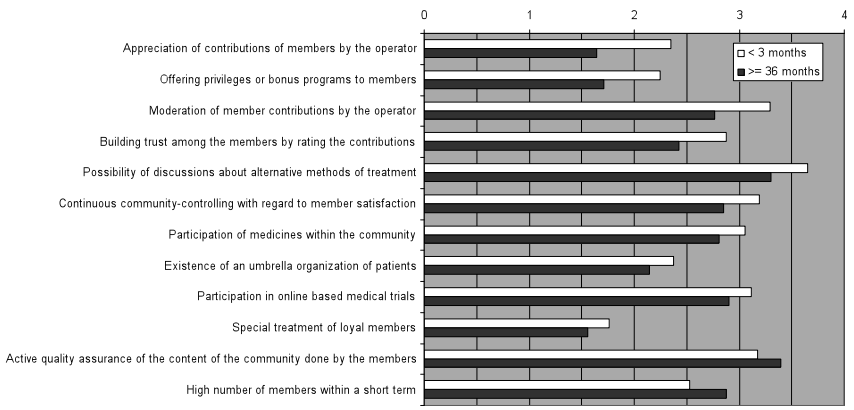


Fig. 2. Main differences in importance to success factors by the view of newbies and experienced members (sorted by difference >= .2) (highly disagree 0 – highly agree 4)

To determine whether there are pair wise associations for the sets of normally distributed variables a “Bivariate Correlations” examination based on “Pearson“ was executed. All correlations w.r.t. the time people are member of a VCHC and the success factors are significant at $p < .001$. In Fig. 2, the main difference (>= .2) in success factors between the view of newbies (member less than 3 months) and experienced members (member more than 36 months) is depicted.

The role of an operator, moderator and medical experts and an umbrella organization is more important for newbies than for experienced members according to the results presented in Fig. 2. Experienced members seem to have more trust in community and most likely are more aware of the self-organization processes and the power of the community. Newbies rely more on formal qualifications (physicians), distinctive roles (operator, moderator) and formal processes (medical trials). Newbies are interested in discussions of alternative treatments and in rankings of contributions. Interviews confirmed that information about alternative treatments attracts visitors.

Experienced members are more interested in other community members (meeting community members offline, number of community members), usability of the website (intuitive user guidance) and in quality assurance done by community members (Active quality assurance of the content of the community done by the members).

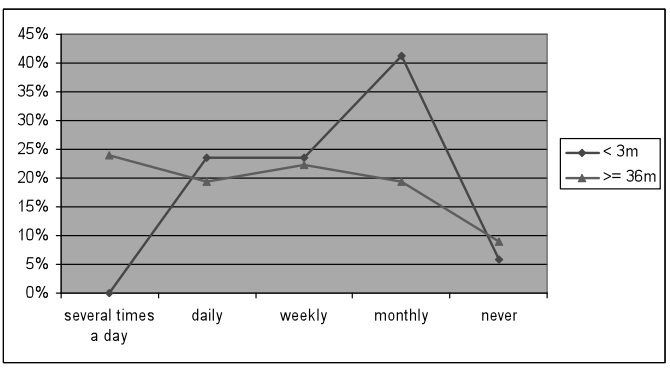


Fig. 3. Distribution of postings. Newbies (< 3m) vs. Experienced members (> 36m)

They seem to be more interested in community and seem to have more confidence in what the community can contribute. Note that the bonding to the VCHC is more important to the experienced members than it is to newbies (e.g. “Does the community play a central role”). A formal organization and the presence of medics is important for newbies as this signals orientation and professionalism. For experienced members the interaction is important.

Let us have a look on the contributions of newbies and experienced members. Experienced members contribute more often than newbies. Fig. 3 depicts the difference from posting “several times a day” (newbies 0%, experienced members 24%) up to posting “monthly” (newbie 42%, experienced members 19%).

Newbies (< 3 months)			Experienced members (>= 36 months)		
Success factor	**Ø**	**Type**	**Success factor**	**Ø**	**Type**
Handling member data sensitively	3,82	S	Handling member data sensitively	3,77	S
That people understand you with your problems	3,71	S	That people understand you with your problems	3,61	S
Possibility of discussions about alternative treatment	3,65	MQA	Stability of the website	3,49	T
Push of research within the field of your disease	3,47	MC	Assistance for new members by experienced members	3,42	S
Assistance for new members by experienced members	3,41	S	Active quality assurance of the content done by the members	3,39	MQA
Statements of the community about medics	3,41	MQA	Establishing codes of behavior(netiquette/guidelines)	3,33	S
Fast reaction time of the website	3,35	T	Fast reaction time of the website	3,31	T
Offering up-to-date and relevant clinical trials	3,29	MC	Sustaining neutrality when presenting and selecting offers	3,30	S
Statements of the community about medical institutions	3,29	MQA	Statements of the community about medical institutions	3,29	MQA
Moderation of member contributions by the operator	3,29	MQA	The feeling to be in a place at home	3,26	S

Table 4. Top Ten of important factors by newbie and experienced member view

Interesting is that distribution of contributions of experienced members (except for "never") is nearly equal at about 20%. There are long-term and active members as well as long-term and passive members.

Social aspects are important to experienced members (Tab. 4). This coincides with the higher participation of experienced members within self-help groups and that they know more people of their VCHC in real life [DL06c; DL06a].

For newbies as well as experienced members "Handling member data sensitively" and "That people understand them" are the most important factors.

Type	Newbies (< 3 months)	Experienced members (>= 36 months)
Altruistic/ Social issues (S)	3	6
Technical issues (T)	1	2
Medical Content (MC)	2	0
Medical quality assurance (MQA)	4	2

Table 5. Summary - Top Ten important factors of newbie and experienced members

The differences in the needs of newbies and experienced members are summarized in Tab. 5. For newbies, social issues as well as medical issues (medical content and medical quality assurance) are in their focus. For experienced members the altruistic and social aspects (S) are dominant. For both, the medical content (MC) within VCHC seems not to be important. This coincides with the theory on virtual communities – contribution of members and social aspects are key.

8 Newly affected vs. long-term affected members

As in the previous section – the differences concerning the attitude of newly affected and long-term affected patients towards success factors is analyzed first.

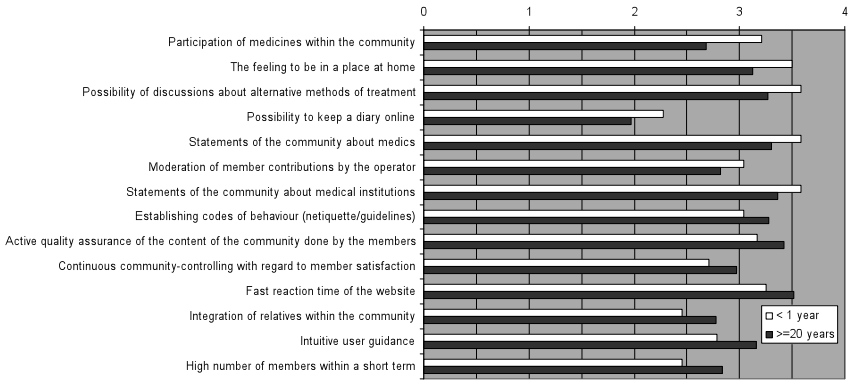


Fig. 4. Main differences in importance to success factors by the view of newly affected vs. Long-term affected members (sorted by difference >=.2) (highly agree 4 – highly disagree 0)

Fig. 4 depicts the main differences (>=.2) between the view of newly affected and long-term affected patients. To determine whether there are pair wise associations for the sets of normally distributed variables a "Bivariate Correlations" examination based on "Pearson" was executed. All correlations w.r.t. the time people are affected by their disease and the success factors are significant at $p < .001$.

Let us first discuss the factors that are more important to the people affected less than a year than to people affected for a very long time. We see that information about medical issues by the community, about treatments and medical institutions, and medics is important. Again, as in the previous section we see the importance of medics, and of formal roles within the VCHC (moderator, neutrality of the community). The feeling at home in the virtual community is also important for newly affected members. For members that are affected for more than 20 years, community aspects, e.g. netiquette, high number of members, the controlling of a community, the contributions of members as well as technical issues about the website (fast reaction time, usability) are more important.

The next step is an analysis of the activity of the two groups. In contrary to the time of membership the time members are affected by their disease does have only little impact to the number of contributions. Fig. 5 illustrates that there is just a small shift from "several times a day" (short-term affected 0%, long-term affected 12%) to daily (short-term affected 33%, long-term affected 21%).

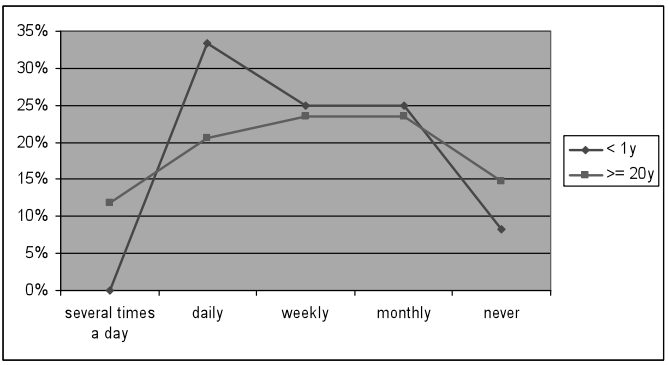


Fig. 5. Distribution of postings. Short-Term (< 1y) vs. Long-term affected (>= 20y)

Note that the ratio contribution/time spent in VCHC of short-term affected members is half of the ratio of long-term affected patients. Thus, short-term affected members spent more time just consuming content.

Short-term affected (< 1 year)			**Long-term affected (>= 20 years)**		
Success factor	**Ø**	**Type**	**Success factor**	**Ø**	**Type**
Handling member data sensitively	3,83	T	Handling member data sensitively	3,91	T
That people understand you with your problems	3,63	S	Stability of the Website	3,61	S
Statements of the community about medics	3,58	MQA	Push of research within the field of your disease	3,56	MC
Possibility of discussions about alternative methods of treatment	3,58	MQA	That people understand you with your problems	3,55	S
Statements of the community about medical institutions	3,58	MQA	Assistance for new members by experienced members	3,52	S
Assistance for new members by experienced members	3,58	S	Fast reaction time of the website	3,52	T
Push of research within the field of your disease	3,50	MC	Statements of the community about medical institutions	3,36	MQA
Stability of the website	3,37	T	Offering up-to-date and relevant clinical trials	3,34	MC
Offering up-to-date and relevant clinical trials	3,33	MC	Encouraging interaction between members	3,21	S
Participation of medics within the community	3,21	MQA	Arranging regular events	3,08	S

Table 6. Top Ten of by short-term and long-term affected member view

For short-term as well as long-term affected members “Handling member data sensitively” and “That people understand them” are the most important factors. Note that 7 out of the 10 most important success factors are in both top ten lists (see Tab. 6). This is an indicator that the time how long members are being affected does not change that much in the needs for e-services.

