International Journal on Advances in Networks and Services

IARIA

2013 vol. 6 nr. 1&2

The *International Journal on Advances in Networks and Services* is published by IARIA.
ISSN: 1942-2644
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Networks and Services
Volume 6, Number 1 & 2, 2013

David Boyle, Tyndall National Institute, University College Cork, Ireland
Mahmoud Brahimi, University of Msila, Algeria
Marco Bruti, Telecom Italia Sparkle S.p.A., Italy
Dumitru Burdescu, University of Craiova, Romania
Diletta Romana Cacciagrano, University of Camerino, Italy
Maria-Dolores Cano, Universidad Politécnica de Cartagena, Spain
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Eduardo Cerqueira, Federal University of Para, Brazil
Patryk Chamuczyński, Radytek, Poland
Bruno Chatras, Orange Labs, France
Marc Cheboldaeff, T-Systems International GmbH, Germany
Kong Cheng, Telcordia Research, USA
Dickson Chiu, Dickson Computer Systems, Hong Kong
Andrzej Chydzinski, Silesian University of Technology, Poland
Hugo Coll Ferri, Polytechnic University of Valencia, Spain
Noelia Correia, University of the Algarve, Portugal
Noël Crespi, Institut Telecom, Telecom SudParis, France
Paulo da Fonseca Pinto, Universidade Nova de Lisboa, Portugal
Philip Davies, Bournemouth and Poole College / Bournemouth University, UK
Carlton Davis, École Polytechnique de Montréal, Canada
Claudio de Castro Monteiro, Federal Institute of Education, Science and Technology of Tocantins, Brazil
João Henrique de Souza Pereira, University of São Paulo, Brazil
Javier Del Ser, Tecnalia Research & Innovation, Spain
Behnam Dezfouli, Universiti Teknologi Malaysia (UTM), Malaysia
Mari Carmen Domingo, Barcelona Tech University, Spain
Daniela Dragomirescu, LAAS-CNRS, University of Toulouse, France
Jean-Michel Dricot, Université Libre de Bruxelles, Belgium
Wan Du, Nanyang Technological University (NTU), Singapore
Matthias Ehmann, Universität Bayreuth, Germany
Wael M El-Medany, University Of Bahrain, Bahrain
Imad H. Elhajj, American University of Beirut, Lebanon
Gledson Elias, Federal University of Paraíba, Brazil
Joshua Ellul, Imperial College, London
Rainer Falk, Siemens AG - Corporate Technology, Germany
Károly Farkas, Budapest University of Technology and Economics, Hungary
Huei-Wen Ferng, National Taiwan University of Science and Technology - Taipei, Taiwan
Gianluigi Ferrari, University of Parma, Italy
Mário F. S. Ferreira, University of Aveiro, Portugal
Bruno Filipe Marques, Polytechnic Institute of Viseu, Portugal
Ulrich Flegel, HFT Stuttgart, Germany
Juan J. Flores, Universidad Michoacana, Mexico
Ingo Friese, Deutsche Telekom AG - Berlin, Germany
Sebastian Fudickar, University of Potsdam, Germany
Stefania Galizia, Innova S.p.A., Italy
Ivan Ganchev, University of Limerick, Ireland
Miguel Garcia, Universitat Politecnica de Valencia, Spain

Emiliano Garcia-Palacios, Queens University Belfast, UK
Gordana Gardasevic, University of Banja Luka, Bosnia and Herzegovina
Marc Gilg, University of Haute-Alsace, France
Debasis Giri, Haldia Institute of Technology, India
Markus Goldstein, DFKI (German Research Center for Artificial Intelligence GmbH), Germany
Luis Gomes, Universidade Nova Lisboa, Portugal
Anahita Gouya, Solution Architect, France
Mohamed Graiet, Institut Supérieur d'Informatique et de Mathématique de Monastir, Tunisie
Christos Grecos, University of West of Scotland, UK
Vic Grout, Glyndwr University, UK
Yi Gu, University of Tennessee, Martin, USA
Angela Guercio, Kent State University, USA
Xiang Gui, Massey University, New Zealand
Mina S. Guirguis, Texas State University - San Marcos, USA
Tibor Gyires, School of Information Technology, Illinois State University, USA
Keijo Haataja, University of Eastern Finland, Finland
Gerhard Hancke, Royal Holloway / University of London, UK
R. Hariprakash, Arulmigu Meenakshi Amman College of Engineering, Chennai, India
Go Hasegawa, Osaka University, Japan
Hermann Hellwagner, Klagenfurt University, Austria
Eva Hladká, CESNET & Masaryk University, Czech Republic
Hans-Joachim Hof, Munich University of Applied Sciences, Germany
Razib Iqbal, Amdocs, Canada
Abhaya Induruwa, Canterbury Christ Church University, UK
Muhammad Ismail, University of Waterloo, Canada
Vasanth Iyer, Florida International University, Miami, USA
Peter Janacik, Heinz Nixdorf Institute, University of Paderborn, Germany
Robert Janowski, Warsaw School of Computer Science, Poland
Imad Jawhar, United Arab Emirates University, UAE
Aravind Kailas, University of North Carolina at Charlotte, USA
Mohamed Abd rabou Ahmed Kalil, Ilmenau University of Technology, Germany
Kyoung-Don Kang, State University of New York at Binghamton, USA
Omid Kashefi, Iran University of Science and Technology, Iran
Sarfraz Khokhar, Cisco Systems Inc., USA
Vitaly Klyuev, University of Aizu, Japan
Jarkko Kneckt, Nokia Research Center, Finland
Dan Komosny, Brno University of Technology, Czech Republic
Ilker Korkmaz, Izmir University of Economics, Turkey
Tomas Koutny, University of West Bohemia, Czech Republic
Evangelos Kranakis, Carleton University - Ottawa, Canada
Lars Krueger, T-Systems International GmbH, Germany
Kae Hsiang Kwong, MIMOS Berhad, Malaysia
KP Lam, University of Keele, UK
Birger Lantow, University of Rostock, Germany
Hadi Larijani, Glasgow Caledonian Univ., UK
Annett Laube-Rosenpflanzer, Bern University of Applied Sciences, Switzerland

Angelos Lazaris, University of Southern California (USC), USA
Gyu Myoung Lee, Institut Telecom, Telecom SudParis, France
Ying Li, Peking University, China
Shiguo Lian, Orange Labs Beijing, China
Chiu-Kuo Liang, Chung Hua University, Hsinchu, Taiwan
Wei-Ming Lin, University of Texas at San Antonio, USA
David Lizcano, Universidad a Distancia de Madrid, Spain
Chengnian Long, Shanghai Jiao Tong University, China
Jonathan Loo, Middlesex University, UK
Edmo Lopes Filho, Algar Telecom, Brazil
Pascal Lorenz, University of Haute Alsace, France
Albert A. Lysko, Council for Scientific and Industrial Research (CSIR), South Africa
Pavel Mach, Czech Technical University in Prague, Czech Republic
Elsa María Macías López, University of Las Palmas de Gran Canaria, Spain
Damien Magoni, University of Bordeaux, France
Ahmed Mahdy, Texas A&M University-Corpus Christi, USA
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
Gianfranco Manes, University of Florence, Italy
Sathiamoorthy Manoharan, University of Auckland, New Zealand
Moshe Timothy Masonta, Council for Scientific and Industrial Research (CSIR), Pretoria, South Africa
Hamid Menouar, QU Wireless Innovations Center - Doha, Qatar
Guowang Miao, KTH, The Royal Institute of Technology, Sweden
Mohssen Mohammed, University of Cape Town, South Africa
Miklos Molnar, University Montpellier 2, France
Lorenzo Mossucca, Istituto Superiore Mario Boella, Italy
Jogesh K. Muppala, The Hong Kong University of Science and Technology, Hong Kong
Katsuhiro Naito, Mie University, Japan
Deok Hee Nam, Wilberforce University, USA
Sarmistha Neogy, Jadavpur University- Kolkata, India
Rui Neto Marinheiro, Instituto Universitário de Lisboa (ISCTE-IUL), Instituto de Telecomunicações, Portugal
David Newell, Bournemouth University - Bournemouth, UK
Armando Nolasco Pinto, Universidade de Aveiro / Instituto de Telecomunicações, Portugal
Jason R.C. Nurse, University of Oxford, UK
Kazuya Odagiri, Yamaguchi University, Japan
Máirtín O'Droma, University of Limerick, Ireland
Rainer Oechsle, University of Applied Science, Trier, Germany
Henning Olesen, Aalborg University Copenhagen, Denmark
Jose Oscar Fajardo, University of the Basque Country, Spain
Constantin Paleologu, University Politehnica of Bucharest, Romania
Eleni Patouni, National & Kapodistrian University of Athens, Greece
Harry Perros, NC State University, USA
Miodrag Potkonjak, University of California - Los Angeles, USA
Yusnita Rahayu, Universiti Malaysia Pahang (UMP), Malaysia
Yenumula B. Reddy, Grambling State University, USA
Oliviero Riganelli, University of Milano Bicocca, Italy
Patrice Rondao Alface, Alcatel-Lucent Bell Labs, Belgium

Teng Rui, National Institute of Information and Communication Technology, Japan
Antonio Ruiz Martinez, University of Murcia, Spain
George S. Oreku, TIRDO / North West University, Tanzania/ South Africa
Sattar B. Sadkhan, Chairman of IEEE IRAQ Section, Iraq
Husnain Saeed, National University of Sciences & Technology (NUST), Pakistan
Addisson Salazar, Universidad Politecnica de Valencia, Spain
Sébastien Salva, University of Auvergne, France
Ioakeim Samaras, Aristotle University of Thessaloniki, Greece
Luz A. Sánchez-Gálvez, Benemérita Universidad Autónoma de Puebla, México
Teerapat Sanguankotchakorn, Asian Institute of Technology, Thailand
José Santa, University of Murcia, Spain
Rajarshi Sanyal, Belgacom International Carrier Services, Belgium
Mohamad Sayed Hassan, Orange Labs, France
Thomas C. Schmidt, HAW Hamburg, Germany
Hans Scholten, Pervasive Systems / University of Twente, The Netherlands
Véronique Sebastien, University of Reunion Island, France
Jean-Pierre Seifert, Technische Universität Berlin & Telekom Innovation Laboratories, Germany
Sandra Sendra Compte, Polytechnic University of Valencia, Spain
Dimitrios Serpanos, Univ. of Patras and ISI/RC ATHENA, Greece
Xu Shao, Institute for Infocomm Research, Singapore
Roman Y. Shtykh, Rakuten, Inc., Japan
Salman Ijaz Institute of Systems and Robotics, University of Algarve, Portugal
Adão Silva, University of Aveiro / Institute of Telecommunications, Portugal
Florian Skopik, AIT Austrian Institute of Technology, Austria
Karel Slavicek, Masaryk University, Czech Republic
Vahid Solouk, Urmia University of Technology, Iran
Peter Soreanu, ORT Braude College, Israel
Pedro Sousa, University of Minho, Portugal
Vladimir Stantchev, SRH University Berlin, Germany
Radu Stoleru, Texas A&M University - College Station, USA
Lars Strand, Nofas, Norway
Stefan Strauβ, Austrian Academy of Sciences, Austria
Álvaro Suárez Sarmiento, University of Las Palmas de Gran Canaria, Spain
Masashi Sugano, School of Knowledge and Information Systems, Osaka Prefecture University, Japan
Young-Joo Suh, POSTECH (Pohang University of Science and Technology), Korea
Junzhao Sun, University of Oulu, Finland
David R. Surma, Indiana University South Bend, USA
Yongning Tang, School of Information Technology, Illinois State University, USA
Yoshiaki Taniguchi, Osaka University, Japan
Anel Tanovic, BH Telecom d.d. Sarajevo, Bosnia and Herzegovina
Olivier Terzo, Istituto Superiore Mario Boella - Torino, Italy
Tzu-Chieh Tsai, National Chengchi University, Taiwan
Samyr Vale, Federal University of Maranhão - UFMA, Brazil
Dario Vieira, EFREI, France
Natalija Vlajic, York University - Toronto, Canada
Lukas Vojtech, Czech Technical University in Prague, Czech Republic

International Journal on Advances in Networks and Services
Volume 6, Numbers 1 & 2, 2013

CONTENTS

Sun Behind Clouds - On Automatic Cloud Security Audits and a Cloud Audit Policy Language

Frank Doelitzscher, Thomas Ruebsamen, Tina Karbe, Martin Knahl, Christoph Reich
Cloud Research Lab
Furtwangen University, Furtwangen, Germany
{Frank.Doelitzscher, Thomas.Ruebsamen, Tina.Karbe, Martin.Knahl, Christoph.Reich}@hs-furtwangen.de

Nathan Clarke
Centre for Security, Communications & Network Research
Plymouth University
Plymouth PL4 8AA, United Kingdom
N.Clarke@plymouth.ac.uk

***Abstract*—Studies show that when it comes to an integration of Cloud computing into enterprises, chief information officers and management still see some dark Clouds on the horizon. The biggest one is the lack of security, which results in distrust and skepticism against the technology, mainly originating from an intransparency of Cloud environments. To increase this transparency, the Cloud Research Lab at Furtwangen University develops the Security Audit as a Service (SAaaS) architecture for Infrastructure as a Service Cloud environments. It is targeted to ensure that a desired security level is reached and maintained within a frequently changing Cloud infrastructure. Despite a traditional security audit, which includes a comprehensive and therefore time-consuming security check of a whole infrastructure, a Cloud security audit needs to be lightweight enough to be executed right after an infrastructure change occurred, and precisely target-oriented to perform an audit of the specific infrastructure components affected by this change. This is called a concurrent security audit. In this paper, a Cloud audit policy language for the SAaaS architecture gets presented. First, the design and implementation of the automated audit system of virtual machine images, which ensures legal and company policies, is described. Second, on-demand deployed software audit agents that maintain and validate the security compliance of running Cloud services, are discussed.**

***Keywords*—Cloud computing, security policies, Cloud audits, agents**

What concerns were expressed during the decision-making process to migrate to the Cloud?

Only asked of respondents whose company has hosted or Cloud-based services in use	Total	No. employees Fewer than 20	20 - 200	More than 200	Public	Private
Data security	82%	77%	85%	82%	89%	78%
Data privacy	69%	79%	63%	68%	70%	69%
Dependency upon internet access (availability and bandwidth)	51%	53%	52%	48%	49%	52%
Fear of loss of control/manageability	46%	35%	48%	54%	53%	43%
Confidence in the reliability of the vendors	36%	35%	33%	39%	25%	42%
Data sovereignty/jurisdiction	33%	37%	31%	30%	30%	34%
Cost of change/migration	31%	33%	33%	29%	26%	34%
Contract lock-in	29%	33%	26%	30%	32%	28%
Lack of clarity of impact of Cloud Services on business processes	27%	16%	28%	34%	32%	24%
Regulatory constraints	26%	16%	30%	30%	28%	25%
Contractual liability for services if SaI's are missed	20%	12%	17%	29%	9%	25%
Confidence in the vendors business capability	19%	19%	19%	20%	17%	20%
Confidence in knowing who to choose to supply service	18%	26%	20%	11%	19%	18%
Confidence in the clarity of charges (i.e. will they be cheaper than on-premise)	18%	16%	13%	23%	19%	17%
Lack of confidence in the business case to need Cloud Services	11%	9%	11%	13%	11%	11%
Lack of clarity in most appropriate Cloud deployment model	10%	7%	13%	9%	8%	11%
Lack of any advice from within the company to adopt	8%	9%	7%	9%	13%	6%
Lack of clarity in most effective service delivery model	7%	5%	7%	9%	8%	7%
Lack of any promotion or awareness by the people we buy IT from	2%	0%	0%	5%	4%	1%
Other	1%	2%	2%	0%	2%	1%
Base	153	43	54	56	53	100

Fig. 1: What concerns were expressed during the decision-making process to migrate to the Cloud? [6]

I. Introduction

This paper is the successor of the conference paper "Incident Detection for Cloud Environments" [1] presented at EMERGING 2011. In addition to the presentation of the Security Audit as a Service (SAaaS) architecture, the comprehensive definition of an automated virtual machine (VM) image audit system and the definition and presentation of a Cloud audit policy language, form a novel contribution for this extended journal paper.

Cloud vendors promise "infinite scalability and resources" combined with on-demand access from everywhere. This lets Cloud users quickly forget that there is still a real IT infrastructure behind a Cloud, and due to virtualization and multi-tenancy the complexity of these infrastructures is even increased compared to traditional data centers. This makes management of service provisioning, monitoring, backup, disaster recovery, security, etc. more complicated and, therefore, there is still a lack of trust in Cloud infrastructures. Enterprise analysts and researchers have identified Cloud specific security problems as the major research area in Cloud computing [2], [3], [4], [5]. The survey "Cloud Adoption and Trends for 2013" which was done amongst 250 CIOs and other IT executives at UK companies and public sector organizations states that security concerns are still the major issue, which hinders a broad industry acceptance of actually utilizing Cloud technologies (see Fig. 1). In fact, even for private Cloud solutions, where an enterprise runs its own Cloud IT infrastructure, security concerns are named as the major obstacles, which proves that there is a general lack of trust in Cloud computing security.

Security is a considerable challenge for Cloud environments due to its characteristics: seamless scalability, shared resources, multi-tenancy, access from everywhere, on-demand availability and third party hosting. Although existing industry recommendations (ITIL), standards (ISO 20000, ISO 27001:5, CobiT) and laws (e.g., Germanys Federal Data Protection Act) provide well established security and privacy rule sets for data center providers, research has shown that additional

regulations have to be defined for Cloud environments [2], [7]. The following examples of Cloud security incidents illustrate the need for Cloud computing security improvements:

- Hackers stole credentials of Salesforce.com's customers via phishing attacks (2007)
- T-Mobile customers lost data due to the "Sidekick disaster" of Microsoft Cloud (2009)
- Botnet incident at Amazon EC2 infected customer's computers and compromised their privacy (2009)
- Hotmail accounts were hacked due to technical flaws in Microsoft software (2010)
- Amazon customer services were unavailable for multiple days and data was lost due to a logical flaw in the Cloud storage design (2011)

Traditionally, IT infrastructure security audits are used to document a data-center's compliance to security best practices and laws. But, the major shortcoming of a traditional security audit is that it only provides a snapshot of an environments' security state at the performed audit time. This is adequate since classic IT infrastructures don't change that frequently. But because of the mentioned Cloud characteristics above, it is not sufficient for auditing a Cloud environment [1].

Beside the frequently changing infrastructure inside a Cloud, there is also a new dynamic in administrative tasks. The number of administrators of a traditional data centre is limited and they all are working under the same company security policy, while installing and maintaining machines. This can be completely different in a Cloud infrastructure. Public marketplaces for exchanging Cloud appliances such as, OpenNebula Marketplace [8], Amazon Web Services EC2 Management Console or the Amazon Web Services Marketplace [9] provide Cloud customers with an easy and efficient way of finding the right virtual machine image. But they also allow users to be administrators of their virtual machines, or upload and share their custom made VM images with other users. Although Cloud providers provide security guidelines [10] on how to prepare an image before releasing it to a marketplace, current research by Balduzzi [11], Bugiel [12] and Meer [13] shows that marketplace images are highly insecure due to old software versions or "forgotten" or restorable security credentials, such as SSH private keys. Users, uploading appliances are usually more or less anonymous. There is no way to easily determine whether a custom appliance is legit or maliciously manipulated. Images could contain rootkits, which are performing passive eavesdropping attacks such as traffic analysis, keylogging or transmission of user's data to external systems for industrial spying [11].

European and German data protection laws increase the necessity for users to re-validate the security status of virtual machines originating from preconfigured images by clearly putting the user who runs the image into responsibility (§3.7 German Data Protection Law, Art. 2d 4, European Guideline 95/46/EG) when data is processed in the Cloud. The Cloud user has to re-validate technical and organizational security measures taken by the Cloud provider initially at the beginning of a Cloud usage and periodically over time (§11 II - 4 German Data Protection Law).

To mitigate these problems, this work proposes the Security Audit as a Service architecture for Infrastructure as a Service (IaaS) Clouds. The contribution of this paper is structured in two parts: A) Description of an automatic Cloud audit architecture based on agents, which react on changes in a Cloud infrastructure. B) A Cloud audit policy language to describe security targets and enable automatic Cloud audits. It is shown that automatic audits of VMs and VM images can increase transparency and therefore security in Cloud computing environments.

In the remainder of this article, Section II - Use Cases of the SAaaS System, introduces the target scenarios, which this work aims to solve. Section III then gives an inside view into the process of automatic virtual machine image audits. Following, Section IV - A Cloud Audit Policy Language presents a comparison of existing security policy languages and introduces CAPL - a policy extension for the Cloud Infrastructure Management Interface (CIMI). Then, the Security Audit as a Service architecture is presented in Section V, which introduces the concept of using distributed agents to perform Cloud audits. An integration of the developed Cloud audit policy language is also shown. How the presented work increases security and transparency in Cloud environments is elaborated in Section VI - Evaluation. Section VII - Related Work, discusses related work on the topic of Cloud specific security problems, Cloud audits and other research similar to Security Audit as a Service, before Section VIII - Conclusion & Outlook wraps-up the paper and gives an outlook into future work.

II. Use Cases of the SAaaS System

The Security Audit as a Service system presented in this work covers three use cases:

1. Automated security audit: In this use case the SAaaS architecture gets used as a Software as a Service (SaaS) solution. It enables users to plan and perform security audits of their IT infrastructure on a regular basis. An audit can consist of regular vulnerability scans of a user's internet exposed systems (not necessarily Cloud instances). Results get automatically evaluated, post-processed and submitted as a security report in a standardized format to the user. Additionally, to simplify black box scans it is imaginable to deposit an entry credential (e.g., a ssh key pair) in the service so that the service can log in and perform internal security scans. While such systems already exist as appliances (e.g., Nessus appliance), especially small SMEs can profit from this service running in a Cloud since they only need to pay per scan. For the Cloud provider this service is valuable since computing resources are only allocated for the duration of a scan. Afterwards, the compute resources are released and made available for different tasks.

2. Monitoring and Audit of Cloud Instances: User VMs running in a Cloud infrastructure are equipped with a SAaaS agent. The user creates security policies defining the behaviour of this VM to be considered "normal", which VM components are to be monitored and how to alert the customer in case of system deviation from the defined manner. The status gets conditioned in a user-friendly format in a Web portal - the SAaaS security dashboard. This continuous

monitoring creates transparency about the security status of a user's Cloud VMs hence increasing the user's trust into the cloud environment.

3. Cloud Infrastructure Monitoring and Audit: The security status of the entire Cloud environment, especially the Cloud management system, access to customer data and data paths are monitored. Usage and communication behaviour profiles are created automatically and continuously analyzed for substantial changes. This way monitoring across different customers is used by the Cloud provider as well as a 3rd party, like a security service provider to ensure the overall cloud security status. Standardized interfaces enable security audits of the Cloud infrastructure, which can lead to a cloud security certification.

III. Auditing VMs of a Cloud

As previously mentioned, using third party appliance images from public marketplaces can pose a significant security risk. Therefore, not only running virtual machines need to be audited in a Cloud environment, but also virtual appliance images, from which virtual machines are created. This section describes requirements (see Section III-A), roles (see Section III-B), audit categories (see Section III-C) and a system architecture (see Section III-D) involved in solving the aforementioned issues. Virtual machine image auditing is regarded as a part of SAaaS use case 2 - Monitoring and Audit of Cloud Instances.

A. VM Auditing Requirements

To be able to automatically audit VM images, it is essential to describe the security and privacy requirements, in a machine understandable way. This is commonly achieved by the definition of security policies, transferring a requirement into a checklist of one or multiple testable conditions. To respect Cloud user's and provider's security requirements, both parties need to be able to create policies. A key factor for the success of such a system is the detailed and distinct definition of security policies. However, this is contrary to a short VM deployment process a Cloud user expects. Therefore, we propose to create a very easily operable, security policy generator, where Cloud users can define security policies in a human way of thinking, such as: "*The VM must be checked for malware*". Such simple policies could be supported by a graphical Web interface with templates utilizing check boxes or drop-down lists. This needs then to be translated into a machine understandable format, which results in the audit checks to be performed. The output of these checks needs to be translated back into a human understandable format, which will form the audit report submitted to the image creator, Cloud provider and image user. In summary, the following important audit requirements can be identified:

- VM images need to be audited in an automatic manner, to provide short response times to an image creator who wants to publish its image.
- The system needs to respect different security requirements from the image creator as well as the Cloud provider.
- The system needs to produce a human understandable output in case an image did not pass the security check, providing the image creator with information about what prohibited the image release so that he is able to fix it.
- Security policies need to be described in a machine understandable way.

B. VM Audit Roles

When it comes to auditing virtual machine appliances, there are a couple of different roles, which need to be considered: *appliance user*, *appliance creator*, *Cloud provider*, *audit service provider* and *audit tool provider*. These roles will be described in more detail in the following.

The **appliance user** is a customer of the Cloud provider obtaining the virtual machine images via the appliance store. The main concern of the appliance user is to make sure, that the VM complies with the company's IT security policy when using third party appliances. Such a security policy may include the necessity of malware checks (e.g., viruses, trojans, spyware, rootkits etc.), checks for undesirable software (e.g., games, file sharing software) but also a more detailed view on the operating system and services configuration of an appliance may be checked. For example, if there exist unprivileged system user accounts for running a Web server or if there are any leftover default passwords, which the appliance creator may have overlooked. The auditing's goal, from an appliance user's point of view, is to make sure a virtual appliance complies to his company's security policy, before the appliance is started and integrated in the company's IT infrastructure.

The **appliance creator** can be the Cloud provider himself or a customer of the Cloud provider. He creates individual VMs and shares them with other Cloud customers using an appliance store. Before publishing virtual appliances, the appliance creator has to make sure that there is no private data, which could compromise privacy (e.g., logs, browser cache, user information like names and addresses), left on the image. Another, often overlooked, aspect are non-securely deleted files on the image's file system. It is often possible to recover such files with little effort using file carving tools, like extundelete [14] or winundelete [15]. The auditing's goal from the appliance creator's point of view is therefore mostly to make sure policies, which prevent the disclosure of sensitive data, are used during auditing of the appliance before it gets published.

The **Cloud provider** provides the technical infrastructure for running the virtual machine image and also runs the appliance store. Providers usually have little or no interest in restricting the creation and publication of virtual machine appliances, as long as there is no violation of laws or terms of use. Such violations may include the intentional distribution of malware, intentionally misconfigured services or any form of illegal content, such as pirated software.

The two remaining roles **audit service provider** and **audit tool provider** have no immediate interest in auditing virtual machine appliances. They merely complete the auditing process by providing additional services and tools. The audit tool provider designs, develops and provides programs and services

TABLE I: Virtual Appliance Audit Categories

		Appliance User	Appliance Creator	Cloud Provider
Security	Malware	x		x
	Undesirable Software	x		
	Account Requirements	x		
	Login Requirements	x		
	Password Strength	x		
	Access Rights	x		x
	Service Misconfiguration	x		x
	Unwanted Service Combination	x		
Privacy	Browser Caches		x	
	Log Files		x	
	History Files		x	
	Insecurely Deleted Files		x	
Legal	Software Licenses	x	x	
	Illegal Content	x	x	x
	Customer Specific Requirements	x		

for auditing virtual machine images. The audit service provider is a specialist in auditing IT infrastructure and therefore has extensive knowledge about auditing procedures and methodologies, which he offers to the Cloud provider. He also provides the Cloud provider with work-flows, recommendations about the tools to use, knowledge about currently emerging threats to security and privacy as well as any additional auditing know-how.

C. Auditing Categories

The auditing categories identified by the authors are **security**, **privacy** and **legal concerns**. Multiple audit cases from these categories can be arbitrarily combined to form an auditing policy. Each of the previously described roles has a different view on the requirements the audit process has to fulfill. Table I illustrates this circumstance. The security category includes requirements regarding the absence of malware and otherwise undesirable software in the virtual appliance. Also, the preconfiguration of the appliance's services, like access rights on a file system level, the combination of services (e.g., mail daemons and network attached storage (NAS) service on the same appliance), insecure default service configurations, the use of insecure default passwords and login requirements (allowing remote administrator access with passwords) are the most common concerns. The privacy category includes mostly requirements, which should help preventing unintentional data loss. This includes leftovers from the appliance setup process, like log files, command line history or insecurely deleted files. Additionally, this category also includes data generated by end-user applications (e.g., browser caches). The legal category includes all sorts of compliance requirements, for example illegal content stored on the image.

The appliance user obviously has much more interest in secure and compliant appliances. One could argue that data loss while using the appliance because of misconfigured services or backdoor programs could be assigned to the privacy category, but this data loss arises because of security problems.

The appliance creator has mostly privacy concerns, when publishing virtual appliances. Checks for the previously described problems need to be thoroughly executed before publishing a virtual appliance. Also, not publishing preinstalled software licenses is a legal problem, which needs to be checked for.

The Cloud provider's main concern is protecting his own infrastructure. Therefore, the categories, which apply are security and legal. Checking virtual appliance images for malware may reduce the risk of malware spreading and hackers using the appliance store as a basis for their attacks. Checking for illegal content may also be necessary because the Cloud provider stores virtual appliances.

D. Automatic Auditing System Overview

Figure 2 shows an overview of our proposed auditing system for virtual appliances.

The *appliance store* is enhanced by an *Automatic Audit System* (AAS) for image auditing. The cloud infrastructure as well as the appliance store are run and maintained by the Cloud provider.

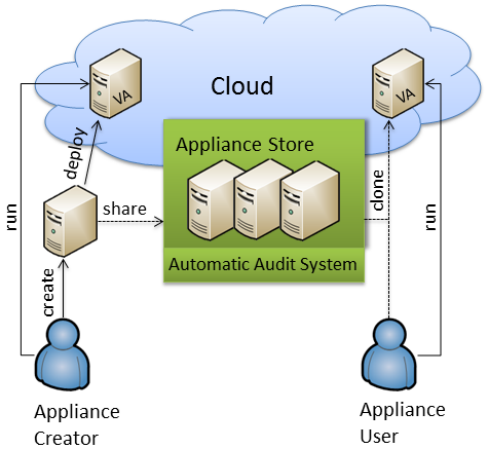


Fig. 2: Appliance Store with Automatic Audit System

A typical process of creating, auditing and sharing virtual appliances is described in the following:

1) The *appliance creator* creates a virtual appliance (VA) (e.g., setting up the operating system and services provided by the appliance). Optionally, the appliance creator can upload a policy document, which describes his requirements for publication of the VA. This policy is evaluated by the AAS. If any policy violations are detected (e.g., there are insecurely deleted files or log files in the VA), the publication process is immediately stopped.
2) The *AAS* audits the virtual appliance using the Cloud provider's policies (see III-C for audit case examples). This step is performed to make sure, the Cloud provider's requirements are met (e.g., protecting the Cloud infrastructure from malicious VAs).

3) On successfully passing the audit process the appliance is deployed in the Cloud and the image is published using the appliance store.
4) The *appliance user* now decides to use the virtual appliance. He therefore uploads his policy to the AAS, which in turn audits the appliance according to the audit cases defined in this policy.
5) If the audit process is passed successfully, the virtual appliance image is cloned by the appliance store and deployed to the cloud.

While this is a rather high level design of an automatic system more details of the technology used is described in the following sections. It becomes clear, that as an enabler a security policy language for virtual machines is needed.

IV. A Cloud Audit Policy Language

This section first describes the requirements established for a Cloud audit policy language. Beneath generic attributes six specific policy scenarios get introduced, which have to be describable with the target language. Afterwards, several already existing security policy languages have been evaluated using the requirements and it is shown why none of them fulfills these requirements. Therefore, a new Cloud Audit Policy Language CAPL will be introduced, which is based on the Cloud Infrastructure Management Interface (CIMI) [16] specification.

A. Security Policy Language Requirements

Apparently, there is a huge number of policy definition languages available (ASL, LaSCO, PDL, XACML, SPARQLE, SSPL, OVAL, etc.) all aiming to model security policies. Furthermore, business process languages (BPEL, WADE, YAWL, ADEPT, etc.) could be applicable as well, which increases the size of languages to choose from. To be able to evaluate, which language fits best a bottom up approach was taken by first defining the policy scenarios, which need to be describable by the security policy definition language.

1.) Policy Scenario Modelling Support The most important criterion in our evaluation process is the ability to model security requirements of Cloud components, such as VMs and their interaction with each other. To have a basis for the evaluation of this criterion a number of important example policy scenarios have been identified:

P1 - Malware: Since malware affects availability, integrity and confidentiality every VM image needs to be checked for viruses and rootkits before being started within a cloud. Running VMs must be checked on a regular basis. The resulting policy could be: “The VM is free of malware”.

P2 - Filesystem changes: Malicious attacks often result in change of the file systems’ content, such as modification of config files or installation of malicious software. Therefore, a Cloud audit policy should allow to define: a) A certain file (or folder) may not be changed at all. Every single change should raise an action. b) Validation of a certain file containing a specific content. Latter is most important in config files, which affect security relevant configurations. The resulting policy could be: “File X may not be changed”.

TABLE II: Policy Example Scenarios

No.	Policy
P1	The VM is free of malware
P2	File X has not been changes
P3	Upscaling of VMs in VM cluster “WWW-Server” is only permitted if average requests per second $\geq$ Y
P4	Port Z is open, allowed protocols: HTTP
P5	Software X must (not) be installed
P6	The VM does not contain any personal information

P3 - VM scalability: One attribute of Cloud environments is flexibility and on-demand availability of resources. Depending on a currently existing demand additional VMs can be added to a certain service cluster (VM upscale) and when demand lowers, VMs can be decommissioned again (VM downscale). But this could also be misused by attackers to compromise the availability of a customer’s Cloud based infrastructure, by downscaling VMs during a high demand period. Contrary, unnecessary upscaling of VMs increases the running costs of a customer. The resulting policy could be: “Upscaling of VMs in VM cluster “WWW-Servers” is only permitted, if average requests per second $\geq$ N”.

P4 - Technical attribute modelling: Security is expressed if the infrastructure complies to certain technical attributes. A very simple rule defines the state of a network port. A port can be closed or open. Same can be used for allowed protocols. The resulting policies could be: “Port 80 is allowed to be open”, “Allowed network protocols: HTTP”

P5 - VM content: A VM contains software, such as an operating system and certain application software. To increase security by banning certain software products or specific versions of a software, to prevent data leakage or just to be compliant to existing software licenses it can be necessary to restrict the existence of software on a VM. The resulting policies could be: “Software X must (not) be installed”.

P6 - Data traces: In case of VM marketplaces, users prepare VM images and offer them at the marketplace (role: appliance creator). It is important, that these images don’t contain any personal information of the VM image creator, such as private key files or passwords, which could lead to a security breach. It is to validate that files are cleared e.g., history file, and critical information is securely wiped (and not be restorable anymore even with file carving tools). The resulting policies could be: “The VM does not contain any personal information”. Table II summarizes the elaborated policy scenarios.

This is just a brief description of the policy language requirements. A detailed description can be found in the bachelor thesis “Design and Development of a Security Policy Language for Automatic Cloud Audits” [17].

In addition to the specific functional requirements, there are several additional generic criteria to be considered when choosing a policy language. To model the scenarios described above the following features must be supported by a security policy language:

- *Monitored Objects* are any kind of entities in the Cloud infrastructure, which shall be monitored (e.g., hosts, virtual machines, files).
- *Logical Policy Operations* can be used to create more complex policies by combining them with logical operators such as *AND* and *OR*.
- *Policy Scoping* By grouping virtual machines or policies the process of creating and managing policies becomes easier. Also, incorporating the ability to define provider-only policies or policies, which can only be used by the provider, may prove to be beneficial.

2.) Technological Support Policies can be described using textual as well as graphical methods. However, the focus of SAaaS will be on a textual description of policies. All language candidates will be analyzed with regards to their technological basis, especially whether they build upon established standards such as XML and JSON or they introduce completely new language formats. Using widely accepted technologies may be beneficial because there already exist a lot of tools such as parsers, interpreters and comprehensive documentation. Custom language formats however can be tailored to the problem domain and might improve flexibility and readability. To ensure a fast adoption by developers and leverage the large amount of tools already available, XML should be the preferred language base.

3. Development Activity Estimation Release cycles of tools, the size of the developing community and the adoption of a language by other projects indicates a high development activity and is an indicator for a future proof implementation.

4. Documentation Quality A comprehensive documentation is essential for understanding and evaluating a policy language. The quality of the documentation is hereby defined by factors such as the logical structure, accessibility, profoundness and consistency.

5. Complexity & Integratability in SAaaS The target language should be complex enough the fulfill all requirements but also generalisable up to a certain point. Too much complexity will affect the ease of learning by cloud administrators and therefore indirectly and negatively influence the utilization of the language, which affects its overall success. Furthermore, it is essential to evaluate if the language can be integrated into the SAaaS architecture, presented later in this paper in Section V.

B. Evaluation of Existing Policy Languages

In this subsection, policy language candidates are evaluated and compared for their suitability as a security policy language in SAaaS.

REI is an OWL based language, developed by Lalana Kagal in 2005. REI allows the definition of management, security, privacy and conversation policies [18]. These policies define the optimal behavior in a problem domain. A policy is hereby defined by the prohibition, permission or the obligation to perform an action on a target. The focus on semantic technology is not needed for SAaaS and introduces needless complexity. Additionally, REI has no practical relevance nor has it spread beyond a PhD thesis, which it was developed for.

Common Information Model (CIM) is a model to describe elements and the relationships between them (such as policies). It addresses most of the SAaaS requirements and would have been a suitable candidate. However, an implementation according to the CIM standard would have gone way beyond the requirements of the SAaaS project. Therefore, CAPL (Section IV-C) only uses parts of CIM for its implementation.

Ponder is a policy specification language, which already features tools and services for policy enforcement and evaluation. One of the main concepts behind Ponder is the general-purpose object management system and message passing paradigm [19]. Here the language is meant to be implemented in a way that the actual decision making process (deciding whether a policy is fulfilled or not) needs to be as close as possible implemented to its data source. In a Cloud scenario this would mean, that the decision engine needs to be implemented on each single machine. In addition to the proprietary language base Ponder Talk, this is a knock-out criteria for the usage of Ponder for our scenario.

LaSCO follows a graph based approach to define policies. Despite this being a rather interesting approach, it introduces a lot of unnecessary complexity. Additionally, the problem of conflict management is not addressed [20] and similarly to REI LaSCO has not spread beyond academic boundaries (one dissertation in [21]), which makes this language unsuitable for SAaaS.