Type	Short-term Affected (< 1 year)	Long-term Affected (>= 24 years)
Altruistic/ Social issues (S)	2	5
Technical issues (T)	2	2
Medical Content (MC)	2	2
Medical quality assurance (MQA)	4	1

Table 7. Summary - Top Ten factors by short-term and long-term affected member view

Tab. 7 summarizes that for short-term affected members quality assurance issues are in their focus, social, technical and content aspects are of minor interest. For long-term affected members medical and social aspects are dominant. The (pure) content does play only a minor role. Interaction and social relations are the key to attract and keep members that are affected for a long time while medical information and in particular the quality assurance done by community members is the key to attract short-term affected members (that most likely have been diagnosed recently and are looking for neutral and authentic information about disease, medical institutions and treatments.

9 Conclusions

The objective of our research is to obtain insights in what is important to virtual health communities and to various subgroups within virtual health communities. The results include the most important aspects in design and services – the success factors for virtual communities. The analysis of the relation between time of membership and time how long people are affected illustrates that today health communities do not attract the newly diagnosed very well and that they do not bind and keep members very well. The study on the differences between success factors of new members and experienced members show the differences in the needs of those two subgroups: for the newly diagnosed, it is medical content and social interaction and for the long term members, social interaction and altruistic motives dominate their perception of what is important to virtual communities. When we compare this to the differences in success factors that originate in the time people are affected by the disease we observe that a clear tendency in that altruistic and social issues are more important to long-term members while medical quality assurance is more in the focus of new members. We conclude that virtual communities indeed need a more differentiated and richer set of services than what currently is part of a community platform as precondition for eventually empowering virtual communities of patients.

This analysis of the demand for e-services has been done for a very special kind of community. Some results of the study coincide with the literature in the field of services or platforms for virtual communities. The study shows differences in demand for e-services for various subgroups and this illustrates the necessity for profound research and requirement specifications for communities in general.

Acknowledgements

We thank Jan-Marco Leimeister and Helmut Krcmar for providing the origin questionnaire "Success Factors of Virtual Communities from the Perspective of Members and Operators", Heiko Hahn, Robert Kösling, Florian Schiessl and Oliver Schütz and Sven Steinfurt for supporting us within this study and the design of services. We want to thank VC operators and VC members for their participation in the study.

References

[BTD02] Borgaonkar, M. R.; G. Townson; M. Donnelly: Providing Disease-Related information worsens Health-Related Quality of Life in Inflammatory Bowel Disease. In: Inflamm Bowel Disease 8 (2002), S. 264-269.

[Bo04] Borgetto, B.: Selbsthilfe und Gesundheit - Analysen, Forschungsergebnisse und Perspektiven. Verlag Hans Huber, Bern 2004.

[DL04] Dannecker, A.; U. Lechner: "Virtual Communities with a Mission" in the Health Care Sector. In: 11th Research Symposium on Emerging Electronic Markets (RSEEM 2004) (2004), S. 115-128.

[DL06a] Dannecker, A.; U. Lechner: An empirical analysis of the demand for e-services for virtual communities of patients. In: 19th Bled eConference (2006), S. 18.

[DL06c] Dannecker, A.; U. Lechner: Success Factors of Communities of Patients. In: 14th European Conference on Information Systems (2006), S. 12.

[Go05] Goldschmidt, P. C.: HIT and MIS: Implications of Health Information Technology and Medical Information Systems. In: Communications of the ACM 48 (2005) 10, S. 69-74.

[HA97] Hagel III, J.; A. G. Armstrong: Net gain: expanding markets through virtual communities. Harvard Business School Press 1997.

[HT05] Hulstijn, J.; Y.-H. Tan: Design Aspects of a Personalized Information System about Healthcare Regulations. In: 12th Research Symposium on Emerging Electronic Markets (RSEEM 2005) (2005), S. 135-149.

[JKG05] Janke, K. H.; B. Klump; M. Gregor; C. Meisner; W. Haeuser: Determinants of life satisfaction in inflammatory bowel disease. In: Inflamm Bowel Disease 11 (2005) 3, S. 272-286.

[KRN03] Kennedy, A.; A. Robinson; E. Nelson; A. Rogers; D. Reeves; M. Sculpoher; G. Richardson; D. Thompson; C. Roberts: A randomised controlled trial to assess the impact of a package comprising a patient-orientated, evidence-based self-help guidebook and patient-centred consultations on disease management and satisfaction in inflammatory bowel disease. In: Health Technology Assessment 7 (2003) 28, S. 140.

[Le04b] Leimeister, J. M.; P. Sidiras; H. Krcmar: Success Factors of Virtual Communities from the Perspective of Members and Operators: An Empirical Study. In: 37th Annual Hawaii International Conference on System Sciences (HICSS'04) (2004), S. 10.

[MF06] Moona, J.; J. Fisher: The Effectiveness of Australian Medical Portals: Are They Meeting the Health Consumers' Needs? In: 19th Bled eConference (2006), S. 12.

[PUC06] Pratt, W.; K. Unruh; A. Civan; M. Skeels: Personal Health Information Management. In: Communications of the ACM 49 (2006) 1, S. 51-55.

[Pr00] Preece, J.: Online Communities: Designing Usability and Supporting Socialbilty. John Wiley & Sons, Inc. 2000.

A Multi Agent System Application to Support Communities of Practice: Preliminary Analysis

Gilson Yukio Sato, Jean-Paul Barthès

CMR UMR 6599 HEUDIASYC, Computer Science Department
Université de Technologie de Compiègne
Centre de Recherche Royallieu BP20529
60205 Compiègne cedex France
gsato@hds.utc.fr
barthes@utc.fr

Abstract: This paper presents a preliminary analysis for applying Multi Agent Systems to Communities of Practice. In this paper, we present some basic issues on Communities of Practice including a definition and some concepts, namely those of identity, trajectory and multi-membership. We analyze the adequacy of the Multi-Agent Systems technology to support Communities of Practice. We show how some characteristics of Communities of Practice can suggest different applications of Multi-Agent Systems, exploring one of the identified possibilities, more specifically the one related with a member's trajectory into and inside a community.

1 Introduction

The concept of Communities of Practice (CoP) has been used by several organizations to handle problems related to knowledge [BD98][BD00][We98][We00a][We00b] [WMS02]. CoPs have been deployed in a set of different contexts. Tech Clubs at Chrysler regroup in a community various specialists who were spread in different car platforms in order to enable knowledge sharing. The high-availability software community at HP succeeded in standardizing the software sales and installation processes. Eli Lilly used a CoP to solve problems related to duplication of effort, technology redundancy and ineffective transfer of work that occurred after having acquired a smaller company [WMS02].

Usually CoPs are not the main activity of a person in an organization. A community member has other activities like managing projects, programming, selling, etc. In this context, it is important that he could participate effectively and efficiently in his communities, which implies that support systems are highly desirable.

Multi-Agent System approaches in particular could provide the adequate technology for supporting such systems. Intelligent agents that are cooperative, proactive and adaptable could perform tasks to alleviate the increased workload of a person participating in a CoP. In this paper we present a preliminary analysis of some of their possibilities.

The paper is organized as follows: in Section 2 we present some basic issues on CoPs. In Section 3 we analyze the factors that we believe make MAS suitable for supporting CoPs. In Section 4, we demonstrate how some characteristics of CoPs can suggest different applications of MAS and we explore one of them. In the last section, we present some final considerations.

2 Communities of Practice

Organizations have been using Communities of Practice (CoPs) as a new approach to manage knowledge [WMS02]. Most organizations started concentrating their efforts on Information Technology (IT), building intranets, knowledge repositories and tools for improving communications. Such an approach gave them important advantages, like shrinking development cycles, shrinking costs and delivering better products or services. Nevertheless, the approach has some limitations [BD98][BD00][FP98][Mc00]. For example, most IT tools can handle "hard" knowledge (the knowledge that can be easily articulated and captured) in an efficient way but cannot do the same with "soft" knowledge (that includes experience and tacit knowledge) [HK02]. In this context, CoPs provide a new interesting framework for managing knowledge.

2.1 Definition

A CoP is defined as "a group of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in the corresponding area by interacting on an ongoing basis." They exist as informal structures in any organisation, whether acknowledged or not. CoPs cannot be established (like a multifunctional team), but can be cultivated. It is possible to create an environment where they can thrive. In such an environment they are properly funded, they have time to develop, people participation is encouraged, their learning is valued and barriers are removed as indicated by Wenger et al. [WMS02].

The structural model of a CoP combines three elements: (i) a domain of knowledge; (ii) a community of people; and (iii) a shared practice. The domain defines a set of issues and legitimizes the community by affirming its purpose and value to its members. The "domain" motivates members´ participation and contribution and helps them defining what activities should be performed. The "community" creates the social fabric of learning and fosters interactions and relationships based on mutual respect and trust. This kind of relationship creates an environment encouraging people to share ideas, to expose their ignorance, to ask questions and to listen carefully.

The “practice” is a set of frameworks, ideas, tools, information, styles, languages, stories and documents that community members share. It represents the knowledge that the community creates, shares and maintains [WMS02]. The concept of CoP has been used to denote a different way to manage knowledge and has been generating interesting results [BD98][BD00][WMS02].

2.2 Some Important Aspects of CoPs

A CoP is a concept belonging to a wider theoretical framework called Situated Learning. A CoP is the place where a process of Legitimate Peripheral Participation takes place. In this process, newcomers become old timers, starting from a peripheral participation to a full participation as shown by Lave and Wenger [LW91]. In this context, the concept of *identity* plays a major role.

The identity of persons belonging to a community can be thought of as their signature. Through the concept of identity, newcomers feel more familiar with some communities than with others. However, when engaging in some of them, they keep forging their identity. A newcomer will learn the community practices, will become more knowledgeable in its domain and will share knowledge in such a way as to belong to this community. Belonging to a community helps to build an identity, because it helps to define what should be known and what can be ignored. Also, the identity is determined by communities to which one belongs or does not belong.

Identities are not static, they change in time. They have trajectories inside communities that represent the past, the present and the future of the members. Analyzing emblematic trajectories could help newcomers to have a glance of their perspectives and could allow old timers to revisit their own history.