Evaluation Overview: Besides the aforementioned languages WS-Trust, IDMEF, SSPL, PAX PDL, CADF and KAoS have also been evaluated. However, none of those languages have proven to be useful for the SAaaS approach, which is why we omit going into further detail. All language criteria are listed in the evaluation summary, depicted in Table III. All knock-out criteria (which do not fulfill our requirements introduced in Section IV-A) are displayed in bold red. It is shown, that none of the evaluated security languages fulfills the established requirements. As a result an own Cloud audit security policy language needed to be developed.

C. Cloud Audit Policy Language (CAPL)

CAPL is an extension of the Cloud Infrastructure Management Interface. Core features like the object model, the protocol and a simplified variant of CIMI classes (e.g., Machine, MachineConfiguration, MachineImage) are inherited by CAPL. However, due to the different focus of CIMI on managing Cloud infrastructures some parts of CIMI have not been adopted in CAPL because they are not required for the SAaaS scenario. A detailed description of CAPL features is provided in the following.

1) User Roles: CAPL uses slightly simplified definitions of the CIMI roles *Cloud Provider* and *Cloud Consumer*. The Cloud Provider manages and provisions Cloud services and possesses full access rights. The Cloud Consumer uses cloud services as well as the service for auditing his virtual machines. The Cloud Consumer has a limited set of access rights, which are required to define policies and triggering audits.

2) Service Interface: CIMI uses a REST based protocol for communication. CAPL adopts the CIMI service interface.

TABLE III: Comparison of existing security policy languages

Evaluation Property	**REI** Support	Details	**CIM** Support	Details	**Ponder** Support	Details	**LaSCO** Support	Details
Technological Base	⊖	OWL Lite	⊕	MOF & UML, XML through WBEM	⊖	**PonderTalk (SmallTalk)**	⊖	Directed graphs
Definition of monitored objects	⊕	Targets, user	⊕	PolicyInSystem	⊕	Managed Object in Subject, Action, Target Syntax	⊕	Knots
Combination of policies	⊕	Denotic objects	⊕	Conditions, PolicyRule, PolicySet	⊕	Obligation policy	⊕	Conjunctions
Area of validity	⊕	Constraints, groups of objects, a single assignment seems difficult	○	Unclear	⊕	Self managed cell	⊕	Domains
Conflict management	⊕	Priorities	⊕	Priorities	⊕	Yes	⊖	**Not implemented**
Last version	○	Updated 2005	⊕	Currently revised (version 2013)	⊕	2011	○	2000
Acceptance	⊖	**Just a PhD thesis work, cited in different paper, no practical application**	⊕	Windows Management Instrumentation[22], SBLIM project[23], IBM[24]	○	Cited in multiple papers	⊖	**Only used in PhD dissertation [21] and paper [20] of LaSCO author**
Community support	⊖	**None**	⊕	None for CIM, but for specific implementations	⊖	**None**	⊖	**None**
Documentation	○	Rough description of classes[18], paper, presentations, Examples[25]	⊕	UML Diagram[26], Policy profile[27]	○	Good examples, but for old Ponder version	○	Only one PhD dissertation
Complexity	⊖	Long training period, complex & nested architecture	⊖	High, due to inconsistencies of different versions	○	Policies are human friendly readable, but developing own is difficult	⊖	Very complex due to its graph based origin
Support of SAaaS policy scenarios	⊕	Yes	⊕	Yes	⊕	Yes	⊕	Yes
Implementation effort	○	Unclear	⊖	**Complete CIM and WBEM implementation necessary**	⊖	Very high, since it brings its own agents	⊖	**Completely different base layer**
Integratability in SAaaS architecture	○	**Semantic of OWL not necessary**	⊕	Yes	⊖	**No, own agents necessary, different philosophy of policy evaluation**	⊖	**No, due to graph based nature**

3) Language Basics: CAPL enhances CIMI by adding several new classes:

- Machine
  The Machine class represents a machine, which shall be audited. CIMI uses Machines only for virtual machines. However, CAPL enhances the scope of Machines and includes host machines running virtual machines because those might be as well targets for audits.
- MachineTemplate
  The MachineTemplate defines the initial configuration of a VM.
- Policy
  Defines a policy rule (e.g., "a virtual machine must not contain malware"), which can be assigned to a machine or a group.
- PolicySet
  A PolicySet contains multiple Policies. Only if all contained conditions of the rules are fulfilled, the PolicySet evaluates to success. A PolicySet may be used like a poliy and attached to machines or groups. Rules contained in a PolicySet may be linked disjunctive or conjunctive (using *AND*/*OR*). This behavior originates from the CIM policy model [28].
- Group
  Groups are used to manage related objects like multiple rules and machines. In such a case all rules of the group apply to all machines.
- RuleType
  RuleType describes what a policy is supposed to check and defines attributes and configurations, which the policy has to set.

An example, which depicts the key features of CAPL, is shown in listing 1. This Policy describes the conditions under which upscaling of Web server VMs is allowed. In this case, it is measured whether upscaling is allowed or not.

Listing 1: CAPL Example

```
<Group xmnls="http://research.cloud.hs-furtwangen.de/capl/">
<id>https://research.cloud.hs-furtwangen.de/CAPLPrototyp/groups/wwwCluster1</id>
<name>Cluster at Loadbalancer1</name>
<refName>WWWCluster1</refName>
<description>Group of web servers at loadbalancer1
</description>
<created>2013-03-03</created>
<updated>2013-03-03</updated>
<enabled>true</enabled>
<machines>
  <machine href="https://research.cloud.hs-furtwangen.de/CAPLPrototyp/rest/machines/6" />
  <machine href="https://research.cloud.hs-furtwangen.de/CAPLPrototyp/rest/machines/7" />
</machines>
</Group>

<Policy xmnls="http://research.cloud.hs-furtwangen.de/capl/">
<id>https://research.cloud.hs-furtwangen.de/CAPLPrototyp/policies/upscale</id>
<name>Upscale is Allowed</name>
<refName>upscaleCluster1</refName>
<description>Upscale is only allowed when requests on machines is higher than 10</description>
<created>2013-03-04</created>
<updated>2013-03-04</updated>
<enabled>true</enabled>
<ruleType href="https://research.cloud.hs-furtwangen.de/CAPLPrototyp/rest/ruleTypes/upscale"/>
<intervalType>no</intervalType>
<targetResource href="https://research.cloud.hs-furtwangen.de/CAPLPrototyp/rest/machines/5"/>
<attribute key="metric">requests per second</attribute>
<attribute key="threshold">10</attribute>
<attribute key="cluster">https://research.cloud.hs-furtwangen.de/CAPLPrototyp/groups/wwwCluster1
</attribute>
</Policy>
```

V. SAaaS ARCHITECTURE

To support the presented use cases of a concurrent audit system for Cloud environments (see Section II), an agent system to monitor Cloud environments is proposed. Before explaining the SAaaS architecture and advantages of agents in detail, we briefly want to explain the whole Security as a Service event processing sequence. To support this, consider the following example.

A. Scenario Used to Explain the SAaaS Architecture

A typical Web application architecture consisting of one or multiple Webserver(s) and a database backend is deployed at VMs in a Cloud. The VMs are logically grouped together to *WWW-Cluster1*. Initially, the Cloud customer's administrator installs the VMs with the necessary software, e.g., Apache web server, MySQL database. After the functional configuration security policies are modelled to describe the target infrastructure state, such as the policies P1 - P3 introduced in Table II. This can be:

A) Technical rules like allowed network protocols and connections between VMs, or that the web server configuration is finished and a notification should be sent if changes to its config files are detected. As a result of these policies software agents called `VM agents` are configured with the necessary tools to monitor these requirements and automatically deployed to the VMs.

B) Business flow related security policies can be created as well, such as a simple scalability policy: "If the cloud management systems gets an upscale event request for components of *WWW-Cluster1*, first the actual load of all web servers needs to be checked. If the average load over all web servers is not higher than a certain threshold, e.g., 100 http connections / Web server, the upscaling gets denied and an alarm gets raised since the event must originate from a systems failure or a successful hacker attack at the Cloud management system. The same scenario works for downscaling in an inverse manner: If a downscaling event for *WWW-Cluster1* gets detected, but the actual load is above a downscale threshold, an alarm gets raised. Here, software agents are additionally located at the Cloud management system and the scaling service.

B. Cloud Audits Using Agents

To generalize this scenario: in the SAaaS architecture a modelled security state of certain components gets monitored by agents, which are deployed at the specific resource, e.g., the Cloud management system (CMS), a Cloud host or a VM.

An agent can be defined as [29]:
"... a software entity, which functions continuously and autonomously in a particular environment ... able to carry out activities in a flexible and intelligent manner that is responsive to changes in the environment ... Ideally, an agent that functions continuously ... would be able to learn from its experience. In addition, we expect an agent that inhabits an environment with other agents and processes to be able to communicate and cooperate with them ..."

Since the agents in the SAaaS architecture are running independently, not necessarily connected to a certain central instance agents can receive data from other instances (e.g., the policy module) and send information to other instances like other SAaaS agents or the SAaaS' event processing system. The "central" event processing system gets itself implemented as an agent, which can be scaled and distributed over multiple VMs.

C. Type of Agents

Agents collecting data are called `Sensor Agents`. If the location of an agent needs to be expressed, they are also titled as `VM Agent` (agent running at a VM), `Host Agent` (agent running on a Cloud host monitoring, the hypervisor) or `CMS Agent` (agent monitoring the Cloud management system). Specific targeted security audits perform specific checks of systems, which are affected by a change within the Cloud environment. These checks are performed automatically by so-called `Audit Agents`. The threshold values to be checked are defined as metrics and get checked in case of an event by `Metric Agents`. Changes to the Cloud infrastructure get detected by sensor agents before sent out the central event processing unit preprocessed and aggregated by a `Event Aggregator Agent`, which also runs on same location as the sensor agent, e.g., a VM. This is important to reduce the overall messages sent to the global Cloud event processing

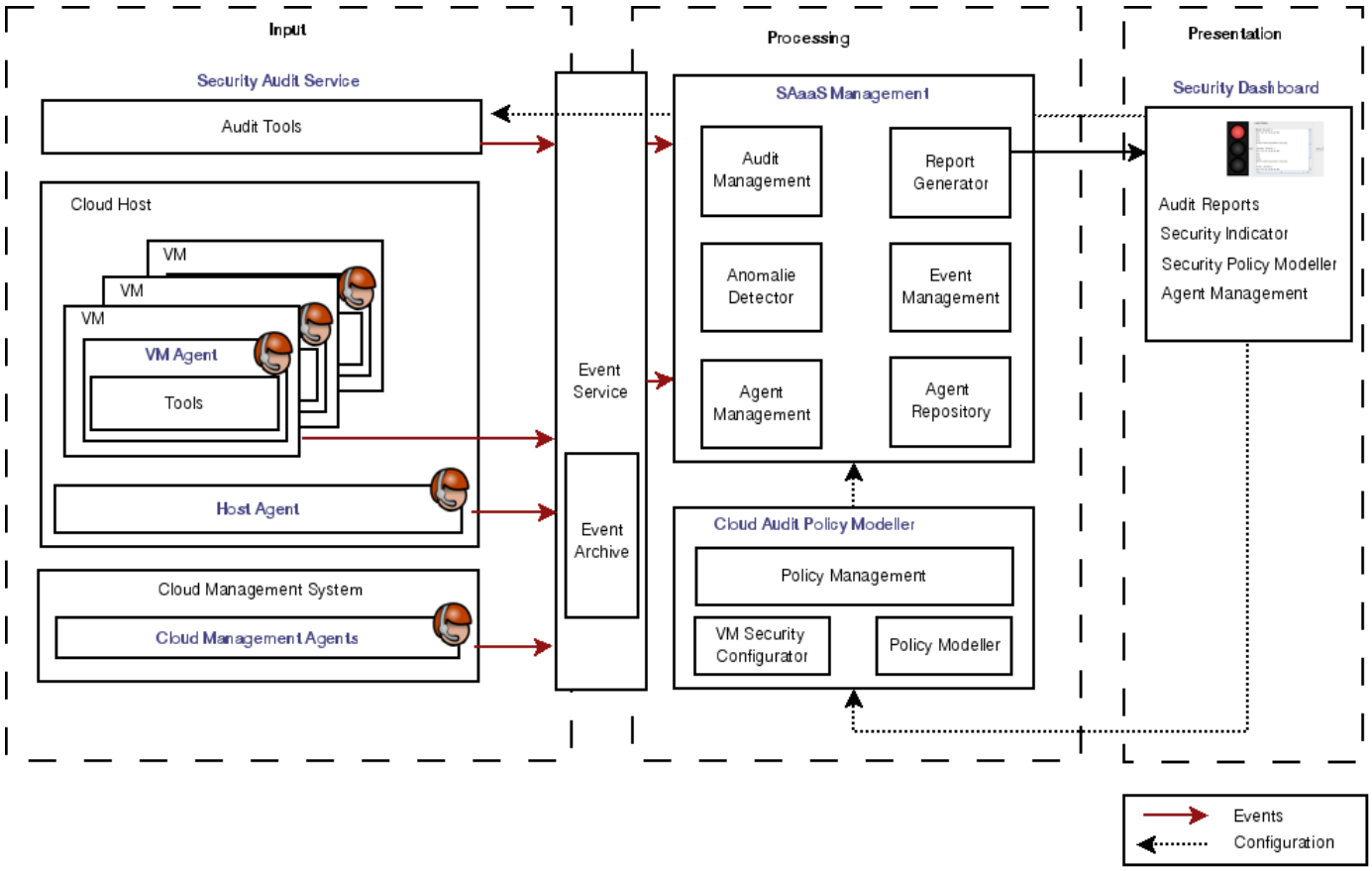


Fig. 3: SAaaS event processing sequence

system especially in large Cloud computing environments. The `Event Aggregator Agent` filters out possible VM dependent events like a started Web application session from *IP 1.2.3.4*. A more abstracted event gets send to the Cloud event processing system to detect (possible) user overlapping security incidences. This could be a message containing the number of not completed Web shop transactions by IP *IP 1.2. 3. 4* to pre-detect a Denial of Service attack.

D. SAaaS Event Processing

Figure 3 gives a high level overview how events are generated, preprocessed, combined and forwarded within the SAaaS architecture. It can be divided into three logical layers: *Input*, *Processing* and *Presentation*.

Input: The SAaaS architecture gets its monitoring information from distributed agents, which are positioned at key points of the cloud's infrastructure to detect abnormal activities in a Cloud environment. Possible key points are: running VMs of Cloud users, the VM hosting systems (Cloud hosts), data storage, network transition points like virtual switches, hardware switches, firewalls, and especially the Cloud management system. A `VM agent` integrates several monitor and policy enforcing tools. Therefore, it loads necessary VM agent plug-ins to interact with stand-alone `tools` like process monitor, intrusion detection system or anti virus scanner. It gets installed on a VM likewise on a Cloud host. A logging component is recording the chronological sequence of occurrences building `audit trails`.

Processing: Each SAaaS agent receives security policies from the security policy modeller component. Through security policies each agent gets a rule set (its intelligence) specifying actions in case of a specific occurrence (e.g., modification of a frozen config file). Thus, every occurrence gets first preprocessed by an agent, which reduces communication between VM agents and Cloud management agent. The `Cloud Audit Policy Modeller` consists of a policy editor and a VM security configurator. An example of a Cloud specific security policy could be: "In case of a successfully detected rootkit attack on a VM running on the same Cloud as a users VM, the user VM gets moved to a different host to diminish the risk of further damage." whereas a security configuration could state: "In case a modification attempt of a file within / etc/php5/ gets detected, deny it and send an email to the Cloud administrator." Cloud audit policies get send from the Policy Management to the Agent Management to configure the corresponding agents. By using the monitoring information of the distributed agents in combination with the security policies a `Cloud behaviour model` is built up for every Cloud user. Cloud audit policies are also used as input for the Cloud management agent to detect user overlapping audit events. Forwarded higher level events are processed by an event processing engine. It is also fed with the modelled security flows from the `Security Policy`

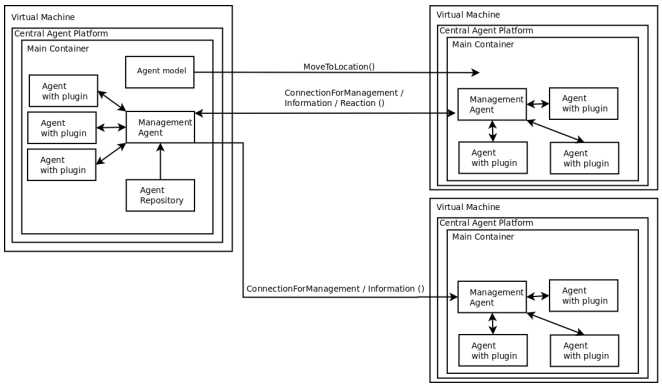


Fig. 4: Basic SAaaS agent design

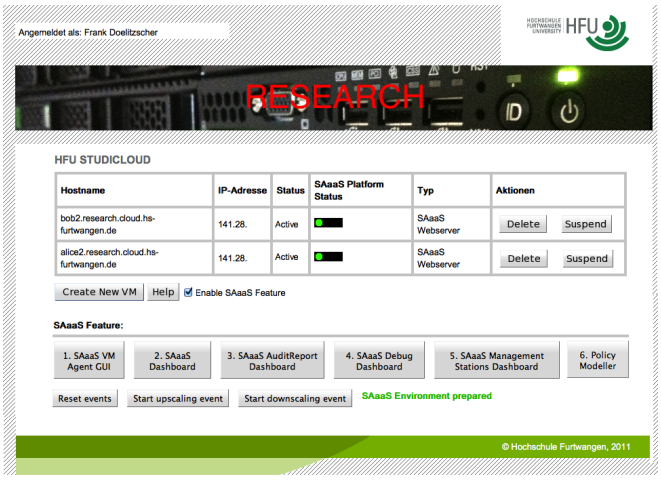


Fig. 5: HFU Cloud management interface

`Modeller` to aggregate information and detect behaviour anomalies. Countermeasures can then be applied to early detect and prohibit security or privacy breaches. The `Report Generator` conditions events, corresponding security status as well as audit report results in a human friendly presentation.

Presentation: As a single interaction point to Cloud users the `Security Dashboard` provides usage profiles, trends, anomalies and Cloud instances' security status (e.g., patch level). Information are organized in different granular hierarchies depending on the information detail necessary. At the highest level a simple three colour indicator informs about a users Cloud services overall status. It also provides a graphical user interface to deploy agents to Cloud instances. Figure 5 shows a part of the security dashboard prototype, which gets described in more detail in the next (but one) Section.

Communication between the distributed agents and the security dashboard is handled by an `Event Service`. Events will use the Agent Communication Language Format (ACL) [30] and are exchanged using a FIPA[31] compliant HTTPS Message Transfer Protocol (HTTPS-MTP) [32]. Events are also stored in an `Event Archive`.

E. How agents can improve incident detection

Incident detection in Cloud environments is a non trivial task due to its characteristics as discussed in Section I. Therefore, it is important to have a high number of sensors capturing simple events. Preprocessed and combined complex events can be generated reducing the possibility of "event storms". Combined with knowledge about business process flows (specified in security policies), it will be possible to detect security incidents while keeping the network load low. The usage of agents delivers this possibility because agents are *independent units* that can be added, removed or reconfigured during runtime without altering other components. Thus, the amount of monitoring entities (e.g., network connections of a VM, running processes, storage access, etc.) of a Cloud instance can be changed without restarting the incident detection system. Simultaneously using agents can *save computing resources* since the underlying business process flow can be taken into account.

While single `sensor agents` can monitor simple events (e.g., user login on VM) and share them with other agents *complex events* can be detected. Given the scenario of a successful unauthorized login of an attacker at a virtual machine `VM2`, misusing a web server's directory to deposit malicious content for instance a trojan. Agent `A1` monitors the user login, agent `A2` detects the change of a directory content and agent `A3` detects a download of a not known file (the trojan). Instead of sending those three simple messages to a central event processing unit a `VM agent` can collect them, conditioning one higher level event message that `VM2` was hijacked. This can result in a predefined action by the `Cloud Management Agent` e.g., moving a hijacked VM into a quarantine environment, alerting the user and simultaneously starting a fresh instance of `VM2` based on its VM image.

By ordering agents in a hierarchical structure and preprocessing of detected events reduces network load originated from the incidents detection system. Furthermore, this makes the system more scalable by reducing data sent to upper system layers. This concept is introduced and successfully used in [33]. Combining events from system deployed agents (e.g., VM agents, host agent) and infrastructure monitoring agents (network agent, firewall agent) incident detection is not limited to either host or network based sensors, which is especially important for the characteristics of Cloud environments. Using agents has advantages in case of a system failure. Agents can monitor the existence of co-located agents. If an agent stops for whatever reasons this stays not undetected. Concepts of asymmetric cryptography or Trusted Platform Module (TPM) technology can be used to guarantee the integrity of a (re-)started agent. If an agent stops the damage is restricted to this single agent or a small subset of connected agents, which are requiring information from this agent.

F. SAaaS Agent Architecture Implementation

For the SAaaS architecture we evaluated existing agent frameworks with the following requirements:

- Agents can be deployed, moved, updated during runtime
- Agent performance
- Open Source, documentation, community support

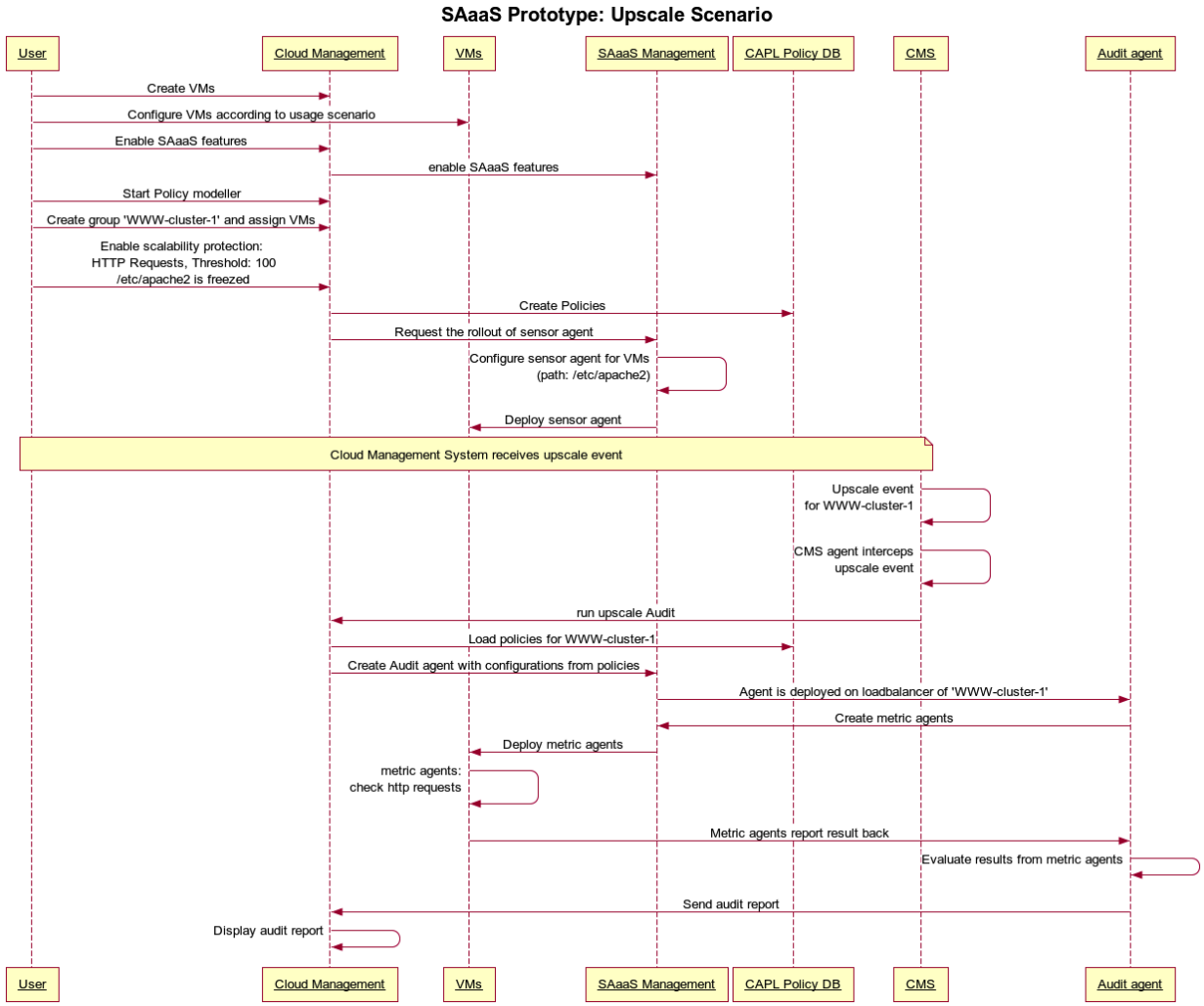


Fig. 6: Automatic security audit in case of upscale event

Since our Cloud environment at HFU's Cloud Research Lab CloudIA [34] is built around the Cloud management system Open Nebula another requirement was the agent programming language: Java. As a result we choose the Java Agent DEvelopment Platform (JADE), which enables the implementation of multi-agent systems and complies to FIPA specifications. Figure 4 illustrates a basic agent architecture. It shows three VM Sensor Agents. Agents live in an agent platform, which provides them with basic services such as message delivery. A platform is composed of one or more Containers. Containers can be executed on different hosts thus achieving a distributed platform. Each container can contain zero or more agents [35]. To provide monitoring functionality a VM agents interacts through agent plugins with stand-alone tools like process monitor, intrusion detection system or anti virus scanner, as depicted in Figure 4. To harness the potential of Cloud computing an agent can be deployed to a VM on-demand according to the policies a user defines. Different agents based on modelled business processes are stored within an agent repository. To be able to move a JADE agent to a running Cloud instance the Inter Platform Mobility Service (IPMS) by Cucurull et al. [36] was integrated. This supports the presented advantage of deploying agents on-demand if a designed business process flow was started (as described in Section V-E).

G. CAPL Integration & SAaaS Prototype

The described Cloud audit policy language, presented in Section IV is seamlessly integrated into the SAaaS architecture. To show how SAaaS can increase transparency in Cloud environments the following prototype scenario was implemented. It is also depicted in Figure 6 and describes a whole SAaaS life cycle:

1) A Cloud user creates three new VMs on the Web based Cloud management interface, depicted in Figure 5. The VMs get configured as a typical Web application installation: two Web servers, which are delivering content from one database server.
2) After VM configuration is finished the user enables the SAaaS features, starts the security policy modeller and groups the VMs into group *"WWW-Cluster1"*.

Then he activates a scalability monitoring policy with a metric of HTTP requests per second and a threshold of 100 for *WWW-Cluster1*. Furthermore, he creates a policy saying that the *"/etc/apache2"* config directory should be considered frozen and therefore be monitored for changes.

3) As a result from enabling the SAaaS features and the policy creation, a sensor agent for filesystem monitoring gets deployed to the VMs of *"WWW-Cluster1"*. It utilizes the linux tool *inotify* [37] to watch the *"/etc/apache2"* directory.
Now let's assume there is a lot of load on the web servers due to a product launch of the user's company. Therefore, the Cloud management system gets an upscale event for *WWW-Cluster1*, which gets intercepted by a SAaaS agent monitoring the CMS.
4) The event provokes the policy and configures a scalability audit agent with a scalability check and deploys it on the SAaas management VM (in the case of a filesystem change event within a web server VM the audit agent would have been deployed to that particular VM).
5) The agent creates new metric agents, which get deployed to the web server VMs of *WWW-Cluster1* to check the current load of HTTP requests. They report the result back to the audit agent. The audit agent evaluates the result and decides, dependent on the average load reported by the metric agents, if the upscale event is okay or not.
6) The results get conditioned into an audit report.

As a first prototype, a two layered agent platform was developed, consisting of a `sensor agent` running inside a VM and a `Cloud management system agent`. Audit reports get displayed in a `Security Dashboard`. Since all Cloud VMs in CloudIA are Linux based, only Open Source Linux tools were considered during our research. Two notification mechanisms were implemented:

a) The tool sends agent compatible events directly to the agent plugin.

b) The tool writes events in a proprietary format into a logfile, which gets parsed by an agent plugin.

As for mechanism a) the filesystem changes monitoring tool *inotify* was used, whereas for mechanism b) *fail2ban* [15], an intrusion prevention framework was chosen. For demo purposes a simple Web frontend was written, which offers to launch several attack scenarios on a VM agents equipped VM in CloudIA. Before/after tests were performed to validate that an attack was detected and (depending on the plugin's configuration) prohibited. A prototype version of the security dashboard, depicted in Figure 7 showing a signal light indicator informed about occurring events. When started, it shows a green light. After launching an attack, the security dashboard indicator light changes its colour to yellow or red. The impact of a monitored event is defined by a severity matrix, shown in Table IV. Each severity value out of the Web server log file gets associated with a certain score. This score gets summed up for all events. Then the quotient gets calculated which is directly connected to the resulting colour.

TABLE IV: Severity matrix for security indicator light

Severity	Value of message	Quotient	Colour
info	0	< 1.4	green
low	1	$<= 1.4$	green
middle	2	$> 1.4 < 2.5$	yellow
high	7	$>= 2.6$	red

H. VM Automatic Audit Integration & SAaaS Prototype

As described in Section III especially for a public Cloud it is necessary to audit VMs. Every time a VM is uploaded to the Cloud an image audit agent is checking the image according to the enterprise policies defined in CAPL.

VI. Evaluation

This section evaluates the presented automatic audit system approach of the SAaaS system. First, it is discussed how SAaaS enhances the auditing of non-running virtual machine images. Therefore, an evaluation scenario from a Cloud customer's perspective is presented. We describe necessary user effort utilizing the presented automatic audit system and if he would take a manual approach. Second, it is evaluated how the presented Cloud audit policy language supports the established requirements, introduced in Section IV-A. Finally, it is elaborated how the presented concurrent audits of the SAaaS system address Cloud specific security issues and therefore enhance Cloud computing security and transparency.

A. Auditing of VM Images

To evaluate the automatic audit system the following scenario is considered: A Cloud customer chooses an online shop virtual appliance (containing a web server and a database) out of a cloud's store. Before transferring actual data to the image the following security policies need to be evaluated: P1: The image must not contain any malware. P2: The image must not run any other software than the web server and the database. P3: The web server and database connection must be configured properly.

1) Customer uses Automatic Audit System Approach: When using the automatic audit system, the appliance user or appliance creator first describes his security policies. Since malware checks are usually a default policy, provided by the Cloud provider, there is no need to model those. Additionally, to simplify black box scans of the proper interaction between web server and database it is imaginable to deposit a predefined default start page, which could be browsed. The automatic audit system will parse the security policies and identifies the necessary audit cases, which are fetched from the database. The audit cases get sorted, dependent on how the checks can be executed. There are two kinds of security audit modes. *Offline VM audits* mount the VM's image and perform audit tasks on it, whereas *online VM audits* launch the VM in a quarantine environment of the cloud. There, audit tasks, which can be only performed on the running VM are executed, such as an analysis of open ports. Again, the results of the single audit cases will be submitted to the parser and saved as mini reports in the audit system's database.

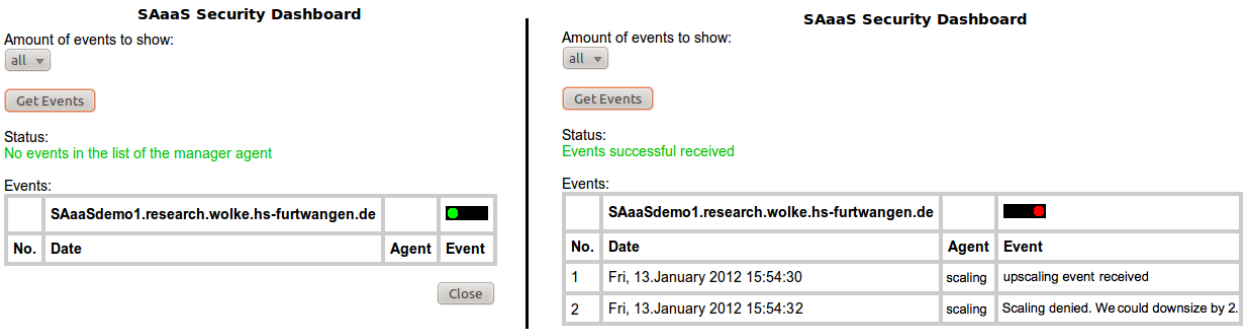


Fig. 7: Cloud security dashboard prototype

At last, the report generator conditions the results of all mini reports. The cloud management system is informed if the overall audit result is *"passed"* or *"failed"*. If the status is passed, the image can be added to the store, otherwise the release will be denied. Nevertheless of the result, the complete audit report will be sent to the appliance creator, to inform about necessary problems to be fixed.

2) Manual Approach: In contrast to the automatic audit system an offline audit of the appliance's image is not immediately possible. This is due to the fact, that the appliance user does not have direct access to images stored in the appliance store. The only two approaches possible are downloading the appliance's image, which enables offline auditing or to limit the audit process to online auditing. This is done by an administrator, who must have sufficient expertise in virtualization technologies, auditing methods and must be an audit tool expert. Additionally, for the sake of reproducibility and documentation, the appliance user has to follow a very well defined auditing process (assuming such a process exists). Downloading VM images and evaluating them offline imposes a significant network overhead on the appliance user as well as the Cloud provider. Manual online auditing can be performed, when a virtual appliance image is already started. The appliance user has to log in to the appliance in order to execute auditing tools and scripts. Additionally, the virtual appliance also has to be checked externally to determine which services are activated, for example by port scanning. Performing port scans on virtual machines executed in the cloud may trigger the Cloud provider's intrusion detection systems or may even be prohibited entirely by the Cloud provider's terms of use.

This demonstrates the overall complexity of a manual virtual appliance auditing process. The automatic audit system delivers the following advantages for appliance creator/user and Cloud provider: Improved security when using 3rd party virtual appliances (appliance user), well documented and formalized audit process (all), customizable, machine-readable audit policies (all) and additional revenue by offering audits as a service (Cloud provider).

B. CAPL Evaluation

Because none of the evaluated languages fulfilled the requirements of SAaaS on a policy language (e.g., missing conflict management, combination of policies), while retaining a reasonable complexity, CAPL was developed, which specifically addresses all those requirements. CAPL is based on XML and extends the Cloud Infrastructure Management Interface CIMI by a definition of security policies. Cloud providers as well as Cloud users are enabled to define policy rules for virtual machines. Table V evaluates CAPL against the requirements, established in Section IV-A. By staying closely to the CIMI standard, it will be possible to define security policy for any CIMI compatible cloud infrastructure. This also increases the compatibility of the proposed SAaaS system. Another advantage of CAPL is its simplicity, since it is tailored to the SAaaS target scenario. However, developing a new policy language also has some negative aspects. Our results are used in the SAaaS project only, which leads to a rather poor acceptance. Also, besides the SAaaS project members there is no community surrounding and developing this language.

C. Cloud Specific Security Issues Addressed by SAaaS

The German Federal Office for Information Security publishes the IT baseline protection catalogues enabling enterprises to achieve an appropriate security level for all types of information. In a comprehensive study [38] on all IT baseline protection catalogues as well as current scientific literature available [2][39][40][4][5], we identified the following Cloud specific security issues as solvable by the presented SAaaS system:

Abuse of Cloud resources
Cloud computing advantages are also used by hackers, enabling them to have a big amount of computing power for a relatively decent price, startable in no time. Cloud infrastructure gets used to crack WPA, and PGP keys as well as to host malware, trojans, software exploits used by phishing attacks or to build botnets like the Zeus botnet. The problem of malicious insiders also exists in classical IT-Outsourcing but gets amplified in Cloud computing through the lack of transparency into provider process and procedure. This issue affects authorisation, integrity, non-repudiation and privacy. Strong monitoring of user activities on all Cloud infrastructure components is necessary to increase transparency. The presented SAaaS use case A) Monitoring and audit of Cloud instances addresses this problem.

Missing security monitoring in Cloud infrastructure
Security incidents in Cloud environments occur and (normally) get fixed by the Cloud provider. But to our best knowledge no Cloud provider so far provides a system, which informs user

TABLE V: Evaluation of CAPL

Requirement	CAPL Supp.	Details
Technological Base	⊕	XML
Definition of monitored objects	⊕	Tailored to problem domain
Combination of policies	⊕	Groups
Area of validity	⊕	CIMI compatible infrastructures
Conflict management	⊕	Included
Last version	○	Under active but internal development
Acceptance	⊖	Not spread beyond SAaaS
Community support	⊖	Only SAaaS
Documentation	○	CAPL documentation [17]
Complexity	⊕	Tailored to problem domain
Support of SAaaS policy scenarios	⊕	Full
Implementation effort	○	Own development
Integratability in SAaaS architecture	⊕	Fully integratable

promptly if the Cloud infrastructure gets attacked, enabling them to evaluate the risk of keeping their Cloud services productive during the attack. Thereby the customer must not necessarily be a victim of the attack, but still might be informed to decide about the continuity of his running Cloud service. Furthermore, no Cloud provider so far shares information about possible security issues caused by software running directly on Cloud host machines. In an event of a possible 0-day exploit in software running on Cloud hosts (e.g., hypervisor, OS kernel) Cloud customers blindly depend on a working patch management of the Cloud provider. The presented SAaaS use case B) Cloud infrastructure monitoring and audit addresses this problem.

Defective isolation of shared resources

In Cloud computing isolation in-depth is not easily achievable due to usage of rather complex virtualization technology like VMware, Xen or KVM. Persistent storage is shared between customers as well. Cloud providers advertise implemented reliability measures to pretend data loss like replicating data up to six times. In contrast, customers have no possibility to prove all these copies get securely erased in case they quit with the provider and this storage gets newly assigned to a different customer. While the presented SAaaS architecture does not directly increase isolation in-depth it adds to the detection of security breaches helping contain its damage by the presented actions.

VII. State of the art - Related Work

To put the presented work described in this paper into perspective, this section first discusses related research work on Cloud security issues, followed by other Cloud security research projects in contrast to SAaaS and the usage of agents to increase security. It then discusses related work regarding security of virtual appliance images. Afterwards, work on security policy languages is elaborated, which are an important part of the proposed work in this paper, too.

A rather high-level, but comprehensive view on the whole topic of Cloud computing security is given by Mather et al in the book "Cloud Security and Privacy" [41]. It provides a very good introduction especially by laying out the necessary groundwork. The most comprehensive survey about current literature addressing Cloud security issues is given by Vaquero et al. in [4]. It categorizes the most widely accepted Cloud security issues into three different domains of the Infrastructure as a Service model: machine virtualization, network virtualization and physical domain. It also proposes prevention frameworks on several architectural levels to address the identified issues.