Identities also grow in space, they cross boundaries among communities. People participate in various communities and they cannot use a specific identity for each of them. Multi-membership is an inherent aspect of identities. This multi-membership could be a bridge between different communities and constitute another way to expand identities as discussed by Wenger [We98][We00a].

The utilization of CoPs to manage knowledge has been already giving results but we think that other aspects like *trajectories* and *multi-membership* should be explored in order to better use the corresponding social structures.

3 Multi-Agent Systems

Multi-agent systems (MAS) have been deployed in various domains, like concurrent engineering, manufacturing [SNB01], knowledge management [BT02][TB02][TB03], computer communications [MV04] or e-commerce. They can be used to intelligently assist users in specialized and generic tasks. Specialized tasks include, among others, network management [MV04] or operation of CAE tools [MYA99]. Generic tasks can also be supported by agents and by MAS. For example: handling information (e.g. retrieving, filtering, synthesizing), making decisions (decision support systems) [Kl01] or capturing lessons learned by a project team [TB03].

3.1 Definition

For this work we considered that an MAS is a system composed by a group of possibly heterogeneous and autonomous agents that share a common goal and work cooperatively to achieve this goal [TB02].

3.2 Why Should MAS Be Applied for Supporting CoPs

MAS technology is adequate to develop applications for CoPs because it offers flexibility to support complex applications. Moreover, MAS offer the possibility to [BT02]:

- Evolve by the addition or suppression of agents,
- Include proactive services that run in the background,
- Interface to legacy software
- Run several options for the same service in parallel.

We think that the flexibility of MAS allows them to be the base for developing applications that are suited to the needs of a specific community. CoPs can differ from one another [WMS02]. For example, some community can privilege the use of asynchronous tools like email or *blogs,* while another will prefer using synchronous tools like chats. Probably, in such cases a good application for the first community is not so useful for the second one (e.g. a meeting scheduler).

Another factor requiring flexibility from the technology is that communities evolve. A Community has a life cycle: it starts, grows, matures and disappears [WMS02]. We think that in this context the tools used in the community should follow suit. In this case, the flexibility and the possibility to evolve inherent to an MAS can help to provide an always up to date set of tools for the community.

The activities of members of CoPs are performed in parallel with other activities like the activities of a project team [WMS02]. In this context, members do not have much time to dedicate to community tasks. An MAS can help them to decrease their workload by performing tasks that can be automated.

Although Hattori et al. [Ha99] do not refer to CoPs specifically; they argue that an MAS is attractive to support networked communities. The distributed character of this kind of community fits in a distributed architecture like the multi-agent architecture. They also mention that the support system should handle the dynamic nature of the community in which members change the way of participating. The last mentioned characteristic is that the individuality of the members is preserved in a community. They suggest that a personal agent should be used to preserve such individuality [Ha99].

Case et al. [Ca01] although not referring to CoPs also argue that intelligent agents are the ideal technological platform to provide services and solutions for building electronic communities because an intelligent agent is cooperative, proactive and adaptable [Ca01].

3.5 Related Works

In this section we describe classified some MAS applications in relation with the goal of supporting communities in general. For this purpose, we used a classification elaborated by Wenger et al. [We05].

A first group of applications, not designed to support communities, could help community members to perform daily activities more efficiently. In this group we find tools to: elaborate individual profiles [Ma01]; generate a personalized newspaper [CS98][GSA04]; support Web browsing [KM00][SMB01] and filter and retrieve information of distributed sources [BHB01][GSA04][JS02][PKB00] [SGK03][SMY02].

Another group could facilitate both synchronous and asynchronous communications among members of a community. In this group one finds tools to: indicate on-line presence [Ha99][Yo03]; schedule meetings [CWW03][LE98]; promote spontaneous synchronous meetings [Na04][RKD03]; provide smart support to meetings [Ch04] [HGG04][HK04]; direct e-mails to appropriate members [LSS04] and analyze and classify discussions boards [Ha99].

A third group could help communities to consult, save and organize information. This information can be stored in a knowledge repository, an organizational memory or in the Internet. In this group we find tools to: manage community bookmarks [KM02] [KLW01]; recommend documents [GP03][MM97]; filter and retrieve information collaboratively [Gl01]; capture lessons learned [TB03]; automate functions in a portal [BT02] and access organizational memory [AM96].

The fourth and last group includes systems designed to support communities. They include tools to: indicate the presence of a community member [GP01]; identify and form communities [Ha99][LDV99][Ro01][SM02][Wa02][Yo03]; support community activities [GP01][LDV99].

Some of the systems or applications contain a user profile in order to provide more personalized services. They are able to perform more personalized information retrieval or to find other persons with similar profiles, aiming at the formation of communities. Some profiles are elaborated by analyzing users' web navigation behavior; others are built by analyzing the documents a given user utilizes. As these profiles characterize the users, they could be considered like "glimpses" of the users' identities.

In the present work, we focus on different aspects of users' identities. We aim at characterizing (at least, partially) each member's trajectory in the community through parameters like the number of posts in a discussion list or the evaluation of the impact of his contribution to the community. In this sense, we think the system we envisage can support communities in a different way.

Analyzing the available systems, we could confirm that MAS are used to implement systems to support communities because of their distributed character and the possibility to offer intelligent services to the users.

4 Applying MAS to CoPs

4.1 Multi-Membership

Some features of CoPs can be explored in order to provide a better support. One of them is *multi-membership*. People usually participate in more than one community and each community contributes to build the identity of its members. For example, in an organization, a person might participate in a community interested in Java programming and also participate in another one interested in project management applied to software. His participation might be different in each community. In the Java community he could be one of the most knowledgeable and experimented member who contributes sharing his knowledge with newcomers. The same person could be a newcomer in the project management community. He could just start to lurk around the more experienced people. He could keep this status or could start participating more actively.

An MAS usually has users' profiles that can provide "a glance" of users' identities. With such profiles an MAS is able to offer intelligent services. For example, based on a good profile, an agent can perform better information retrieval. However, multi-membership raises some issues. Should the user have one profile for each community to which she belongs? If she has various profiles probably she will spend time managing them or will need to shift among them during her work. Maybe it is the only option for somebody who participates in CoPs supported by different systems. But should it be so in an organization with a single system? Is it possible to utilize only one profile for all communities in which a person participates? We think that such issues should be explored in order to develop better systems for supporting CoPs.

4.2 Trajectories

Another aspect that should be explored is the evolution of identities in time. Surely profiles should be dynamic. But we think that even dynamic profiles cannot represent the evolution of an identity inside the community.

In order to better explain our notion of trajectories, we are going to use an analogy between a formal association (e.g., a non profit organization) and a community. Usually to become member of a formal association, the candidates must subscribe. Paying the fee, the new members become eligible for some services. For example, they can receive newsletters; be invited to events promoted by the association or run for the presidency of the association.

In a distributed community, usually candidates do not pay any fee but probably would subscribe to a discussion forum. They are then entitled to access the community document repository or to participate in chats with other members. A community, like an association, allows different levels of participation. An association is usually managed by executive committee. Its members who are more engaged in the activities of the association. For example, they organize workshops or conferences, publish newsletters, and try to attract new members. Other members participate in the conferences organized by the association and sometimes even help in organizing them. There are also members who read the newsletters or the proceedings of the conferences and participate sometimes in the workshops promoted by the association. In a distributed community there is no executive committee but members who participate more frequently and intensively, who form the *core group* of the community. Such a group is responsible for organizing events like chats and for animating the community in, say, suggesting new topics or new activities. The core members amount to generally 10 to 15 percent of the whole community [WMS02]. Other members, active members, of distributed communities participate in the activities, like the chats or the discussion forums, but without the same regularity and intensity as the core members. Active members of a community are 15 to 20 percent of the members of a community [WMS02]. Members of an association who read the newsletter could be compared with the peripheral members of a community. Such peripheral members observe the interactions among core and active members through the discussion forums and in their own way learn the practice of the community as the process goes by.

In an association, usually the members of the committee are elected for a term. So after some time, new elections are called and a new committee is appointed to manage the association for another term. In a community, the core group can be changing at any time. A core member could loose interest in the domain as it shifts and leave the community. Peripheral members could become more active as they start participating more regularly. As they get more involved in the activities of the community, they move towards the centre and can end up in the core group.

As they change their level of participation, members describe trajectories. Such trajectories can be in the direction of the core but they can also indicate that a member is disengaging from the community. As CoP elements change (the domain, the practice or the community) the members change their level of participation describing a trajectory that points towards the outside of the community. In an association, some members can also adopt a peripheral level of participation when a new executive committee is elected.

Monitoring individual trajectories could seem secondary to monitoring the whole community, but individual trajectories represent in some extent how a community is doing. For example, becoming a core member is an important process in a community because it indicates that the member has learnt the practice of the community. Trajectories towards the core of the CoP indicate that members are learning, which is a vital process for the entire community.

Given the importance of trajectories, we think that they must be explored in order to support CoPs effectively. We consider that following trajectories could contribute to the development of CoPs in several ways:

- Trajectories can indicate how well members of a given community are learning;
- Various members with trajectories in direction of the core are a signal of the vitality of the community.
- Remarkable trajectories could help newcomers to project their future participation [WMS02].
- Trajectories can help old-timers to analyze their past participation [WMS02].

Certainly, it is a challenge to determine which indicators can best represent a trajectory. Quantifiable indicators such the number of posts in a discussion list, the number of contributions in a *blog* and the number of documents posted in a repository could be useful. A process of peer assessment could be used to follow a trajectory too. But qualitative clues should be more representatives of the trajectories, like, the quality of the contributions or the impact of a document.

At this point, we can envisage some MAS applications for supporting CoPs. One of them is an agent that could survey some quantitative indicators of participation. This agent could monitor the number of contributions in the various forms of communication used by the community.

For example, if the community uses email, *blogs* and chats for communication, such an agent could count the number of emails posted by a member, the number of his interventions in the chat, the entries in his own *blogs* and the commentaries posted in other members' *blogs*. The agent could also verify in how many chat sessions each member participated and how often a given *blog* is accessed.

Such indicators cannot measure the participation of each member, but they could provide some clues on how a member is participating to the community coordinator. For example, if the participation is low, he could try to promote a face-to-face meeting to verify if there is a problem.

In a CoP, usually, peer recognition is highly valued. An agent can help to manage spontaneous demonstrations of recognition. For example, if a member of the core group recognizes that the contribution of a peripheral member was valuable, this fact should be saved and considered because it could mean that the degree of participation of such a peripheral member is changing.