Pearson [42] proposes several software design guidelines for delivering Cloud services taking privacy into account, such as using a privacy impact assessment, allowing user choice and providing feedback. While Chen et al. state in [5] that many IaaS-related Cloud security problems are problems of traditional computing solved by presented technology frameworks it also demands an architecture that enables "mutual trust" for the Cloud user and Cloud provider. Both papers confirm and complete the list of Cloud specific security issues identified by previous members of our research group, presented in [43]. Furthermore, they identified a demand for a two-way trust enabling architecture for Cloud infrastructures and the ability of "choosable security primitives with well considered defaults" [5]. The SAaaS architecture, proposed in this paper is targeted to provide this mutual trust. SAaaS' security audit policy language enables the user to define its own security policy and to choose from a spectrum of security subsystems as demanded by [5].

Zamboni et al. present in [44] show traditional Intrusion Detection Systems (IDS) can be enhanced by using autonomous agents. They confirm the advantages of using autonomous agents in regards to scalability and system overlapping security event detection. In contrast to our SAaaS architecture their research is focusing on the detection of intrusions into a relatively closed environment whereas our work applies to an open (cloud) environment where incidents like abuse of resources needs to be detected. Mo et al. introduce in [45] an IDS based on distributed agents using the mobile technology. They show how mobile agents can support anomaly detection thereby overcoming the flaws of traditional intrusion detection in accuracy and performance. The paradigm of cooperating distributed autonomous agents and its corresponding advantages for IDS' is shown by Sengupta et al. in [46]. The presented advantages apply for our SAaaS agents as well [1].

Amazon's Cloud platform Elastic Compute Cloud (EC2) allows users to create and share virtual machine images. Balduzzi et al. [11] analyzed the security risks of running third party images. The work gives a good insight about the current risk, which comes from pre-configured Cloud appliances. After the investigation of over 5000 Amazon Amazon Machine Images (AMIs), they found that 98% of Windows AMIs and 58% of Linux AMIs contain software with critical vulnerabilities [11]. Furthermore, two VM images were infected with malware, two were configured to write logs to an external machine, 21.8% contained leftover credentials that would allow a third party to remotely log into a machine [11].

Their approach is to start AMIs on EC2 and then scan them for security problems and privacy risks. However, their system is not intended to be used as an auditing service. They execute a predefined set of security and privacy checks and provide no way of customizing policies, which will be supported by the work proposed in this paper.

Wei et al. have worked on the problem of securing virtual machine images [47], and propose the *Mirage Image Management System* as a mitigation solution. Mirage is run by the Cloud provider to secure virtual machine images in his repositories. Mirage incorporates an access control framework and image filters, which remove unwanted information automatically from an image on publish/retrieve time. It incorporates a change tracking system and repository maintenance services, like periodic virus scanning. However, providing the appliance user and creator with a facility for easily creating custom policy definitions is not part of their work.

Schwartzkopf et al. present an "Update Checker", which investigates the up-to-dateness of installed software within VM images before they get launched in a Cloud environment [48]. They provide a graphical alerting method within the Cloud management system to inform the user about outdated software within a certain image. However, their approach only focuses on Linux images so far, which are being mounted and only checks for outdated software. The automatic audit system proposed in this work supports a wider spectrum of security or privacy checks through the concept of flexible security policies and offline and online investigation of VMs. However, the work proposed in [48] could be used to optimize the offline checks proposed in this paper.

Al Morsy and Faheem identified the need for automated policy enforcement systems [49]. Although, a lot of different security policy definition languages exist (e.g., LaSCO, XACML, SPARQL, etc.), it is shown that each of those has different limitations in terms of policy constraints. Therefore, Morsy and Faheem propose a policy automation framework including a new language called Standard Security Policy Language (SSPL), which tries to simplify the process of creating machine-readable security policies. The results of their policiy language analysis will be confirmed by the security policy language evaluation presented in this work. We further show why the developed Standard Security Policy Language does not meet the requirements of our work.

VIII. Conclusion & Outlook

In this paper, we introduced the Security Audit as a Service architecture to mitigate the shortcomings traditional audit systems suffer to audit Cloud computing environments. It was shown that SAaaS provides automatic auditing of virtual machine images according to custom user-defined policies. The results are reduced security and privacy risks, a well defined and reproducible audit process, as well as good documentation of results, when using 3rd party virtual machine appliances, while keeping the required technical understanding of the audit process to a reasonable minimum. The description of audit requirements in a Cloud audit policy language, allows abstraction of the audit requirements to a level, where even non-technical experts should be able to transfer company security and privacy policies into audit policy documents. A prototype was presented where user enable an automatic evaluation of an upscaling event. Furthermore, the advantages of using agents as a source for sensor information were shown. By utilizing lightweight, on-demand deployable agents it is possible to perform specific targeted audits every time a change within a user's Cloud infrastructure is performed. The current status of the work implements this on a user basis, but the system will be even more valuable when customer-overspanning events will be evaluated.

Therefore, as for future work an anomaly detection module will be developed, which is targeted to learn "normal" usage behaviour of Cloud instances by their users. As a result, this will enable the SAaaS system to detect anomalies within a Cloud infrastructure. For the presented Cloud audit policy language CAPL the extension of the CAPL schema by CIMI required objects is planned. This will enable the SAaaS system to audit any CIMI compatible infrastructures and the SAaaS system will be Cloud provider interoperable. Also, the implementation of a graphical user interface for the security policy modeller is planned. Thus, user will be able to easily define policies. Furthermore, it is planned to extend the CAPL class `MetaResource`, which provides a functions catalog of CAPL policies. User then could access this catalog to get automatically information about necessary parameters of a policy they like to define. At last, a security evaluation of the SAaaS system is planned to prove if it imposes new security risks to a Cloud environment.

Acknowledgment

This research is supported by the German Federal Ministry of Education and Research (BMBF) through the research grant number 01BY1116.

References

[1] F. Doelitzscher, C. Reich, M. Knahl, and N. Clarke, "Incident Detection for Cloud Environments," in *Proceedings of the Third International Conference on Emerging Network Intelligence (EMERGING 2011)*, no. 978-1-61208-174-8, 2011, pp. 100–105.

[2] Cloud Security Alliance, "Security Guidance for Critical Areas of Focus in Cloud Computing v2.1," 12 2009.

[3] European Network and Information Security Agency, "Cloud Computing Security Risk Assessment," Tech. Rep., 2009.

[4] L. Vaquero, L. Rodero-Merino, and D. Moran, "Locking the Sky: A Survey on IaaS Cloud Security," *Computing*, vol. 91, pp. 93–118, 2010.

[5] Y. Chen, V. Paxson, and R. H. Katz, "What's New About Cloud Computing Security?" EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-5, 01 2010.

[6] Cloud Industry Forum, "Cloud Adoption and Trends for 2013," vol. 08, 2013.

[7] F. Doelitzscher, C. Reich, and A. Sulistio, "Designing Cloud Services Adhering to Government Privacy Laws," in *Proceedings of 10th IEEE International Conference on Computer and Information Technology (CIT 2010)*, 2010, pp. 930–935.

[8] OpenNebula. (2012, July) OpenNebula Marketplace. [Online]. Available: http://marketplace.c12g.com/appliance-Accessed:10.06.2013

[9] Amazon. (2012, July) AWS Marketplace. [Online]. Available: http://aws.amazon.com/marketplace-Accessed:10.06.2013

[10] Amazon Web Services. (2012, July) How To Share and Use Public AMIs in A Secure Manner. [Online]. Available: http://aws.amazon.com/articles/0155828273219400-Accessed:10.06.2013

[11] M. Balduzzi, J. Zaddach, D. Balzarotti, E. Kirda, and S. Loureiro, "A Security Analysis of Amazon's Elastic Compute Cloud Service," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, ser. SAC '12. New York, NY, USA: ACM, 2012, pp. 1427–1434.

[12] S. Bugiel, S. Nürnberger, T. Pöppelmann, A.-R. Sadeghi, and T. Schneider, "Amazonia: When elasticity snaps back," in *Proceedings of the 18th ACM conference on Computer and communications security*, ser. CCS '11. New York, NY, USA: ACM, 2011, pp. 389–400.

[13] N. A. Haroon Meer. (2009) Clobbering the Cloud, part 4 of 5. [Online]. Available: http://www.defcon.org/images/defcon-17/dc-17-presentations/defcon-17-sensepost-clobbering_the_cloud.pdf-Accessed:10.06.2013

[14] N E Case. (2012, July) Extundelete. [Online]. Available: http://extundelete.sourceforge.net-Accessed:10.06.2013

[15] WinRecovery. (2012, July) Winundelete. [Online]. Available: http://www.winundelete.com-Accessed:10.06.2013

[16] Distributed Management Taskforce Inc. (DMTF). Cloud Infrastructure Management Interface (CIMI). Accessed: 10.06.2013. [Online]. Available: http://www.dmtf.org/standards/cloud-Accessed:10.06.2013

[17] T. Karbe, "Design and Development of an Audit Policy Language for Cloud Computing Environments," Cloud Research Lab - University of Apllied Sciences Furtwangen, Tech. Rep., 2013. [Online]. Available: http://wolke.hs-furtwangen.de/publications/theses

[18] L. Kagal. (2004) Rei Ontology Specifications. [Online]. Available: http://www.csee.umbc.edu/~lkagal1/rei/-Accessed:10.06.2013

[19] Imperial College London. (2013, March) Ponder2. [Online]. Available: http://www.ponder2.net-Accessed:10.06.2013

[20] J. A. Hoagland, R. Pandey, R. P, and K. N. Levitt. A Graph-based Language for Specifying Security Policies.

[21] James A. Hoagland, "Specifying and Implementing Security Policies Using LaSCO, the Language for Security Constraints on Objects," Ph.D. dissertation, University of California Davis, 2000. [Online]. Available: http://seclab.cs.ucdavis.edu/projects/arpa/LaSCO/dis/dissertation.pdf

[22] MS TechNet. (2004, 09) Windows Management Instrumentation. [Online]. Available: http://www.microsoft.com/germany/technet/datenbank/articles/600682.mspx-Accessed:10.06.2013

[23] SBLIM. (2009) SBLIM Project Wiki. [Online]. Available: http://sourceforge.net/apps/mediawiki/sblim/index.php?title=MainPage-Accessed:10.06.2013

[24] IBM. IBM Director. [Online]. Available: http://publib.boulder.ibm.com/infocenter/eserver/v1r2/index.jsp?topic=%2Fdiricinfo%2Ffqm0ccommoninfomodel-Accessed:10.06.2013

[25] L. Kagal. (2004) Rei Examples. [Online]. Available: http://www.csee.umbc.edu/~lkagal1/rei/examples/univ/-Accessed:10.06.2013

[26] DMTF Policy Working Group. CIM Schema Final Documentation. [Online]. Available: http://dmtf.org/sites/default/files/cim/cim_schema_v2340/cim_schema_2.34.0Final-Doc.zip-Accessed:10.06.2013

[27] Distributed Management Taskforce Inc. (DMTF). (2007, 02) Policy Profile. [Online]. Available: http://www.dmtf.org/sites/default/files/standards/documents/DSP1003.pdf-Accessed:10.06.2013

[28] ——. Cim schema - policy model. [Online]. Available: http://www.wbemsolutions.com/tutorials/CIM/cim-model-policy.html-Accessed:10.06.2013

[29] J. M. Bradshaw, *An Introduction to Software Agents*. Cambridge, MA, USA: MIT Press, 1997, pp. 3–46.

[30] Foundation for Intelligent Agents. (2002) Fipa acl message structure specification. [Online]. Available: http://www.fipa.org/specs/fipa00061/SC00061G.html-Accessed:10.06.2013

[31] IEEE Computer Society standards organization. Founddation for Intelligent Physical Agents - FIPA. http://www.fipa.org/. [Online]. Available: http://www.fipa.org/-Accessed:10.06.2013

[32] JADE Tutorials. HTTP MTP for Inter-Platform Communication. [Online]. Available: http://jade.tilab.com/doc/tutorials/JADEAdmin/HttpMtpTutorial.html-Accessed:10.06.2013

[33] S. Staniford-chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagl, K. Levitt, C. Wee, R. Yip, and D. Zerkle, "GrIDS - A Graph Based Intrusion Detection System For Large Networks," in *In Proceedings of the 19th National Information Systems Security Conference*, 1996, pp. 361–370.

[34] A. Sulistio, C. Reich, and F. Doelitzscher, "Cloud Infrastructure & Applications - CloudIA," in *Proceedings of the 1st International Conference on Cloud Computing (CloudCom'09)*, Beijing, China, 2009.

[35] David Grimshaw. JADE Administration Tutorial. [Online]. Available: http://jade.tilab.com/doc/tutorials/JADEAdmin-Accessed:10.06.2013

[36] J. Cucurull, R. Marti, G. Navarro-Arribas, S. Robles, B. Overeinder, and J. Borrell, "Agent mobility architecture based on IEEE-FIPA standards," *Computer Communications*, vol. 32, no. 4, pp. 712 – 729, 2009.

[37] inotify. - monitoring file system events. [Online]. Available: http://linux.die.net/man/7/inotify-Accessed:10.06.2013

[38] F. Doelitzscher and M. Ardelt and M. Knahl and C. Reich, "Sicherheitsprobleme für IT Outsourcing basierend auf Cloud Computing," *HMD - Praxis der Wirtschaftsinformatik*, vol. 281, 10 2011.

[39] Cloud Security Alliance, "Top Threats to Cloud Computing V1.0," 2010, https://cloudsecurityalliance.org/topthreats.html, 06.09.2011.

[40] European Network and Information Security Agency, "Cloud Computing Security Risk Assessment," Tech. Rep., 11 2009.

[41] T. Mather, S. Kumaraswamy, and S. Latif, *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*. O'Reilly Media, Inc., 2009.

[42] S. Pearson, "Taking Account of Privacy when Designing Cloud Computing Services," in *Proceedings of the ICSE Workshop on Software Engineering Challenges of Cloud Computing*, Vancouver, Canada, May 23 2009.

[43] F. Doelitzscher, C. Reich, M. Knahl, A. Passfall, and N. Clarke, "An Agent Based Business Aware Incident Detection System for Cloud Environments," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 1, no. 1, p. 9, 2012.

[44] J. Balasubramaniyan, J. Garcia-Fernandez, D. Isacoff, E. Spafford, and D. Zamboni, "An Architecture for Intrusion Detection Using Autonomous Agents," in *Computer Security Applications Conference, 1998, Proceedings., 14th Annual*, dec 1998, pp. 13 –24.

[45] Y. Mo, Y. Ma, and L. Xu, "Design and Implementation of Intrusion Detection Based on Mobile Agents," in *IT in Medicine and Education, 2008. ITME 2008. IEEE International Symposium on*, dec. 2008, pp. 278 –281.

[46] J. Sen, I. Sengupta, and P. Chowdhury, "An Architecture of a Distributed Intrusion Detection System Using Cooperating Agents," in *Computing Informatics, 2006. ICOCI '06. International Conference on*, june 2006, pp. 1 –6.

[47] J. Wei, X. Zhang, G. Ammons, V. Bala, and P. Ning, "Managing Security of Virtual Machine Images in a Cloud Environment," in *Proceedings of the 2009 ACM workshop on Cloud computing security*, ser. CCSW '09. New York, NY, USA: ACM, 2009, pp. 91–96.

[48] R. Schwarzkopf, M. Schmidt, C. Strack, and B. Freisleben, "Checking Running and Dormant Virtual Machines for the Necessity of Security Updates in Cloud Environments," in *Cloud Computing Technology and Science, 2011 IEEE Third International Conference on*, 2011, pp. 239 –246.

[49] M. Al-Morsy and H. Faheem, "A new standard security policy language," *Potentials, IEEE*, vol. 28, no. 2, pp. 19 –26, march-april 2009.

Effect of Radio Wave Obstruction by Obstacles on Performance of IEEE 802.16j Wireless Multi-Hop Relay Networks

Go Hasegawa*, Yuuki Ise†, Yoshiaki Taniguchi*, and Hirotaka Nakano*
*Cybermedia Center, Osaka University,
1–32, Machikaneyama, Toyonaka, Osaka, Japan
Email: {hasegawa, y-tanigu, nakano}@cmc.osaka-u.ac.jp
†Graduate School of Information Science and Technology, Osaka University,
1–5, Yamadaoka, Suita, Osaka, Japan
Email: y-ise@ist.osaka-u.ac.jp

***Abstract*—In IEEE 802.16j networks, radio wave interference between wireless links must be taken into account when radio resources are assigned to network links. The protocol model, which defines the transmission and interference ranges as circles, is well-known as one of the major radio interference models. Although a lot of related studies on IEEE 802.16j networks use the protocol model, they do not consider the presence of obstacles. In this paper, we first investigate the performance of IEEE 802.16j networks considering the effect of obstacles. For that purpose, we define an obstacle model and extended protocol model for accommodating obstacles, where radio waves propagation is obstructed by obstacles. Next, the performance of IEEE 802.16j networks is evaluated through simulation experiments using the obstacle model and the extended protocol model. Additionally, we present a method for estimating the performance of IEEE 802.16j networks with obstacles. Then, we use multiple regression analysis based on simulation results and construct regression equations for network service ratio and power-to-throughput ratio. By evaluating the accuracy of the developed equations based on real-world environment, we confirm that the service ratio can be estimated with a high degree of accuracy when the distribution density of obstacle is small.**

Keywords-IEEE 802.16j, wireless multi-hop networks, relay network, obstacles, radio wave blocking

I. Introduction

With the rapid progress of networking technologies, the demands for broadband access network environment is growing at various places such as home, office, and public areas. Wireless communication technologies are important to accommodate such services and users' demands. IEEE 802.16j [1][2] has attracted much attention to satisfy such demands, providing wider-area broadband wireless access environment. The IEEE 802.16j protocol utilizes multi-hop wireless networks for extending the network service area [3][4][5].

Generally, IEEE 802.16j wireless multi-hop relay networks (hereinafter, relay networks) consist of two types of nodes: gateway and relay nodes. As shown in Figure 1, there is a wired connection between the gateway node and an external network, while relay nodes communicate with the gateway node through wireless links. These nodes construct a tree topology where the root is the gateway node and there is a wireless multi-hop transmission path from any relay node to the gateway node [6][7][8]. A client terminal can access the external network by connecting to one of these nodes whose service area covers the client terminal [9]. One advantage of relay networks is that it is possible to extend the network service area by adding relay nodes without additional wired network facilities. In other words, the relay networks can provide wireless access environment by using multi-hop relaying to the area to which the radio waves cannot be directly reached from the gateway node. Therefore, the relay network is considered as possible networking technologies especially for thinly-populated regions and the area where the radio wave of gateway node is hard to reach due to underground and shades of buildings.

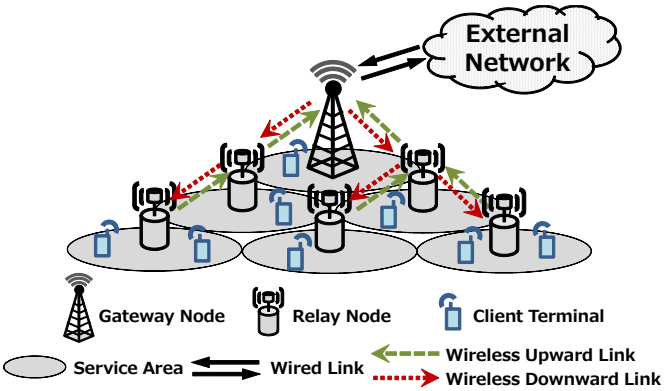


Figure 1. IEEE 802.16j multi-hop relay networks

In relay networks, obstacles, which refer in this paper to physical objects such as residential, office and commercial buildings, largely affect the connectivity between relay nodes and radio wave interference among wireless links. For example, even when two nodes exist in the transmission

range of each other, the obstacles can prevent the nodes from communicating with each other. Furthermore, obstacles can reduce the size of the service area. On the other hand, the obstacles can increase network performance by reducing the occurrence of radio wave interference since the interference range is limited by the obstacles. Therefore, since the obstacles have both advantage and disadvantage on the network performance, it is important to consider the presence of obstacles and their influence on radio wave propagation for assessing the network performance.

In wireless networks, there is a general problem related to radio wave interference. Specifically, multiple nodes that exist in the interference range of each other cannot successfully transmit the radio waves at the same time [10][11]. To avoid this problem, relay networks use an Orthogonal Frequency Division Multiple Access (OFDMA) protocol [12], which gives radio resources to network links as transmission opportunities [13]. Previous studies on relay networks have focused on preventing radio wave interference by introducing concepts such as link scheduling [14][15][16] and power control [17][18][19].

The protocol model [20], which defines the transmission and interference ranges as circles, is well-known as one of the major radio interference models. In the protocol model, whether a transmission succeeds or encounters interference depends on only the distance between nodes, which are obtained through comparison with the transmission and interference ranges of other nodes. Therefore, the connectivity between relay nodes and radio wave interference among network links can be easily determined. Although a lot of studies on relay networks use the protocol model [21][22][23][24][25], they did not consider the presence of obstacles.

In this paper, extending the paper [1], we investigate the performance of IEEE 802.16j networks considering the effect of obstacles. First, we define an obstacle model which determines location and size of obstacle in the network. Then, we extend the function of the protocol model for accommodating obstacles. Specifically, radio waves propagation is obstructed by the obstacles in the extended protocol model. Next, we consider both positive and negative aspects for the relay network due to radio wave obstruction. Using the obstacle model and the extended protocol model, the performance of relay networks is evaluated through simulation experiments in terms of network service ratio and power-to-throughput ratio.

Furthermore, we propose a method for estimating the performance of relay networks. In the network construction process, it is helpful to estimate the network performance before the network is actually constructed. For example, the performance estimation enables the number of nodes deployed in the network or the transmission range of the nodes to be adjusted in order to achieve pre-determined performance goals such as service ratio, network throughput, and power consumption. For this purpose, by using multiple regression (polynomial regression) analysis of numerical results of simulation experiments, the regression equations are derived to estimate the performance of relay networks. We confirm the effectiveness of the estimation method by evaluating the accuracy of regression equations. Specifically, we conduct simulation experiments based on the real-world environment to assess the estimation accuracy of the proposed method.

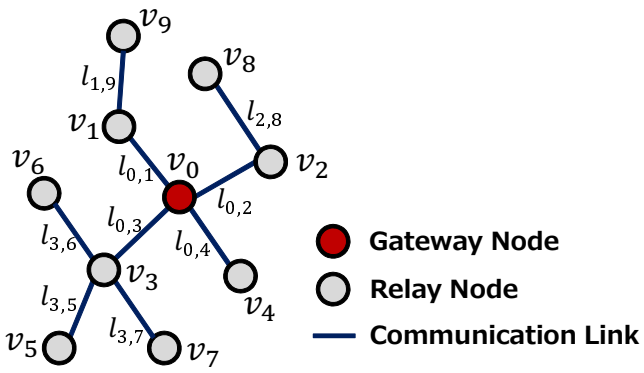


Figure 2. Network topology

The rest of this paper is organized as follows. In Section II, we introduce the network model of the relay networks. Section III introduces the obstacle model and the extended protocol model. The simulation results based on the proposed models are given in Section IV. Then in Section V, we present a method for estimating the performance of relay networks and evaluate the effectiveness of method. Finally, Section VI concludes this paper and describes future work.

II. System model

This section describes the network model, the radio interference model, and time slot assignment mechanism utilized in this paper.

A. Network model

The network is assumed to consist of N nodes, where v_i $(0 \leq i \leq (N-1))$ denotes both the i-th node and the point of the node's location in a field. One node in the network, denoted as v_0, serves as the gateway node, and the remaining nodes function as relay nodes, constructing a network topology that describes the communication between all nodes in the form of a directed graph. In relay networks, the gateway node is connected to an external network, and the relay nodes communicate with the gateway node either directly or via other relay nodes along the path between the relay node and the gateway node. There are two kinds of communications: upward communication and downward communication. In upward communication, data is transferred from relay nodes toward the gateway

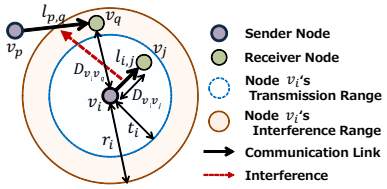


Figure 3. Radio interference based on the protocol model

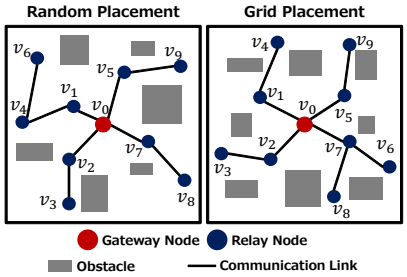


Figure 4. Obstacle model

node. Conversely, data is transferred from the gateway node toward relay nodes in downward communication. The path is determined by a routing algorithm, and the directed graph is constructed as a tree structure whose root is the gateway node v_0. Note that we do not consider the user clients which connect to the relay node.

Figure 2 shows an example of a network topology where a gateway node and nine relay nodes are deployed. In the figure the link between nodes means two directed links. In the figure, the red circle indicates the gateway node, gray circles indicate relay nodes, and solid lines indicate communication links respectively. Here, a communication link from node v_i to node v_j is denoted as $l_{i,j}$.

B. Radio interference model

In this paper, the propagation and interference of radio waves are modelled by the protocol model [20]. The transmission and interference range of node v_i is defined as the following sets of points.

$$\mathcal{T}_i = \{p \mid D_{v_i p} \leq t_i\} \quad (1)$$

$$\mathcal{R}_i = \{p \mid D_{v_i p} \leq r_i\} \quad (2)$$

Here, D_{ab} is the the distance between a and b. In addition, t_i and r_i represent the transmission range and the interference range of node v_i, respectively. In general, $r_i > t_i$, and the ratio of the interference range to the transmission range for node v_i is set to be between around 2 and 4 depending on the environment [26].

Based on the protocol model, the conditions to determine the succeess of transmission and the occurrence of the interference are as follows. Node v_j can receive a transmission from node v_i when $v_j \in \mathcal{T}_i$ is satisfied. Figure 3 shows an example of radio interference between communication links. In this figure, there are four nodes maintaining two communication links $l_{i,j}$ and $l_{p,q}$. The protocol model defines the radio interference between $l_{i,j}$ and $l_{p,q}$ based on the distances between the four vertices v_i, v_j, v_p, and v_q. When $v_q \in \mathcal{R}_i$ is satisfied, link $l_{i,j}$ interferes with link $l_{p,q}$.

C. Time slot assignment mechanism

The IEEE 802.16j protocol uses the OFDMA mechanism to control the ability of nodes to transmit by assigning radio resources for transmission. In the OFDMA mechanism, radio resources are divided along both frequency and time dimensions. For simplicity, in this paper, each divided radio resource is regarded as a time slot. In the relay networks, time slots are assigned to wireless links as transmission opportunities. Then, different time slots are assigned to wireless links that interfere with each other in order to prevent radio wave interference. On the other hand, multiple links can communicate simultaneously within the same time slot as long as the time slot is assigned to the links that do not interfere with each other. This mechanism is known as spatial reuse of wireless resource [27][28].

The performance of relay networks can be improved by spatial reuse with concurrent transmissions since such an approach reduces the total number of time slots assigned to all communication links in the network. The time slot assignment problem with consideration of spatial reuse is regarded as a vertex coloring problem [29] of the conflict graph [30]. In the conflict graph, a vertex represents a link in the network and an edge between two vertices is constructed when the corresponding links interfere with each other, and time slots can be assigned to links in the network by allocating different colors to adjacent vertices in the conflict graph. However, since the vertex coloring problem is known to be NP-hard [31][32], heuristic algorithms have been proposed for solving the problem [33][34][35][36]. In this paper, we use the method proposed in [35] to assign time slots to links for performance evaluation.

III. Proposed Models

This section introduces an obstacle model utilized in this paper, followed by the extension of the protocol model for accommodating the effect of the obstacles on radio wave propagation. Finally, using the obstacle model and extended protocol model, the influence of obstacles on network performance is discussed.

A. Obstacle model

Figure 4 depicts the obstacle model, where rectangular obstacles are deployed in the field. In the model, two placement patterns are considered. In random placement, the obstacles are deployed at random in the field. In grid placement, on the other hand, the obstacles are deployed

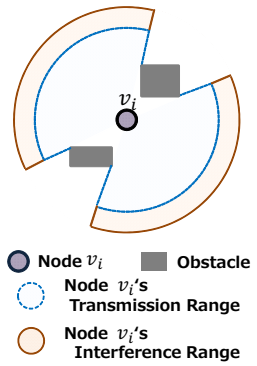


Figure 5. Extended protocol model

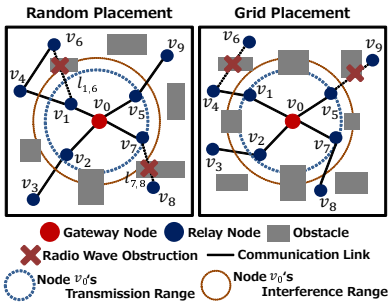


Figure 6. Combined image of the obstacle model and the extended protocol model

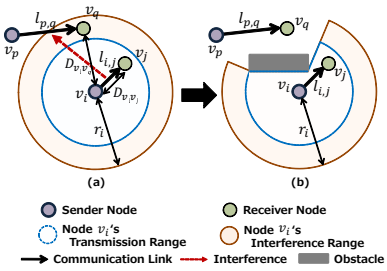


Figure 7. The influence of obstacles on radio wave interference

in a grid pattern. In both placements, the obstacles are not deployed at the center of the field, where the gateway node is located. The side lengths of the obstacles are chosen at random from within a certain range, and the obstacles are placed parallel to the field. The relay nodes cannot be deployed at locations occupied by obstacles. The height of the obstacles is ignored since the network model is constructed in a plane. Here, the following function $O(k)$ is defined whether or not a point k in the field is occupied by obstacles.

$$O(k) = \begin{cases} 1 & \text{point } k \text{ is occupied by obstacles} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

B. Extended protocol model

In general, the effects of obstacles on radio waves include obstruction, reflection, and diffraction. We consider only radio wave obstruction in this paper since it has great effects on relay network performance.

For evaluating the effects of obstacles, we extend the protocol model considering radio wave obstruction by obstacles. The modified model is crefered to as extended protocol model. In this model, the transmission and interference range of node v_i are also defined as sets of points. The sets $\mathcal{T}_i'$ and $\mathcal{R}_i'$ are described as follows.

$$\mathcal{T}_i' = \{p' \mid p' \in \mathcal{T}_i \text{ and } C(v_i, p') = 0\} \quad (4)$$
$$\mathcal{R}_i' = \{p' \mid p' \in \mathcal{R}_i \text{ and } C(v_i, p') = 0\} \quad (5)$$

Here, $C(a, b)$ is the following function that determines the presence of obstacles between two points a and b.

$$C(a, b) = \begin{cases} 1 & \exists k(D_{ak} + D_{bk} = D_{ab} \text{ and } O(k) = 1) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where D_{ak}, D_{bk}, and D_{ab}, means the distance between points a and b.

Figure 5 shows an example of the extended protocol model. The limitation of the transmission and interference range of node v_i by obstacles is confirmed in this figure.

C. Effect of obstacles on relay network performance

Figure 6 shows the combined image of the obstacle model and the extended protocol model. The effect of obstacles on the connectivity between relay nodes is represented in this figure. For example, in the left panel of Figure 6, $l_{1,6}$ and $l_{7,8}$ become disconnected due to radio wave obstruction by obstacles. In this case, v_6 can connect to the network via v_4. On the other hand, v_8 is completely disconnected from the network by another obstacle. This is a negative aspect of obstacles in relay networks, owing to the increased number of isolated nodes and the higher average hop count between relay nodes and the gateway node.

On the other hand, Figure 7 shows an example of a beneficial effect of radio wave obstruction. Although in Figure 7(a) $l_{i,j}$ and $l_{p,q}$ interfere with each other, by adding obstacles as shown in Figure 7(b), the interference range of v_i is limited and v_q is not affected by the interference from v_i. As a result, the addition of an obstacle allows these two links to transmit simultaneously, which is a positive aspect of obstacles.

As described above, obstacles entail both advantages and disadvantages in terms of network performance. Therefore, it is important to consider the presence of obstacles and their influence on radio wave propagation for assessing the network performance.

IV. Performance evaluation and discussions

In this section, we evaluate the influence of obstacles on the performance of relay networks through simulation experiments by using the obstacle model and the extended protocol model proposed in the previous section. We utilized the simulator built in our labortory since there is no existing simulator which can simulate thedetailed behavior of IEEE 802.16j networks with obstacles.

A. Evaluation settings

In the simulation experiments, one gateway node was placed at the center of a 1×1 square area, and 99 relay nodes were distributed at random locations in the field. The transmission range was set to 0.15, 0.20, 0.25, 0.30, or 0.35, where all relay and gateway nodes utilize the same value for each experiment. The ratio of the interference range to the transmission range was set to 2.0 for all nodes. Note that we have confirmed the setting of this ratio does not change the overall tendency of the following results. A network topology was constructed such that the hop count between the gateway node and each relay node was minimized. The number of obstacles was set to be from 0 to 250 at intervals of 25 in the case of random placement. For the grid placement, we choose one the following placement patterns: 3×3 (8 obstacles), 5×5 (24 obstacles), and 7×7 (48 obstacles). The length of the sides of each obstacle was set to a random value between 0.004 and 0.04, assuming that the obstacles are placed in 2000m$\times$2000m area of real-world environment. We determine the traffic demand from relay nodes to the gateway nodes according to the Voronoi diagram for each relay node, where we assume the user clients are distributed uniformly in the area and they generate the same amount of traffic to the gateway node.

We observed the *service ratio* and the *power-to-throughput ratio* as network performance metrics. The service ratio is the ratio of the area where the relay network can provide service to the overall field area, excepting the area of the obstacles. The power-to-throughput ratio is the value of the total power consumption of the nodes divided by the gateway throughput. Here, the total power consumption is simply defined as the sum of the squares of the transmission ranges of all connected nodes, and the gateway throughput is defined as the ratio of the number of time slots assigned to links directly connected to the gateway node against the number of slots assigned to all communication links in the network. We focused only on the downward communication and conducted 100,000 iterations of the simulation experiments for each set of parameter settings and the all results were divided according to the number of connected nodes excluding isolated nodes, and the average values were used for performance evaluation.

B. Effect of the number of obstacles

Figure 8 depicts the service ratio and the power-to-throughput ratio in the random placement pattern as a function of the number of connected nodes when the transmission range is set to 0.20. The x-axis of graph means the number of connected nodes. Note that the x-axis value of less than 100 means that there are some nodes disconnected from the network due to radio wave obstruction by obstacles.

As shown in Figure 8, both metrics increase as the number of connected nodes increases, but there are differences in their increase tendency. The service ratio in Figure 8(a) decreases as the number of obstacles increases regardless of the number of connected nodes since radio wave propagation is obstructed by obstacles. On the other hand, the power-to-throughput ratio in Figure 8(b) does not show such a simple trend. When the number of obstacles increases from 0 to 150, the power-to-throughput ratio decreases, whereas in the case of the increase in the number of obstacles from 150 to 250, the power-to-throughput ratio increases together with the number of obstacles. The reason for this is as follows.

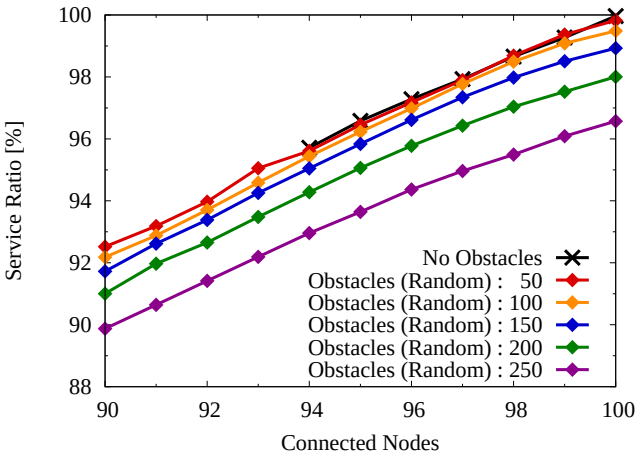
When the number of obstacles increases from 0 to 150, multiple wireless links in the network come to be able to transmit simultaneously due to the limitation of the interference range by obstacles. As a result, the network throughput is improved. On the other hand, as the number of obstacles increases further, the links between nodes are likely to become disconnected due to the radio wave obstruction by obstacles. It leads to the decrease in the network throughput.

C. Effect of the obstacle placement pattern

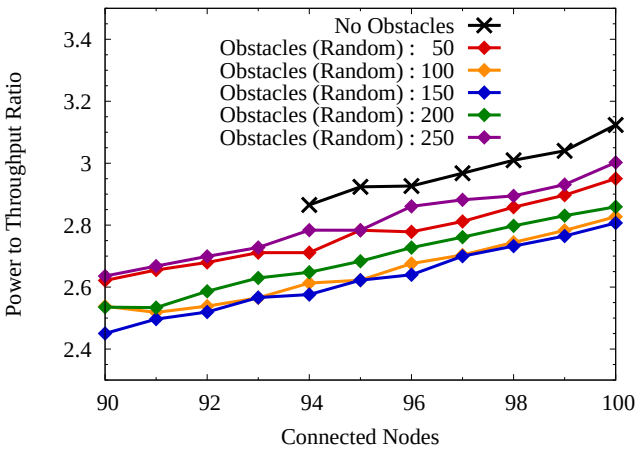
Figure 9 represents the effects of obstacle placement pattern on the network performance where the results of random and grid placement patterns with similar number of obstacles. In terms of service ratio, the obstacle placement pattern and the number of obstacles have little impact as shown in Figure 9(a). This is because the number of obstacles is small as compared with Figure 8, and the service ratio is mainly dependent on the number of connected nodes. On the other hand, Figure 9(b) shows that the decrease trend of the power-to-throughput ratio as the number of obstacles increases is different in the two placement patterns, and when comparing the same number of obstacles in both patterns, the grid placement pattern shows lower power-to-throughput ratio. This is because in the grid placement pattern the number of connected nodes is barely affected since the obstacles are regularly spaced. As a result, the network throughput is improved and the power-to-throughput ratio decreases since the interference is reduced effectively due to the presence of obstacles.