In some circumstances, peer recognition can be induced. For example, an agent could start a process of assessment of a peripheral member's participation, triggered by a quantitative indicator as the number of posted emails. This agent could be used to synthesize the contributions of a member and send them to other members in order to get an evaluation.

In this way, quantitative indicators and qualitative assessments could be combined to represent a trajectory. Surely, it is not a complete representation, but we think it could help the coordinator and the core group to observe the dynamics of the community.

The MAS technology could help in this kind of task. Moreover it could use intelligence to perform the tasks. In this context, we think that the MAS technology can contribute significantly to supporting CoPs.

4.3 Some Practical Considerations

We are currently developing our agents using the OMAS (Open Multi-Agent System) platform. Such a platform allows us to develop two types of agents:

- Service Agents (SAs) that provide specific services like handling documents and performing web searches.
- Personal Assistants (PAs) that interface the users to the system [BT02][TB02] [TB03].

In the following, we assume that in the near future each user will be connected to the outside world through a Personal Assistant. At the moment we plan to have an additional specialized agent, called Trajectory Agent (TA), to manage the trajectory of a community member. Such an agent is an SA that will work mostly with the user's Personal Assistant, in other words, a Staff Agent for the Personal Assistant. It should save and present the information concerning a user's trajectory. We envisage the use of one Trajectory Agent for each community in which a given user participates. In this case, multi membership implies the use of various Trajectory Agents working for the user's Personal Agent.
Trajectory Agents can be used to indicate an affiliation to a community, e.g. whenever a user downloads such an agent to become a community member. Instead of paying a subscription, like in formal associations, candidates would download a Trajectory Agent to become member of a specific community.

All Trajectory Agents of a given user should be able to communicate and provide him, through his Personal Assistant, information about his participation in the various communities. This information would be obtained from other Service Agents.

The Service Agents should perform tasks like the analysis of emails, *blogs* and chats and the management of the peer recognition system. In this way, a Service Agent can monitor the number of messages and replies a user sends in a discussion forum. Another Service Agent could monitor how many times a *blog* is read and who the members that post comments in this blog are. A third Service Agent could monitor the participation in the community chat sessions, etc. All the Service Agents should be able to communicate with the user's appropriate Staff Agent. As OMAS is an open platform, other SAs can be added to provide different kinds of information characterizing a trajectory.

We expect that the coordinator of the community will need a Trajectory Agent for his own trajectory and an agent that allows him to follow the trajectories of other members. Such agents can be used to also monitor important changes in the participating patterns. For example, the coordinator should be able to ask his Personal Assistant information about the activities of a particular member during a given period of time. His Personal Assistant should contact the member's Staff Agent and ask for the information. The Personal Assistant should be able to ask another Service Agent to format and to present the information.

We envisage as a next desirable step to introduce Service Agents for analyzing the content of the contributions, allowing a more precise representation of a trajectory. We consider that although quantitative indicators could help community coordinator and core members to monitor the activity in community, other types of indicator are also desirables. For example, a question in the discussion board that becomes a FAQ in the community's home page represents an important contribution. Members that contribute in this way should be acknowledged and this should appear in some way in their trajectories.

5 Final Considerations

This paper presented a preliminary analysis of some possibilities to apply MAS for supporting CoPs. Participation in CoPs should not absorb too much time because CoP members are basically team members developing projects. Consequently the participation in CoPs should be facilitated. One way of doing this is to provide adequate technological tools. MAS with their flexibility and possibility of intelligent behavior seem to be a promising technology to support participation in CoPs. In particular, one of the aspects in which MAS could be employed is to monitor the TRAJECTORY of a member inside a community. Following this trajectory could help to assess the participation of a member and the vitality of a community. Because our prototype is not yet available, we cannot give concrete results. However, the preliminary analysis presented at this stage constitutes a theoretical framework to develop some ideas about using MAS technology.

Acknowledgement

Gilson Yukio Sato is supported by the Programme AlBan, the European Union Programme of High Level Scholarships for Latin America, scholarship no.E04D032545BR and by the Parana State Federal University of Technology (UTFPR-Brazil).

References

[AM96] Ackerman, M.S.; McDonald, D.W.: Answer Garden 2: Merging Organizational Memory with Collaborative Help. In: Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work, Boston, Massachusetts, ACM Press, 1996; pp. 97-105.

[BT02] Barthès J.P.; Tacla, C.A.: Agent-Supported Portals and Knowledge Management in Complex R&D Projects. Computers in Industry, 2002; pp. 48, 3-17.

[BD98] Brown J. S.; Duguid, P.: Organizing knowledge. In: California Management Review, spring 1998, 40(3); pp. 90-111.

[BD00] Brown J.S.; Duguid, P.: The social life of information, vol. 1, 1 ed. Boston: Harvard Business School Press, 2000.

[BHB01] Budzik, J.; Hammond, K.J.; Birnbaum, L.: Information access in context. In: Knowledge-Based Systems, vol. 14, 2001; pp. 37-53.

[Ca01] Case, S.; Azarmi, N.; Thint, M.; T. Ohtani: Enhancing e-communities with agent-based systems. In: Computer, vol. 34, 2001, pp. 64-69.

[Ch04] Chen, H.; Perich, F.; Chakraborty D.; Finin, T.; Joshi A.: Intelligent Agents Meet Semantic Web in a Smart Meeting Room. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems – Volume 2. IEEE Computer Society, New York, N.Y., 2004; pp. 854-861.

[CS98] Chen, L.; Sycara, K.: WebMate: a personal agent for browsing and searching. In: Proceedings of the second international conference on Autonomous agents, Minneapolis, Minnesota. ACM Press, 1998; pp. 132-139.

[CWW03] Chun, A.; Wai, H.; Wong, R.Y.M.: Optimizing agent-based meeting scheduling through preference estimation. In: Engineering Applications of Artificial Intelligence, vol. 16, 2003; pp. 727-743.

[FP98] Fahey, L.; Prusak, L.: The eleven deadliest sins of knowledge management. In: California Management Review, spring 1998, 40(3); pp. 40-54.

[Gl01] Glance, N. S.: Community search assistant. In Proceedings of the 6th international conference on Intelligent user interfaces, Santa Fe, New Mexico. ACM Press, 2001; pp. 91-96.

[GP03] Gomez-Sanz, J.J.; Pavon, J.: Personalized information dissemination using agent organizations. In: Proceedings of the Ninth IEEE Workshop on Future Trends of at Distributed Computing Systems, 2003- FTDCS 2003, 2003; pp. 38-44.

[GP01] Gräther, W.; Prinz, W.: The social web cockpit: support for virtual communities. In: Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work. Boulder, Colorado, USA. ACM Press, 2001; pp. 252-259.

[GSA04] Godoy, D.; Schiaffino, S.; Amandi, A.: Interface agents personalizing Web-based tasks. In: Cognitive Systems Research, vol. 5, 2004; pp. 207-222.

[HGG04] Harper, L.D.; Gertner, A.S.; Guilder, J.A.V.: Perceptive assistive agents in team spaces. In: Proceedings of the 9th international conference on intelligent user interface, Funchal, Madeira, Portugal. ACM Press, 2004; pp. 253-255.

[Ha99] Hattori, F.; Ohguro, T.; Yokoo, M.; Matsubara, S.; Yoshida, S.: Socialware: multiagent systems for supporting network communities. In: Commun. ACM, vol. 42, 1999; pp. 55-61.

[HK04] Hellenschmidt, M.; Kirste, T.: Software solutions for self-organizing multimedia-appliances. In: Computers & Graphics, vol. 28, 2004; pp. 643-655.

[HK02] Hildreth, P.; Kimble, C.: The duality of knowledge. In: Information Research, 2002, 8.

[JS02] Ji, Y.G.; Salvendy, G.: A metadata filter for intranet portal organizational memory information systems. In: International Journal of Human-Computer Studies, vol. 56, 2002; pp. 525-537.

[KM02] Kanawati R.; Malek, M.: A multi-agent system for collaborative bookmarking. In: Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 3, Bologna, Italy. ACM Press, 2002; pp. 1137-1138.

[KM00] Keeble, R.J.; Macredie, R.D.: Assistant agents for the World Wide Web intelligent interface design challenges. In: Interacting with Computers, vol. 12, 2000; pp. 357-381.

[Kl01] Klusch, M.: Information agent technology for the Internet: A survey. In: Data & Knowledge Engineering, 2001, 36; pp. 337-372.

[KLW01] Koch, M.; Lacher, M.; Worndl, W.: The CommunityItemsTool-interoperable community support in practice. Proceedings of the Tenth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2001. WET ICE 2001; pp. 368-373.

[LW91] Lave J.; Wenger, E.: Situated Learning: Legitimate peripheral participation. Cambridge University Press, Cambridge, 1991.

[LE98] Lemaître C.; Excelente, C.B.: Multi-agent network for cooperative work. In: Expert Systems with Applications, vol. 14, 1998, pp. 117-127.

[LDV99] Lieberman, H.; van Dyke N.; Vivacqua, A.: "Let's browse: a collaborative browsing agent," Knowledge-Based Systems, vol. 12, 1999, pp. 427-431.

[LSS04] Loia, V.; Senatore, S.; Sessa, M.I.M.I.: Combining agent technology and similarity-based reasoning for targeted E-mail services. In: Fuzzy Sets and Systems, vol. 145, 2004, pp. 29-56.

[Ma01] Maglio, P.P.; Campbell, C.S.; Barrett, R.; Selker, T.: An architecture for developing attentive information systems. In: Knowledge-Based Systems, vol. 14, 2001, pp. 103-110.

[MV04] Manvi, S.S.; Venkataram, P.: Applications of agent technology in communications: a review. In: Computer Communications, 2004, 27; pp. 1493–1508

[MM97] Marsh, S.; Masrour, Y.: Agent augmented community-information: the ACORN architecture. In: Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research Toronto, Ontario, Canada IBM Press, 1997; pp. 17

[Mc00] McDermott, R.: Why information technology inspired but cannot deliver knowledge management. In (Lesser, E.L.; Fontaine, M.A.; Slusher, J.A. Eds.): Knowledge and communities. Butterworth-Heinemann, Boston, 2000; p. 21-35.

[MYA99] Miyoshi, A.; Yagawa, G.; Sasaki, S.: An interface agent that actively supports CAE beginner users in performing analyses. In: Advances in Engineering Software, 1999, 30; pp. 575–579

[Na04] Nakanishi, H.: FreeWalk: a social interaction platform for group behaviour in a virtual space. In: International Journal of Human-Computer Studies, vol. 60, 2004; pp. 421-454.