D. Effect of the transmission range

Figure 10 shows the relationship between the service ratio and the power-to-throughput ratio when the transmission range of the nodes is set to 0.15, 0.20, 0.25, 0.30, and

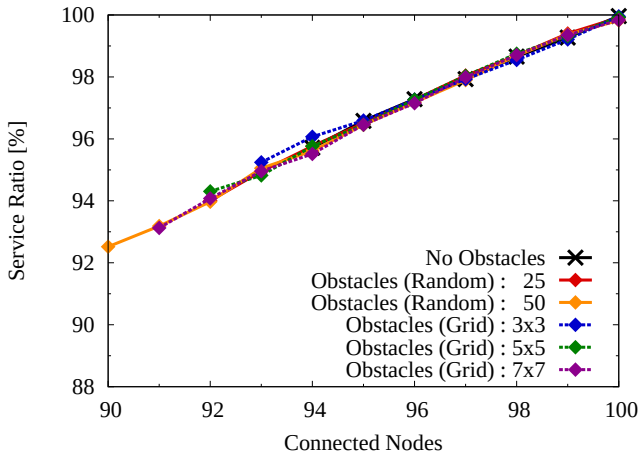


(a) Service ratio

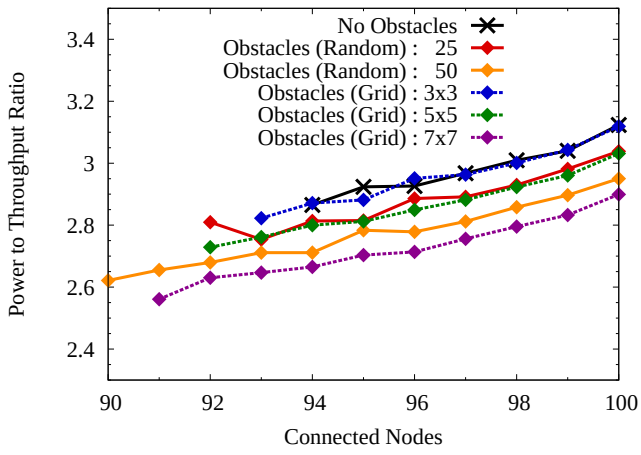


(b) Power-to-throughput ratio

Figure 8. Effect of the number of obstacles with random placement pattern



(a) Service ratio



(b) Power-to-throughput ratio

Figure 9. Effect of the obstacle placement pattern

0.35. In this case, 200 obstacles are placed with random placement pattern. In the graph, the number of connected nodes is indicated for each plot, where it increases from left to right and the rightmost point indicates the average value when the number of connected nodes is 100.

We focused on the plots denoted with squares in the figure when the number of connected nodes is 95. Here, there are two methods for improving the service ratio. One involves increasing the transmission range of each node, and the other involves deploying additional nodes in the network. As shown in Figure 10, when the transmission range becomes large, the power-to-throughput ratio increases rapidly while the service ratio increases. On the other hand, by deploying additional nodes in the field, the service ratio can be enhanced with a small increase of the power-to-throughput ratio. Also, we can see from this figure that the power-to-throughput ratio does not so increased as transmission power increases, compared with the case of no obstacles where the power is proportional to the square of the transmission range. This is because of the another effect of the obstacles. However, we conclude that the deployment of additional nodes can improve the service ratio more effectively than the increase in the transmission range of nodes.

V. Estimation of Network Performance

As described in Section IV, the performance of relay networks is largely affected by the various network parameters such as the number of connected nodes, a transmission range of each node, and distribution density of obstacles. In this section, we propose a method for estimating the performance of relay networks on the basis of the regression analysis of the simulation results. The accuracy of the analysis is then evaluated using both obstacle model and the real-world environment.

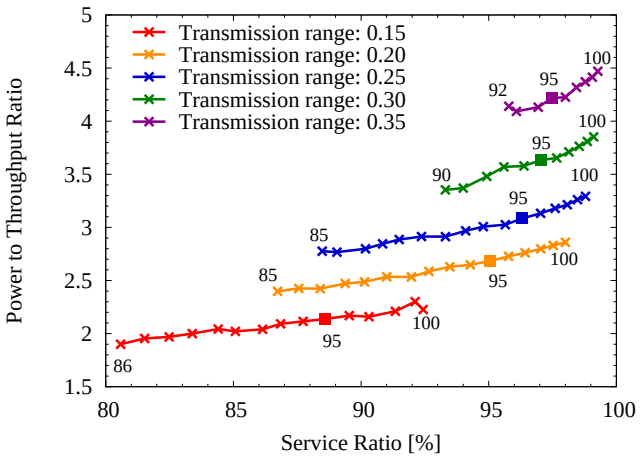


Figure 10. Effect of transmission range on network performance

A. Regression analysis of simulation results

As a method for estimating network performance, regression equations are derived based on the simulation results presented in the previous section. Specifically, the equations for the service ratio and the power-to-throughput ratio are denoted as $S(n,t,d)$ and $P(n,t,d)$, where n, t, and d represent the number of connected nodes in the network, the transmission range of the nodes, and the distribution density of obstacles, respectively. We take there three parameters since they affect the performance of the relay network considered in this paper. The distribution density of obstacles is defined as the ratio of the area of obstacles to the overall area. All parameters are normalized to fall within the range between 0 and 1 based on the maximum values in the simulation experiments. By using these network parameters and the network performance as explanatory variables and objective variables, respectively, multiple regression analysis is conducted.

We conducted Microsoft Excel for regression analysis. As the result, we obtain the following equations to estimate the service ratio and the power-to-throughput ratio.

$$S(n,t,d) = 7.78 + 92.54n + 34.95t^2 - 27.9d \quad (7)$$

$$P(n,t,d) = -1.21 + 2.42n + 9.25t - 2.14d \quad (8)$$

B. Accuracy of the regression equations

In order to examine the accuracy of the regression equations, the estimation values from the equations are compared with the simulation results. In detail, for each experiment, the obstacle placement pattern is first determined based on the obstacle model or the real-world environment. Next, on the basis of each obstacle placement pattern, simulation experiments are conducted, where the number of nodes is 100, and the transmission range of each node is set from 0.15 to 0.35. As a result, the service ratio and the power-to-throughput ratio are obtained as simulation results. Moreover, by using the regression equations in Equations (7) and (8), the estimation values are calculated based on each parameter setting utilized in the simulation experiments. Finally, the estimation accuracy of the regression equations is evaluated through comparison with the simulation results and estimation values. Here, as a metric of the accuracy, the relative error E_r in Equation (9) is utilized, where V_r and V_s indicate the simulation result and the estimation value, respectively.

$$E_r = \frac{|V_r - V_s|}{V_s} \quad (9)$$

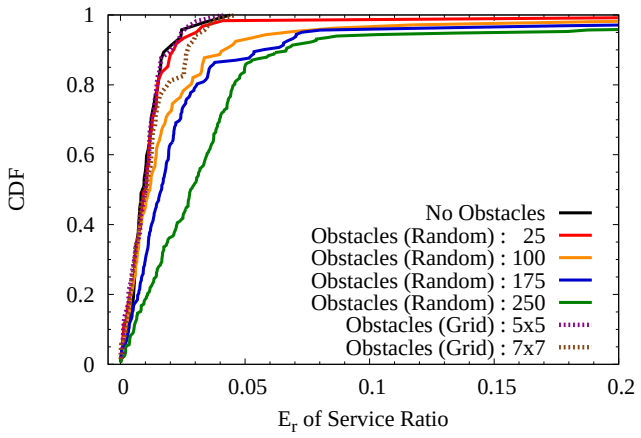
1) Evaluation results based on the obstacle model: Figures 11(a) and 11(b) depict the distributions of the relative error for the service ratio and the power-to-throughput ratio, respectively, with several values for the number of obstacles in the case of both placement patterns. From the figures, it is observed that the accuracy of the equations is high regardless of the obstacle placement pattern. However, the figures also represent that the increase in the number of obstacles leads to deterioration of the accuracy in both metrics. This is because when the number of obstacles is large, the obstacles have greater impact on these network performance than expected.

Therefore, these results show that the proposed methods can provide accurate estimates of network performance without simulation experiments when the number of obstacles is small.

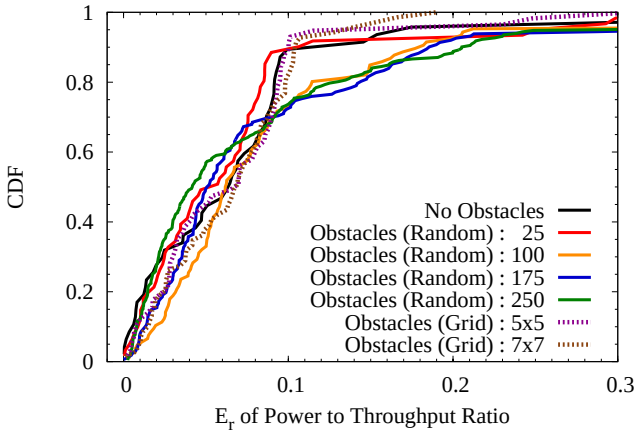
2) Evaluation results based on the real-world environment: Next, the accuracy of the equations is evaluated based on the real-world environment. For this purpose, we obtained obstacle placement patterns from maps in the real world by using Google Maps API [37]. In detail, 2000m×2000m square areas are randomly chosen from the field, ranging from Osaka Prefecture to Mie Prefecture in Japan, which satisfies the conditions that the latitude ranges from 34.5 to 35.5 degrees north and the longitude ranges from 135.5 to 136.5 degrees east, which corresponds to the residential area of the north part of Osaka, Japan. The fields whose distribution density of obstacles is from 1% to 25% are utilized for evaluation.

Figure 12 shows the distribution of relative error of the service ratio and the power-to-throughput ratio, where each plot represents the average of the results when the number of connected nodes is more than 30 in the simulation experiments. Although, in terms of the service ratio, the relative error is within 0.1 when the distribution density of obstacle is small, it becomes large as the distribution density of obstacle increases due to the biased tendency of distribution of obstacles. As a result, when the distribution density of obstacles is small, the service ratio can be estimated with a high degree of accuracy using the proposed method in real-world environment.

On the other hand, the relative error for the power-to-throughput ratio varies as shown in Figure 12(b) even when

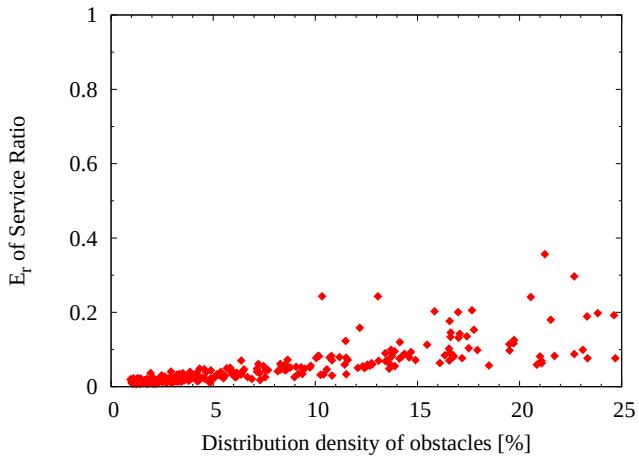


(a) Service ratio

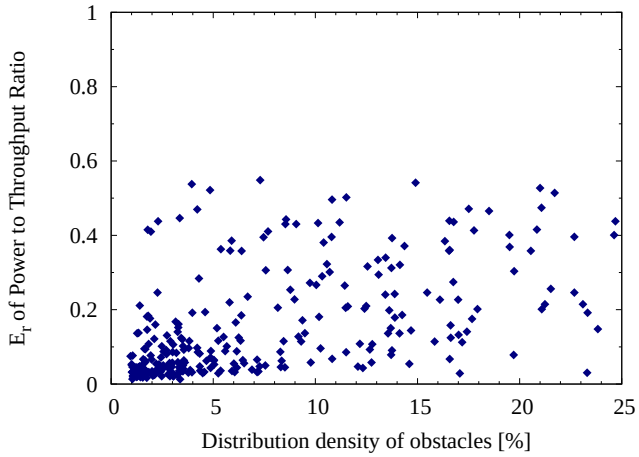


(b) Power-to-throughput ratio

Figure 11. Distribution of relative error for the obstacle model



(a) Service ratio



(b) Power-to-throughput ratio

Figure 12. Distribution of relative error for the real-world environment

the distribution density of obstacles is small. The possible reason is as follows. In the field where there is a lot of obstacles concentrating around the gateway node, the link directly connected to the gateway node is likely to become disconnected due to the radio wave obstruction by obstacles. Although the regression equations are constructed based on simulation results including such situations, the decrease in the number of links of the gateway node has a great impact on the throughput beyond the expectation especially when the distribution density of obstacles is large. Therefore, the accuracy of the proposed method becomes worse in terms of the power-to-throughput ratio.

Figure 13 represents the examples of this case, where black part means the obstacles. 1Although the distribution density of obstacles in both cases is around 17%, the estimation accuracy is different. In Figure13(a), the relative error is 0.0285, while it is 0.471 for Figure 13(b), where many obstacles located at the center area of the field.

VI. Conclusions

In this paper, the performance of IEEE 802.16j multi-hop networks was investigated by considering the presence of obstacles. We first defined the obstacle model which determines location and size of each obstacle in the network, and extended the protocol model for determining the connectivity and interference relationships considering radio wave obstruction by obstacles. Simulation experiments using the proposed models revealed that the deployment of additional relay nodes improves the service ratio more effectively than an increase in the radio transmission range of each relay node. We also revealed the effects of the network parameters, such as the number of connected nodes, a transmission range of each node, and distribution density of obstacles, on the performance of IEEE 802.16j networks.

In addition, a method for estimating the performance of relay networks on the basis of regression analysis was also

(a) Lower estimation error (E_r = 0.0285) (b) Larger estimation error (E_r = 0.471)

Figure 13. Examples of the real-world environment

proposed. By comparing between the simulation results and the estimation values derived from the regression equations, we confirmed that the equations can yield an accurate estimation of network performance when the number of obstacles is small. This results help us estimate the performance of relay networks with obstacles in designing step. However, the accuracy of the proposed equations was deteriorated in terms of the power-to-throughput ratio using the real-world environment even when the number of obstacles is small, since the links directly connected to the gateway node are likely to become disconnected due to the radio wave obstruction by obstacles.

Future work will be directed toward applying more precise radio interference models, including signal-to-interference-plus-noise ratio (SINR) model in order to consider other effects of obstacles, such as reflection and diffraction of radio waves. We need to evaluate the validity of the regression model and to consider more detailed obstract model which accommodate the reflection and diffraction of radio waves. We also plan to evaluate of the proposed model with packet-level performance metric such as packet delivery delay and packet loss ratio. Furthermore, we need to apply the proposed scheme to multi-carrier OFDMA system, whereas in this paper we implicitly assume the single-carrier OFDMA where the radio resources are only time slots. Node replacement for obtaining better performance based on the proposed regression model is another interesting topic.

REFERENCES

[1] Y. Ise, G. Hasegawa, Y. Taniguchi, and H. Nakano, "Evaluation of IEEE 802.16j relay network performance considering obstruction of radio waves propagation by obstacles," in *Proceedings of ICNS 2012*, Mar. 2012.

[2] IEEE Std 802.16j, *IEEE standard for local and metropolitan area networks, Part 16: Air interface for fixed broadband wireless access systems, Amendment 1: Multihop relay specification*, June 2009.

[3] S. W. Peters and R. W. H. Jr, "The future of WiMAX: Multihop relaying with IEEE 802.16j," *IEEE Communications Magazine*, vol. 1, pp. 104–111, Jan. 2009.

[4] V. Genc, S. Murphy, Y. Yu, and J. Murphy, "IEEE 802.16j relay-based wireless access net- works: An overview," *IEEE Wireless Communications*, vol. 15, no. 5, pp. 56—–63, Oct. 2008.

[5] D. Kumar and N. Nagarajan, "Technical issues in IEEE 802.16j mobile multi-hop relay (mmr) networks," *European Journal of Scientific Research*, vol. 65, no. 4, pp. 507–—533, Dec. 2011.

[6] M. Okuda, C. Zhu, and D. Viorel, "Multihop relay extension for WiMAX networks - Overview and benefits of IEEE 802.16j standard," *Fujitsu Scientific and Technical Journal*, vol. 44, no. 3, pp. 292–302, Jan. 2008.

[7] F. E. Ismael, S. K. S. Yusof, and N. Fisal, "An efficient bandwidth demand estimation for delay reduction in IEEE 802.16j MMR WiMAX networks," *International Journal of Engineering*, vol. 3, no. 6, pp. 554–564, Jan. 2010.

[8] B. Lin, P. Ho, L. Xie, and X. Shen, "Optimal relay station placement in IEEE 802.16j networks," in *Proceedings of IWCMC 2007*, Aug. 2007.

[9] D. Niyato, E. Hossain, D. I. Kim, and Z. Han, "Joint optimization of placement and bandwidth reservation for relays in IEEE 802.16j mobile multihop networks," in *Proceedings of IEEE ICC 2009*, pp. 4843–4847, Jun. 2009.

[10] G. Zhou, T. He, J. A. Stankovic, and T. Abdelzaher, "RID: Radio interference detection in wireless sensor networks," in *Proceedings of INFOCOM 2005*, pp. 891–901, Mar. 2005.

[11] A. P. Subramanian, M. M. Buddhikot, and S. Miller, "Interference aware routing in multi-radio wireless mesh networks," in *Proceedings of WiMesh 2006*, pp. 55–63, Sep. 2006.

[12] S. Chieochan and E. Hossain, "Adaptive radio resource allocation in OFDMA systems: A survey of the state-of-the-art approaches," *Wireless Communications and Mobile Computing*, vol. 9, no. 4, pp. 513—–527, Apr. 2009.

[13] D. Ghosh, A. Gupta, and P. Mohapatra, "Scheduling in multihop WiMAX networks," *ACM SIGMOBILE Mobile Computing and Communication Review*, vol. 12, pp. 1–11, Apr. 2008.

[14] C.-Y. Hong and A.-C. Pang, "3-approximation algorithm for joint routing and link scheduling in wireless relay networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 856–861, Feb. 2009.

[15] D. Ghosh, A. Gupta, and P. Mohapatra, "Adaptive scheduling of prioritized traffic in IEEE 802.16j wireless networks," in *Proceedings of WiMob 2009*, pp. 307–313, Oct. 2009.

[16] S. Yang, C. Kao, W. Kan, and T. Shih, "Handoff minimization through a relay station grouping algorithm with efficient radio-resource scheduling policies for IEEE 802.16j multihop relay networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2185—–2197, Jun. 2010.

[17] V. Genc, S. Murphy, and J. Murphy, "Analysis of transparent mode IEEE 802.16j system performance with varying numbers of relays and associated transmit power," *IEEE Wireless Communications & Networking Conference*, pp. –, Apr. 2009.

[18] A. Singh and V. Potdar, "Torpid mode: Hybrid of sleep and idle mode as power saving mechanism for IEEE 802.16j," in *Proceedings of IEEE WAINA 2010*, Apr. 2010.

[19] J. Liang, Y. Wang, J. Chen, J. Liu, and Y. Tseng, "Energy-efficient uplink resource allocation for IEEE 802.16j transparent-relay networks," *Computer Networks*, vol. 55, no. 16, pp. 3705—–3720, Jun. 2011.

[20] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, pp. 388–404, Mar. 2000.

[21] X. Meng, K. Tan, and Q. Zhang, "Joint routing and channel assignment in multi-radio wireless mesh networks," in *Proceedings of ICC 2006*, Jun. 2006.

[22] H. Venkataraman, A. Krishnamurthy, P. Kalyampudi, J. McManis, and G.-M. Muntean, "Clustered architecture for adaptive multimedia streaming in WiMAX-based cellular networks," in *Proceedings of the World Congress on Engineering and Computer Science*, vol. 2, pp. 753–758, Oct. 2009.

[23] P. Thulasiraman and X. Shen, "Interference aware subcarrier assignment for throughput maximization in OFDMA wireless relay mesh networks," in *Proceedings of ICC 2009*, pp. 14–18, June. 2009.

[24] C. Cicconetti, I. F. Akyildiz, and L. Lenzini, "Bandwidth balancing in multi-channel IEEE 802.16 wireless mesh networks," in *Proceedings of INFOCOM 2007*, vol. 5, pp. 6–12, May. 2007.

[25] Y. Lu and G. Zhang, "Maintaining routing tree in IEEE 802.16 centralized scheduling mesh networks," in *Proceedings of 16th International Conference on Computer Communications and Networks 2007*, pp. 240–245, Aug. 2007.

[26] W. Wang, Y. Wang, X. Y. Li, W. Z. Song, and O. Frieder, "Efficient interference-aware TDMA link scheduling for static wireless networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking*, pp. 262–273, Sep. 2006.

[27] L. Kleinrock and J. Silvester, "Spatial reuse in multihop packet radio networks," *Proceedings of the IEEE*, vol. 75, no. 1, pp. 156–167, Jan. 1987.

[28] L.-W. Chen, Y.-C. Tseng, D.-W. Wang, and J.-J. Wu, "Exploiting spectral reuse in routing, resource allocation, and scheduling for IEEE 802.16 mesh networks," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 301–313, Jan. 2009.

[29] M. V. Marathe, H. Breu, H. B. Hunt, S. S. Ravi, and D. J. Rosenkrantz, "Simple heuristics for unit disk graph," *Networks*, vol. 25, pp. 59–68, Sep. 1995.

[30] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," *Wireless Networks*, vol. 11, pp. 471–487, Jul. 2005.

[31] S. Khanna, N. Linial, and S. Safra, "On the hardness of approximating the chromatic number," *Combinatorica*, vol. 20, no. 3, pp. 393–415, Mar. 2000.

[32] B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit disk graphs," *Discrete mathmatics*, vol. 86, no. 1–3, pp. 165–177, Dec. 1990.

[33] W. Klotz, "Graph coloring algorithms," *Mathematical report TU-Clausthal*, vol. 5, pp. 1–9, May. 2002.

[34] V. A. Kumar, M. V. Marathey, S. Parthasarathyz, and A. Srinivasan, "Algorithmic aspects of capacity in wireless networks," *ACM SIGMETRICS Performance Evaluation Review*, vol. 33, pp. 133–144, Jan. 2005.

[35] R. Ishii, G. Hasegawa, Y. Taniguchi, and H. Nakano, "Time slot assignment algorithms in IEEE 802.16 multi-hop relay networks," in *Proceedings of ICNS 2010*, pp. 265–270, Mar. 2010.

[36] J. Tang, G. Xue, C. Chandler, and W. Zhang, "Link scheduling with power control for throughput enhancement in multihop wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 3, pp. 733–742, May 2006.

[37] Google Maps API, available at http://code.google.com/apis/maps, accessed on 22nd May 2013.

An Algorithm for Variability Identification by Selective Targeting

Anilloy Frank
Institute of Technical Informatics,
Graz University of Technology
Inffeldgasse 16, 8010 Graz, Austria
Email: frankanilloy@gmail.com

Eugen Brenner
Institute of Technical Informatics,
Graz University of Technology
Inffeldgasse 16, 8010 Graz, Austria
Email: brenner@tugraz.at

***Abstract*—Large companies have large embedded software systems, where common and reusable software parts are distributed in various interrelated subsystems that also have lots of uncommon and non-reusable parts. The approach finds software parts that may or may not be reusable in a particular application engineering project. It is the task of application engineering to figure out whether the identified components and variants are directly reusable and reuse them in application engineering. In Software Product Lines, the identified reusable common and variable components should be generalized and stored into asset bases. In real life, it may be too much effort and costs to generalize application level assets into domain assets and it is just more feasible to try to find reusable common and variable components directly from existing applications. The proposed approach is selectively targeting the component-feature model instead of an inclusive search to improve the identification. We explore the components and their features from a predefined component node list and the features node vector respectively.**

***Keywords*—Design Tools; Embedded Systems; Feature Extraction; Software Reusability; Variability Management.**

I. INTRODUCTION

This paper is an extension of the conference paper [1], and aims at providing a greater insight into the algorithm for managing software variants of embedded systems. It presents a semi-automatic approach to identify reusable parts.

Embedded systems are microcontroller-based systems built into technical equipment mainly designed for a dedicated purpose. Communication with the outside world occurs via sensors and actuators [2]. Although this definition implies that embedded systems are used as isolated units, from 2006 it is observed that there is a trend to construct distributed pervasive systems by connecting several embedded devices as indicated by Tanenbaum and van Steen [3].

The current development trend in automotive software is to map software components on networked Electronic Control Units (ECU), which includes the shift from an ECU based approach to a function based approach. Also according to data presented by Ebert and Jones [4] up to 70 electronic units are used in a car containing embedded software, which is responsible for the value creation of the car and consists of more than 100 million lines of object code.

Ebert and Jones presents data about embedded software, stating that the volume of embedded software is increasing between 10 and 20 percent per year as a consequence of the increasing automation of devices and their application in real world scenarios.

An industrially accepted approach in the automotive applications is Model Based Software Engineering (MBSE). Model-Driven Engineering (MDE) is the use of models as the main artifacts during the software development and the maintenance process. Model Driven Software Development (MDSD) is typically realized in a distributed system environment.

Most MDSD approaches follow the Model Driven Architecture (MDA) concept. In this concept, we start with the specification of a platform independent model, this is then transformed to a platform specific model by applying several generators. The layered Meta Object Facility (MOF) approach is used for creating the models. This approach is also used as the basis for the Unified Modeling Language UML.

While MDSD facilitates models for the abstract specification of system architectures, their platform specific artifacts are often realized by applying Component Based Software Engineering (CBSE) techniques. Models become artifacts to be maintained along with the code, by using model transformations and code generation.

MDE is related with the Object Management Group (OMG) initiatives, Model-Driven Architecture (MDA®) and Model-Driven Development (MDD®), which argue that the use of models as the main artifact on software development will bring benefits on software reuse, documentation, maintenance, and development time.

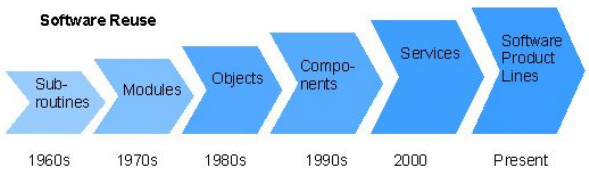


Fig. 1. Software reuse history.

Reuse of automotive embedded software is difficult, as it is typically developed for a small ECU that lacks both processing speed and memory of a general purpose machine. Moreover, the complexity of the embedded software is dramatically increasing. In view of this complexity, achieving the required reliability and performance is one of the most challenging problems [5].

Figure 1 shows a short history of the usage of reuse in software development. In the 1960s, reuse of software started with subroutines, followed by modules in the 1970s and objects in the 1980s. Around 1990 components appeared, followed by services at about 2000. Currently, Software Product Lines (SPL) are state of the art in the reuse of software. Today, many different approaches exist to the implementation of Software Product Lines, but the complexity still remains at unmanageable proportions.

Complexity management has become a vital factor in an organization. To save costs a company needs to minimize internal complexities arising from numerous factors like large products portfolios, regulations that necessitate component variations in different regions, requiring components from external sources like Original Equipment Manufacturers (OEMs), requirements for meeting certifications, and Virtual Organizations (VOs) [6]. It is also necessary to satisfy the range of customer requirements which determines external complexity. The dynamics involved is due to three major factors:

- *Globalization:* For companies to be present in all major markets and to be competitive the requirements of customers with different cultural, technological, economic, and legal backgrounds needs to be incorporated in products.
- *Evolving Technology:* With a need to reduce the time-to-market, technology is evolving at an extremely fast pace. The trend to launch new products quickly in the market is increasing, which necessitate for enhanced technology as well as convergence of technologies [7][8].
- *Increasing market influence:* The customers influence to determine a product's features and price is inducing the manufacturers to provide more and more product variants.

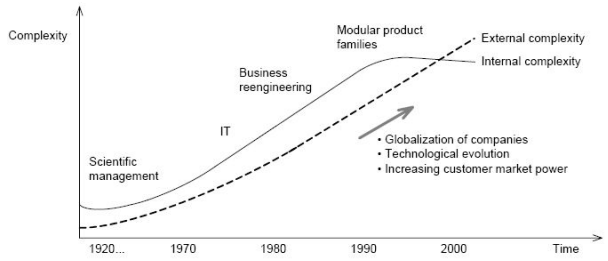


Fig. 2. Evolution of complexities [9].

Figure 2 depicts numerous methods and tools introduced in the past to limit the impact of rising external complexity onto internal complexity in manufacturing, information management, and processes.

With globalization, evolving technology, and increasing market influence the complexity revolving in reuse of embedded software is becoming extremely unmanageable mainly due to large number of variants. The proposed strategy is to introduce a variability identification layer that intends to facilitate software reuse. We start by analyzing the model structure. Based on this we form a concept to extract an element list to facilitate the identification of variability. The implementation section describes algorithm fragments of the different functional blocks. The evaluation of the proposed strategy is based on a technically advanced adaptation of a formal mathematical model [10], which is beyond the scope of this paper.

The rest of this paper is organized as follows. In Section II, a brief summary of related work by other authors is given, while in Section III, enumerates the objectives of the approach. Section IV presents the concept and approach, algorithm fragments, and evaluation. Section V discusses the contributions, while Section VI draws conclusions and future work.

II. Related work

Usually, the product governs processes, manufacturing and information. The product is an interface between external and internal complexity. Designing modular products and applying module variants results in product families [11]. The interfaces between these modules need to be clearly specified. To address modular product families from a holistic perspective it needs to be managed in development and realization across the entire life cycle.

With so many modular product families now being in place, the following observations however indicate the following:

- *Increasing number of variants:* The number of variants continues to rise and is unmanageable in most companies. Due to cannibalization effects, new variants often do not substantially increase sales but only lead to redistribution from standard to special products. As a result, increased costs are not passed on to the selling price and the profit margin decreases [12].
- *Insufficient decision basis:* Many of the complexity effects cannot be captured using traditional accounting techniques, e.g., overhead calculation. The widely-used methods and lack of technical knowledge on the consequences can be misleading when it comes to decisions in variant management.
- *Unsuitable Methodologies:* Modular product families are treated with the same mechanisms as single products, which is unsuitable. Modular product families require a different approach to variant management than single products as interfaces and interactions among modules is crucial.

Planning a standardized architecture within an organization may address a part of these problems and facilitate reuse. With constantly changing requirements within the set of products, the variability needs to evolve. Many embedded systems are implemented with a set of alternative function variants to adapt to the changing requirements. Major challenges are in identifying the commonality of functionality, where the designs involve variability (ability to customize). In addition to variants, versions/releases of functional blocks also play an important role for the effective management over the entire product cycle.

Figure 3 depicts a scenario where well established software components tested for performance, safety, and reliability

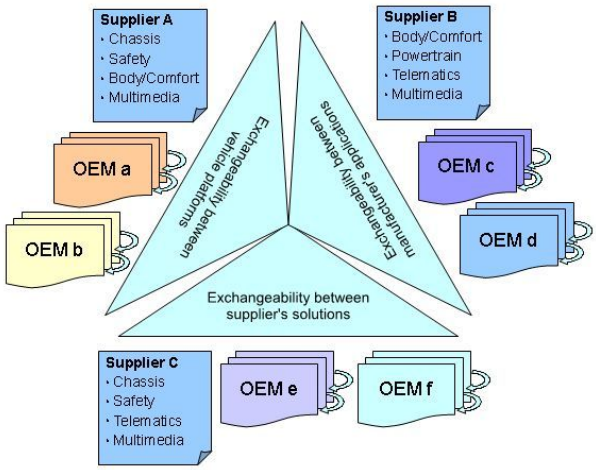


Fig. 3. External components are a hindrance to variability management.

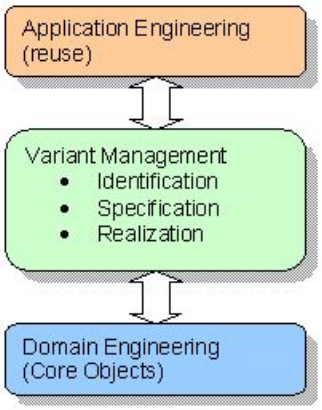


Fig. 4. Variability management in product lines.

procured from external sources and Original Equipment Manufacturers (OEMs) are causes for a hindrance in managing variability.

For achieving large-scale software reuse, reliability, performance, and rapid development of new products, a software product-line (SPL) is an effective strategy. A SPL is a family of products sharing the same assets allowing the derivation of distinct products within the same application domain.

An SPL is a set of software-intensive systems that share a common set of features for satisfying a particular market segment's needs. SPL can reduce development costs, shorten time-to-market, and improve product quality by reusing core assets for project-specific customizations [13][14].

The SPL approach promotes the generation of specific products from a set of core assets, domains in which products have well defined commonalities and variation points [15].

Enabling variability in software consists in delaying decisions at different software abstraction levels, ranging from requirements to runtime. The object-oriented approach to implement variability is based on the development of a frameworks of reusable software components described by a set of classes and by way instances of those classes collaborate.

One of the fundamental activity in Software Product Line Engineering (SPLE) is Variability Management (VM). Throughout the SPL life cycle, VM explicitly represents variations of software artifacts, managing dependencies among variants and supporting their instantiations [13].

To enable reuse on a large scale, SPLE identifies and manages commonalities and variations across a set of system artifacts such as requirements, architectures, code components, and test cases. As seen in the Product Line Hall of Fame [16], many companies have adopted this development approach.

As depicted in Figure 4, SPLE can be categorized into domain engineering and application engineering [17][18]. Domain engineering involves design, analysis and implementation of core objects, whereas application engineering is reusing these objects for product development.

Activities on the variant management process involves variability identification, variability specification and variability realization [19].

- The Variability Identification Process will incorporate feature extraction and feature modeling.
- The Variability Specification Process is to derive a pattern.
- The Variability Realization Process is a mechanism to allow variability.

To enable identification of variability for software components in a distributed system within the automotive domain [20][21], we enlist the specifications below:

- *Specification of components by compatibility*
  The product is tested using software functions of a certain variant and version. These products may exhibit compatibility issues between functional blocks, whilst using later version of the function may fail to perform as expected.
- *Extract, identify, and specify features*
  To enable parallel development, it is necessary to be able to extract features, and to identify and specify the functional blocks in the repository based on architecture and functionality.
- *Usability and prevention of inconsistencies*
  A process that tracks usability and prevents inconsistencies due to deprecate variants and versions in the repository is required.
- *Testing mechanism for validations*
  A testing mechanism for validations in order to maintain high quality for components and its variants has to be established.
- *Mechanism for simplified assistance*
  The developer has to be assisted by a process to intelligently determine whether a functional block or its variant should exist in the data backbone to avoid redesign of existing functions, thereby improving productivity.

Although variability management is recognized as an important aspect for the success of SPLs, there are not many solutions available [22]. However, there are currently no commonly accepted approaches that deal with variability holistically at architectural level [23].

Based on the challenges discussed and the concluded related work presented, the following objectives can be derived.

III. OBJECTIVES OF THIS APPROACH

- *Objective 1: Support heterogeneous models containing hierarchically embedded software components containing the complete specification of specific functionality to foster reuse.*
  Breaking down the models into several components and logical clustering of components of the modeled software is not targeted. In contrast, the proposed methodology enables the identification of commonalities of components in heterogeneous models. For deployment and reuse purposes several partial models are treated as one artifact. Furthermore, the architecture should support reuse of these artifacts for the development of new functionalities.
  The challenge of the realized system of artifact heterogeneity should be based on existing component technologies that provides mature techniques, that are a consequence of the application independent and generic definition of the system specific components and ensures the portability of the proposed system on other platforms.

- *Objective 2: Enable dynamic configuration.*
  Each subsystem is modeled and simulated using a domain-specific simulation tool, while the co-simulation platform handles the coupling between these subsystems that enables holistic simulation of a system.
  The challenge for identifying variability of software components validating to numerous schemata of respective simulation tools and dynamically loading of plug-ins for specific set of components adhering to respective schemata at execution time in model interpretation architecture.

- *Objective 3: Enable shared usage of resources.*
  A scenario depicting the concept of virtual organization should have a clear method to tackle resource access, validation and verification of specific models.

IV. ACTIVITIES FOR VARIABILITY IDENTIFICATION

Models confirming to numerous tools like ESCAPE®, EAST-ADL®, UML® tools, SysML® specifications and AUTOSAR® were considered. Although this concept is not limited to automotive domain alone.

A. Project analysis

An analysis of the models exhibits a common architecture. Figure 5 depicts the textual representation that underlies several graphical models. The textual representation usually is given in XML, which strictly validates to a schema. A heterogeneous modeling environment may consist of numerous design tools, each with its own unique schema, to offer integrity and avoid inconsistencies. Developed projects have to be strictly validated to the schemas of these tools.

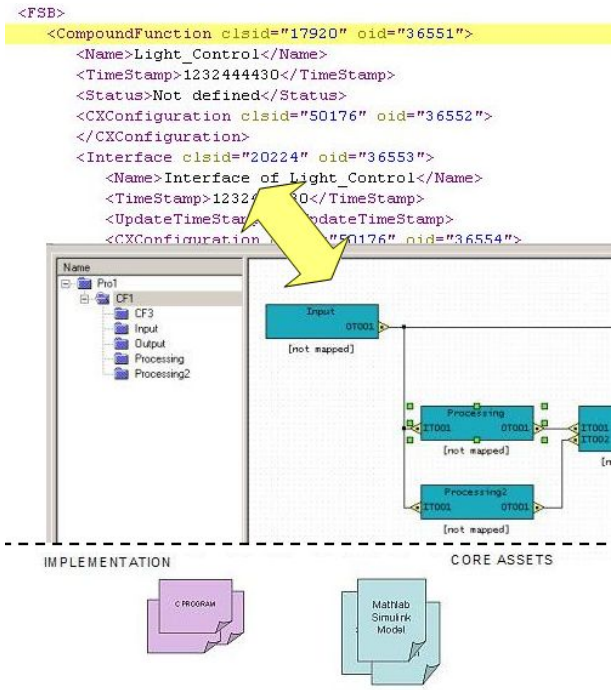


Fig. 5. Mapping textual and graphical representations.

Fig. 6. XML Nodes that are not significant for variability.

A closer examination of the nodes in the textual representation of models depicted in Figure 6 reveals some interesting information. The nodes outlined in rectangles provide important information regarding the identity, specification, physical attributes, etc. of a component, but are insignificant from the perspective of variant.

B. Concept and approach

The basic concept to identify variability is depicted in Figure 7.

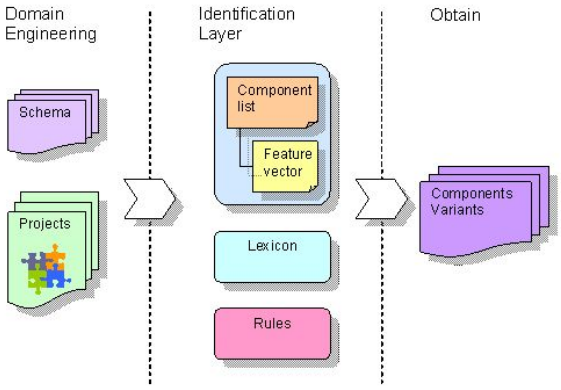


Fig. 7. Basic Concept.