[PKB00] Pierre, S.; Kacan, C.; Probst, W.: An agent-based approach for integrating user profile into a knowledge management process. In: Knowledge-Based Systems, vol. 13, 2000; pp. 307-314.

[RKD03] Raybourn, E.M.; Kings, N.; Davies, J.: Adding cultural signposts in adaptive community-based virtual environments. In: Interacting with Computers, 2003, 15; pp. 91-107

[Ro01] De Roure, D.; Hall, W.; Reich, S.; Hill, G.; Pikrakis, A.; Stairmand, M.: MEMOIR - an open framework for enhanced navigation of distributed information. In: Information Processing & Management, vol. 37, 2001, pp. 53-74.

[SGK03] Shakshuki, E.; Ghenniwa, H.; Kamel, M.: An architecture for cooperative information systems. In: Knowledge-Based Systems, vol. 16, 2003; pp. 17-27.

[SMY02] Shaw, N.G.; Mian, A.; and Yadav, S.B.: A comprehensive agent-based architecture for intelligent information retrieval in a distributed heterogeneous environment. In: Decision Support Systems, vol. 32, 2002; pp. 401-415.

[SNB01] Shen, W.; Norrie D.H; Barthes, J.P.: Multi-Agent Systems for Concurrent Intelligent Design and Manufacturing, Taylor and Francis, London, UK, 2001.

[SMB01] Sorensen, C.; Macklin, D.; Beaumont, T.: Navigating the World Wide Web: bookmark maintenance architectures. In: Interacting with Computers, vol. 13, 2001; pp. 375-400.

[SM02] Sumi Y.; Mase, K.: Supporting the awareness of shared interests and experiences in communities. In: International Journal of Human-Computer Studies, vol. 56, 2002, pp. 127-146.

[TB02] Tacla, C.A.; Barthès, J.P.: A Multi-Agent Architecture for Knowledge Management Systems. In: Proceedings of the Second IEEE International Symposium on Advanced Distributed Computing Systems, Guadalajara, México, Nov. 13-15, 2002; pp. 1-12.

[TB03] Tacla, C.A.; Barthès, J.P.: A Multi Agent System for Acquiring and Sharing Lessons Learned. In: Computers in Industry, 2003, 52; pp. 5-16.

[Wa02] Wang, F.: Self-organising communities formed by middle agents. In: Proceedings of the first international joint conference on Autonomous agents and multi-agent systems: part 3. Bologna, Italy. ACM Press, 2002; pp. 1333-1339.

[We98] Wenger, E.: Communities of practice: learning, meaning and identity. Cambridge university press, Cambridge, 1998.

[We00a] Wenger, E.: Communities of practice: the key to knowledge strategy. In (Lesser, E.L.; Fontaine, M.A.; Slusher, J.A. Eds.): Knowledge and communities. Butterworth-Heinemann, Boston, 2000; pp. 3-20.

[We00b] Wenger, E.: Communities of practice and social learning systems. In: Organization, May 2000, 7(2); pp. 225-246.

[WMS02] Wenger, E.; McDermott, R.; Snyder, W.M.: Cultivating communities of practice: a guide to managing knowledge, vol. 1, 1 ed. .Harvard Business School Press, Boston, 2002.

[We05] Wenger, E.; White, N.; Smith, J.D.; Rowe, K.: Outiller sa communauté de pratique. In (Langelier, L. Ed): Travailler, apprendre et collaborer en réseau. Guide de mise en place et d'animation de communautés de pratique intentionelles. Quebec: CEFRIO, 2005, pp. 47 - 66.

[Yo03] Yoshida, S.; Kamei, K.; Ohguro, T.; Kuwabara, K.: Shine: a peer-to-peer based framework of network community support systems. In: Computer Communications, vol. 26, 2003; pp. 1199-1209.

Stemming Strategies for European Languages

Jacques Savoy

Computer Science Dept.
University of Neuchatel
Rue Emile Argand 11
CH - 2009 Neuchatel (Switzerland)
Jacques.Savoy@unine.ch

Abstract: In this paper, we describe and evaluate different general stemming approaches for the French, Portuguese (Brazilian), German and Hungarian languages. Based on the CLEF test-collections, we demonstrate that light stemming approaches are quite effective for the French, Portuguese and Hungarian languages, and perform reasonably well for the German language. Variations in mean average precision among the different stemmers are also evaluated and are sometimes found to be statistically significant.

1 Introduction

In order to perform various text tasks such as text mining, information retrieval, entity extraction, or Web-based lexical statistics, we usually need to transform the words as they appear into their corresponding root or stem forms. Such a procedure is called stemming. In information retrieval (IR), when indexing documents or requests it is assumed that the application of a stemmer is a good practice. For example, when a query contains the word "plane," it seems reasonable to also retrieve documents containing the related word "planes."

As a first approach to designing a stemmer, we begin by removing only inflectional suffixes so that singular and plural word forms (e.g., "cars" and "car") or feminine and masculine variants (e.g., "actress" and "actor") will conflate to the same root. Stemming schemes that remove only morphological inflections are termed as "light" suffix-stripping algorithms, while more sophisticated approaches have also been proposed to remove derivational suffixes (e.g., '-ition', '-able' in "recognition," "recognizable," and "recognize"). Those suggested by Lovins [Lov68] or by Porter [Por80] are both typical examples for the English language

Stemming schemes are usually designed to work with general text. Certain stemming procedures may also be especially designed for a specific domain (e.g., in medicine) or a given document collection, such as that developed by Xu & Croft [XC98], which uses a corpus-based approach. This more closely reflects the language usage (including word frequencies and other co-occurrence statistics), instead of a set of morphological rules in

which the frequency of each rule (and therefore its underlying importance) is not precisely known.

While the English language has already been the object of various stemmer studies, this is not true of other European languages, for which stemmers and appropriate evaluation studies are not readily available. This paper is divided as follows: Section 2 presents some related works, while Section 3 depicts the main characteristics of our test-collections. Section 4 briefly describes the IR models used during our experiments. Section 5 discloses the various stemming approaches suggested, and in Section 6 they are evaluated from various perspectives. The main findings of this paper are presented in Section 7.

2 Related Work

In the IR domain we usually assume that stemming is an effective method of enhancing retrieval effectiveness through conflating several different word variants into a common form (the n-gram indexing strategy [MM04] is a typical exception to this rule). Most stemming approaches are based on morphological rules for the language involved (see [Lov68] or [Por80] for the English language). In such cases, suffix removal is also controlled through the adjunct of quantitative restrictions (e.g., '-ing' would be removed if the resulting stem had more than 3 letters as in "running," but not in "king") or qualitative restrictions (e.g., '-ize' would be removed if the resulting stem did not end with 'e' as in "seize"). Moreover, certain ad hoc spelling correction rules are also applied to improve conflation accuracy (e.g., "running" - 'ing' gives "run" and not "runn").

Simple stemming procedures can process text quickly but by ignoring word meanings they tend to make errors, usually due to over-stemming (e.g., "general" becomes "gener", and "organization" is reduced to "organ") or to under-stemming (e.g., the words "create" and "creation" do not always conflate to the same root). Thus the use of an on-line dictionary has been suggested in order to produce better conflations [KJ04].

Compared to other languages with more complex morphologies [Spr92], English stemmers are quite simple and to reduce their error rate, we may be consider using a dictionary [Sav93]. For those languages having more complex morphological structures, a deeper analysis may be required (e.g., for the Finnish language [KJ04]). Moreover, for other European languages only a few stemming procedures have been suggested, and those schemes available usually apply only to languages that are most spoken. For the African languages (except for Arabic) no stemming procedures are readily available while for the Asian languages, stemming is not always useful. In Chinese for example, morphological variations are usually not indicated (e.g., by a suffix as in Indo-European languages). In Japanese Hiragana characters are mainly used to write grammatical words (e.g., "do", "and", "of"), and inflectional endings (e.g., possessive, subject or object markers) for verbs, adjectives and nouns. Thus the removal of Hiragana characters is a simple process that may be viewed as a stemming procedure.

When analyzing IR stemming performance, Harman [Har91] demonstrated that no statistically significant improvements could be achieved through applying three different stem-

ming strategies, namely that of Lovins [Lov68], Porter [Por80] and another basic stemmer that conflates English singular and plural word forms (based on three rules). A query-by-query analysis revealed that stemming did affect the performance yet the number of queries depicting improvements was nearly equal to the number of queries showing degradation in performance. Other studies [Hul96] limited to only one language (usually English) showed modest improvement when using a stemmer and came to similar conclusions when using one search strategy: The use of a stemmer resulted in only modest improvement, and when compared to an approach ignoring stemming, the difference was not always statistically significant.

It was also surprising to see that during the last CLEF evaluation campaigns [PM05] (see Web site at `www.clef-campaign.org`), only a few stemmers were suggested by other participants and little effort was made to compare stemmers. For example, when evaluating the two statistical stemmers used for five languages, Di Nunzio *et al.* [DNO04] showed for each of these languages there were variations in relative retrieval performance. This means that any given stemming approach may work well for one language, yet poorly for another. When compared to statistical stemmers, Porter's stemmers seemed to work slightly better. For German, Braschler & Ripplinger [BR04] showed that for short queries stemming may enhance mean average precision by 23%, compared to 11% for longer queries. Finally, Tomlinson [Tom04] evaluated the differences between Porter's stemmer and the lexical stemmer (based on a dictionary of the corresponding language). He found that for the Finnish and German languages, the lexical stemmer tended to produce statistically better results, while for seven other languages the performance differences were small and insignificant.

From these facts, the following questions thus arise: 1) Does stemming affect IR performance for European languages other than English, or is the impact of stemming negligible due to their more complex morphology? 2) For these languages, are light stemming approaches less effective than more complex suffix-stripping algorithms? The rest of this paper provides answers to these questions.

3 Test Collections

In our experiments we used the CLEF 2005 corpora made up of newspaper and news agency articles, namely *Le Monde* (French), *SDA* (French), *Público* (Portuguese), *Folha* (Brazilian), *Magyar Hirlap* (Hungarian). The German collection is part of the GIRT corpora and composed of bibliographic records extracted from various sources in the social sciences. A typical record in this German corpus consists of a title, an abstract and a set of manually assigned descriptors. See Kluck [Klu04] for a more complete description of this corpus.

As shown in Table 1, both the French and Portuguese corpora have roughly the same size (487 MB vs. 564 MB), with the German ranking third and the Hungarian fourth, both in size (105 MB) and in number of documents (49,530). The Portuguese corpus has also a larger mean size (212.9 indexing terms/document) or median number of terms per doc-

	French	Portuguese	Hungarian	German
Size	487 MB	564 MB	105 MB	326 MB
# documents	177,452	210,734	49,530	151,319
mean number of terms	178	212.9	142.1	89.6
median size (# terms)	126	171	88	95
# queries	50	50	50	50
# rel. doc./query	50.7	58.1	18.8	86.9

Table 1: Some statistics from our test collections (CLEF)

ument (171) than does the French collection (mean = 178, median = 126). This mean value is slightly smaller for the Hungarian corpus (mean = 142.1, median = 88) and smallest for the German collection (mean = 89.6, median = 95). During the indexing process, we retained the logical sections allowed by CLEF evaluation campaigns. For the German collection, we applied a decompounding procedure [Sav04], and retained compound words and their components in document or topic representations. Compound words (e.g., handgun, worldwide) are widely used in German and can lead to more difficulties they do for the English language. "Computersicherheit" for example is composed of "Computer" + "Sicherheit" (security) and could also appear separately (e.g., "die Sicherheit mit Computern"). Finally, although accents were removed this process may have accidentally conflated words with different meanings into the same form (e.g., in French the word "tâche" (task) and "tache" (mark, spot)).

Each topic was structured into three logical sections comprising a brief title, a one-sentence description, and a narrative part specifying the relevance assessment criteria. In this study, we used the shortest query formulation in order to reflect a more realistic search context. Based on the topic title only, the query had a mean size of 2.8 search terms for the French collection, 2.6 for the Portuguese, 2.2 for the Hungarian and 1.7 for the German.

The available topics covered various subjects (e.g., "Brain-Drain Impact," "Internet Junkies," or "Creutzfeldt-Jakob Disease") and included both regional ("Deutsche Bank Takeovers") and international coverage ("Microsoft Competitors").

As shown in Table 1, the mean number of relevant items per query for the French and Portuguese collection has a relatively similar value (50.7 and 58.1 respectively), but this value is lower for the Hungarian corpus (18.8), a collection whose size is only one quarter that of the French corpus. The mean number of relevant articles per request for the German test-collection was clearly higher, at 86.9

4 IR Models

In order to obtain a broader view of the relative merit of the various retrieval models and stemming approaches, we used seven vector-space schemes and two probabilistic models. First we adopted the classical $tf\ idf$ model, in which the weight attached to each indexing

ntc	$w_{ij} = \frac{tf_{ij} \cdot idf_j}{\sqrt{\sum_{k=1}^{t}(tf_{ik} \cdot idf_k)^2}}$
atn	$w_{ij} = idf_j \cdot \left[\frac{0.5 + 0.5 \cdot tf_{ij}}{max\ tf_{i.}}\right]$
ltn	$w_{ij} = [ln(tf_{ij}) + 1] \cdot idf_j$
dtn	$w_{ij} = [ln(ln(tf_{ij}) + 1) + 1] \cdot idf_j$
ltc	$w_{ij} = \frac{[ln(tf_{ij})+1] \cdot idf_j}{\sqrt{\sum_{k=1}^{t}([ln(tf_{ik})+1] \cdot idf_k)^2}}$
dtu	$w_{ij} = \frac{[ln(ln(tf_{ij})+1)+1] \cdot idf_j}{(1-slope) \cdot pivot + (slope \cdot nt_i)}$
Lnu	$w_{ij} = \frac{\frac{ln(tf_{ij})+1}{ln\left(\frac{l_i}{nt_i}\right)+1}}{(1-slope) \cdot pivot + (slope \cdot nt_i)}$
Okapi	$w_{ij} = \frac{(k_1+1) \cdot tf_{ij}}{K + tf_{ij}}$ with $K = k_1 \cdot \left[(1-b) + b \cdot \frac{l_i}{avdl}\right]$

Table 2: Various Weighting Schemes

term was the product of its term occurrence frequency (or tf_{ij} for indexing term t_j in document D_i) and its inverse document frequency (or $idf_j = ln(n/df_j)$, where n indicates the number of documents in the corpus, and df_j the number of documents in which the term t_j appears). To measure similarities between documents and requests, we computed the inner product after normalizing indexing weights (model denoted "doc=ntc, query=ntc" or "ntc-ntc").

Other variants might also be created, especially in cases when the occurrence of a particular term in a document was deemed a rare event. Thus, it might be good practice to assign more importance to the first occurrence of this word, as compared to any successive, repeating occurrences. Therefore, the tf component might be computed as the $ln(tf) + 1$ (model "doc=ltc, query=ltc") or as $0.5+0.5 \cdot [tf\ /\ max\ tf\ in\ D_i]$. Of course, other weighting formulae could also be used for documents and requests, leading to different weighting combinations (see Table 2 where the length of document D_i is denoted by nt_i, and $avdl$, b, k_1, $pivot$ and $slope$ are constants.). We might also consider that a term's presence in a shorter document would provide stronger evidence than in a longer document, leading to more complex IR models; for example the IR model denoted by "doc=Lnu" [BS96], "doc=dtu" [SP99].

In addition to these vector-space schemes, we also considered probabilistic models such as the Okapi model [RB00]. As shown in Table 2, this model includes some constants fixed as b=0.7, k_1=1.5 (French), b=0.7, k_1=1.5 (Portuguese), b=0.75, k_1=1.2 (Hungarian), and b=0.5, k_1=1.2 (German), while $avdl$ indicates the mean document length (values are given in Table 1). As a second probabilistic approach, we implemented the GL2 approach taken from the *Divergence from Randomness* (DFR) framework [AvR02], based on combining the two information measures formulated below:

$$w_{ij} = Inf_{ij}^1(tf) \cdot Inf_{ij}^2(tf) = -log_2\left[Prob_{ij}^1(tf)\right] \cdot (1 - Prob_{ij}^2(tf))$$

where w_{ij} indicates the indexing weight attached to term t_j in document D_i, $Prob^1_{ij}(tf)$ is the probability of finding tf occurrences of the indexing unit t_j in the document D_i. On the other hand, $Prob^2_{ij}(tf)$ is the probability of encountering a new occurrence of t_j in the document given that we have already found tf occurrences of this indexing unit. Within this framework, the GL2 model is based on the following formulae:

$$Prob^1_{ij}(tf) = [1/(1+\lambda_j)] \cdot [\lambda_j/(1+\lambda_j)]^{tfn_{ij}} \text{ with } \lambda_j = tc_j/n \quad (1)$$

$$Prob^2_{ij}(tf) = tfn_{ij} \,/\, (tfn_{ij}+1) \text{ with} \quad (2)$$

$$tfn_{ij} = tf_{ij} \cdot \log_2 [1 + ((C \cdot mean\ dl)/l_i)] \quad (3)$$

where l_i the number of indexing terms included in the representation of D_i, tc_j represents the number of occurrences of term t_j in the collection, C is a constant fixed at 1.25, and $mean\ dl$ (mean document length) depends on the corpus (values given in Table 1).

5 Stemming Strategies

In our point of view it is important to develop a simple approach, one that does not require a dictionary or any other sophisticated data structures or processing. We also believe that a good IR system stemming procedure should focus mainly on nouns and adjectives, thus ignoring various verb forms (although past participles could be an exception to this rule). Given this assumption, our stemming approach tried to remove morphological variations associated with number (singular vs. plural), gender (masculine or feminine), and various grammatical cases (nominative, accusative, ablative, etc.). For verbal forms we ignored variations which are usually too numerous, while for adjectives we did not attempt to remove comparative and superlative suffixes (e.g., "larger," "largest"), forms that are less frequently used.

An analysis of the grammar of any given language however usually reveals numerous inflectional rules, some of which are used for only one or a few words (e.g., "child" and "children" or "foot" and "feet" in English). As for those languages having morphologies more complex than English, we could develop an even simpler stemmer, based only on a few but frequently used rules. For French, such a stemming approach (label "S-stemmer") is defined as follows.

```
For words of six or more letters
   if final letters are '-aux' then replace -aux by -al
   if final letter is '-x' then remove '-x',
   if final letter is '-s' then remove '-s',
   if final letter is '-r' then remove '-r',
   if final letter is '-e' then remove '-e',
   if final letter is '-é' then remove '-é',
   if final two letters are the same, remove final letter
```

For example, the word "chevaux" (horses) is reduced to "cheval" (horse) and the words

”baronnes” (baronesses) or ”barons” are all reduced to the same stem ”baron”. As a variant for French, we would suggest removing other inflections and also certain derivational suffixes. Labeled ”UniNE” in our experiments, this stemming method is composed of 27 rules (see `www.unine.ch/info/clef/`).

For Portuguese, our suggested stemmer tries to remove inflections attached to both nouns and adjectives, based on rules for the plural form (10 rules) and feminine form (13 rules). In Portuguese as in English the usual plural form is obtained by adding an ’-s’ (e.g., ”amigo” and ”amigos” (friend)). This suffix is also used for adjectives. There are of course various exceptions to the general rule (e.g., ”mar” and ”mares” (sea), ”fuzil” and ”fuzis” (gun), and for the adjective ”fácil” (easy), its plural form is ”fáceis”). The feminine form is usually obtained by replacing the final ’-o’ by an ’-a’ (e.g., ”americano” and ”americana”), but there are various exceptions to be taken into account (e.g., ”inglês” (British) becomes ”inglesa” in the feminine, ”leão” (lion) becomes ”leoa” and ”professor” gives ”professora”)

For German our suggested stemmer incorporates 11 rules to remove both plural forms and grammatical case endings (e.g., those usually used to indicate the genitive case by employing an ’-s’ or ’-es’ as in ”Staates” (of the state), ”Mannes” (of the man)). In German the plural form is denoted using a variety of endings such as ’-en’ (e.g., ”Motor”, ”Motoren” (engine)), ’-er’, ’-e’ (e.g., ”Jahr”, ”Jahre” (year)) or ’-n’ (e.g., ”Name”, ”Namen”). Plural forms also use diacritic characters (e.g., ”Apfel” (apple) becomes ”Äpfel”) or in conjunction with a suffix (e.g., ”Haus” and ”Häuser” (house)). Also frequently used are the suffixes ’-en’ or ’-n’ to indicate grammatical cases or for adjectives (e.g., ”ï einen guten Mann” (a good man) in the accusative singular form).