The left side is a set of projects that have software components hierarchically embedded. These projects validate to the corresponding schemas. The middle layer is an identification layer with three functional blocks. A set of component lists is derived from the node list in the schema. Similarly, a feature vector is derived from the schema that corresponds to components. The second block is a customized parser that generates a relevant lexicon from the set of software components within a project. The third block is a set of rules (viz., mandatory, optional, exclude) to govern the identification of variability.

The basic concept can be extended to obtain a working model for the identification of variants. The work flow is depicted in Figure 8. The top layer here represents the domain or core assets. The middle layer is a semi-automatic identification layer for variants. A component list and a feature vector is derived manually from the schema of the project; a collection of elements that represent components and their descriptive features that significantly contribute to the identification of the component's variant.

The workflow can be further extended to adapt a heterogeneous environment, which consist of projects developed using several modeling and simulation tools. The identification layer is separated into two parts. Numerous component lists and feature vectors can be derived for each distinct schema as depicted in Figure 9, whereas a common lexicon and common rules govern the identification process.

C. Implementation

In this section, several key aspects of the implementation are discussed. The focus is to describe the architecture of the identification layer, which forms the intermediate layer for adapting the core assets from domain engineering into application engineering.

The related approaches put on view a need for a generic methodology in identification of software components developed using several design tools.

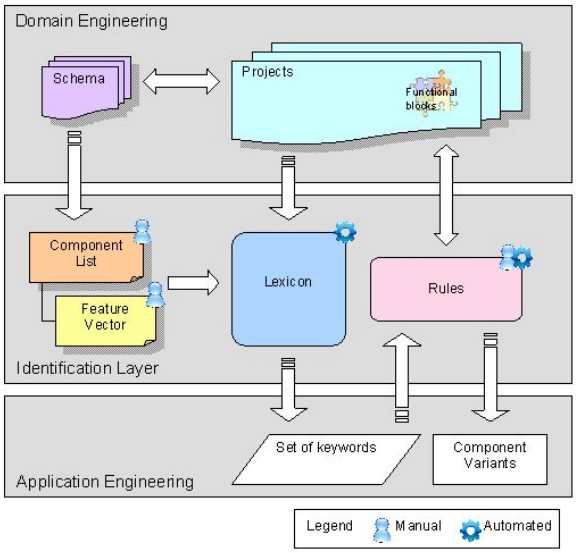


Fig. 8. Work flow of the identification process.

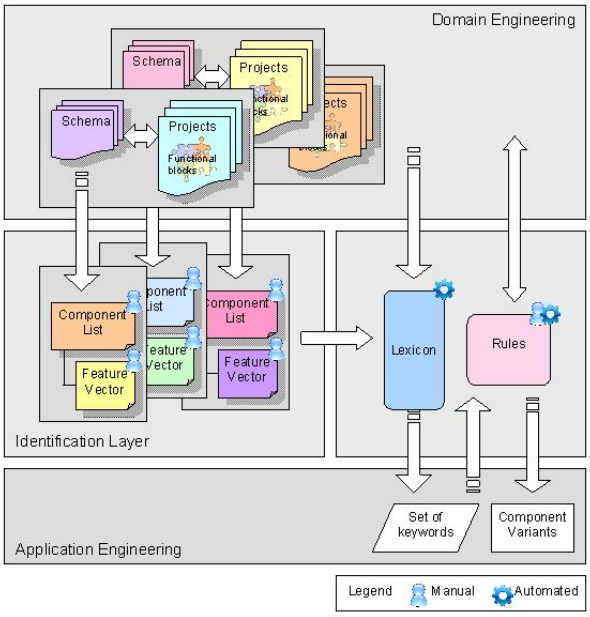


Fig. 9. Work flow of the identification process for heterogeneous systems.

1) Component list and feature vectors: As the project structure for each tool is well defined and strictly validated with corresponding schemata, these schemata can be used as basis for deriving the list that can identify components.

The results are summarized in Table I.

a) Component list

An example of the list of elements that characterize components derived manually from the schemata for design tool ESCAPE [24] is

```
"CompoundFunction HWFunction
SWFunction Parameter"
+ " StructureElement SWBubbleType
ParameterType ParameterTypeTerminal
IntDataType FloatDataType
TimeDataType AliasDataType
VariantDataType HWFunctionType
TypeInterface FunctionTypeTerminal
HWTypeTerminal"
+ " StructureElement DeviceMapping
DeviceType BusCAN BusSegment
MappedFunction"
```

The list is a delimited string with whitespace or any other delimiter.

A tool supporting multi-functional structures like ESCAPE has three views: Functional structure builder (FSB), Function type builder (FTB) and Hardware Structure Builder (HSB). Each view can have an independent list

- Component list for FSB
  FSB facilitates to build the structure of the model.

  ```
  "CompoundFunction HWFunction
  SWFunction Parameter"
  ```

- Component list for FTB
  FTB provides defining hardware and software types.

  ```
  "StructureElement SWBubbleType
  ParameterType ParameterTypeTerminal
  IntDataType FloatDataType
  TimeDataType AliasDataType
  VariantDataType HWFunctionType
  TypeInterface FunctionTypeTerminal
  HWTypeTerminal"
  ```

- Component list for HSB
  HSB that allows networking ECUs and mapping the software functions.

  ```
  "StructureElement DeviceMapping
  DeviceType BusCAN BusSegment
  MappedFunction"
  ```

Similarly, in a heterogeneous modeling environment each modeling tool will have its own schemata, and a corresponding list may be derived for each tool.

b) Feature vector

Similarly, the elements that characterize features of the software components are also derived manually from the schemata, which forms the feature vector and are enlisted below

```
"Name LongName DEScription
ConnectionSegment SourceTerminal
SinkTerminal Interface
CompoundTerminal HWTerminal
SWTerminal Input DataType"
```

c) Algorithm to identify components within projects

Using the string described in Section IV-C1a that characterize the software components nodes list within a project, the following algorithm can be devised.

componentListString ← string described in Section IV-C1a;

Nodes ← doc.GetElementsByTagName("*");

for each **Node** in the **Nodes** (Length(L_n) $\geq$ 1), do

  if **Node.name** in **componentListString**, then

    componentList ← **Node.name**;

The order for matching the software components is $O(N)$.

The prototype dataset used for evaluation of this algorithm contained a total of 32909 nodes, of which only 1583 matches were the software components.

Similarly, using the string described in Section IV-C1b that characterize the features within software components, the following algorithm can be devised.

featureVectorString ← string described in Section IV-C1b;

Nodes ← **componentList**;

for each **Node** in the **Nodes** (Length(L_c) $\geq$ 1), do

  if **Node.name** in **featureVectorString**, then

    featureList ← **Node.name**;

The order for determining the corresponding features within the software components is $O(N)$.

From the prototype dataset a total of 13353 nodes matches to the feature vector were found.

The results are summarized in Table II.

2) Lexicon: A simple customized parser has been devised which automatically extracts words from the text within the software components and features that match the component list and feature vector respectively.

lexiconList ← NULL;

Nodes ← **componentList** ∪ **featureList**;

for each **Node** in the **Nodes** (Length(L_{cf}) $\geq$ 1), do

  wordList ← split(**Node.innerText**, delimiter) ;

  for each **word** in the **wordList** (Length(L_w) $\geq$ 1),do

    if **word** not in **lexiconList**, then

      lexiconList ← **word**;

      lexiconList.frequency ← 1;

else

lexiconList.frequency ←

lexiconList.frequency+1;

End For;

A more sophisticated parser that discards non-words will further improve the Lexicon.

The Lexicon assists the user to choose from a set of relevant words along with their frequencies thereby improving the user experience.

3) Rules: In every case, a full match of software components to specification sets is not desired, but in many instances specification sets contain elements that are mandatory (contains all), optional (one or more) and exclude (omit). Providing rules to execute these features enhances the performance in the identification process.

ruleContainAll ←

Specification subset with Contain-all elements;

ruleOptional ←

Specification subset with Optional elements;

ruleExclude ←

Specification subset with Exclude elements;

Nodes ← **componentList** ∪ **featureList**;

for each **Node** in the **Nodes** (Length(L_{cf}) ≥ 1), do

wordList ← split(**Node.innerText**, delimiter) ;

for each **word** in the **wordList** (Length(L_w) ≥ 1),do

if **word** not in **ruleExclude**, then

if **word** in **ruleContainAll**, then

variantList ← **word**;

elseif **word** in **ruleOptional**, then

variantList ← **word**;

End For;

Using the rules enables to narrow down to a more realistic list of variants that matches the specification set.

4) Transforming naming convention: Moreover, the naming convention within an organization also lead to ambiguity in the identification of components when the number is large.

a) Naming convention

A list for a naming convention for a distributed business process is illustrated below

```
"WorkSpace DOMain GRouP PRoJect
FunctionBlock PartNo VARiant"
```

b) Algorithm to transform names

The string described above characterizes the naming convention within an organization, the scattered software components can be organized by splitting the names along a delimiter and transforming them into a hierarchical structure, then the following algorithm can be devised:

nameConv ← List described in Section IV-C4a;

SWcompNameList ← doc.readCompName("*");

for each **SWcompNameConv** in

SWcompNameList (Length(L_{nc}) ≥ 1), do

SWcompNameSplit ←

split(**SWcompNameConv.name**, delimiter);

for each **SWcompNamePart** in

SWcompNameSplit (Length(L_{sn}) ≥ 1), do

if not exist **SWcompNamePart0**, then

RootElementNode ← **SWcompNamePart**;

else

ParentElementNode ← **RootElementNode**;

for each **SWcompNamePart** in

ParentElementNode.ChildNodes

(Length(L_{cn}) ≥ 1),do

if not exist **SWcompNamePart**, then

ParentElementNode.addChildNode ←

SWcompNamePart;

else

ParentElementNode ←

ParentElementNode.ChildNodes;

End For;

This algorithm can be further extended to assist the user to identify, search, and construct the names and display them as a hierarchy. A procedure to navigate and simplify the construction of such names will enable the user to quickly build long names uniformly over the entire project.

TABLE I. SUMMARY OF ELEMENTS IN SCHEMA OF THE SAMPLE DATA SET

Schema	
Description	**Count**
Total elements collection	171
Components list	23
Features vector	12

TABLE II. SUMMARY OF ELEMENTS IN PROJECT OF THE SAMPLE DATA SET

Project		
Description	**Count**	**Category**
Total elements	32909	all
Components	1583	23
Features within components	13353	12

D. Evaluation

A prototype of the architecture presented here has been implemented. The case studies targeted the design of model-based software components firstly in an industrial use case where the project model was developed using the design tool ESCAPE® [24], and secondly in a case study targeting the execution of specific paradigms based on the naming convention of AUTOSAR® [25].

The number of elements in schema and project of sample data set that was used to evaluate the implementation is summarized in Table I and Table II, respectively. It consists of a total of 32,909 elements. Of these a total of 1583 elements signify components which are categorized into 23 categories that form the Component List is summarized in Table III, where as a total of 13353 elements that signify features which are categorized in 12 categories that form the Feature Vector is summarized in Table IV.

Three different approaches were adopted to evaluate and determine the performance with respect to matches.

TABLE III. COMPONENT LIST DERIVED FROM SCHEMA

Component List	
Description	**Count**
CompoundFunction	58
HWFunction	182
SWFunction	46
Parameter	6
StructureElement	50
SWBubbleType	130
ParameterType	5
ParameterTypeTerminal	8
IntDataType	14
FloatDataType	2
TimeDataType	1
AliasDataType	1
VariantDataType	4
HWFunctionType	46
TypeInterface	181
FunctionTypeTerminal	580
HWTypeTerminal	91
StructureElement	50
DeviceMapping	10
DeviceType	4
BusCAN	3
BusSegment	3
MappedFunction	108
	1583

TABLE IV. FEATURE VECTOR DERIVED FROM SCHEMA

Feature Vector	
Description	**Count**
Name	7500
LongName	0
Description	0
ConnectionSegment	537
SourceTerminal	538
SinkTerminal	538
Interface	292
CompoundTerminal	269
HWTerminal	292
SWTerminal	302
Input	1543
DataType	1542
	13353

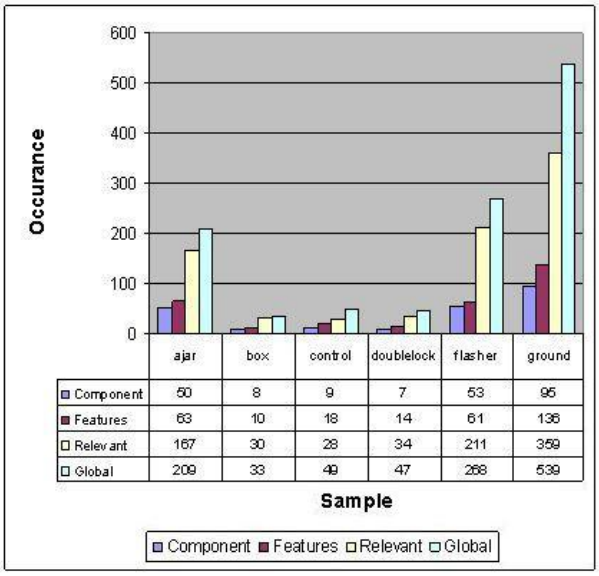


Fig. 10. Occurrence graph for a single element specification set.

- **Evaluation using a single element specification set**
  The first experiment was conducted on a single element specification set. A group of ten sets formed the input to determine the result set in both comprehensive (global) search and selective search as illustrated in Figure 10.
  The notion of comprehensive search is used, when scanning all occurrences of the specification set within projects, irrespective of whether they are components or features of those components. This can return a result set that contains false matches.
  The pattern of the results displayed similar behavior.

  Observations
  - The comprehensive search yields a result set that contains every occurrence of the specification set, even if these nodes do not characterize a component.
  - The nodes representing components yield a result set which is somewhat realistic, though these do not epitomize the complete set desired. This is often observed when the component nodes do not match, but their features collectively match the specification set.

- These nodes along with the feature set yield a more elaborate result set. A match contained by any node in a set of features would result in representing the component to which it belongs.

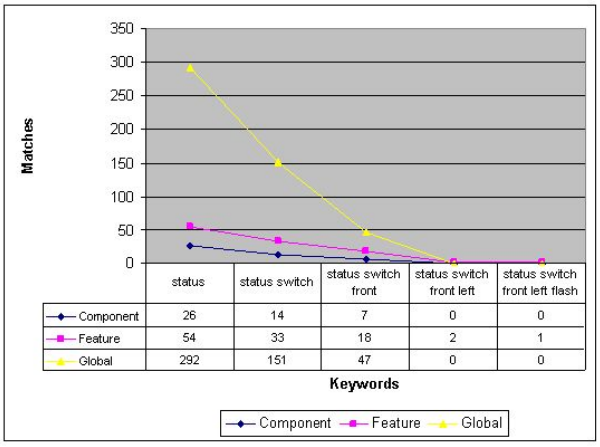


Fig. 11. Occurrence graph for multiple element specification sets.

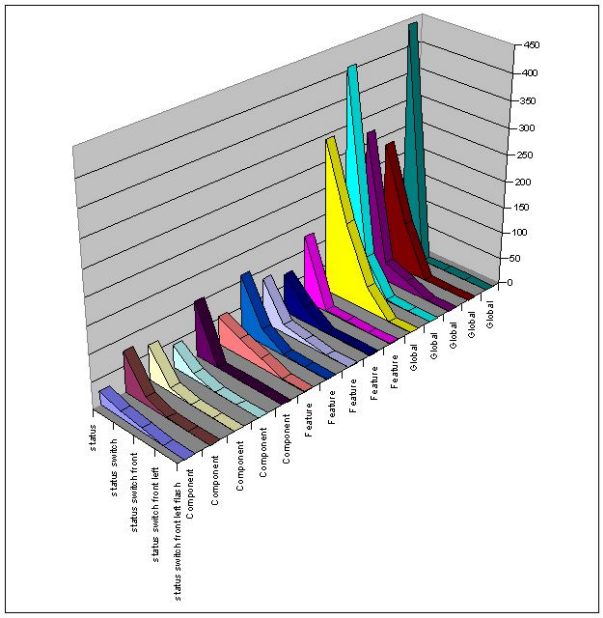


Fig. 12. Occurrence graph for different starting points.

- **Evaluation using multiple element specification set**
The second experiment was conducted using one up to seven element specification sets as a group illustrated in Figure 11.

  Observations
  - The comprehensive search often yielded large result sets, as it searches in individual nodes that are treated as atomic.
  - The exhibited behavior is similar to the varying size of the specification set. As observed in Figure 11, the selective component-feature search result set demonstrates a value when the size of specification set exceeds 3, because in this case the matches take place across the boundary of the feature within the component. On the other hand, the other methods return null result set as the search is only within the boundary of the element.
  - For any given size of specification set, the selective component-feature search returns a much smaller result set and is more precise.
  - Convergence is optimal with a specification set of size 3. If the size of the specification is too large, the result may be null for both methods as shown in Figure 11.

- **Evaluation using different starting points for elements in specification sets**
The third experiment was conducted searching for elements within specification sets using different starting points. Figure 12 depicts the result sets in comprehensive search and selective search.
To determine the effect of different starting points, a multiple-element specification set was used, where the orders of the elements were changed to obtain five sets.
The result set for this exhibits the same pattern as the two experiments above.

V. CONTRIBUTIONS OF THIS APPROACH

- *Contribution 1: Model-based Variability Management for Complex Embedded Networks.*
The concept of Model-based Variability Management is proposed in the paper, which contemplates on the definition of a problem and specification of the cases. Furthermore the concept specified is used for feature extraction to extract spatial, functional, and name for the realization of new functionality. These models has been evaluated for data models in IV-D.

- *Contribution 2: A generic approach to envisage the identification of variability.*
The primary mechanism for determining commonality, allowing dynamic extension in the identification of variability of software components which are embedded in hierarchical model confirming to numerous tools like ESCAPE®, EAST-ADL®, UML® tools, SysML® specifications, and AUTOSAR®. The approach is based on the adaption of a formal mathematical model presented in the publication [10].

- *Contribution 3: An approach to visualize, navigate and simplify the unintelligible naming conventions.*
Mapping highly indecipherable naming conventions and transposing to hierarchical structures using predetermined delimiters, to assist the user to identify, search, and construct these names, comfortably displaying them as hierarchy, as well as having a procedure to navigate and simplify the construction of such names.

VI. Conclusion

Managing variants is of utmost importance in today's large software bases as they reflect legal constraints, marketing decisions, and development cycles. As these software bases often grew from different sources and were developed by different teams using different tools it is in many cases very complicated if not nearly impossible to find artifacts that might be variants, both for historical reasons as for development purposes.

The algorithms presented here reflects both the capability to match keywords and to reflect the structure that characterizes a component. It can be applied directly to application engineering for identification of software component variants. Furthermore, it may also be applied for variability identification of software components in core assets of domain engineering in SPL. Our proposed method is capable of both aspects and therefore helps the developer to find matches even in large and heterogeneous databases.

The developed prototype is itself independent of a specific tool as it works on textual descriptions that typically are available in XML. The future work may comprise to extend the concept to specify and verify reusable components.

References

[1] A. A. Frank and E. Brenner, "Variability identification by selective targeting of significant nodes," ICCGI 2012, The Seventh International Multi-conference on Computing in the Global Information Technology, 2012, pp. 148-153.

[2] C. Ebert and J. Salecker, "Guest editors' introduction: embedded software technologies and trends," Software, IEEE, Vol 26(3), 2009, pp. 14-18.

[3] A. S. Tanenbaum and M. Van Steen, "Distributed Systems: principles and paradigms (2nd Edition)," Prentice Hall, 2006.

[4] C. Ebert and C. Jones, "Embedded software: facts, figures, and future," Computer, IEEE Vol 42(4), 2009, pp. 42-52.

[5] D. Kum, G. Park, S. Lee, and W. Jung, "AUTOSAR migration from existing automotive software," The Proceedings of International Conference on Control, Automation and Systems, 2008, pp. 558–562.

[6] I. Foster and C. Kesselman, "The Grid: blueprint for a new computing infrastructure," Elsevier Series in Grid Computing. Morgan Kaufmann, second edition, 2004, pp. 672.

[7] B. L. Bayus, "Are product life cycles really getting shorter?" Journal of Product Innovation Management, Vol. 11 (4), 1994, pp. 300-308.

[8] S. Poole and M. Simon, "Technological trends, product design and the environment," Design Studies, Vol. 18 (3), 1997, pp. 237-248.

[9] A. Ericsson and G. Erixon, "Controlling design variants modular product platforms," Society of Manufacturing Engineers, Dearborn, MI, 1999.

[10] A. A. Frank and E. Brenner, "A generic approach for the identification of variability," ENASE2012, 7th International Conference on Evaluation of Novel Approaches to Software Engineering, 2012, pp. 167-172.

[11] T. W. Simpson, "Product platform design and customization: status and promise," Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol.18 (1), 2004, pp. 3-20.

[12] P. Child, R. Diederichs, F. H. Sanders, and S. Wisniowsk, "The management of complexity," Sloan Management Review, Vol. 33 (1), 1991, pp. 73-80.

[13] P. Clements and L. Northrop, "Software Product Lines: practices and patterns," Addison-Wesley, 2007.

[14] H. Gomaa and D. L. Webber, "Modeling adaptive and evolvable Software Product Lines using the variation point model," The Proceedings of the 37th Hawaii international Conference on System Sciences, 2004.

[15] E. Oliveira, I. Gimenes, and J. Maldonado, "A variability management process for software product lines," CASCON 2005, The conference of the Centre for Advanced Studies on Collaborative research, 2005, pp. 225 - 241.

[16] Product line hall of fame, "http://splc.net/fame.html," retrieved: 02,2013.

[17] F. Bachmann and P. C. Clements, "Variability in Software Product Lines," Technical Report -CMU/SEI-2005-TR-012, 2005.

[18] J. Bosch, "Design and use of Software Architectures: adopting and evolving a product-line approach," Addison-Wesley, 2000.

[19] L. A. Burgareli, Selma, S. S. Melnikoff, and G. V. Mauricio Ferreira, "A variation mechanism based on Adaptive Object Model for Software Product Line of Brazilian Satellite Launcher," ECBS-EERC 2009, First IEEE Eastern European Conference on the Engineering of Computer Based Systems, 2009, pp. 24-31.

[20] A. A. Frank and E. Brenner, "Model-based variability management for complex embedded networks," ICCGI 2010, The Fifth International Multi-conference on Computing in the Global Information Technology, 2010, pp. 305-309.

[21] A. A. Frank and E. Brenner, "Strategy for modeling variability in configurable software," PDES 2010, The 10th IFAC workshop on Programmable Devices and Embedded Systems, 2010, pp. 88-91.

[22] P. Heymans and J. Trigaux, "Software product line: state of the art," Technical report for PLENTY project, Institut d'Informatique FUNDP, Namur, 2003.

[23] M. Galster and P. Avgeriou, "Handling variability in software architecture: problem and implications," WICSA 2011, Ninth Working IEEE/IFIP Conference on Software Architecture, 2011, pp. 171-180.

[24] ESCAPE, "http://www.gigatronik2.de/index.php?seite=escape_produktinfos_de&navigation=3019&root=192&kanal=html," retrieved: 11,2012

[25] AUTOSAR, "http://www.autosar.org/download/conferencedocs/03_AUTOSAR_Tutorial.pdf," retrieved: 02,2013

Adaptive Fractal-like Network Structure for Efficient Search of Targets at Unknown Positions and for Cooperative Routing

Yukio Hayashi and Takayuki Komaki,
Japan Advanced Institute of Science and Technology
Email: yhayashi@jaist.ac.jp, takayuki.k@mocha.ocn.ne.jp

***Abstract*—From viewpoints of complex network science and biological foraging for communication networks, we propose a system model of scalable self-organized geographical networks, in which the proper positions of nodes and the network topology are simultaneously determined according to population. The fractal-like network structure is constructed by iterative divisions of rectangles for load balancing across nodes, in order to adapt to territory changes. In numerical simulations, we show that, for searching targets concentrated around high population areas, the naturally embedded fractal-like structure by population has higher efficiency than the conventionally optimal strategy on a square lattice. The adaptation of network structure to the spatial distribution of realistic communication requests gives such a high performance.**

Keywords -self-organized design, spatially inhomogeneous communication requests, random walk, Lévy flight, cooperative message ferries.

I. INTRODUCTION

The size and complexity of communication and transportation networks are growing year by year for the increase of users, communication requests, mobility, and technological innovations in real-world sensing or handling of rich contents. For the design and control of growing networks, scalability, adaptivity, and self-organization will be more required in the near future. Here, we consider the adaptive network structure [1] to be suitable for searching a target on the network embedded in a space with population density, because searching (or routing) is one of the important basic tasks to establish a connected path on a network.

Many network infrastructures: power grids, airline networks, and the Internet, are embedded in a metric space, and long-range links are relatively restricted [2], [3] for economical reasons. The spatial distribution of nodes is neither uniformly at random nor on a regular lattice, which is often assumed in the conventional network models. In real data, a population density is mapped to the number of router nodes on Earth [2]. Similar spatially inhomogeneous distributions of nodes are found in air transportation networks [4] and in mobile communication networks [5]. Thus, it is not trivial how to locate nodes within a space using patterns of points. Point processes in spatial statistics [6] provide models for irregular patterns of points in urban planing, astronomy, forestry, or ecology, such as spatial distributions of rainfall, germinations, plants, and animals. The processes assume homogeneous Poisson and Gibbs distributions to generate a pattern of random packing or independent clustering, and to estimate parameters of competitive potential functions in a territory model for a given statistical data, respectively. However, rather than random pattern and statistical estimation, we focus on a self-organized network infrastructure by taking into account realistic spatial distributions of nodes and communication requests. In particular, we aim at developing adaptive and scalable networks by adding the links between proximity nodes according to the increase of communication requests. Because a spatial distribution of communication requests affects the proper positions of nodes, which control both the load of requests assigned to each node (e.g., assigned at the nearest access point of node as a base-station from a user) and the communication efficiency depending on the selection of routing paths.

Thus, we propose a scalable self-organized geographical network, considering an interrelation of routing algorithms in computer science, biological foraging, and complex network science. Complex network science is a groundbreaking science that has emerged from a physical society about ten years ago for understanding the common network structures in social, technological, and biological systems [7], [8], [9], and the fundamental generation mechanisms. We show that the naturally embedded fractal-like structure in the proposed network [10] is suitable for searching inhomogeneously distributed targets more efficiently than the square lattice tracked by the Lévy flights, which is known as an optimal biological search for homogeneously distributed targets [11]. Moreover, we investigate the performance for a cooperative routing method in the fractal-like network, as an extension of the conference paper [1]. We emphasize that a spatially inhomogeneous distribution of communication requests is important [10] for a realistic situation according to population density, and that an adaptive network structure to population is self-organized.

The organization of this paper is as follows. For starting a discussion, in Section II, we mention the miminimum

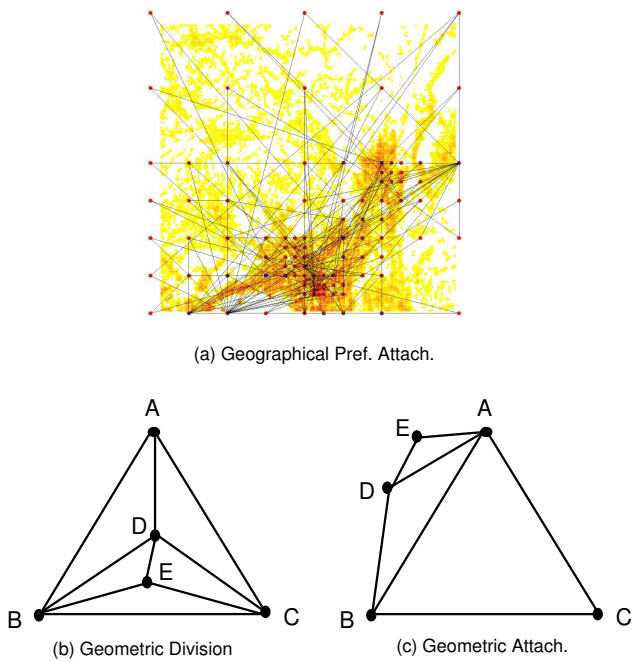
necessity of related works. Because we treat several different topics and the interrelations of network self-organization, routing methods by a determinant or stochastic walker, biological search strategy, and cooperative routing by multiple walkers. To avoid confusion, each detail is explained in the most related section. In Section III, we briefly review the conventional geographical network models, and propose a new model based on iterative divisions of rectangles. In Section IV, we show the basic search performance of a random walk in the fractal-like network structure for targets at unknown positions. In Section V, we investigate the routing property by using cooperative agents on the network structure, comparing that with the Lévy flights on a square lattice. In Section VI, we summarize these results and mention further studies.

II. Related Work For Routing and Searching

For routing in ad hoc networks, global information, e.g., a routing table in the Internet, cannot be applied, because many nodes and connections between them are likely to change over time. Although there are many protocols [12], [13] for energy saving, mobile networks, GIS-based location awareness, QoS, and wireless sensor devices, we restrict strongly related ones to our discussion.

In early works on computer science, some decentralized routing methods were developed to reduce energy consumption in sensor or mobile networks. However they lead to the failure of guaranteed delivery [14]; in the flooding algorithm, multiple redundant copies of a message are sent and cause congestion, while greedy and compass routings may occasionally fall into infinite loops or into a dead end. We do not need to be particular about these simple and energy saving methods in the current and future technologies. At least, it is better to guarantee the delivery. In complex network science, other no-failure efficient decentralized routing methods have been also proposed. The stochastic methods by using local information of the node degrees and other measures are called preferential [15] and congestion-aware [16], [17] routings as extensions of a uniformly random walk.

Decentralized routing has a potential performance to search a target whose position is unknown in advance. Since this situation looks like foraging, the biological strategy may be useful for the efficient search. We are interested in a relation between the search and the routing on a spatially inhomogeneous network structure according to population. Many experimental observations for biological foraging found the evidence in favor of anomalous diffusion in the movement of insects, fishes, birds, mammals and human being [11]. As a consistent result, it has been theoretically analyzed for a continuous space model that an inverse square distribution of flight lengths is an optimal strategy to search sparsely and randomly located targets on a homogeneous space [18]. The discrete space models on a regular lattice [19] and the defective one [20] are also discussed. Such behavior is called *Lévy flight* characterized by a distribution function $P(l_{ij}) \sim l_{ij}^{-\mu}$ with $1 < \mu \leq 3$, where l_{ij} is a flight length between nodes i and j in the stochastic movement for any direction. The values of $\mu \geq 3$ lead to Brownian motions, while $\mu \to 1$ to ballistic motions. The optimal case is $\mu \approx 2$ for maximizing the efficiency of search. Here, we assume that the mobility of a node is ignored due to a sufficiently slow speed in comparison with the communication process. In current or future technologies, wide-area wireless connections by directional beams will be possible, the modeling of unit disk graph with a constant transmission range is not necessary. Thus, as a system model, we consider efficient search and routing on an adaptive fractal-like network structure to spatially inhomogeneous communication requests.



(a) Geographical Pref. Attach.

(b) Geometric Division

(c) Geometric Attach.

Figure 1. Typical construction methods of geographical networks. At each time step, a new node is added at a random position. Then, (a) the new node i is linked to an existing node j chosen with a probability $\Pi_j \propto d_{ij}^{-\alpha} pop_j^{\beta} k_j^{\gamma}$, where α, β, and γ are real parameters, d_{ij} denotes the distance between nodes i and j, pop_j denotes the population in the node j's territory, and k_j is its degree. The territory is a merged area of mesh blocks, which nearest access node is j. The gradation (from while, yellow, orange, to red) of background mesh blocks on a $L \times L$ square lattice is proportional to the value of population given from a census data. The case of $\alpha = \beta = 0$ and $\gamma > 0$ is the degree based model, and the case of $\alpha, \gamma > 0$ and $\beta = 0$ is a combination of degree and distance based model. On the other hand, the new node (D or E) is set at random (b) in a chosen triangle or (c) outside of a chosen edge, and linked to (b) the three nodes of the randomly chosen triangle or (c) both ends of the randomly chosen edge. The initial configuration consists of triangles.

III. Geographical Networks

We introduce geographical network models proposed in complex network science, which aims to elucidate a fun-

damental mechanism for generating an efficient network structure in a distributed manner.

A. Conventional Models

Geographical constructions of complex networks have been proposed so far. Figure 1 (a) and (b)(c) shows the typical methods. It is well known that the preferential attachment [7] is fundamental to construct a scale-free (SF) network that follows a power law degree distribution found in many real systems [21], [22]. As a generation mechanism of geographical SF networks, a spatially preferential attachment is applied in some extensions [23], [24], [25], [26], [27] from the topological degree based model [28] to a combination of degree and distance based model (Figure 1(a)). However, this construction tends to have many long links, which are wasteful. The original degree based preferential attachment is known as "rich gets richer" rule that means a higher degree node tends to get more links. It is a surprising thing that inhomogeneous complex network structures emerge from such a simple rule. On the other hand, geometric construction methods have also been proposed (Figure 1(b)(c)). They have both small-world [29] and SF structures generated by a recursive growing rule for the division of a chosen triangle [30], [31], [32], [33] or for the attachment aiming at a chosen edge [34], [35], [36] in random or hierarchical selections. Here, small-world means that the average path length counted by hops between two nodes is very small as $O(\log N)$ even in a large size N defined by the total number of nodes. These geometric models are proper for the analysis of degree distribution due to the regularly recursive generation process. Although the position of a newly added node is basically free as far as the geometric operations are possible, it has no relation to population. Considering the effects of population in a geographical network is necessary to self-organize a spatial distribution of nodes that is suitable for socioeconomic communication and transportation requests. Moreover, in these geometric methods, narrow triangles with long links tend to be constructed, and adding only one node per step may lead to exclude other topologies from the SF structure. Unfortunately, SF networks are extremely vulnerable against the intentional hub attacks [37]. We should develop other models of self-organized networks distinct from the conventional models; e.g., a better network without long links can be constructed by subdivisions of equilateral triangles, which is a well balanced (neither fat nor thin) shape for any directions as shown in Figure 2(a). In the network without long links, a node with a small degree does not become hub, therefore the attack vulnerability of connectivity disappears.

B. Generalized MSQ Network

Thus, we have considered the multi-scale quartered (MSQ) network model [38], [39]. It is based on a stochastic construction by a self-similar tiling of primitive shape.

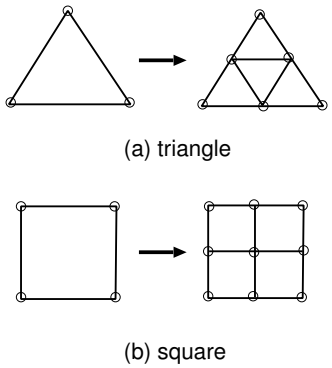


Figure 2. Basic process of the division.

Figure 2(a)(b) shows the basic process of division in the tiling of equilateral triangle or square. At each time step, a face is chosen proportionally to the population in the space. Then, the chosen face is divided into four smaller equilateral triangles or squares. This process is repeated. The MSQ networks without hub nodes have several advantages such as the strong robustness of connectivity (due to the small degrees) against node removals by random failures and intentional attacks, the bounded short path as $t = 2$-spanner [40], and the efficient face routing by using only local information. The t-spanner means that the length of shortest distance path (defined by the sum of link lengths on the path) between nodes u and v is bounded at most t times the Euclidean distance d_{uv} of the corresponding straight line between them. In the face routing, the shortest distance path can be found on the edges of faces that intersect the straight line, since the MSQ network is planner, which is also suitable for avoiding the interference among wireless beams. Furthermore, the MSQ networks are more efficient (economic) with shorter link lengths and more suitable (tolerant) with lower load for avoiding traffic congestion [39] than the state-of-the-art geometric growing networks [30], [31], [32], [33], [34], [35], [36] and the spatially preferential attachment models [23], [24], [25], [26], [27] with various topologies ranging from river to SF geographical networks. However, in the MSQ networks, the position of a new node is restricted on the half-point of an edge of the chosen face, and the link length is proportional to $(\frac{1}{2})^H$ where H is the depth number of iterative divisions. Thus, from square to rectangle, we generalize the division procedure as follows. Figure 3 illustrates it.

- Step 0: Set an initial square, in which the candidates of division axes are the segments of an $L \times L$ lattice (Figure 3(a)).
- Step 1: At each time step, a face is chosen with a probability proportional to the population counted in the face covered by mesh blocks of a census data (Figure 3(b)).
- Step 2: Four smaller rectangles are created from the

division of the chosen rectangle by horizontal and vertical axes. For the division, two axes are chosen by that their cross point is the nearest to the population barycenter of the face (Figure 3(c)).

Step 3: Return to Step 1, while the network size N does not exceed a given size.

Note that the maximum size N_{max} depends on the value of L; the iteration of division is finitely stopped, since the extreme rectangle can not be divided any longer when one of the edge lengths of rectangle is the initial lattice's unit length. We use the population data on a map in $80km^2$ of 160×160 mesh blocks ($L = 160$) provided by the Japan Statistical Association. Of course, other data is possible.

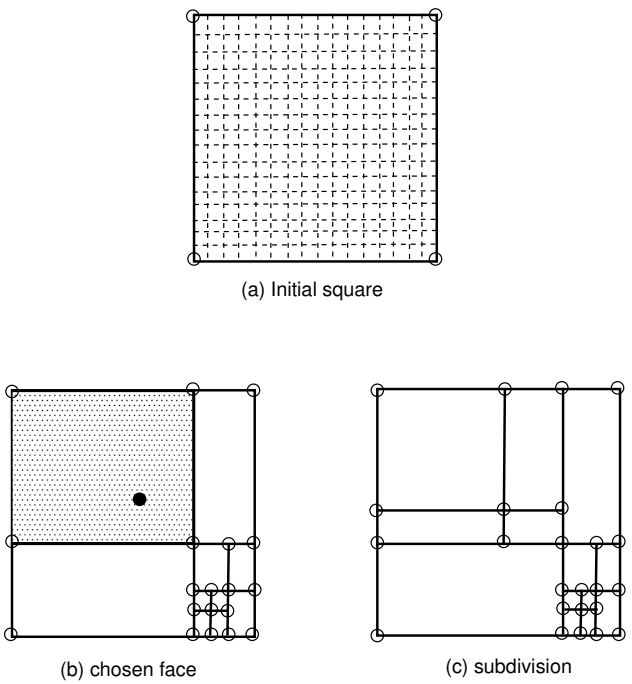


Figure 3. Division procedure in generating a generalized MSQ network. (a) Initial configuration: the outer square of 4 nodes and 4 links. For division, dashed-lines represent the segments defined by the edges of $L \times L$ mesh blocks. To each mesh block whose right-bottom corner is at $1 \leq x \leq L$ and $1 \leq y \leq L$, a value of population $pop_{x,y}$ is assigned by a census data. (b) As an example, a (shaded) face f is chosen with a probability $\propto \sum_{x,y \in A_f} pop_{x,y}$ at the 4th time step. Where A_f denotes the set of x-y coordinate values included in the face f. (c) Then, the horizontal and vertical axes, which cross point is nearest to the population varycenter (black filled circle) of face f, are selected, and divide the chosen face into four smaller ones.