As with Finnish, Hungarian makes use of a greater number of grammatical cases (usually 18) than does German (four cases). Each case has its own unambiguous suffix however; e.g. the noun ”house” (”hàz” in nominative) may appear as ”hàzat” (accusative case, as in ”(I see) the house”), ”hàzakat” (accusative plural case, as in ”(I see) the houses”), ”hàzamat” (”ï my house”) or ”hàzamait” (”... my houses”). In this language the general construction used for nouns is as follows: ’stem’ ’possessive marker’ ’plural’ ’case’ as in ’hàz’ + ’ak’ + ’at’ (in which the letter ’a’ is introduced to facilitate better pronunciation because ”hàzkt” could be difficult to pronounce). Our suggested ”UniNE” stemming procedure for the plural in this language is based on two rules, plus there are 17 rules for removing various possessive suffixes and 21 rules for removing case markers. In a lighter stemming procedure, we would ignore the possessive marker (under the assumption that such suffixes are infrequently used and in an effort to reduce the number of conflation errors). Moreover, in order to automatically remove the most frequent cases we would apply only 13 rules.

Compared to the 260 rules used by Lovins or the 60 by Porter in their stemmers proposed for the English language, the stemmers we suggest could be viewed as light stemmers for languages having more complex morphologies than English. These stemmers are available at `www.unine.ch/info/clef/`. As an alternative to our light stemmers, we might also employ a more aggressive stemmer, taken from among those found within Porter’s family (available for the French, Portuguese and German languages at `snowball.tartarus.org/`). In the next section, we will evaluate these various stem-

ming approaches and their resultant retrieval effectiveness.

6 Evaluation

To measure retrieval performance, we adopted mean average precision (MAP) as computed by TREC_EVAL. To determine whether or not any given search strategy might be better than another, we applied a statistical test. More precisely, we stated the null hypothesis (denoted H_0) specifying that both retrieval schemes achieved similar performance levels (MAP), and this hypothesis would be rejected at a significance level fixed at α = 5% (two-tailed test). In this paper we have underlined any statistically significant differences that result from a two-sided non-parametric bootstrap test [Sav97].

6.1 IR Models Evaluation

Based on this evaluation methodology, Table 3 depicts the MAP for the French or Portuguese collections, using different stemming approaches. The same information is given in Table 4 for the Hungarian and German corpora. These experiments show that the Okapi probabilistic model usually produces the best retrieval performance (depicted in bold) across the different languages. The Hungarian corpus without stemming is an exception to this finding, for which the MAP difference between the "dtu-dtn" approach (0.1980) and the Okapi model (0.1957) is not however statistically significant (and thus we did not underline this value). Moreover, when considering the French, Portuguese and German corpora, the differences between the Okapi model and other IR models are statistically significant.

For the Hungarian corpus, the difference between the two probabilistic schemes (GL2 and Okapi) and the two best performing vector-processing models ("Lnu-ltc" and "dtu-dtn") is not statistically significant.

6.2 Nonstemming vs. Stemming

In this section we would like to apply a different point of view in order to verify whether or not a given stemming procedure might statistically improve mean average precision. To verify the effectiveness of this approach we adapted retrieval performance without stemming as the baseline (MAP depicted under the label "None" in Tables 3 and 4). For the French collection, all three stemming approaches performed better statistically than the baseline, for the nine IR models. After averaging the percentage of enhancement across these nine models, we found an average increase of 35% when using the UniNE stemmer, 30.5% with Porter's scheme, and 27.3% for the "S-stemmer".

With the Portuguese and German corpora, we found similar conclusions; with the two

	Mean average precision						
Model	French None	French UniNE	French S-stem.	French Porter	Portug. None	Portug. UniNE	Portug. Porter
Okapi	**0.2260**	**0.3045**	**0.2858**	**0.2978**	**0.2238**	**0.2873**	**0.2610**
GL2	0.2125	0.2918	0.2739	0.2878	0.2182	0.2755	0.2502
Lnu-ltc	0.2112	0.2933	0.2717	0.2808	0.1989	0.2611	0.2296
dtu-dtn	0.2062	0.2780	0.2611	0.2758	0.2096	0.2571	0.2189
atn-ntc	0.2088	0.2755	0.2603	0.2695	0.2049	0.2458	0.2128
ltn-ntc	0.1945	0.2466	0.2402	0.2371	0.1758	0.2149	0.1831
lnc-ltc	0.1545	0.2233	0.2080	0.2131	0.1519	0.1811	0.1607
ltc-ltc	0.1461	0.1975	0.1879	0.1922	0.1433	0.1625	0.1415
ntc-ntc	0.1462	0.1918	0.1807	0.1758	0.1344	0.1553	0.1422

Table 3: MAP of various IR models applying different stemming strategies (French & Portuguese corpus)

	Mean average precision					
Model	Hungarian None	Hungarian Light	Hungarian UniNE	German None	German UniNE	German Porter
Okapi	0.1957	**0.2988**	**0.3076**	**0.3552**	**0.3931**	**0.4058**
GL2	0.1883	0.2905	0.2964	0.3464	0.3805	0.3934
Lnu-ltc	0.1887	0.2913	0.2868	0.3357	0.3638	0.3793
dtu-dtn	**0.1980**	0.2857	0.2900	0.3357	0.3671	0.3826
atn-ntc	0.1794	0.2651	0.2755	0.3381	0.3653	0.3789
ltn-ntc	0.1919	0.2556	0.2567	0.3184	0.3421	0.3573
lnc-ltc	0.1616	0.2188	0.2153	0.2757	0.2983	0.3032
ltc-ltc	0.1675	0.2207	0.2183	0.2575	0.2773	0.2891
ntc-ntc	0.1713	0.2162	0.2079	0.2510	0.2649	0.2759

Table 4: MAP of various IR models applying different stemming strategies (Hungarian & German corpus)

stemming procedures always performing statistically better than the search done without stemming. When computing percentages of the MAP differences across the nine IR models, we found the UniNE stemmer would improve MAP by 22% on average for the Portuguese collection and by 8.4% for the German corpus. Using the same baseline, Porter's stemmer improved MAP by 7.7% on average for the Portuguese collection, and by 12.4% for the German corpus.

For the Hungarian corpus, both stemming approaches improved the MAP when compared to the nonstemming approach (on average by 42.8% for UniNE stemmer, and 42.2% for the light stemming scheme). Both stemmers did indeed improve MAP statistically compared to an indexing scheme that ignored stemming.

6.3 Comparing Different Stemmers

It is assumed that stemming usually improves retrieval performance (even though the difference might not always be statistically significant) on the one hand, and on the other, different stemmers tend to produce similar results. To investigate this issue we compared the retrieval effectiveness produced by the various stemmers.

Using the "S-stemmer" retrieval performance as a baseline, for the French collection Porter's stemmer improved by 2.5% on average (computed from the nine IR models). These differences are however not statistically significant. The UniNE stemmer showed an average enhancement of 6%, and this difference was statistically significant for the Okapi, GL2, and "dtu-dtn" IR schemes. While performance differences between Porter and UniNE always favored the second (+3.5% in average), these variations were not however statistically significant.

For Portuguese, the situation is relatively similar. Using the UniNE stemmer as a baseline, Porter's approach resulted in lower MAP (-11.8% in average across the nine IR models). Moreover, for the 5 IR models, the differences were also statistically significant. Thus for both French and Portuguese, different stemmers would result in IR performances with statistically significant differences. Moreover, for these languages at least a light stemming approach seemed to be more effective than a stemming approach that tried to remove more suffixes.

For German, Porter's stemmer provided better retrieval performance than did the UniNE scheme (average difference of 3.7% over all IR models). The difference between these two stemming schemes however was never statistically significant. Finally for Hungarian, the difference between the two suggested stemming methods is very small (0.3% on average), and not statistically significant.

When performing high precision searches, we assumed that the light stemming approach would produce better results. To verify this hypothesis, we computed the retrieval precision for five documents from the French corpus, and then compared the three stemming approaches (mean precision depicted in Table 5). This data did not show any enhancement over the light stemming approach (evaluation given under the label "S-stemmer") or a scheme ignoring stemming (under the label "None"). The other two stemming approaches

	Precision after 5 documents			
	None	S-stemmer	UniNE	Porter
Okapi	**0.5040**	0.5280	**0.5480**	**0.5400**
GL2	0.4840	0.5200	0.5280	0.5240
Lnu-ltc	0.4960	**0.5320**	0.5200	0.5160
dtu-dtn	0.4320	0.4720	0.4840	0.4720
atn-ntc	0.4800	0.5120	0.5040	0.5120
ltn-ntc	0.4560	0.4840	0.4600	0.4720
lnc-ltc	0.3960	0.4480	0.4280	0.4240
ltc-ltc	0.3240	0.3520	0.3480	0.3680
ntc-ntc	0.3360	0.3640	0.3600	0.3600

Table 5: Mean precision after 5 documents (French corpus)

did however seem to show better results. The differences in performance between the "S-stemmer" and the others were never statistically significant.

7 Conclusion

We have proposed and analyzed various stemming approaches based on four different languages, and our experiments have demonstrated that the Okapi probabilistic model produces the best retrieval performance. Moreover, the differences between the Okapi and other IR models are statistically significant for the French, Portuguese and German corpora.

A second set of experiments clearly shows that a stemming procedure improves retrieval effectiveness for those European languages belonging to either the Latin (French, Portuguese), Germanic (German) or Finno-Ugrian (Hungarian) families. For these same four European languages, differences in retrieval performance are significant from a statistical point of view and favor searches performed with a stemmer.

When comparing different stemming strategies, it seems that a light stemming approach (one that tries to automatically remove the most frequently used inflectional suffixes) produces better MAP than does a more aggressive stemmer. Moreover, for some IR models, the difference between these two stemming schemes could be statistically significant and in favor of a light stemming solution. For the German and the Hungarian languages, the performance difference between the stemmers is not statistically significant. Finally, based on our experiments we cannot confirm that a light stemmer would be more effective for high precision searches, at least for the French language.

Acknowledgments. This research was supported in part by the Swiss National Science Foundation under Grant #200020-103420.

References

[AvR02] G. Amati and C.J. van Rijsbergen. Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. *ACM - Transactions on Information Systems*, 20:357–389, 2002.

[BR04] M. Braschler and B. Ripplinger. How Effective is Stemming and Decompounding for German Text Retrieval? *IR Journal*, 7:291–316, 2004.

[BS96] Singhal A. Mitra M. Buckley, C. and G. Salton. New Retrieval Approaches using SMART. In *Proceedings TREC-4*, pages 25–48, 1996.