It is worth noting that the positions of nodes and the network topology are simultaneously determined by the divisions of faces within the fractal-like structure. There exists a mixture of sparse and dense parts of nodes with small and large faces. Moreover, while the network is growing, the divisions of faces perform a load balancing of nodes in their adaptively changed territories for the increase of population. Such a network is constructed according to a spatially inhomogeneous distribution of population, which is proportional to communication requests in a realistic space. In the following, we show the naturally embedded fractal-like structure is suitable for searching targets. Moreover, we apply the good property to a routing task in Section V.

IV. Basic Search Performance

As a preliminary, we consider the preferential routing [15] which is also called α-random walk [41]; The forwarding node j is chosen proportionally to K_j^α by a walker in the connected one hop neighbors $\mathcal{N}_i$ of its resident node i of a walker (packet), where K_j denotes the degree of node j and α is a real parameter. We assume that the start position of walker is set to the nearest node to the population barycenter of the initial square. Figure 4 shows the length distribution of visited links. The dashed lines in log-log plot suggest a power law, for which the exponents estimated as the slopes by a mean-square-error method are 2.336, 2.315, and 2.296 for $\alpha = 1, 0, -1$, respectively. These values are close to the optimal exponent $\mu \approx 2$ [18], [19] in the Lévy flight on a square lattice. The exponents for the α-random walks slightly increase as the network size N becomes larger. Here, the case of $\alpha = 0$ shows the length distribution of existing links in a network. Since the stationary probability of incoming at node j is $P_j^\infty \propto K_j^{1+\alpha}$ [42], especially at $\alpha = 0$, each of the connected links to j is chosen at random by the probability $1/K_j$ for the leaving from j, therefore a walker visits each link at the same number. Figure 5 shows that the frequency of visited links by the α-random walks at $\alpha = \pm 1$ is different even for the degrees 3 and 4 in a generalized MSQ network. On the thick lines, a walker tends to visit high population (diagonal) areas colored by orange and red in the case of $\alpha = 1$, while it tends to visit low population peripheral (corner) areas in the case of $\alpha = -1$. Thus, the case of $\alpha = 1$ is expected to selectively cover high population areas, which has a lot of communication requests in cities. Note that the absolute value of α should be not too large, since a walker is trapped a long time between high/low degree nodes as the phenomena does not contribute to the search of targets.

We investigate the search efficiency for the α-random walk on a generalized MSQ network, and compare the efficiency with that for Lévy flights on a $L \times L$ square lattice with periodic boundary conditions [19]. As shown in Figure 6, a walker constantly looks for targets (destination nodes of packets) scanning on a link between two nodes in the generalized MSQ network. If a target exists in the vision area of r_v for the up/down/left/right directions from the center position, a walker gets it and return to the position on the link for continuing the search on the same direction. When more than one target exist in the area, a walker gets all of them successively in each direction, and return to the position. Only at a node of rectangle, the search direction is changeable along one of the connected links.

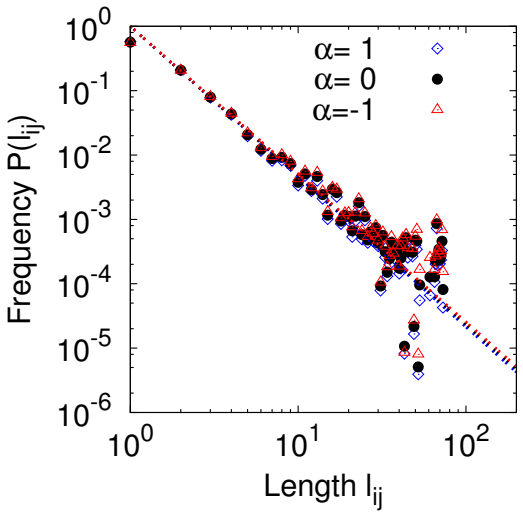


Figure 4. Length distribution of visited links on generalized MSQ networks by an α-random walker in 10^6 time steps. The marks of blue diamond, black circle, and red triangle correspond to the cases of $\alpha = 1, 0, -1$, respectively. These results are obtained by the average of 100 networks for $N = 2000$.

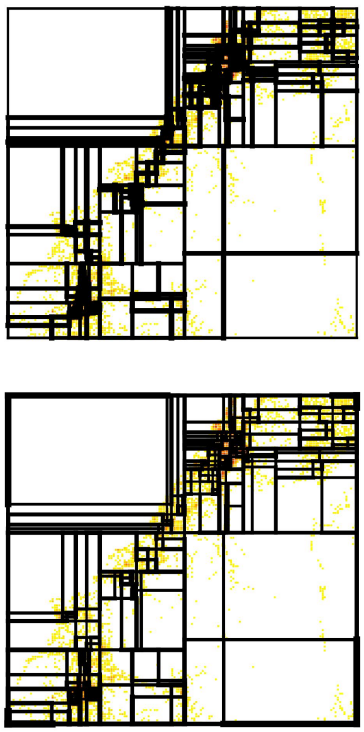


Figure 5. Visualization examples of the visited links by α-random walks at $\alpha = 1$ (Top) and $\alpha = -1$ (Bottom) on a generalized MSQ network for $N = 500$. The thickness of link indicates the frequency of visiting in 10^6 time steps. From light to dark: white, yellow, and orange to red, the color gradation on a mesh block is proportionally assigned to the population data. Many nodes represented as cross points of links concentrate on high population (dark: orange and red) areas on the diagonal direction. In the upper left and lower right of square, corner triangle areas lighted by almost white are the sea of Japan and the Hakusan mountain range.

Thus, the search is restricted on the edges of rectangle in the generalized MSQ network. While the search direction of a Lévy flight on the square lattice [19] is selectable from four directions of horizontal and vertical at all times after getting a target in the scanning with the vision area of r_v, moreover, the length of scan follows $P(l_{ij}) \sim l_{ij}^{-\mu}$, $l_{ij} > r_v$. We set a target at the position chosen proportionally to the population around a cross point in $(L+1)^2$, for which the population is defined by the average of four values in its contact mesh regions. In particular, we discuss the destructive case [19]: once a target is detected by a walker, then it is removed and a new target is created at a different position chosen with the above probability. Similar results to below in this section are obtained for the non-destructive case [10].

The search efficiency [18], [19], [20] is defined by

$$\eta \stackrel{\text{def}}{=} \frac{1}{M} \sum_{m=1}^{M} \frac{N_s}{L_m}, \tag{1}$$

$$\lambda \stackrel{\text{def}}{=} \frac{(L+1)^2}{N_t 2 r_v}, \tag{2}$$

where L_m denotes the traversed distance counted by the lattice's unit length until detecting $N_s = 50$ targets from the total N_t targets in the mth run. We consider a variety of $N_t = 60, 100, 200, 300, 400$, and 500 for investigating the dependency of the search efficiency on the number N_t of targets. The quantity λ represents the mean interval between two targets for the scaling of efficiency by target density. We set $M = 10^3$ and $r_v = 1$ for the convenience of simulation. Intuitively, the sparse and dense structures according to the network size N have the advantage and disadvantage in order to raise the search efficiency in the generalized MSQ network. Although the scanned areas are limited by some large rectangle holes as N is small, a walker preferably visits the high population areas that include many targets. While the scanned areas are densely covered as N is large, the search direction is constrained on long links of a collapse rectangle, therefore it is rather hard for a walker to escape from a local area in which targets are a few.

We compare the search efficiency of α-random walks on the generalized MSQ networks with that of the Lévy flights on the square lattice. Figure 7 shows typical trajectories until detecting $N_s = 50$ targets. On the generalized MSQ network and the square lattice, a walker tends to cover a local area with high population and a wider area, respectively. Without wandering in peripheral wasteful areas, the generalized MSQ network has a more efficient structure than the square lattice for detecting many targets concentrated on the diagonal areas. Here the exponent $\mu = 1.8$ of Lévy flight corresponds to the slope of $P(l_{ij})$ in the generalized MSQ network at the

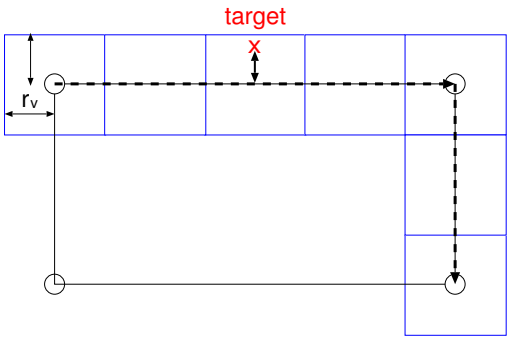
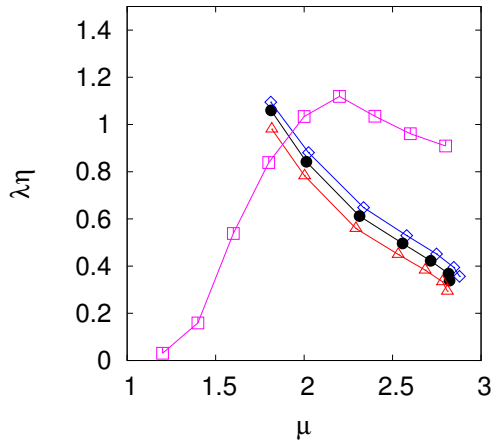


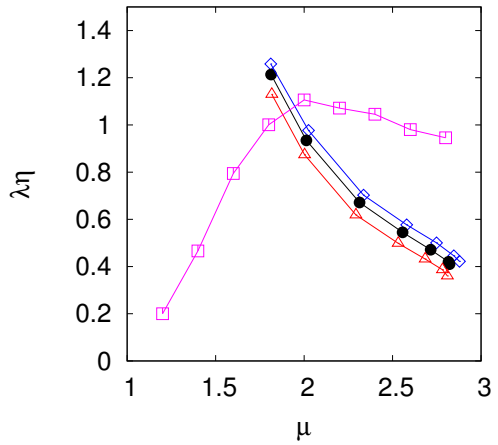
Figure 6. Searching in a generalized MSQ network. Each blue square represents a vision area, and is scanned (from left to right, from top to bottom in this example) by the walker on an edge between two nodes (denoted by circles) of a rectangle. For a target in the area, the walker moves to get it and returns on the link.

Figure 7. Trajectories of a random walk (Top) at $\alpha = 0$ on a generalized MSQ network for $N = 500$ and of a Lévy flight (Bottom) for $\mu = 1.8$ on the square lattice with periodic boundary conditions until detecting $N_s = 50$ targets in $N_t = 200$. Black circle, red circles, and gray rectangle marks denote the start point at the population barycenter, the existing targets, and the removed targets after the detections, respectively. Note that a walker can travel back and forth on a link in the connected path.

optimal size $N = 500$ for the search efficiency. As shown in Figure 8(a)(b), the generalized MSQ networks of $N = 500$ (the diamond, circle, and triangle marks are sticking out at the left) have higher efficiency than the square lattice (the rectangle mark). For the cases with many nodes of $N \geq 1000$, the efficiency decreases more rapidly than that of the Lévy flight, however this phenomenon means that an extremely large network size is wasteful and unnecessary to get a high search performance in generalized MSQ networks. When the number N_t of targets increases in cases from Figure 8(a) to (b), the curves are shifted up, especially for the generalized MSQ networks. The peak value for $N_t = 200$ is larger than the optimal case of the Lévy flight at $\mu = 2.0$. Therefore denser targets to that extent around $N_t = 200$ is suitable, although a case of larger $N_t > 300$ brings down the search efficiency even for inhomogeneously distributed targets.



(a) $N_t = 100$



(b) $N_t = 200$

Figure 8. The scaled efficiency $\lambda\eta$ vs. the exponent μ. The marks of blue diamond, black circle, and red triangle correspond to the cases of $\alpha = 1, 0, -1$, respectively, in which the increasing values of μ are estimated for generalized MSQ networks at $N = 500, 1000, 2000, 3000, 4000, 5000,$ and 5649: N_{max} from left to right. The magenta rectangle corresponds to the case of Lévy flights on the square lattice. These results are obtained by the average of 100 networks.

In more details, Figure 9 shows the effect of the number N_t of targets on the search efficiency $\lambda\eta$. The efficiency firstly increases, then reaches at a peak, and finally decreases

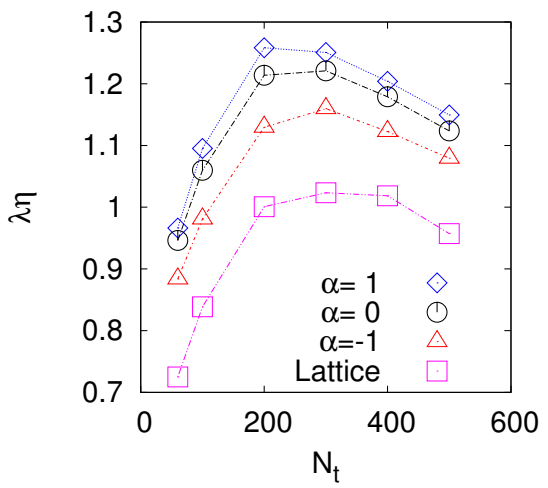


Figure 9. The number N_t of targets vs. the scaled efficiency $\lambda\eta$ of α-random walks on the generalized MSQ networks for $N = 500$ and of the corresponding Lévy flights for $\mu = 1.8$ (see Figure 8) on the square lattice. The maximum (optimal) efficiency appears in $N_t = 200 \sim 300$. These results are obtained by the average of 100 networks.

for setting more targets. This up-down phenomenon is caused by a trade-off between L_m and N_t in Eqs. (1) and (2). Please note that the case of size $N < 500$ is omitted for the generalized MSQ networks. Because sometimes the process for detecting targets until N_s is not completed, moreover, the variety of link lengths is too little to estimate the exponent as a slope of $P(l_{ij})$ in the log-log scale. In other words, the estimation is inaccurate because of the short linear part.

V. Routing by Message Ferries

When a communication network is often disconnected but resilient due to node mobility, limited radio power, node or link failure, etc., it is known as a Delay/Disruption Tolerant Network (DTN) where a mobile device or software agent temporary stores and carries local information for forwarding messages until an end-to-end route is re-established or re-generated. It is used in disasters, battlefields, and vehicular communications. There are many protocols in the concept of DTN routings [43], [44]. A message ferrying scheme is one of the DTN routing strategies, in which a device or agent called ferry stores, carries, and forwards messages in partitioned ad hoc networks. It is classified into a single ferry [45] or multiple ferries, stationary or mobile node, node-initiated or ferry-initiated moving to communicate [46], single-route or multi-routes, and node relaying or ferry relaying [47] according to the protocol components: a movement of ferry, interactions between node and ferry or between ferries, how much and which type of local information is stored in a ferry or at a node, and so on.

We focus on cooperative multiple ferries as software agents, because the ferries interact asynchronously through a mediator node sharing partial information or exchanging it for their routing tasks. This method can avoid the problem of very rare encounter between ferries because of their random walks. During a routing, we assume a network is fixed to distinguish the effects by the network structure and by the disconnections on the performance, since we consider the network structure is a primal factor to control a ferry's movement. In addition, we distinguish between how to determine a route and how to deliver a message (data packets), thus we do not care whether or not a ferry should move with its message. The appropriative delivery depends on the ability of devices, the amount of message, and communication environment. However, our approach will be applicable to an opportunity networking with node mobility. Note that a Lévy walk of a single ferry is applied for searching on a continuous space with the Euclidean distance in order to maximize the opportunity of encounter with the destinations of mobile nodes [48], though the problem setting is different from ours. In the following, we consider only the case of $\alpha = 0$ in the α-random walks, to simplify the discussion, since the difference for the cases of $\alpha = \pm 1$ is small.

A. Multiple Message Ferries Routing

We explain the outline of routing algorithm. Note that a ferry has no vision area in the routing problem unlike the searching problem discussed in the previous section. In addition, a walker moves to get many targets in the searching problem, while multiple ferries move to cooperatively find paths between source and destination nodes in the routing problem.

Initially, there is no label at each node. Communication requests by different pairs of source s and destination d are labeled at a node. In other words, as a mediator between ferries, each node handles more than one requests that are carried from ferries. Similarly, a ferry can carry more than one requests. Without global information, a node visited by a ferry A at time t memorizes the node $n_A(t-1)$ for each ferry in the connected neighbors, where $n_A(t-1)$ denotes the visited node by A at $t-1$. While a ferry memorizes a set of passed links as the history in a limited size of buffer. Thus, using only partial information, a path between any two nodes is found as follows.

1) **Comm. Request:** A pair of s and d nodes is chosen proportionally to the population counted in the territory of each node (defined by the nearest access point) for a census data referred in Subsection III. The generation rate R is the number of generated s-d pairs in the network per time step.
   After the generation at node s, the communication request $REQ(s, d)$ is put on hold until a ferry encounters it for the carrying, because each node is fixed.

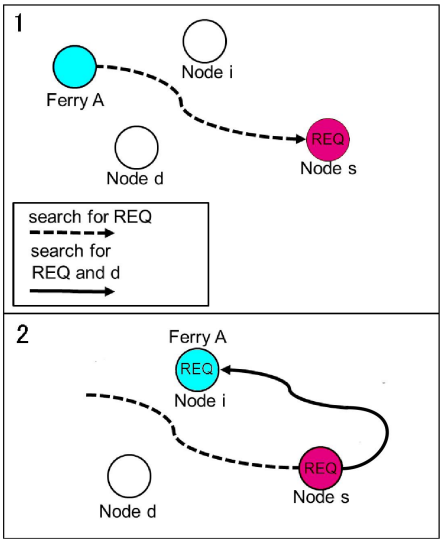


Figure 10. Ferry's modes. (Top) In free-mode, a ferry A encounters the source node s for a request: $REQ(s,d)$ on its walk. (Bottom) In search-mode after the encounter, the ferry A carries the request $REQ(s,d)$ to other nodes, e.g., i, for searching the destination node d.

2) **Ferry's Action:** At each time step, each ferry walks at random on a generalized MSQ network, storing the passed links into its stack-based buffer. Using a set of the stored links, it interacts with the visited node as mentioned in 3).
A ferry has two modes: free and search as shown in Figure 10.
   - Free-Mode: This mode is preliminary to searching, but may contribute to a cooperation between ferries: please see 4). When a ferry of free-mode encounters a source node s or requests handled and labeled at a visiting node, the mode is changed to search-mode.
   - Search-Mode: A ferry of search-mode carries a set $\{REQ(s,d)\}$ of requests to a node i through the random walk.
     Moreover the ferry A asks whether or not the node i knows the $REQ(s,d)$ in its label. If the answer is "NO," the visiting node i is labeled by the $REQ_A(s,d)$. Here, the suffix $_A$ is added to distinguish which ferry carries the $REQ(s,d)$ for backtracking in the path finding. This inquiry is tested for all requests carried by the ferry.
     Of course, if the visiting node is d, then a path between s and d is found: please see 5).

3) **Node Mediator:** A ferry interacts with the visiting node at a time. For each request $REQ(s',d')$ handled at the node, the node asks whether or not the ferry has a link to d' in the buffer stored as the visiting history. If the answer is "YES," go to 4).
   Some requests which the ferry does not have are copied from the node to the ferry for the carrying.

4) **Cooperation:** As shown in Figure 11, when a ferry B, which has a link to d in its buffer, visits a node i labeled by $REQ_A(s,d)$, a path between s and d is found. Because the existence of label $REQ_A(s,d)$ means that the node i is already visited by another ferry A, which comes from s (In more detail, a path from i to s is obtained from the node's information through switching ferries A, C, ..., Z, which carry the $REQ(s,d)$ via intermediate nodes from s to i', from i' to i'', ..., and to i.).

5) **Path Finding:** A path between s and d is found in a subgraph, which consists of the links (including a link to the destination d) in the ferry's buffer and the backward connections of $\{n_A(t-1)\}$ nodes for the ferry A until reaching the source s.
   If a ferry A starting from s visited d, then the ferry's buffer is unnecessary for the path finding. In other words, this case has no cooperation, or is equivalent to the case of zero buffer size.

After the find of a path between s and d, the multiple labels of $REQ_A(s,d), REQ_B(s,d), \ldots$ by ferries $A, B, \ldots$ at a node are deleted in a distributed manner, if the forward connections of $n_A(t+1), n_B(t+1), \ldots$ nodes are memorized for each related ferry $A, B, \ldots$ to the request $REQ(s,d)$.

We investigate the average time step and movement distance until a ferry encounters a source node s from the generation of $REQ(s,d)$. On a movement, the distance means the sum of link lengths or flight lengths counted by the unit of the square lattice. Figure 12(a)(b) shows that both the time and the distance decrease as the number m of ferries increase. The slope near $1/m$ is consistent with the effect of speed-up in parallel walks [49]. Figure 13(a)(b) shows the average time and distance for a path finding. They roughly follow the $1/m$ property, but their slopes depend on the buffer sizes. As the buffer ratio becomes large, both the time and the distance decrease by the effect of cooperation between ferries. These results do not depend on the size N and the packet generation ratio R.

Table I
RELATION OF THE NETWORK SIZE N AND THE BUFFER SIZE FOR THE RATIO BR:2.0. NOTE THAT THE TOTAL NUMBER OF LINKS IS PROPORTIONAL TO N IN A GENERALIZED MSQ NETWORK.

N	buffer size
500	165
1000	333
2000	673
3000	1019

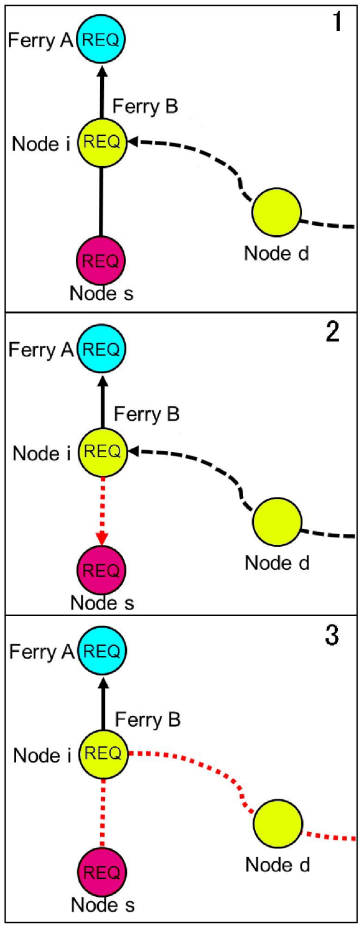
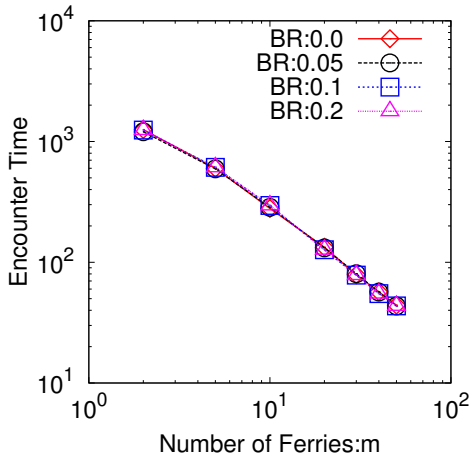
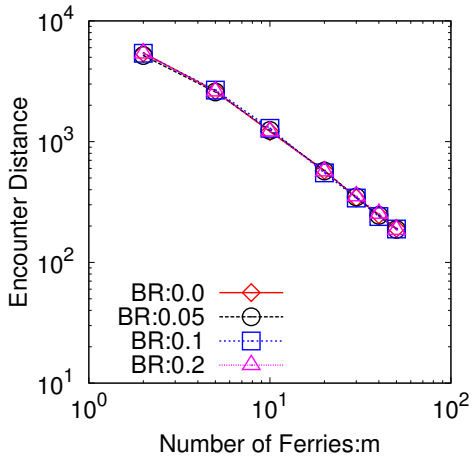


Figure 11. Cooperation of two ferries. When the ferry B visits a node i, which was visited by the ferry A with the request $REQ(s,d)$, a path between s and d is found from the concatenation of A's information and B's information in the situations: (Top) 1: The ferry B notifies the visiting experience at d to the node i, (Middel) 2: It backtracks the links from i to s on the red line by using $\{n_A(t-1)\}$ in the token relay via nodes, (Bottom) 3: From the subgraph that consists of the above gathered links and the B's link set, a path between s and d is calculated, e.g, by a criterion of the shortest distance. In this case, the ferry B does not have the $REQ(s,d)$, however it already visited d and stored the set of links connected to d into its buffer.



(a) Time



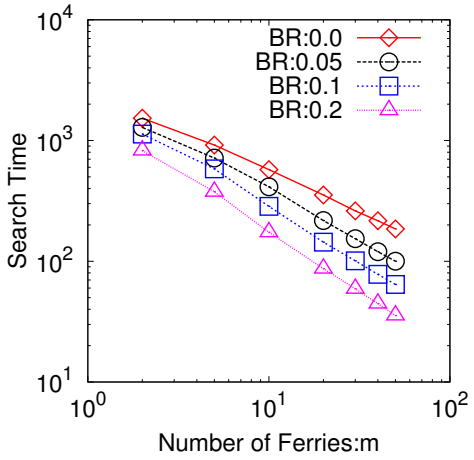
(b) Distance

Figure 12. Average time and distance over 50 realizations until a ferry of free-mode encounters a node s at which a request $REQ(s,d)$ is generated. BR denotes the buffer ratio as the maximum stored size of links in a ferry to the total number of links in the network. Although the encounter time does not depend on the values of BR, because a cooperation of ferries does not start, they are marked to be compared with the subsequent result in Figure 13. Here, the generation rate of requests is $R = 0.01$, and the network size is $N = 1000$.

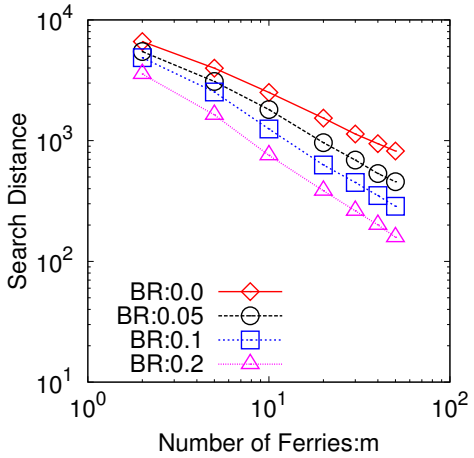
B. Comparison with Lévy Flights

In the cooperative message ferries scheme, we compare the performance of routing by random walks on a generalized MSQ network with that by Lévy flights on a $L \times L$ square lattice with periodic boundary conditions. The lattice is a background virtual space to determine a ferry's movement according to the Lévy flight. Note that only part 2) **Ferry' Action** is replaced in the routing algorithm for the Lévy flight version.

We investigate the average behavior over 50 realizations for each case of $m = 20$ ferries (in order to save computation time) in the combinations of the exponent μ of Lévy flights or the corresponding size of generalized MSQ networks and the buffer ratio (BR). The following simulation conditions are applied in both cases of random walks on a generalized MSQ network and Lévy flights on the lattice. For the generation of a communication request with rate $R = 0.01$, a s or d node is not able to set all lattice points but restricted on the node of a generalized MSQ network, and chosen proportionally to the population in the territory of each node in the network. Because a ferry that walks on the network can not visit any lattice point, in contrast, a ferry that moves according to the Lévy flight can visit any node in the network. The BR is set as 0.2 based on the results in Figures 12 and 13. Note that a larger buffer size tends to be effective in the routing in both short time and distance, however it

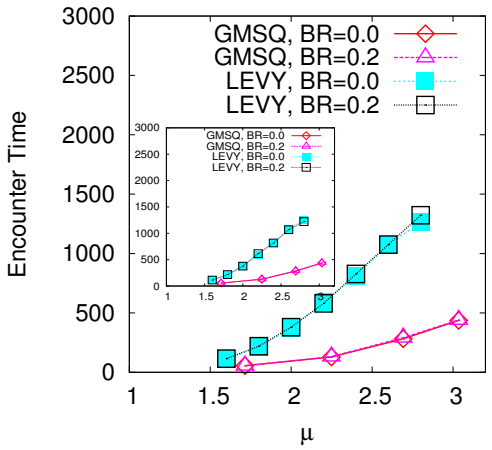


(a) Time

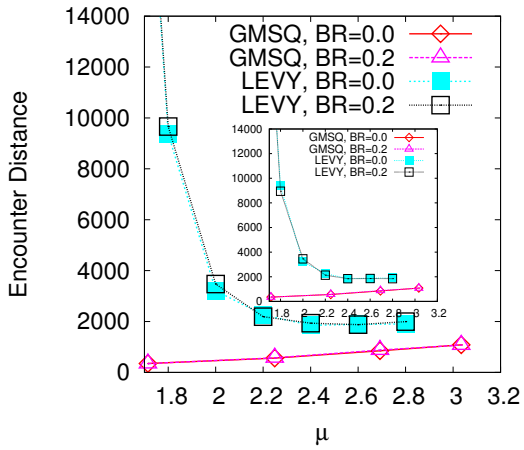


(b) Distance

Figure 13. Average time and distance over 50 realizations until a ferry finds a path between s and d from the encounter with a request $REQ(s,d)$. BR denotes the buffer ratio as the maximum stored size of links in a ferry to the total number of links in the network. Here, the generation rate of requests is $R = 0.01$, and the network size is $N = 1000$.



(a) Time

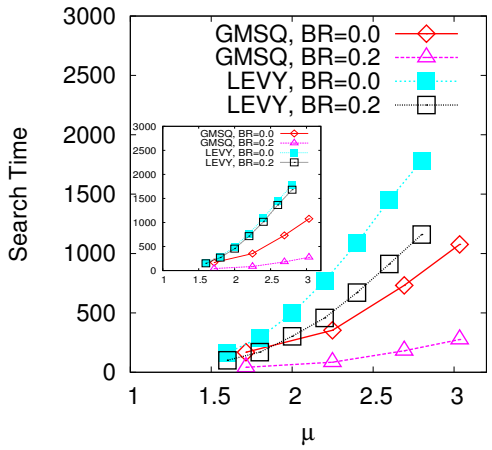


(b) Distance

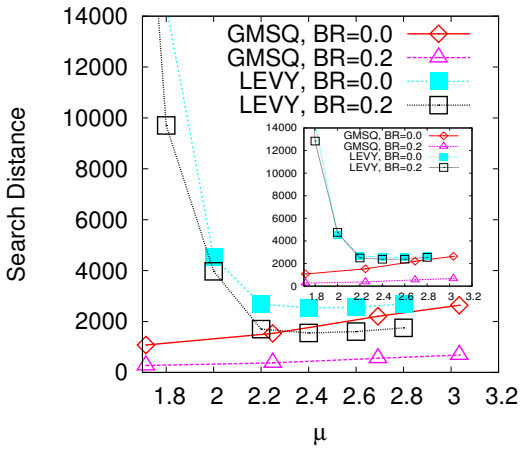
Figure 14. Average encounter time and distance over 50 realizations for GMSQ: random walks on the generalized MSQ networks and LEVY: Lévy flighst on the lattice. Inset shows the case of spatially sparser distributions of communication requests for LEVY. The lines of GMSQ are duplicated.

gives more load for a ferry to store and carry the information of a large number of links. Table I shows the relation of the network size N and the buffer size for the same BR:0.2. In the generalized MSQ networks, the slopes of $P(l_{ij})$ in log-log plot correspond to $\mu = 1.414, 2.248, 2.692$, and 3.034 for $N = 500, 1000, 2000$, and 3000, respectively. These values of μ slightly differ from the example shown in Section IV because of using other area in the census data, however the obtained results are consistent.

Figure 14(a)(b) shows the average time step and distance until a ferry of free mode encounters a node s. The time step increases as the value of μ is larger, since the length of movement in one hop tends to be small on the dense network and on a Brownian motion. By the above effect, the distance also increases as the value of μ is larger. In the Lévy flights, the part of extremely large distance for $\mu < 2$ is due to a ballistic motion, and the shortest distance is obtained around $\mu = 2.4$. Remember that such a U-shape graph of μ vs. distance is obtained as the inverse U-shape graph of μ vs. scaled efficiency in Figure 8. Here, for the Lévy flights of $\mu = 1.6 \sim 2.8$, the positions of s-d nodes are set at the nodes of the generalized MSQ networks of a large size $N = 3000$, while they are set at the nodes of them of a small size $N = 500$ in Inset. We call these positions POS-N3000 and POS-N500, corresponding to spatially dense and sparse distributions of communication requests. Figure 15(a)(b) shows the average time step and distance until a ferry finds a path between s and d from the encounter with $REQ(s,d)$. Similar behavior to Figure 14(a)(b) is obtained, although there are dependences on the BR; The time step and distance become shorter, as the BR is larger. In both



(a) Time

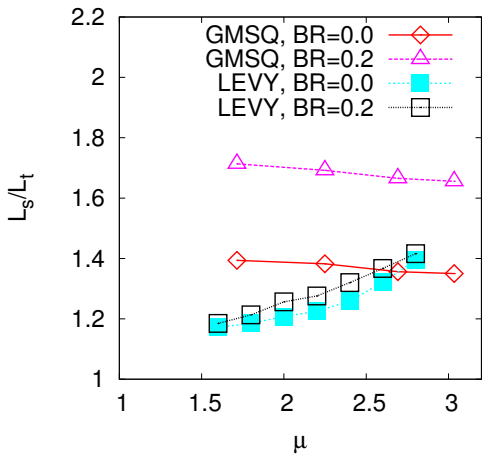


(b) Distance

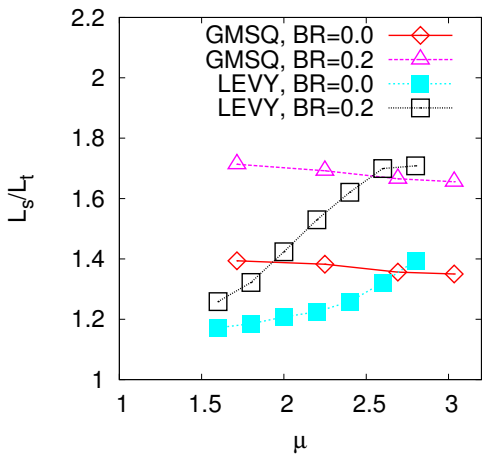
Figure 15. Average search time and distance over 50 realizations for GMSQ: random walks on the generalized MSQ networks and LEVY: Lévy flights on the lattice. Inset shows the case of spatially sparser distributions of communication requests for LEVY. The lines of GMSQ are duplicated.

Figures 14 and 15, we emphasize that the cases of GMSQ show shorter time and distance than the cases of LEVY. The most efficient size is $N = 500$ corresponding to the smallest μ plotted at the left end in the figures. We note that, depending on the situation of movements of ferries and locations of destination nodes, the links memorized in the buffer of a ferry maybe not work well, since hundreds of times are spent for the encounter and the search, especially for a large μ.

The ratio of the path lengths obtained by the routing and by the shortest distance on the generalized MSQ network is between 1.1 and 1.8 as shown in Figure 16(a)(b). For the reason that the case for BR:0.2 is worse than the case for BR:0 without cooperations between ferries, the higher ratio of L_s/L_t is caused from less information used for finding a path as shown in Figure 17(a)(b). In addition, Figure 17(a)(b) shows that the average number of used links in the subgraph for calculating a path is between 20% and 40% of the total number of links. There is a trade-off: In Figure 17(a) for BR:0.2, GMSQ is slightly better than LEVY for using less information, however the path length is longer as shown in Figure 16(a). Figure 18(a)(b) shows that the average number of requests carried by a ferry is less than 6 in the cases of GMSQ, and smaller than that in the cases of LEVY. Thus, in the cases of GMSQ for BR:0.2, a routing path is obtained at most 1.4 times longer than the shortest distance by using only partial information about 20% of the total number of links, in average.



(a) POS-N500

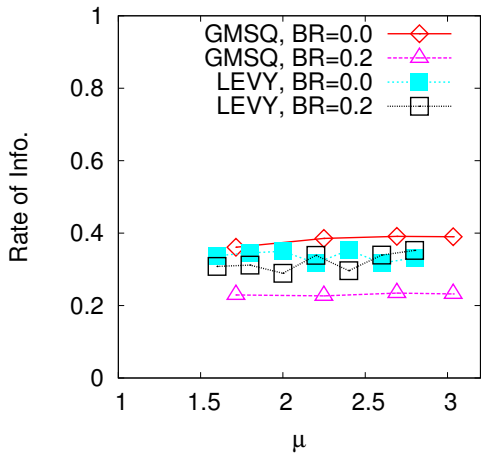


(b) POS-N3000

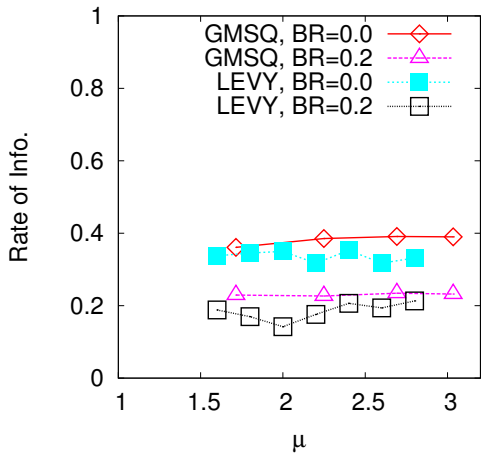
Figure 16. Ratio of the path length L_s obtained by the routing and the shortest distance path L_t. For LEVY, the spatially (a) sparse and (b) dense distributions of communication requests are generated on the nodes of the generalized MSQ networks for $N = 500$ and $N = 3000$, respectively. Here, L_s is between 80 and 120, therefore L_t is in the same order.