[DNO04] Ferro N. Melucci M. Di Nunzio, G.M. and N. Orio. Experiments to Evaluate Probabilistic Models for Automatic Stemmer Generation and Query Word Translation. In Braschler M. Gonzalo J. Kluck M. Peters, C., editor, *Comparative Evaluation of Multilingual Information Access Systems*, Lecture Notes in Computer Science: Vol. 3237, pages 220–235, Heidelberg, 2004. Springer.

[Har91] D. Harman. How Effective is Suffixing? *Journal of the American Society for Information Science*, 42:7–15, 1991.

[Hul96] D. Hull. Stemming Algorithms: A Case Study for Detailed Evaluation. *Journal of the American Society for Information Science*, 47:70–84, 1996.

[KJ04] Laurikkala J. Jarvelin K. Korenius, T. and M. Juhola. Stemming and Lemmatization in the Clustering of Finnish Text Documents. In *Proceedings of the ACM-CIKM*, pages 625–633, 2004.

[Klu04] M. Kluck. The GIRT Data in the Evaluation of CLIR Systems - From 1997 Until 2003. In Braschler M. Gonzalo J. Peters, C. and M. Kluck, editors, *Comparative Evaluation of Multilingual Information Access Systems*, Lecture Notes in Computer Science: Vol. 3237, pages 376–390, Heidelberg, 2004. Springer.

[Lov68] J.B. Lovins. Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*, 11:22–31, 1968.

[MM04] P. McNamee and J. Mayfield. Character N-gram Tokenization for European Language Text Retrieval. *IR Journal*, 7:73–97, 2004.

[PM05] Clough P.D. Gonzalo J. Jones G.J.F. Kluck M. Peters, C. and B. Magnini. *Multilingual Information Access for Text, Speech and Images: Results of the Fifth CLEF Evaluation Campaign*. Lecture Notes in Computer Science: Vol. 3491. Springer, Berlin, 2005.

[Por80] M.F. Porter. An Algorithm for Suffix Stripping. *Program*, 14:130–137, 1980.

[RB00] Walker S. Robertson, S. E. and M. Beaulieu. Experimentation as a Way of Life: Okapi at TREC. *Information Processing & Management*, 36:95–108, 2000.

[Sav93] J. Savoy. Stemming of French Words Based on Grammatical Category. *Journal of the American Society for Information Science*, 44:1–9, 1993.

[Sav97] J. Savoy. Statistical Inference in Retrieval Effectiveness Evaluation. *Information Processing & Management*, 33:495–512, 1997.

[Sav04] J. Savoy. Report on CLEF-2003 Monolingual Tracks: Fusion of Probabilistic Models for Effective Monolingual Retrieval. In Braschler M. Gonzalo J. Peters, C. and M. Kluck, editors, *Comparative Evaluation of Multilingual Information Access Systems*, Lecture Notes in Computer Science: Vol. 3237, pages 322–336, Heidelberg, 2004. Springer.

[SP99] Choi J. Hindle D. Lewis D.D. Singhal, A. and F. Pereira. AT&T at TREC-7. In *Proceedings TREC-7*, pages 239–251, 1999.

[Spr92] R. Sproat. *Morphology and Computation*. The MIT Press, Cambridge, 1992.

[Tom04] S. Tomlinson. Lexical and Algorithmic Stemming Compared for 9 European Languages with Humminbird^{TM}SearchServer at CLEF 2003. In Braschler M. Gonzalo J. Kluck M. Peters, C., editor, *Comparative Evaluation of Multilingual Information Access Systems*, Lecture Notes in Computer Science: Vol. 3237, pages 286–300, Heidelberg, 2004. Springer.

[XC98] J. Xu and B. Croft. Corpus-Based Stemming Using Cooccurrence of Word Variants. *ACM-Transactions on Information Systems*, 16:61–81, 1998.

GI-Edition Lecture Notes in Informatics

P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.

P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.

P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.

P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.

P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.

P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.

P-8 Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.

P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.

P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.

P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.

P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.

P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.

P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Busi-ness 2002.

P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.

P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund.

P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund (Ergänzungsband).

P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.

P-22 Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.

P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit

P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen

P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003

P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web

P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin

P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen

P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems

P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications

P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometrics and Electronic Signatures

P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003

P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft

P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)

P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)

P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik

P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik

P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003

P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003

P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004

P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing

P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste

P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistance, Scalability, Transactions – Database Mechanisms for Mobile Applications

P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services

P-45 Bernhard Rumpe, Wofgang Hesse (Hrsg.): Modellierung 2004

P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment

P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society

P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications

P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven

P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik

P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004

P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration

P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Adhoc-Netze Medienintegration

P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government

P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen

P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality

P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments

P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung

P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen

P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit

P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications

P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005

P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web

P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik

P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)

P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)

P-69 Robert Hirschfeld, Ryszard Kowalcyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005

P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)

P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005

P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment

P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): “Heute schon das Morgen sehen“

P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology

P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture

P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz

P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006

P-78 K.-O. Wenkel, P. Wagner, M. Morgenstern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel

P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006

P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce

P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS´06

P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006

P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics

P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications

P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems

P-86 Robert Krimmer (Ed.): Electronic Voting 2006

P-87 Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik

P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006

P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur

P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006

P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006

P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1

P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2

P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen

P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies

P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices

P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik

P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik

P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten

P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments

P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)

P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme

P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007

P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications

P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures

P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1

P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2

P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik

P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis

P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)

P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics

P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007

P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1st International Working Conference on Business Process and Services Computing BPSC 2007

P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and manegement The 4th International Conference on Grid Service Engineering and Management GSEM 2007

P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on ehealth 2007

P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications

P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges

P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik

P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)

P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)
9th Workshop on Parallel Systems and Algorithms (PASA)
Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA

P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)
Unternehmens-IT: Führungsinstrument oder Verwaltungsbürde
Referate der 28. GIL Jahrestagung

P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)
10th Workshop Software Reengineering (WSR 2008)

P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)
Modellierung 2008

P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)
Sicherheit 2008
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
2.-4. April 2008
Saarbrücken, Germany

P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)
Sigsand-Europe 2008
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems

P-130 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung

P-131 Robert Krimmer, Rüdiger Grimm (Eds.)
3rd International Conference on Electronic Voting 2008
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)
DeLFI 2008:
Die 6. e-Learning Fachtagung Informatik

P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 1

P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 2

P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)
Didaktik der Informatik – Aktuelle Forschungsergebnisse

P-136 Andreas Beyer, Michael Schroeder (Eds.)
German Conference on Bioinformatics GCB 2008

P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2008: Biometrics and Electronic Signatures

P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)
Synergien durch Integration und Informationslogistik
Proceedings zur DW2008

P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)
Industrialisierung des Software-Managements
Fachtagung des GI-Fachausschusses Management der Anwendungsentwick-lung und -wartung im Fachbereich Wirtschaftsinformatik

P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)
IMF 2008 - IT Incident Management & IT Forensics

P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)
Modellierung betrieblicher Informations-systeme (MobIS 2008)
Modellierung zwischen SOA und Compliance Management

P-142 R. Bill, P. Korduan, L. Theuvsen, M. Morgenstern (Hrsg.)
Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung

P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.)
Software Engineering 2009
Fachtagung des GI-Fachbereichs Softwaretechnik

P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.)
Datenbanksysteme in Business, Technologie und Web (BTW)

P-145 Knut Hinkelmann, Holger Wache (Eds.)
WM2009: 5th Conference on Professional Knowledge Management

P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung
4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)

P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.)
Business Process, Services Computing and Intelligent Service Management
BPSC 2009 · ISM 2009 · YRW-MBP 2009

P-148 Christian Erfurth, Gerald Eichler, Volkmar Schau (Eds.)
9th International Conference on Innovative Internet Community Systems
I²CS 2009

P-149 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
2. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

P-150 Jürgen Münch, Peter Liggesmeyer (Hrsg.)
Software Engineering
2009 - Workshopband

P-151 Armin Heinzl, Peter Dadam, Stefan Kirn, Peter Lockemann (Eds.)
PRIMIUM
Process Innovation for Enterprise Software

P-152 Jan Mendling, Stefanie Rinderle-Ma, Werner Esswein (Eds.)
Enterprise Modelling and Information Systems Architectures
Proceedings of the 3rd Int'l Workshop EMISA 2009

P-153 Andreas Schwill, Nicolas Apostolopoulos (Hrsg.)
Lernen im Digitalen Zeitalter
DeLFI 2009 – Die 7. E-Learning Fachtagung Informatik

P-154 Stefan Fischer, Erik Maehle Rüdiger Reischuk (Hrsg.)
INFORMATIK 2009
Im Focus das Leben

P-155 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2009:
Biometrics and Electronic Signatures
Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

P-156 Bernhard Koerber (Hrsg.)
Zukunft braucht Herkunft
25 Jahre »INFOS – Informatik und Schule«

P-157 Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber, Peter Stadler (Eds.)
German Conference on Bioinformatics 2009

P-158 W. Claupein, L. Theuvsen, A. Kämpf, M. Morgenstern (Hrsg.)
Precision Agriculture Reloaded – Informationsgestützte Landwirtschaft

P-159 Gregor Engels, Markus Luckey, Wilhelm Schäfer (Hrsg.)
Software Engineering 2010

P-160 Gregor Engels, Markus Luckey, Alexander Pretschner, Ralf Reussner (Hrsg.)
Software Engineering 2010 – Workshopband
(inkl. Doktorandensymposium)

P-161 Gregor Engels, Dimitris Karagiannis Heinrich C. Mayr (Hrsg.)
Modellierung 2010

P-162 Maria A. Wimmer, Uwe Brinkhoff, Siegfried Kaiser, Dagmar Lück-Schneider, Erich Schweighofer, Andreas Wiebe (Hrsg.)
Vernetzte IT für einen effektiven Staat
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2010

P-163 Markus Bick, Stefan Eulgem, Elgar Fleisch, J. Felix Hampe, Birgitta König-Ries, Franz Lehner, Key Pousttchi, Kai Rannenberg (Hrsg.)
Mobile und Ubiquitäre Informationssysteme
Technologien, Anwendungen und Dienste zur Unterstützung von mobiler Kollaboration

P-165 Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.)
10th International Conference on Innovative Internet Community Systems (I²CS) – Jubilee Edition 2010 –

P-166 Paul Müller, Bernhard Neumair,
Gabi Dreo Rodosek (Hrsg.)
3. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

The titles can be purchased at:

Köllen Druck + Verlag GmbH
Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn
Fax: +49 (0)228/9898222
E-Mail: druckverlag@koellen.de