VI. Conclusion

We have considered a scalable self-organized geographical network by iterative divisions of rectangles for load

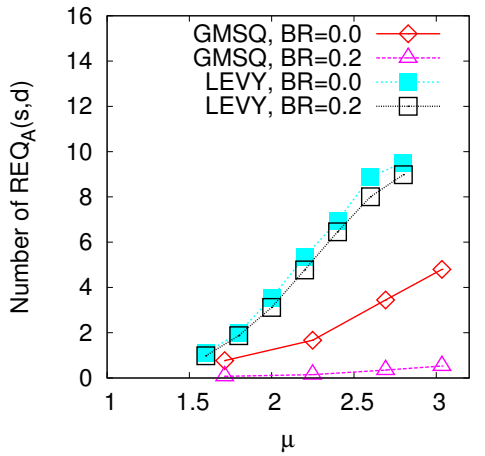


(a) POS-N500

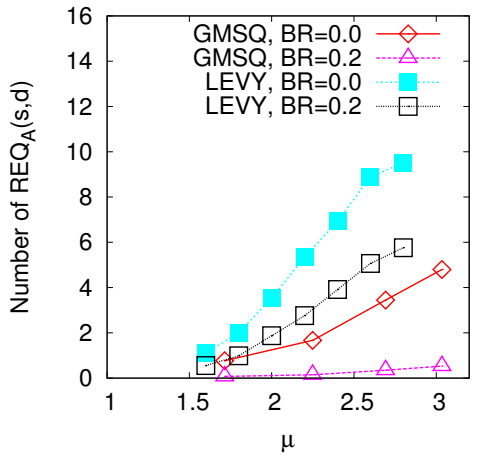


(b) POS-N3000

Figure 17. Rate of partial information: (the average number of used links in the subgraph for calculating a path) divided by (the total number of links). For LEVY, the communication requests are generated on the spatially (a) sparse and (b) dense networks.



(a) POS-N500



(b) POS-N3000

Figure 18. Average number of communication requests carried by a ferry. For LEVY, the requests are generated on the spatially (a) sparse and (b) dense networks.

balancing of nodes in the adaptive change of their territories according to the increase of communication requests. In particular, the spatially inhomogeneous distributions of population and the corresponding positions of nodes are important. For the proposed networks [1], [10], we have investigated the search efficiency in the destructive case [19] with new creations of target after the detections, and shown that the α-random walks [17], [41] on the networks within a small size have higher search efficiency than the Lévy flights known as the optimal strategy [18], [19] for homogeneously distributed targets on the square lattice with periodic boundary conditions. One reason for the better performance is the anisotropic covering of high population areas.

Furthermore, we apply the good property to the decentralized routing by cooperative message ferries. The key point is also the adaptation of network structure to the spatial distributions of source and destination which are inhomogeneous according to a population data. As the merit, by using only a simple protocol based on random walks, the naturally embedded fractal-like sparse structure contributes to the search of targets and to the find of a path efficiently in such a realistic situation of spatially distributed communication request.

However, we must take care of the size. Our method on the generalized MSQ networks within a small size shows better performance in both time and distance than the Lévy flight version using only partial information of links. For the scale up issues, since the performance goes down as the size is larger, we should make various ideas to keep the appropriate size ($N \approx 500$), e.g., by the enhancement of processing power at a node, instead of distribution of load in a large size. The performance for both encounter and search

times will be improved by considering further methods of how to cooperate with ferries and nodes.

On the other hand, the message ferrying scheme is generally applicable to a temporal network, in which the positions of nodes and connections between them are changed in a short time. It is interesting to study such cases for the generalized MSQ networks with temporal disconnections. Since a human mobility pattern resembles to the Lévy flights, a good performance of the proposed cooperative routing will be expected for a temporal network that consists of multi-hop mobile communication equipments, although how to treat the temporal disconnections caused from node mobility will be one of the important issues. Instead of the message ferry scheme, it is worth to investigate the performance of other DTN routing methods on the generalized MSQ network. For more rigorous discussions about the performance, statistical tests [50] may be useful to clarify the applicability of the proposed method. The limitation for the applicability will also depend on future technologies of wireless devices.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for their valuable comments. This research is supported in part by Grant-in-Aide for Scientific Research in Japan, No.21500072 & No.25330100.

REFERENCES

[1] Y. Hayashi, "Adaptive Fractal-like Network Structure for Efficient Search of Targets at Unknown Positions," *Proc. of the 4th International Conference on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE)*, pp.63-68, ISBN:978-1-61208-219-6, 2012.

[2] S. H. Yook, H. Jeong, and A.-L. Barabási, "Modeling the internet's large-scale topology," *PNAS*, Vol.99, No.21, pp.13382–13386, 2002.

[3] M. T. Gastner and M. E. J. Newman, "The spatial structure of networks," *Eur. Phys. J. B*, Vol.49, No.2, pp.247–252, 2006.

[4] R. Guimerà, S. Mossa, A. Turtschi, and L. Amaral, "The worldwide air transportation network: Anormalous centrality, community structure, and cities' global roles," *PNAS*, Vol.102, No.22, pp.7794–7799, 2005.

[5] R. Lambiotte, V. Blondel, C. de Kerchove, E. Huens, C. Prieur, Z. Smoreda, and P. Dooren, "Geographical dispersal of mobile communication networks," *Physica A*, Vol.387, pp.5317–5325, 2008.

[6] D. Stoyan, W. Kendall, and J. Mecke, *Stochastic Geometry and its Applications (2nd Eds.)*, John Wiley & Sons, 1995.

[7] A.-L. Barabási, *Linked: The New Science of Networks*. Perseus, Cambridge, MA, 2002.

[8] M. Buchanan, *Nexus: Small Worlds and the Groundbreaking Theory of Networks*. W.W.Norton, New York, 2002.

[9] M.E.J. Newman, A.-L. Barabási, and D.J. Watts (Eds.), *The structure and dynamics of NETWORKS*. Princeton University Press, Princetion and Oxford, 2006.

[10] Y. Hayashi, "Rethinking of Communication Requests, Routing, and Navigation Hierarchy on Complex Networks -for a Biologically Inspired Efficient Search on a Geographical Space-," In T. Bilogrevic, A. Rezazadeh, and L. Momeni (Eds.), Networks -Emerging Topics in Computer Science, Chapter 4, pp. 67-88, iConcept Press, 2012.

[11] G. M. Viswanathan, M. G. E., Luz, E. P. Raposo, and H. E. Stanley, *The Physics of Foraging -An Introduction to Random Searches and Biological Encounters*, Cambridge University Press, 2011.

[12] I. Stojmeńović (Eds.), Handbook of Wireless Networks and Mobile Computing, John Wiley & Sons, 2002.

[13] A. Boukerche (Eds.), Handbook of Algorithms for Wireless Networking and Mobile Computing, Chapman & Hall, 2006.

[14] J. Urrutia, "Routing with Guaranteed Delivery in Geometric and Wireless Networks," *in Handbook of Wireless Networks and Mobile Computing, I. Stojmenović (Ed.)*, Chapter 18, John Wiley & Sons, 2002.

[15] W.-X. Wang, B.-H. Wang, C.-Y. Yin, Y.-B. Xie, and T. Zhou, Traffic dynamics based on local routing protocol on a scale-free network. *Phys. Rev. E*, Vol.73, pp.026111, 2006.

[16] B. Danila, Y. Yu, S. Earl, J. A. Marsh, Z. Toroczkai, and K. E. Bassler, "Congestion-gradient driven transport on complex networks," *Phys. Rev. E*, Vol.74, pp.046114, 2006.

[17] W.-X. Wang, C.-Y. Yin, C.-Y., G. Yan, and B.-H. Wang, "Integrating local static and dynamic information for routing traffic," *Phys. Rev. E*, Vol.74, pp.016101, 2006.

[18] G. M. Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E., da Luz, E. P. Raposo, and H. E. Stanley, "Optimizing the success of random searches," *Nature*, Vol.401, pp.911–914, 1999.

[19] M. C. Santos, G. M. Viswanathan, E. P. Raposo, and M. E. da Luz, "Optimization of random search on regular lattices," *Phys. Rev. E*, Vol.72, pp.046143, 2005.

[20] M. C. Santos, G. M. Viswanathan, E. P. Raposo, and M. E. da Luz, "Optimization of random searches on defective lattice networks," *Phys. Rev. E*, Vol.77, pp.041101, 2008.

[21] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Mod. Phys.*, Vol.74, No.1, pp.47–97, 2002.

[22] N. E. J. Newman, "The Structure and Function of Complex Networks," *SIAM Review*, Vol.45, No.2, pp.167–256, 2003.

[23] R. Xulvi-Brunet and I. Sokolov, "Evolving networks with disadvantaged long-range connections," *Phys. Rev. E*, Vol.66, pp.026118, 2002.

[24] S. S.Manna and P. Sen, "Modulated scale-free network in euclidean space," *Phys. Rev. E*, Vol.66, pp.066114, 2002.

[25] P. Sen and S. S. Manna, "Clustering properties of generalized critical euclidean network," *Phys. Rev. E*, Vol.68, pp.026104, 2003.

[26] A. K. Nandi and S. S. Manna, "A transition from river networks to scale-free networks," *New J. Phys.*, Vol.9, pp.30, 2007

[27] J. Wang and G. Provan, "Topological analysis of specific spatial complex networks," *Advances in Complex Systems*, Vol.12, No.1, pp.45–71, 2009.

[28] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, Vol.286, pp.509–512, 1999.

[29] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, Vol.393, pp.440–442, 1998.

[30] Z. Zhang, S. Zhou, Z. Su, T. Zou, and J. Guan, "Random sierpinski network with scale-free small-world and modular structure," *Euro. Phys. J. B*, Vol.65, pp.141–148, 2008.

[31] T. Zhou, G. Yan, and B.-H. Wang, "Maximal planar networks with large clustering coefficient and power-law degree distribution," *Phys. Rev. E*, Vol.71, pp.046141, 2005.

[32] Z. Zhang and L. Rong, "High-dimensional random apollonian networks," *Physica A*, Vol.364, pp.610–618, 2006.

[33] J. P. K. Doye and C. P. Massen, "Self-similar disk packings as model spatial scale-free networks," *Phys. Rev. E*, Vol.71, pp.016128, 2005.

[34] L. Wang, F. Du, H. P. Dai, and Y. X. Sun, "Random pseudofractal scale-free networks with small-world effect," *Eur. Phys. J. B*, Vol.53, pp.361–366, 2006.

[35] H. D. Rozenfeld, S. Havlin, and D. ben Avraham, "Fractal and transfractal scale-free nets," *New J. of Phys.*, Vol.9, pp.175, 2007.

[36] S. N. Dorogovtsev, A. V. Goltsev, and J. F. F. Mendes, "Pseudofractal scale-free web," *Phys. Rev. E*, Vol.65, pp.066122, 2002.

[37] R. Albert and A.-L. Barabási, "Error and attack tolerance of complex networks," *Nature*, Vol.406, pp.378–382, 2000.

[38] Y. Hayashi, "Evolutionary construction of geographical networks with nearly optimal robustness and efficient routing properties," *Physica A*, Vol.388, pp.991–998, 2009.

[39] Y. Hayashi and Y. Ono, "Geographical networks stochastically constructed by a self-similar tiling according to population," *Phys. Rev. E*, Vol.82, pp.016108, 2010.

[40] M. I. Karavelas and L. J. Guibas, "Static and kinetic geometric spanners with applications," In *Proc. of the 12th ACM-SIAM Symposium on Discrete Algorithms*, pp. 168–176, 2001.

[41] Y. Hayashi and Y. Ono, "Traffic properties for stochastic routing on scale-free networks," *IEICE Trans. on Communication*, Vol.E94-B(5), pp.1311–1322, 2011.

[42] J. D. Noh and H. Rieger, "Random walks on complex networks," *Phys. Rev. Lett.*, Vol.92, No.11, pp.118701, 2004.

[43] H. Shah, "Routing Enhancement Specific to Mobile Environment Using DTN," *International Journal of Computer Theory and Engineering*, Vol.3, No.4, pp. 537-542, 2011.

[44] R. J., D'Souza and J. Jose, "Routing Approaches in Delay Tolerant Networks: A Survey," *International Journal of Computer Applications*, Vol.1, No.17, pp. 8-14, 2010.

[45] W. Zhao, and M. H. Ammar, "Message Ferrying: Proactive Routing in Highly-partitioned Wireless Ad Hoc Networks," *Proc. of the 9th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS)*, pp.308-314, 2003.

[46] W. Zhao, M. H. Ammar, and E. Zegura, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks," *Proc. of the 5th ACM International Symposium on Mobile Ad Hoc Networking (MOBIHOC)*, pp.187–198, 2004.

[47] W. Zhao, M. H. Ammar, and E. Zegura, "Controlling the Mobility of Multiple Data Transport Ferries in a Delay-Torerant Network," *Proc. of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pp.1407–1418, 2005.

[48] M. Shin, S. Hong, and I. Rhee, "DTN Routing Strategies using Optimal Search Patterns," *Proc. of the 3rd ACM Workshop on Challenged Networks (CHANTS)*, pp.27–32, 2008.

[49] N. Alon, C. Avin, M. Koucky, G. Kozma, Z. Lotker, and M. R. Tuttle, "Many Random Walks Are Fster Than One," *Combinatorics, Probability and Computing*, Vol.20, No.4, pp.481–502, 2011.

[50] A. Clauset, C. R. Shalizi, M .E. J. Newman, "Power-Law Distributions in Empirical Data," *SIAM Review*, Vol.51, pp.661–703, 2009.

Generic Middleware for User-friendly Control Systems in Home and Building Automation

Armin Veichtlbauer
Josef Ressel Center for User-Centric
Smart Grid Privacy, Security and Control
Salzburg University of Applied Sciences
Puch/Salzburg, Austria
armin.veichtlbauer@en-trust.at

Thomas Pfeiffenberger
Advanced Networking Center
Competence Field e-Tourism
Salzburg Research Forschungsgesellschaft
Salzburg, Austria
thomas.pfeiffenberger@salzburgresearch.at

***Abstract*—In the field of Home Automation and Building Automation systems, the lack of interoperability of subsystems constitutes a major problem, especially for the integration of subsystems of different vendors. In order to overcome this drawback, our research group developed a concept of a generic control framework, which allows for integration of heterogeneous subsystems in an easy to control manner. This control framework contains functions to provide a dependable and secure control system for various Home Automation respectively Building Automation applications. To achieve that, the framework must be able to handle multiple users with different access rights using a variety of applications, as well as multiple devices (sensors, actuators, controllers, PCs, switches, routers, etc.) with different algorithmic roles. As a proof of concept, selected functions of this framework have been implemented and tested at a local test site. In this paper, we outline the architecture of the framework, describe the centerpiece of this architecture (i.e., the middleware layer), and show some results of the validation process.**

***Keywords*—Home Automation; Communication Infrastructure; User Control; Generic Interfaces**

I. INTRODUCTION

As stated in [1], Home Automation (HA) and Building Automation (BA) systems usually consist of a variety of different sensors and actuators (field level / field zone) as well as control devices (automation level / station zone), which are interconnected via several field bus technologies, like European Installation Bus (EIB), Modbus, Local Operating Network (LON), Digital Addressable Lighting Interface (DALI), etc. Alternatively, radio or powerline communication may be used to reduce mounting costs, especially for already existing surroundings. The management level / operation zone, if existing, supervises and controls the automation tasks; in many cases this is realized via web-based services in order to allow a remote control of the automation applications, possibly using smartphones [2].

The market for HA and BA solutions has been rapidly growing in recent years; yet in most cases buildings are not equipped with an integrative solution from a system provider, but with individual solutions for different building automation applications [3]. The lack of interoperability of these heterogeneous solutions prevents the shared use of existing equipment, e.g., information from access control systems (like the number of persons in certain parts of a building) could be a valuable input for evacuation support systems in cases of danger, but is usually not accessible due to the proprietary nature of both solutions. Especially for home users, which do not aim to afford an industrial sized solution for HA, this situation is very unsatisfactory, as the management of distinct island solution is not only a cost factor, but also uncomfortable - both the costs and the lack of user friendliness have been identified as big market barriers for HA [4].

A. Research Goals

Our approach to overcome the mentioned drawbacks was to define a framework, which uses open protocols and generic standards at every communication layer according to the OSI reference model [6] and at every level of the automation pyramid. Thus, every control application supporting these standards can use the functionality of our HA/BA framework without the need for individual adaptations. We conducted a thorough requirements analysis to determine the functions, which had to be added to these underlying technologies in order to form a working solution. Based upon this analysis we derived our architectural model, which we referred to as "X-Model", consisting of infrastructure, middleware, and application layer respectively. The middleware layer was designed as a convergence layer on All-IP [7] basis, which allowed us for keeping the framework architecture simple, while facilitating the integration of several applications of different vendors as well as the use of different network infrastructures.

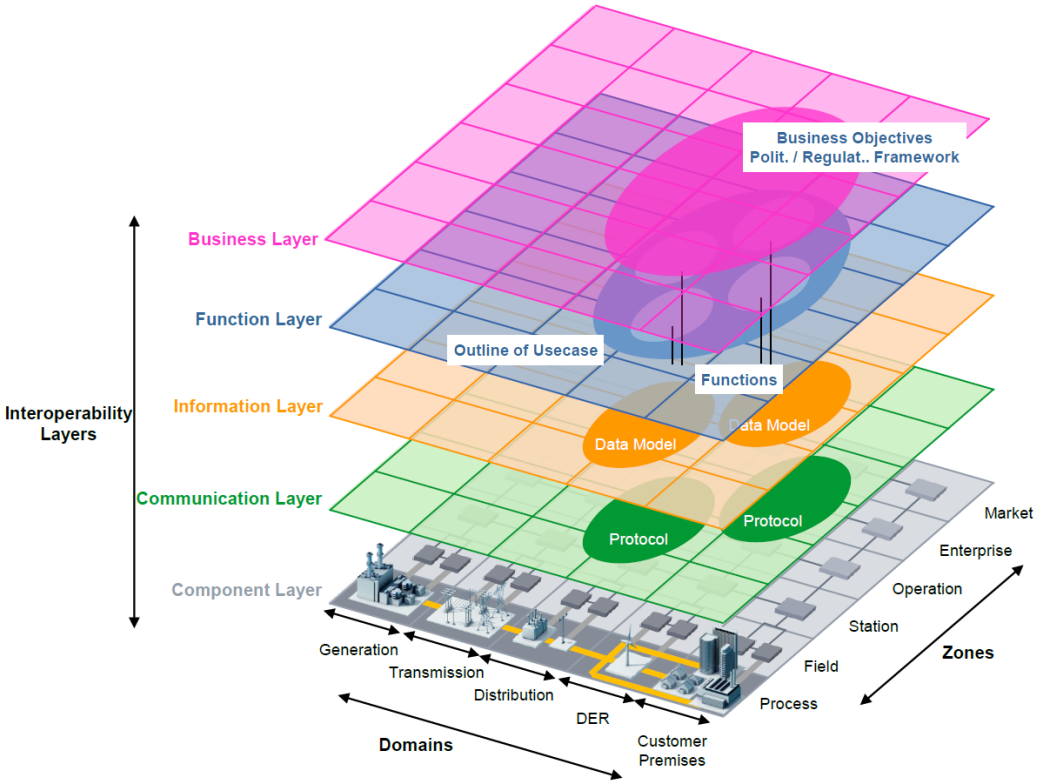


Figure 1. M/490 Smart Grid Architecture Model [5]

In the already finished research project "ROFCO" (Robust Facility Communication) [8], which was funded by the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT), we developed the generic control architecture for use in a HA/BA surrounding. We implemented selected middleware functions and tested them using applications like lighting control and blinds control [1]. Hereby, the implementation of these applications as well as the setup of the testbed infrastructure have been performed for validation purposes only; conceptually, these parts formed the test environment for the actual proof of concept, i.e., the middleware.

During the current work in the "Josef Ressel Center for User-Centric Smart Grid Privacy, Security and Control" (also referred to as "EnTrust") [9], which is funded by the Austrian Federal Ministry of Economy, Family and Youth (BMWFJ), we use this architecture to deploy Smart Grid applications like demand response or energy monitoring as well as for health monitoring in an Ambient Assisted Living (AAL) environment. Obviously, Smart Grid applications induce additional requirements compared to stand alone HA/BA systems, especially regarding security and privacy, since data is exchanged with external parties like utilities.

The architectural concept of the X-Model, however, seems suitable also for this extended functionality [10]. In terms of the M/490 standardization mandate [11] of the European Commission for the Smart Grid area, our approach complies with the customer premises domain (i.e., HA) of the M/490 Smart Grid Architecture Model (SGAM) [5], as shown in Figure 1. This current work of our research group is to be published in further follow up papers.

B. *Scope of Paper*

In this paper, we will give details about the architectural framework depicted as X-Model with a special focus on the core functionalities, i.e, the middleware layer. This layer contains functions to provide sufficient dependability [12], especially for highly safety relevant applications like evacuation guidance in case of emergencies. This paper extends our conference paper [1], where we presented the basic points of our architecture as well as some validation issues, by providing additional information about the theoretical background of this architecture, i.e., the requirements, design principles, and the specification of our solution. Hereby, we will pay special attention to the middleware layer of our architecture, as in our X-Model approach

this is the core part containing the business logic; conceptionally, this part has to be able to work with the whole plethora of possible applications on the higher abstraction layer, as well as with all common HA/BA infrastructures on the lower abstraction layer.

We will start giving an overview about the work of other research groups in that area, and will also investigate the state of the art in industrial solutions in HA/BA. We will then work out the functional and non-functional requirements for a generic control solution, resulting in our architectural approach. We describe the three layered architecture we propose for a generic HA/BA control framework (X-Model). Then, we give a brief description of the testbed infrastructure we used for validating the middleware functionality including the specification of network parameters and participating devices and HA appliances. After that, we will give a detailed description of the business logic in our middleware layer, containing the core functions of our X-Model architecture to ensure dependability in our framework. Here, we define the roles, which have to be implemented by the participating devices, and make an assessment of several potential solutions we could use to fulfil the ascertained middleware requirements. This is followed by some implementation issues and a short overview of the tests we conducted at our testbed in order to validate our approach. We conclude with an outlook and some open research questions for future work.

II. Related Work

The heterogeneity of HA/BA solutions has been identified as a potential barrier for HA/BA technologies since about the turn of the millennium [13] [14]. Big vendors may offer integrative solutions, e.g., "Total Building Solutions" from Siemens [15] or "Raumtalk" from ABB [16], yet based on proprietary communication and control technologies. Several research teams have tried to overcome this barrier by proposing interoperability features for HA/BA systems, e.g., via gateways between field bus technologies [13], or by providing complete HA/BA architectures for interoperable HA/BA applications [2] [17]. For communication infrastructures, the idea of using the IP standard is not new [14].

A fully integrated approach, however, requires solutions for the whole automation pyramid, i.e., on every level of the control process: setting and getting values at field level, performing a control task at automation level, and supervising this at management level. A standardised middleware for that purpose needs to provide more than just IP communication; especially, a generic modelling of BA objects and variables is inevitable. For that purpose, the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE) defined the Building Automation and Control Networks (BACnet) standard [18]. With BACnet, complete HA/BA environments could be built based on one generic technology [19]; yet in reality this approach has several drawbacks:

- The computational power required by the BACnet protocol suite is rather high, thus many field layer devices are not able to implement the BACnet stack, i.e., these devices have to be integrated via gateways.
- The support of the very common IP protocol is weak, as it is not part of the native BACnet stack. A work around named BACnet-IP is provided, i.e., a tunneling of BACnet messages through an IP network.
- State-of-the-art network management concepts like Quality of Service (QoS) are not supported with BACnet, which is especially critical with the use of safety or security relevant control applications [20], as they require very high dependability standards, especially concerning availability of communication infrastructure.

The definition of the Object Linking and Embedding (OLE) for Process Control - Unified Architecture (OPC-UA) standard [21], which is already commonly used for the control of industrial production [22], may help to overcome these shortages. OPC-UA is an interoperability standard originally based on Microsoft's Distributed Component Object Model (DCOM) standard, which facilitates reading and writing access to distributed field components (OPC Servers), which can be used by industrial Supervisory Control and Data Acquisition (SCADA) applications (OPC Clients) for their respective control tasks. By using OPC-UA in combination with TCP [23] as transport protocol it is possible to integrate IP networks and all the QoS mechanisms existing for the TCP/IP protocol stack. Some academic implementations of OPC-UA for HA/BA systems are already existing, e.g., the solutions of the TU Vienna [24]. Yet the requirements for end systems still are rather high, resulting in the necessity to provide gateways to legacy systems containing older devices with not sufficient computational power.

In BA, the use of industrial SCADA systems, which contain drivers for many different BA solutions, is a feasible approach and thus offered by BA vendors, e.g., [25] [26]. As a consequence, a suitable device for the management level (capable of running the SCADA software) has to be used, i.e., in most cases a device having the same computational power as a PC. This seems no problem for BA; for HA, however, such a supervising device at management level embodies a barrier for spreading the market widely - for HA, smaller, cheaper

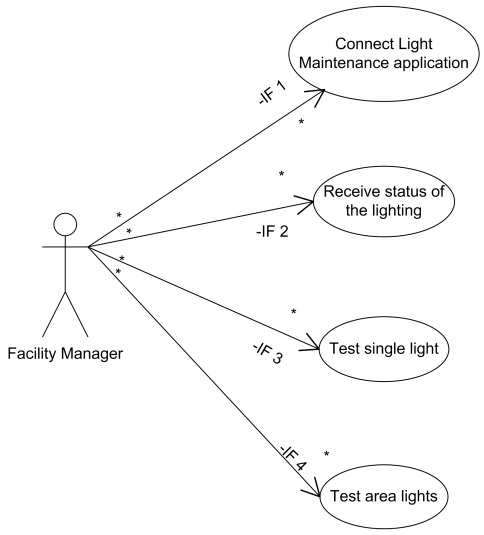


Figure 2. ROFCO Use Case Light Maintenance

and easier to deploy solutions have to be found, i.e., lightweight SCADA systems that can be addressed as web services or work as smartphone apps. Such systems are sometimes referred to as "Mini-SCADA" and are offered to end-users of HA/BA systems, but partially also to other stakeholders like energy utilities [27].

As pointed out in [28], interoperability issues are still an unsolved problem in HA, and constitute thus an important market barrier for HA solutions. For the Smart Grid area, the need for standardization has been clearly identified, e.g., in [29], but from the point of a HA customer, a smart building contains a variety of applications, which have to be included in a trusted user domain [30]. In that context, safety and security topics are of notable interest in order to produce saleable solutions [31], as open systems are always prone to outages [32] in consequence of improper use or even planned attacks.

There are some further research activities in the area of HA/BA systems. These include topics as control strategies and technologies [33], as well as performance issues [34]. Besides the technical research fields there are multiple socio-economic research activities, focussing on the potential impacts of the studied technologies on end-users.

III. REQUIREMENTS

During the requirements engineering process, we identified user stories in cooperation with the ROFCO project partners, especially with the Techno-Z Salzburg, which hosted the testbed for the validation of our approach. Hereby we were considering the interests of different stakeholders, e.g., fire fighters, public authorities, or end users. We then extracted the respective use cases from the user stories and depicted them in the Unified Modelling Language (UML), as shown in Figure 2. From the explored use cases we derived the general requirements, which we then broke down to concrete technical requirements.

A. Requirements Analysis

The challenge of the requirements analysis for our intended generic dependable HA/BA solution, which we called the "Dependable HA/BA Framework" (DHF), was to support the different and complex requirements of a variety of heterogeneous HA/BA applications. Conceptually, all thinkable HA/BA applications must be included in order to provide the required genericity. Yet as the requirements engineering process was based on use cases, we had to choose applications controlling typical HA/BA appliances, but not too similar and thus providing an as complete range of requirements as possible. At the end, we decided to base the requirement analysis of the DHF on three potential HA/BA applications:

- Lighting Control
- Blinds Control
- Evacuation Support

The first two applications also built the basis for our validation process (see Section VIII); the last application, however, was important for the requirements analysis in order to assess additional non-functional (quality) requirements, especially regarding safety and reliability [35]. As mentioned, the use of our architecture in the Smart Grid area creates further requirements. These are currently explored and thus not part of the original requirements engineering process described here.

In the following, we describe the requirements engineering process based on the exemplary application Lighting Control. First, the Lighting Control user story was defined in cooperation with the Techno-Z Salzburg as mentioned above. Since different user types (stakeholders) are involved, the user story contains different roles and activities based on appropriate authorization mechanisms. Roles define the rights to perform simple atomic activities, like receiving or sending messages from a user interface to some control units, sensors, or actuators in the DHF. Thereto the different components must support authentication, authorization, and encryption. To integrate already installed systems to the DHF, mediators are used to adopt and translate the respective messages. For this user story, we derived appropriate use cases by grouping atomic activities to expedient units. The resulting use cases cover not only direct lighting control in the building (on/off or dimming of certain

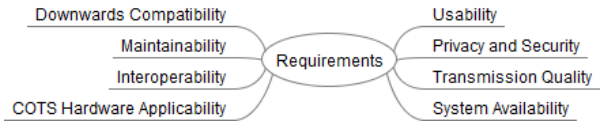


Figure 3. General Requirements

lights), but also procedures for evacuation situations, maintenance, holidays, alarms or personalized control procedures (e.g., on/off or dimming of a user-defined group of lights). For instance, the use case Light Maintenance depicted in Figure 2 consists of the following atomic procedures:

- Connect the Light Maintenance application to the DHF.
- Receive the status of the lighting (on/off - dimming status - failure) in the configured area.
- Set a single light (on/off - dimming) and check the status.
- Set a group of lights (on/off - dimming) and check the status.

B. General Requirements

After having defined the use cases for the aforementioned applications, we derived the general requirements on our DHF, as depicted in Figure 3.

- *Downwards Compatibiliy*
  First of all, the support of legacy systems must be guaranteed, as the acceptance and the price of new systems built from the scratch would prevent an economic exploitation of the solution. This holds simply for the fact, that existing parts of HA/BA systems have to be reused to keep the costs as low as possible, and that users might tend to use solutions they already know.
- *Maintainability*
  The whole system has to be easily maintainable and configurable. Most important, the integration of new devices must be working in a plug and play manner as far as possible. Clear enough, by having a rights management concept [36] limiting the use of devices, applications and data to users with respective rights, some configuration tasks will be unavoidable. All necessary configurations have to be performed in a user-friendly way, and supported by suitable tools, like wizards, as far as possible. As the degree of automation shall be adjustable, this may include decision support systems. For instance, when including a new sensor, the rights management system could provide suggestions about the users' rights by assessing the existing rights of similar sensors.
- *Interoperability*
  One of the most crucial requirements is interoperability, i.e., devices from different vendors must be integrated seamlessly to guarantee an easy access to the whole functionality for the respective users. This is ensured by the use of standards and open protocols, most important by the use of the IP as basic network layer protocol. Proprietary solutions should not be used as far as possible, and if it is unavoidable due to a lack of open solutions, the interfaces to these proprietary parts have to be defined clearly. Some proprietary solutions provide at least open application programming interfaces (APIs), on top of which our functionality could be realized.
- *Applicability of COTS Hardware*
  A main requirement of our system is to use commercial off-the-shelf (COTS) hardware. As the hardware must support high reliability and calculable availability, the mean time between failure (MTBF) and the mean time to repair (MTTR) metrics of each hardware device must be known in order to derive the system's overall availability. For authentication and authorisation well established mechanisms have to be used, such as ITU-T X.501 [37] or IETF Radius/Diameter [38] [39]. Encryption is a further main requirement to establish a secure connection over a distributed heterogeneous communication system. For the underlying network functionalities, classical network devices like Cisco switches and routers [40] are used. Address management and routing are based on IP, routing metrics [41] must be supported.
- *Usability and User-friendliness*
  A basic quality requirement of our middleware is to provide means to control several appliances (e.g., electric lighting) for different types of stakeholders (e.g., end users, home owners, etc.). This includes freedom of choice for using more or less automation: For instance, user A might want to have a fully automatic control of room temperature, which is configured once and then working continuously, whereas user B wants to manually control the room temperatures in order to have a greater flexibility. Although there are no commonly accepted metrics for user-friendliness, the integration of customer choice mechanisms in HA/BA seems indispensable in order to raise user acceptance [42].

- *Privacy and Security*
  The use of open systems, which are accessible via Internet to enhance user-friendliness, has some drawbacks concerning privacy and security. As it is not possible anymore to build closed ecosystems, which are per definition not accessible to potential fraud, we have to face unexpected and unauthorized use of system resources, up to the possibility of attacks, e.g., denial of service attacks damaging safety functions, or intrusions to get access to private data. This is especially risky for distributed systems, e.g., energy sharing communities in settlements. For instance, the exact knowledge of energy consumption of a household could be used to identify the currently watched TV program [43]. Thus, a complete authentication, authorization, and accounting (AAA) system in connection with a suitable encryption technology is necessary to enable authorized access only. Furthermore, countermeasures against potential attackers and methods of ensuring the privacy of data (e.g., data aggregation) have to be considered.

- *Data Transmission Quality*
  An overall requirement in a dependable infrastructure is to guarantee the transmission capacity and the transmission quality. Thereto some Quality of Service mechanisms in the communication infrastructure are required, such that the different network components and applications are able to label the data packets according to the transmission quality requirements. Luckily, IP supports the labelling of the packet by using the so called "Type of Service" field [7].

- *System Availability and Reliability*
  Last but not least the required dependability [12] of the intended solution has to be guaranteed, in terms of availability and reliability [44]. The availability can be assured by a process life cycle management according to [45], defining availability metrics dependent on applications' risk parameters like probability, avoidance possibility, frequency and consequences. Reliability is issued by several testing methods; for the validation of our prototype we used functional tests of the implemented components, yet this was not the core of our research, as the realized prototype works basically as proof of concept. Thus, for validation of commercially saleable solutions a much more exhaustive testing process would be required in order to facilitate the keeping of existing standards and regulations (see Section VIII).

C. Technical Requirements

From these high level requirements we derived concrete (functional and non-functional) technical requirements for the DHF. The non-functional requirements basically concern the quality of the underlying communication infrastructure, which we take as given in order to be compatible to existing solutions. This quality is assessed in terms of:

- Bit Error Rate (BER)
- Redundand Networkpaths
- Attack Robustness
- Catastrophe Robustness
- Data Packet Prioritization
- Deterministic Delay Bounds
- Network Size (number of end devices)
- Data Rate
- Range (Link length)

The functional requirements concern the necessary functionality of the DHF for users in order to perform their monitoring and control tasks in a secure manner. Thereto a rights management is indispensable, as different users (and user types) may share access to the same appliances. Thus we have derived the following functional requirements:

- Sensor/Actuator Interaction: Means to collect sensor data and to apply control strategies to actuators
- Data Structure and Representation: Means to represent, store, and query data used to control several appliances in a HA/BA environment
- Signing and Encryption: Means to label data and to avoid unauthorized use thereof
- Authentication and Authorization: Means to enable the identification of users with respective access rights
- Registration and Discovery: Means to manage devices, applications, and users combined with automated detection of changes
- Notification and Alarming: Means to notify users in case of the fulfilling of defined conditions and to throw alarms in case of unexpected conditions like limit violations
- Abstract Address Scheme: Means to identify and address devices in a unique manner
- Heartbeat / Keepalive: Means to check whether crucial system parts are up and running

IV. ARCHITECTURE

After having finished the requirements engineering process, the resulting technical requirements for dependable generic HA/BA systems could be grouped in two layers: infrastructure requirements and middleware requirements [46]. This resulted in a layered approach, where the infrastructure functionality can be separated from the middleware functionality and the application themselves, which use the middleware and infrastructure functions.

Moreover, as our goal was to integrate different applications as well as different infrastructures, this would result in a N:M relationship in case that each application would have to run on each infrastructure. In order to avoid that, we had to introduce a convergence layer in the core of our architecture, thus forming what we called the X-Model.

Basically, this is a three layered approach as shown in Figure 4, where the middle layer serves as convergence layer, which can be used by all considered HA/BA applications, and which uses several considered infrastructure technologies (i.e., those that are suitable to meet the infrastructure requirements as defined in the requirements analysis):

- An infrastructure layer (INF), which embodies all the necessary networking functionalities and end devices for our control architecture
- A middleware layer (MID), which provides appropriate dependability [44] [47] means on an end-to-end basis
- An application layer (APP), which is responsible for the distributed control tasks of the applications using our architecture

A. Infrastructure Layer

As for the network infrastructure, we intended to use an All-IP solution, which is “Layer 2 agnostic”, i.e., that is able to run on a variety of lower layer technologies, including those field bus systems, which are common in the area of HA/BA. By this strategy it was possible to natively integrate numerous devices, as long as they are able to speak IP and are able to deploy the dependability functionality of our middleware. SCADA systems, e.g., “Zenon” from our project partner Copa-Data [48], can thus be integrated by providing an open software interface containing IP sockets. Due to this openness several SCADA manufacturers may share different end devices and data servers; i.e., our solution provides a holistic concept to integrate global dependability means, opposed to currently available island solutions. Thus, a “dependability domain” is generated, which is realized by our DHF.

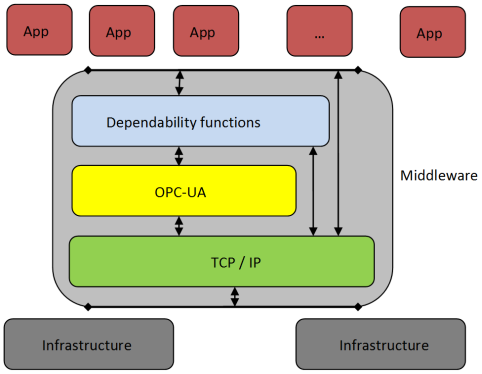


Figure 4. Generic Architecture (X-Model)

However, the integration of legacy components, which are not IP capable, could be done only via gateways, as shown in Figure 5. At this place, information loss can not be avoided completely, as the legacy devices do not necessarily support all required parameters. As a consequence, the guarantees for dependability can be made only for the natively integrated components. In spite of this drawback, the use of legacy components may enrich the dependability domain, e.g., by the integration of additional sensors - yet these components are not an integral part of the dependability domain. In this case, the parameter mapping has to be defined at the respective gateway, which is then providing these data in a dependable manner for all system integrated applications, thus providing added value. The other direction, i.e., the control of actuators outside the dependability domain, is also possible in principle, yet the dependability properties can then be mapped only partially, depending on the mechanisms of the legacy components. In both cases, the scope of the dependability domain ends in the gateways.

B. Middleware Layer

The main goal of the generic architecture was to ensure dependability [44], i.e., robustness, reliability, availability, maintainability, safety and security. For instance, by ensuring interoperability in the way that applications should have access to the whole network and sensor/actuator infrastructure, the danger of potential misuse arises; this implicates the necessity to define appropriate security means in order to avoid damages. Safety relevant applications require high standards of reliability, availability and robustness. Thus, the core functionality of the middleware layer was to provide appropriate means to facilitate and document the fulfillment of these dependability requirements within the dependability domain, i.e., the scope of the control architecture consisting of natively integrated and fully functional devices.

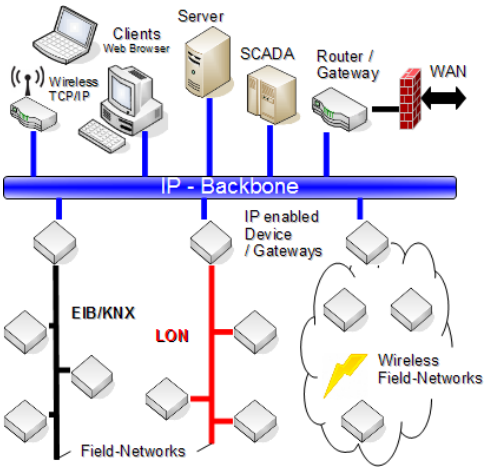


Figure 5. Generic Network Infrastructure

C. Application Layer

The application layer comprised several control logics, e.g., implemented by a Programmable Logic Controller (PLC) or a Direct Digital Control (DDC), at automation level as well as supervisory tools for end users at the management level (SCADA / Mini-SCADA), both functionalities based on working middleware implementations. Consequently, the control logics should run on devices, which are capable to host the complete middleware, as otherwise the dependability can not be fully ensured. Legacy controllers could be operated in a way, that they provide information to the dependability domain (which can for instance be evaluated and visualized by a SCADA system), but were not an integral part of the dependability domain.

According to the layered approach, following issues had to be done in parallel after finishing the design phase:

- First, we had to specify a network / hardware architecture, which was able to meet the identified infrastructure requirements.
- Second, we had to define the middleware functionality and to determine, which functions thereof we wanted to implement with our framework prototype.

These questions are addressed in the next two sections; this is followed by some implementation issues, as well as a description of the validation process and its results. The validation process comprises the setup of a real-world testbed according to the infrastructure specification, the conduction of necessary functional tests with the implemented prototype, as well as an evaluation of results.

V. INFRASTRUCTURE

As our framework should work with all multi-vendor infrastructures fulfilling our requirements, our aim was not to implement yet another infrastructure technology, but to choose suitable existing solutions. Thus, the functionalities of potential infrastructure technologies, e.g., providing appropriate link layer mechanisms, have been assumed as given. For validation purposes we had to set up a testbed infrastructure suitable to provide all required mechanisms for testing our proof of concept implementation (test environment); yet this actual proof of concept contained only middleware functions (system under test).

Basically, infrastructure technologies consist of two parts: the participating devices (which we intended to use as they are in order to ensure optimal compatibility with existing HA/BA solutions), and the lower layer network functionality (which is specified within the OSI reference model [6]).

A. Testbed Network

With given functional properties, we still had to assess the potential communication infrastructure technologies regarding their non-functional properties, i.e., the fulfilling of quality requirements, before setting up the testbed network physically. We had identified four potential infrastructure technologies, which could be used as a basis for the testbed we intended to set up at the test site of our ROFCO project partner Techno-Z Salzburg: Ethernet, Wi-Fi, ZigBee and Powerline.

Table I shows the matching of the quality requirements for these communication infrastructure technologies. The mentioned All-IP approach of our architecture guarantees the required Layer 2 agnosticism by definition [7]; furthermore IP is a protocol that had proved its ability to work in generic network systems for decades (and thereby functioning with a variety of different PHY and MAC layer protocols according to the OSI reference model [6]). Thus, it was a quite logic decision to use an All-IP approach for our HA/BA architecture. As a consequence, we could choose the concrete Layer 2 technology freely, provided that the chosen technologies meet our above defined requirements.

As Ethernet provides good quality regarding the BER metric, as well as convincing scalability properties, we decided to use it as base technology, extended with a WiFi access point in order to provide the required redundancy. Additionally we installed fiber channels to connect the different buildings of the test site. The usage of this combination of communication technologies as network infrastructure for our testbed kept the installation effort low, as Ethernet cabling was already present in all buildings of the test site.

Table I. INFRASTRUCTURE REQUIREMENTS

	Ether net	IEEE 802.11	Zig Bee	Power line
Bit Error Rate (BER)	++	– –	–	+
Redundand Networkpaths	+	++	–	– –
Attack Robustness	+	– –	– –	++
Catastrophe Robustness	– –	+	+	–
Data Packet Prioritization	++	++	+	–
Deterministic Delay Bounds	+	– –	+	–
Network Size [# end devices]	2^{48}	2^{48}	64k	2-50
Data Rate [Mbit/s]	10-1000	11-54(600)	0.02-0.25	10-200
Range [m]	100	1-100	1-100	200-300

Figure 6 shows the network topology of the testbed, which expanded over three buildings (3, 10, 12) at the Techno-Z. It was basically composed of two class C IP subnets:

- The management subnet of the Techno-Z used in Building 10 and 12
- The control subnet from the ROFCO laboratory at Building 3

In both subnets we used switches with two redundant GBIC ports, thus connecting both subnets with redundant fiber connections between Building 3 and Building 10. A third switch in the ROFCO laboratory built the interface to the various ROFCO servers. As part of the robustness concept these (manageable) switches were configured with the spanning tree (STP) mechanism. Due to the security concept two Virtual Local Area Networks (VLAN) I and II were configured on these three main switches, i.e., the devices connected to these switches could be run in both VLANs.

Both subnets were connected with respective company networks (Techno-Z and Salzburg Research) via a router/firewall combination. For further security issues an internal sniffer was installed to monitor the traffic inside the control and management subnets. Both functionalities, along with an intrusion detection system (IDS), could be performed by using the "MF-Security-Gateway" [49] from the ROFCO project partner Underground8.

B. Testbed Components

Besides defining the network parts of our infrastructure, we had to address the question of end devices. Whereas we had been free in the choice of network components (only provided that they meet our requirements), we had to use existing devices for the respective control tasks we wanted to perform in the validation of our prototype, since the project's system context (and thus the applications we used within this context) was defined by the Techno-Z as host of our testbed. As technology park the Techno-Z expressed its project interests in very concrete facility management tasks, which we formulated as UML Use Cases during the requirements analysis. Each building at the Techno-Z is equipped with different BA systems, e.g., a Somfy system to control blinds and a Sauter system to control the lighting and all heating, ventilation, and air conditioning (HVAC) components via EIB/KNX. In the following, we describe those components, which we have researched as part of the heterogeneous ROFCO testbed, grouped to their location.

- *Somfy Control, Building 10*
  To control the blinds of the Buildings 10 to 15, the Somfy blind control was separated into three zones. In zone one, a single Somfy control system at the 3rd Floor regulated the whole blinds for Building 10. At this place a controller of our project partner cTrixs called "cTrixs Base Unit" (CBU) [50] was installed, which served as gateway between the blind circuit (over relay control and digital I/Os) and the Ethernet wiring, which offered the connection to the switch in the ground floor.
- *Facility Management Room, Building 12*
  For managing the BA systems for the Techno-Z complex, a control computer was situated in the facility management room in Building 12 on the ground floor. On this computer the Sauter BA system (which includes the HVAC capabilities) or the Designa access control systems were visualized. Also the fire alarm center was located in this room.
- *Engineering Room, Building 12*
  The Sauter BA system, the EIB lighting system and the central switch were located in the engineering room at the ground floor in Building 12. The entire building is wired from this switch. For the ROFCO network a port on the central switch was reserved and activated. There was also the possibility to configure VLANs on this Catalyst 2950 switch. A second cTrixs controller provided the interface to the EIB lighting in the congress room in Building 12; it was connected to the central switch and to the EIB bus to control the lights at the ground floor.
- *ROFCO Laboratory, Building 3*
  The laboratory was equipped with a cTrixs Application Server (CAPS) and a Zenon Server from Copa-Data with master/backup function.

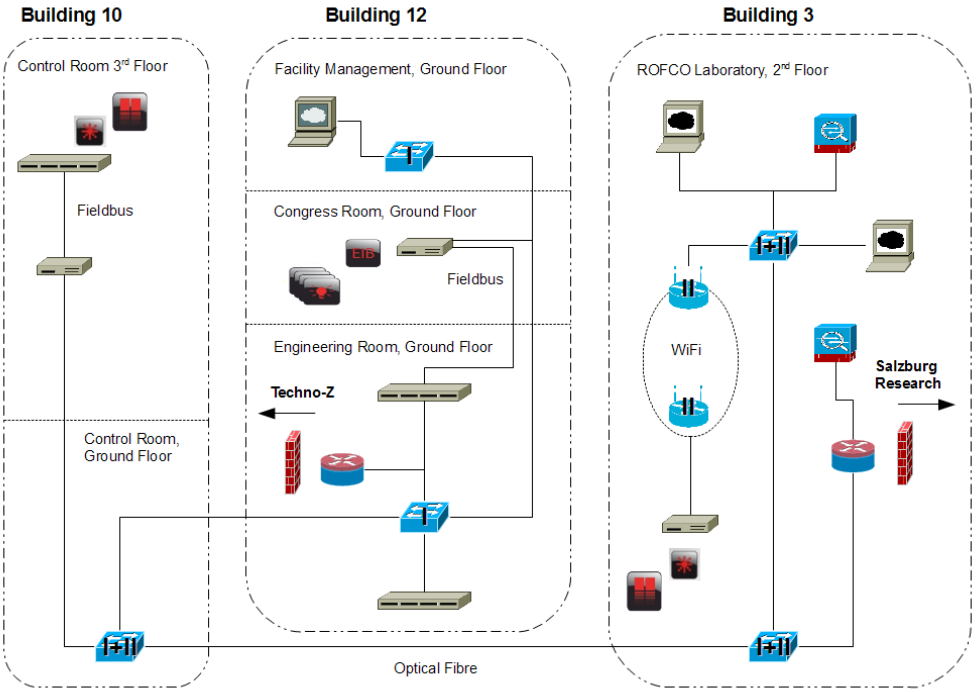


Figure 6. Testbed at Techno-Z Salzburg

With the Zenon SCADA software the use cases we considered in ROFCO could be visualised and controlled. The CAPS was used as a central server for the cTrixs controllers.

At the ground floor in Building 12, the lighting was not fully represented in the Techno-Z's building management; the same applied to some blinds control functions (e.g., open all blinds at one side of the building simulateously). Thus, the respective data points and functions were implemented and visualised on the CAPS and Zenon surfaces and controlled via cTrixs controllers. In Building 3, the blinds were handled by an IP-enabled cTrixs controller, but in opposition to the solution in Building 10, the connection was done directly via analog outputs and relays, and not via EIB. A Wireless Local Area Network (WLAN) bridge has been installed to transmit data to the controller.

VI. MIDDLEWARE

According to the outcome of the requirements analysis and the architecture design process, the middleware layer has to provide means to establish a dependable end-to-end communication between different entities, thus supporting independent distribution of control information between different end systems. This includes not only availability and safety of end-to-end communication, but also an information security and rights management concept [36] [32]. Furthermore, the middleware layer comprises added value: generic data structures (e.g., SensorML), supervising functions, etc. These concepts are detailed in the following.

The middleware layer can make use of the underlying infrastructure layer, which is guaranteeing for the meeting of the lower layer requirements, i.e., requirements for devices and communication links between them. In opposition to that, the middleware layer addresses end-to-end concerns only. It is feasible to address some properties at both layers: For instance, link layer security measures may prevent unauthorized listening on the channel, whereas transport layer security provides end-to-end encryption and authentication to prevent man-in-the-middle attacks. This may be redundant, but relying on link layer security measures is risky, as one unsecured link would jeopardize the whole security concept.

A. Survey of Base Technologies

The targeted functionality is addressed by a number of existing technologies, from commercial products to open protocol standards. Therefore, a new implemen-

Table II. MIDDLEWARE REQUIREMENTS

	OPC -UA	Modbus -TCP	SIP	Soap WSSec
Sensor/Actuator Interaction	y	y	n	n
Data Structure & Representation	y	n	n	n
Signing and Encryption	y	n	y	y
Authentication & Authorization	y	n	y	y
Registration & Discovery	y	n	y	y
Notification & Alarming	y	n	y	y
Abstract Address Scheme	y	n	y	y
Heartbeat / Keepalive	y	n	n	y
Further Robustness Features	y	y	n	n

tation from the scratch seemed an unfavorable solution, taking into account limited resources of research projects. In order to find middleware functions, which were supporting our requirements and which could be integrated into our prototype by providing an appropriate application programming interface (API), we conducted an analysis of some promising solutions and evaluated their applicability for our approach.

Hereby, supporting our middleware requirements does not mean, that the respective technology implements the complete desired functionality, but that it supports the realization of it on top of its API. For instance, the support of the "data structure and representation" requirement means, that it is possible to define objects within a technology, e.g., representing sensor data, but not that for all thinkable sensors corresponding objects are already defined.

Thus, the examined technologies should provide mechanisms to realize all the required functions, but not the implementation of the respective functions itself. As potential open accessible technologies for providing at least parts of the required middleware functions, we identified four candidates: OPC-UA, Modbus/TCP, SIP (Session Initiation Protocol) and SOAP with WS-Security.

Table II matches these candidate technologies with the identified functional requirements for the dependability middleware. As result of the comparison of potential technologies we decided for the use of OPC-UA as generic communication and management protocol [51], which seems to provide a good basis to create a generic control architecture.

B. Entities and Roles

In order to realize the intended dependability means, we had to define the respective business logic. As mentioned, these functions may use an underlying infrastructure fulfilling all lower layer requirements and an OPC-UA stack with API as a basis for the new implementation.

As our approach was to provide a complete definition of the conceptual part (yet only implementing selected functions for validation purposes) we had to perform a comprehensive modelling of the desired functions within our dependability domain. For that purpose we had first to define the entities and roles within the DHF. The entities can be identified with the devices participating in the DHF:

- Sensors
- Actuators
- PLCs, DDCs
- PCs
- Mobile devices (smartphones, tablets)
- Active network devices (routers, switches)
- Data storages
- Communication hardware (cables, antennas)
- Embedded systems (Plug PCs, boards)

Sensors and actuators are data sources and sinks respectively; PLCs and DDCs are used for control tasks at automation level, PCs and embedded PCs also for visualization (SCADA), smartphones and tablets the same with less complexity; network devices and communication hardware provide the infrastructure functionality. The entities realize several distinguishable roles, which incorporate the logically independent parts of the whole functionality:

- Client
- Server
- Registrar
- Mediator

The clients (e.g., sensors, PLCs, smartphones) communicate and exchange information with the server. The server (e.g., a PC or embedded board) stores information about the clients and serves thus as a data base. Servers support the possibility to present the information in OPC-UA style. To be allowed to participate in the DHF, all defined parts (clients, servers) must register at the registrar. The registrar provides interfaces for authentication and authorization to the DHF. To communicate with a non-DHF entity, mediators (basically these are gateways, which are able to represent the data structures of the non-DHF part in a DHF compatible manner) map

all relevant information between DHF entities and non-DHF entities.

Network devices (switches, routers, etc.) do not have a functional role regarding the DHF's middleware and are thus considered transparent. To integrate QoS, service classes are defined for the different requirements of the supported applications and triggered by the end systems (clients).

By having defined the roles, the required functionality of the DHF middleware could now be assigned to these roles. In the following subsection, we concentrate on the registrar, as this is the core element for a generic framework, allowing for the integration of multiple clients, servers, and mediators into one framework.

C. Access Rights Management

The main purpose of the X-Model is to enable multiple applications, which are triggered by multiple users, to get access to all DHF devices. This implies the necessity for an access rights management, which is able to assign respective access rights to applications and users, and to enforce the keeping of these access rights. The basic idea is, that the DHF registrar manages the mapping of registered applications and registered devices [36] [32]. Thereto the registrar has not only to provide means to register for new devices and applications respectively to update the registered information for existing ones, but also to decide for appropriate access rights, i.e., it serves as a "Policy Decision Point" (PDP). Figure 7 shows a scheme of the registration process for client and server devices at the registrar.

Of course the access rights assignment can not be performed fully automated, yet a definition of application types respectively user types makes it possible to map access rights not only to individuals, but to groups with similar roles within the system. For instance, flat owners in a house with multiple parties may have less control rights than fire fighters in case of emergencies. These groups need to be assigned the respective access rights only once then. The classification of devices and applications respectively users has still be done manually, thus the system needs a human operator to control the admission to the DHF and to assign appropriate access rights, i.e., the authentication has to be done on a non-technical basis.

Once the registration process is finished, the registered entities are provided with appropriate keys to communicate directly with the peering entity. As every communication has to be encrypted anyway in order to ensure privacy and security of exchanged data, the distribution of decryption keys according to the defined access rights is a way to ensure, that only entities with respective access rights can read this data. This can go

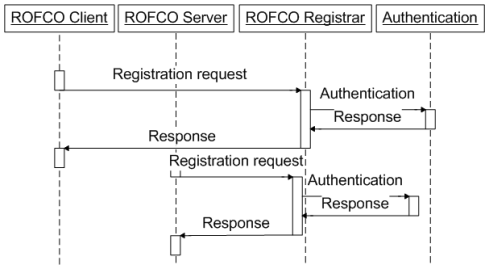


Figure 7. Registration Process

so far, that different entities with different keys can be provided with the same encrypted sensor data, and can decrypt these data with their respective keys in different resolutions, i.e., the different keys represent different authorization to access data. A further authentication with technical means does not have to be performed, as the ownership of the keys is bound to the registration process and therefore secured, provided that the keys are not illegally distributed by the owners.

The big advantage of this solution is the performance, as communication between entities with respective access rights does not require the invocation of central authorities. As resources are especially limited for HA solutions, the concern of performance issues might be unignorable in practice. Yet this is bought by a considerable disadvantage: Key revocation is not possible within such a scheme. The only way to deal with that is to provide access rights only for a defined time, with the necessity to renew admission and thus the key distribution after expiration of granted access rights.

For simplicity reasons, the key distribution may be performed by the registrar [36] [32], which is then also constituting the "Policy Enforcement Point" (PEP). Conceptually, these two functions may also be separated, which could be necessary if performance is still critical, depending on the scale of a DHF realization and the computational power of the registrar device. In opposition to our original intension we decided for symmetric encrpytion for our DHF concept, again for performance issues; only for key exchange asymmetric encryption, i.e., public key encryption, is used.

To complete our security portfolio, further functions have to be addressed: To detect misbehaviour or outages of end systems, keep alive messages are sent during normal operation. Anomaly detection is used to find faulty messages and traffic in the system [49]. With traffic monitoring this traffic can be detected and isolated from the system. Last but not least the triggering of alarms and notifications is not only possible with limit violations from sensor data, but also with peculiar communication attempts.

D. Quality Assurance

To identify potential failures in the design and the application life cycle in the whole DHF and to evaluate potential effects and countermeasures, a Failure Modes and Effects Analysis (FMEA) [52] has been conducted. The FMEA gives an overview about which parts of the DHF are most likely to fail, but also which parts have the most important impacts in case of failures. The FMEA basically consists of following steps:

- A system analysis identifies the parts of the DHF; this can be derived from the design phase, but it also has to take into account external influence factors from the respective system context.
- A function analysis identifies the functions, their allocation to the system parts, and their interoperations and dependencies; thus, critical components can be identified.
- A failure analysis identifies potential failures within the system and allocates them to respective system elements; furthermore, possible reasons for these failures are named.
- A risk analysis identifies the probability of failures, the probability of detection of these failures and the potential impacts; with these factors a risk priority number is calculated and so the most "dangerous" failures can be identified.
- Finally countermeasures and system optimization measures are derived, which shall either minimize the occurrence of failures or the probability of not detecting the failures in time or the potential damage, which could be caused by the considered failure.

Unsurprisingly, outages of controllers have been identified as most dangerous failures, as controllers of HA appliances are not realized in a redundant manner in most cases and thus constituting "single points of failures". The same applies to SCADA systems, yet outages of SCADA systems do not have such immediate consequences, as the several control applications for HA appliances may work autonomously for a certain time. The exact results of the FMEA are documented in the deliverables of the ROFCO project, but not publicly available.

VII. Implementation

As a proof of our concept, we implemented some basic functions prototypically, using the free OPC-UA stack from the OPC foundation [53]. Based on the interests of our partners in the project ROFCO those parts of the framework were implemented, which promised the most direct benefit to them, while still serving as a good basis for our validation process:

1) OPC-UA connection between SCADA system and different control devices (lighting, blinds)
2) Gateway between dependability domain and legacy components
3) Implementation of security (Authentication and Authorization) based on OPC-UA

A. OPC-UA Connection

In our testbed installation, appliances like blinds and lighting were controlled via cTrixs controllers. In order to facilitate communication between SCADA systems and the controllers on OPC-UA basis, parts of the OPC-UA stack had to be implemented at both sides, i.e., the CBUs and the supervising SCADA system. As SCADA system we used Copa-Data's Zenon [48] and the CAPS from cTrixs. Hereby the main focus was to exchange information over an OPC-UA interface by using the OPC-UA information model. Thereto we implemented some selected OPC-UA object types (base object, server objects and the event types). A further focus had been given on the integration of Java and C based OPC-UA libraries into the considered SCADA systems.

B. Legacy Gateway

To interconnect legacy components with the system, it was necessary to map and translate data from the legacy components. Status information about the legacy component had to be stored in an object on the gateway, which represented the properties of the legacy component in the system. The required registration at the system and the mapping of the information exchange had to be handled by the representing object. The implementation of the gateway functionality was based on the OPC-UA ANSI C library and the cTrixs controller communication protocol, which is again based on UDP. As the cTrixs controller was able to map and translate OPC-UA information to EIB/KNX components [54], we used the CBU as our gateway, which controlled the respective EIB appliances (blinds and lighting).

C. Authentication and Authorization

The authentication service was based on an X.509 architecture [55]. The distribution of the key pairs had to be secured by using public key methods to avoid potential leaks in the security concept. The registration and authorization service was supported by an openLDAP infrastructure, which provided a service to register and configure the roles of the different participating devices. The registration of the role and security properties of all devices was stored in an XML configuration file.

Again, the communication between registered users, sensors or gateways is based on the OPC-UA protocol. For legacy devices, encrypted messages can be sent to a gateway by using the OPC-UA communication protocols and interfaces. The gateway can then make a lookup in internal lists or at the registrar in order to decide whether or not to accept the communication from the device. Thus, only messages are accepted, which can be identified by authorized devices.

VIII. VALIDATION

To develop a dependable system, it is a basic precondition to use well established and standardized methods for verification and validation. These methods are based on several different standards, e.g., IEC 61508 [45]. In this paper we concentrate on the validation steps of the ROFCO project. The validation strategy is based on pre-defined use cases, derived from the HA/BA applications Lighting Control and Blinds Control (see Section III). During the course of the project these use cases were adapted to needs and requirements. Thus we have achieved an iterative product life cycle process during the project lifetime in order to enhance the quality of the DHF. The requirement engineering process and the product life cycle process are based on the ISO/IEC 12207-2008 standard [56].

A. Validation Process

As mentioned in Section III, user stories have been used to describe the use case in such a way, that all stakeholders could understand the requirements and the interaction with the DHF. For requirement gathering the verbal description of the use case and the discussion with the stakeholders improved the understanding for the developers. Like in an agile software development process, each single use case had to be validated. Based on the verbal description and the UML Use Case Diagram of each use case we defined the respective test cases. Each test case definition contains attributes, such as verbal test description, pre-conditions, post-conditions, and planned test results, as defined in [57]. The actual test results have been documented in the ROFCO deliverables.

For the exemplary test case Light Maintenance, which is derived directly from the respective use case Light Maintenance as described in Section III, the test case definition looks as follows:

- Test case description: This test case validates the use case Light Maintenance. Thereto the respective application Light Maintenance has to be invoked within the DHF, the status of the lighting in the configured area has to be received, different values (on/off or dimming values) have to be set for single lights and defined groups of lights.
- Pre-conditions: The whole DHF system is installed, the lighting system is installed and configured, the Light Maintenance application is running
- Post-conditions: The Light Maintenance application is still running within the DHF (such that it is possible to re-start this test case several times)
- Planned test results: Status of the lighting is shown correctly, the on/off switches and the dimming controls work correctly for single lights and groups of lights

The test cases form the building blocks of the whole validation process. They have been used in different phases of the validation: During the pre-tests, they helped to identify and fix some misconfigurations in the controller setup and the network configuration. During the final validation trial of the prototype at the Techno-Z Salzburg (see Section V), they have been used to validate

- the control functionality of examined appliances,
- the interworking of the different proprietary HA/BA subsystems, and
- the robustness of the infrastructure and services.

Timing constrains and time criticalities have not been explored so far, yet for future research activities it will be important to address these topics in order to ensure the practical use of the DHF.

B. Validation Results

For both parts of the validation process (pre-tests at module test and integration test level and validation trial at system test level) standardized sets of test cases (“test suites”) have been defined and executed. The test suites have been defined for different parts of the system development process, and are thus constituting an accompanying test process:

- Validation of the developed software components
- Validation of the installed network components
- Validation of the network communication protocols

For instance, the following test suites have been defined for the validation of the network communication protocols:

- Tests of static and dynamic address configurations
- Tests of different routing configurations
- Network link availability tests
- Network device availability tests
- End device reachability tests

Single test cases can be part of one or more of these test suites. For instance, the exemplary test case Light Maintenance is part of the end device reachability test suite. For each test suite, all listed test cases have been executed at least once. If the actual test results were consistent with the planned test results, the test verdict was set to pass, otherwise fail. The test verdict for the whole test suite was set to pass, if and only if all test cases of the test suite achieved positive test verdicts.

Whereas during the pre-test some of the test suites failed, i.e., the respective functionality had to be fixed, the final validation trial yielded only positive test verdicts. Thus, the validation process showed the feasibility of our approach as expected. The interworking of heterogeneous building automation systems based on our X-Model is therefore a potential solution of the mentioned interoperability problems; yet further validation steps are still to be done: First of all, test cases concerning performance and timing issues should be identified and conducted in order to validate the real time capability of the DHF. Furthermore, the definition and execution of test cases derived from the HA/BA application Evacuation Support would help to validate the dependability of the system under test.

IX. Conclusion and Further Work

As a result of our validation trial, we proved the feasibility of our approach, as we were able to access the control devices using different OPC-UA clients. We were able to implement getter and setter functions for the data points of lighting and blinds control in different building units. Furthermore, we developed a dependability concept based on availability calculations according to IEC 61508 [45] functional safety standard and assessed the system relevant risks with an FMEA (see Section VI).

A possible barrier for a wide adoption of our approach in future commercial solutions, especially for the smaller scaled HA market, are the relatively high requirements on the used devices. In order to be able to proceed all the session and rights management data as well as the OPC-UA stack the devices need a certain minimum of computational power; for practical reasons this can not be guaranteed in all cases. Here this can be counteracted by the use of gateways to those legacy systems, which are not able to implement a native OPC-UA connection, yet this limits the beneficiaries of our system to a more narrow system border. However, future developments have to be observed accurately, as the progress of computational power in embedded devices may make this drawback obsolete in a few years. Especially, the market spread of smart phones, which may serve as control devices and user interfaces, brings new chances to HA solutions on IP basis. Another open issue is the influence of the building of communities of households, which will need further research (see Section X). For instance, community based energy optimization applications, but also regulatory aspects, e.g., the EU directive to install smart meters in households (Directive on internal markets 2009/72/EC [58]), may have market implications regarding the use of comprehensive HA systems.

As indicated in [1], the integration of HA/BA appliances with Smart Grids is the main topic of our future research activities. We have just started to test the integration of our X-Model in Smart Grid environments by using Smart Grid applications (like demand response, energy monitoring and health monitoring) with our system approach; yet the challenge will be to ensure the interoperability and collaboration of several HA/BA systems in bigger communities in order to ensure optimization at different scales. Thereto more efforts will be necessary to provide unique control architectures and generic interfaces; moreover the algorithmic side of system optimization (e.g., regarding energy efficiency) has to be addressed in our further research.

Other potential research activities could deal with topics like security and safety. Security will become an even bigger issue than now for two reasons: First, openness requires security means to avoid misuse, and besides all barriers we expect open solutions to spread more widely in future; second, the trend to build communities leads to larger systems with more participants (stakeholders), which exchange privacy and security sensitive data. Safety is already a big issue in BA; if safety solutions get affordable and technically realizable in HA environments, a spread to this market segment is foreseeable, thus research has to deal with this topic.

Industrial solutions can be expected for Mini-SCADA systems on top of dependable frameworks, which do not only provide a one-stop-shop for HA/BA control functionality to the user, but also an easy to use Human Machine Interface (HMI) in order to further increase user-friendliness of HA/BA control. The integration of IP as convergence layer for HA/BA systems is widely accepted in industry now, yet the openness of middleware functions upon IP is still an open issue. Here, the standardization bodies like CEN or ISO are requested to define open standards, which are accepted by the industry; this process is far from being finished.

ACKNOWLEDGMENTS

The work described in this paper was conducted during the project "Robust Facility Communication" (ROFCO), which was funded by the Austrian Federal Ministry for Transport, Innovation and Technology (BMVIT), and in the "Josef Ressel Zentrum for User-Centric Smart Grid Privacy, Security and Control", which is funded by the Austrian Federal Ministry of Ecomomy, Family and Youth (BMWFJ).

REFERENCES

[1] A. Veichtlbauer, T. Pfeiffenberger, and U. Schrittesser, "Generic control architecture for heterogeneous building automation applications," in *Proceedings of the 6th International Conference on Sensor Technologies and Applications (SensorComm 2012)*, Rome, August 2012, pp. 148–153.

[2] K. Charatsis, A. Kalogeras, M. Georgoudakis, J. Gialelis, and G. Papadopoulos, "Home / Building Automation Environment Architecture Enabling Interoperability, Flexibility and Reusability," in *Proceedings of the IEEE International Symposium on Industrial Electronics 2005 (ISIE 2005)*, vol. 4, Jun. 2005, pp. 1441–1446.

[3] F. Ferreira, A. Osorio, J. Calado, and C. Pedro, "Building Automation Interoperability – A Review," in *Proceedings of the 17th International Conference on Systems, Signals and Image Processing (IWSSIP 2010)*, 2010, pp. 158–161.

[4] M. Ciesielska and F. Li, "The connected home: From market barriers to business model solutions," in *Building the e-World Ecosystem*, ser. IFIP Advances in Information and Communication Technology, T. Skersys, R. Butleris, L. Nemuraite, and R. Suomi, Eds. Springer Verlag, Oct. 2011, vol. 353, pp. 189–199.

[5] *Smart Grid Reference Architecture*, CEN/Cenelec/ETSI Smart Grid Coordination Group Std., Nov. 2012.

[6] *ISO/IEC 7498-1:1994 Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*, International Standards Orginization (ISO) Std., 1994, Accessed: 2013-02-25. [Online]. Available: http://www.iso.org/iso/catalogue_detail. htm?csnumber=20269

[7] J. Postel, *Internet Protocol – DARPA Internet Program Protocol Specification*, RFC 791, IETF Std., Sep. 1981.

[8] Salzburg Research Forschungsgesellschaft. (2012) ROFCO – Robust Facility Communication. Accessed: 2012-06-19. [Online]. Available: http://www.salzburgresearch.at/en/projekt/rofco_en/

[9] Salzburg University of Applied Sciences. (2013) en-trust – Josef Ressel Center for User-Centric Smart Grid Privacy, Security and Control. Accessed: 2013-02-04. [Online]. Available: http://www.en-trust.at/

[10] A. Veichtlbauer, D. Engel, F. Knirsch, O. Langthaler, and F. Moser, "Advanced metering and data access infrastructures in smart grid environments," in *Proceedings of the 7th International Conference on Sensor Technologies and Applications (SensorComm 2013)*, Barcelona, Aug. 2013, (accepted).

[11] M. S. Jimenez, *Smart Grid Mandate*, European Commission Directorate-General for Energy Std., 2012.

[12] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. Vol. 1 No.1, pp. 11–33, 2004.

[13] J. P. Thomesse, "Fieldbuses and interoperability," *Control Engineering Practice*, vol. 7, iss. 1, pp. 81–94, Jan. 1999.

[14] E. Finch, "Is IP everywhere the way ahead for building automation?" *Facilities*, vol. 19, iss. 11/12, pp. 396–403, 2001.

[15] Siemens AG. (2011) Total Building Solutions für intelligente Gebäude – Siemens Building Technologies. Accessed: 2012-06-19. [Online]. Available: http://www.industry.siemens.de/buildingtechnologies/de/de/total-building-solutions/Seiten/total-building-solutions.aspx

[16] ABB Asea Brown Boveri Ltd. (2012) Raumtalk – Building Automation over IP. Accessed: 2012-04-11. [Online]. Available: http://www.abb.at/cawp/deabb201/24d156e58bc98443c125720b0025238d.aspx

[17] W. Granzer, W. Kastner, G. Neugschwandtner, and F. Praus, "A Modular Architecture for Building Automation Systems," in *Proceedings of the 6th IEEE International Workshop on Factory Communication Systems (WFCS '06)*, Jun. 2006, pp. 99–102.

[18] American Society of Heating, Refrigerating and Air-Conditioning Engineers Inc., "BACnet - A Data Communication Protocol for Building Automation and Control Networks," ANSI/ASHRAE Standard 135-2004, 2004.

[19] D. Snoonian, "Smart buildings," *Spectrum, IEEE*, vol. 40, pp. 18–23, Aug. 2003.

[20] U. Schrittesser, "Synthese von redundanten vermaschten wlan," Master's thesis, Salzburg University of Applied Sciences, Jun. 2008, in German.

[21] CAS. (2010) OPC Unified Architecture. Accessed: 2012-06-19. [Online]. Available: http://www.commsvr.com/UAModelDesigner/Index.aspx

[22] W. Mahnke, S.-H. Leitner, and M. Damm, *OPC Unified Architecture*. Springer-Verlag Berlin Heidelberg, 2009.

[23] J. Postel, *Transmission Control Protocol – DARPA Internet Program Protocol Specification*, RFC 793, IETF Std., 1981.

[24] A. Fernbach, W. Granzer, and W. Kastner, "Interoperability at the Management Level of Building Automation Systems: A Case Study for BACnet and OPC UA," in *Proceedings of the 16th IEEE Conference on Emerging Technologies and Factory Automation (ETFA '11*, Sep. 2011.

[25] RESI Informatik & Automation GmbH. (2013) Resi SCADA 2D. Accessed: 2013-02-25. [Online]. Available: http://www.resi.cc/wordpress/prestashop/product.php?id_product=59

[26] ETM Professional Control GmbH. (2013) Simatic WinCC Open Architecture. Accessed: 2013-02-25. [Online]. Available: http://www.etm.at/index_e.asp?id=2&sb1=&sb2=&sb3=&sname=&sid=&seite_id=6

[27] Cooper Power Systems. (2010, Oct.) Mini-SCADA solution. Accessed: 2013-02-04. [Online]. Available: http://www.cooperindustries.com/content/dam/public/powersystems/resources/library/1100_EAS/B110007341.pdf

[28] P. E. Rovsing, P. G. Larsen, T. S. Toftegaard, and D. Lux, "A reality check on home automation technologies," *Journal of Green Engineering*, pp. 303–327, 2011.

[29] M. Tariq, Z. Zhou, J. Wu, M. Macuha, and T. Sato, "Smart grid standards for home and building automation," in *Proceedings of the 2012 IEEE International Conference on Power System Technology (POWERCON 2012)*, 2012.

[30] T. S. Hjorth and R. Torbensen, "Trusted domain: A security platform for home automation," *Computers & Security*, vol. 31, no. Issue 8, pp. 940–955, Nov. 2012.

[31] W. Granzer, W. Kastner, G. Neugschwandtner, and F. Praus, "Security in Networked Building Automation Systems," in *Proceedings of the 6th IEEE International Workshop on Factory Communication Systems (WFCS '06)*, Torino, Jun. 2006, pp. 283–292.

[32] C. Probst and A. Veichtlbauer, "Security Features of a Generic Sensor Data Acquisition System," in *Proceedings of the 6th International Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet 2010)*, Bodrum, Turkey, Oct. 2010, pp. 78–81.

[33] T. I. Salsbury, "A Survey of Control Technologies in the Building Automation Industry," in *Proceedings of the 16th IFAC World Congress*, vol. 16, part 1, Prague, Czech Republic, Jul. 2005.

[34] S. Makarechi and R. Kangari, "Research methodology for building automation performance index," *International Journal of Facility Management*, vol. 2, no. 1, 2011.

[35] A. Veichtlbauer and T. Pfeiffenberger, "Dynamic evacuation guidance as safety critical application in building automation," in *Proceedings of the 6th International Conference on Critical Information Infrastructure Security (Critis 2011)*, Lucerne, Switzerland, Sep. 2011.

[36] C. Probst, "Konzeptionierung eines Benutzermanagements für den Zugriff auf vertrauliche Daten von IP fähigen Sensornetzen," Master's thesis, University of Applied Sciences Salzburg, May 2010, in German.

[37] International Telecommunication Union. (2008) X.501. Accessed: 2012-06-19. [Online]. Available: http://www.itu.int/rec/T-REC-X.501

[38] C. Rigney, A. C. Rubens, W. A. Simpson, and S. Willens, *Remote Authentication Dial In User Service (RADIUS)*, RFC 2865, IETF Std., Jun. 2000.

[39] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, *Diameter Base Protocol*, RFC 3588, IETF Std., Sep. 2003.

[40] Cisco Systems, Inc. (2013, Feb.) Cisco Systems, Inc. Cisco Systems, Inc. Accessed: 2013-02-25. [Online]. Available: http://www.cisco.com/

[41] R. Baumann, S. Heimlicher, M. Strasser, and A. Weibel, "A survey on routing metrics," Computer Engineering and Networks Laboratory, ETH-Zentrum, Switzerland, Tech. Rep., Feb. 2007, accessed: 2012-06-19. [Online]. Available: http://www.baumann.info/public/tik262.pdf

[42] S. Shao, M. Pipattanasomporn, and S. Rahman, "Grid integration of electric vehicles and demand response with customer choice," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 543–550, Mar. 2012.

[43] U. Greveler, B. Justus, and D. Löhr, "Multimedia content identification through smart meter power usage profiles," in *Proceedings of the 2012 International Conference on Information and Knowledge Engineering (IKE'12)*, Las Vegas, USA, Jul. 2012, pp. 383–390.

[44] G. Panholzer, A. Veichtlbauer, P. Dorfinger, and U. Schrittesser, "Simulation of a robust communication protocol for sensor data acquisition," in *Proceedings of the 6th International Conference on Wireless and Mobile Communications (ICWMC 2010)*, Valencia, Spain, Sep. 2010, pp. 145–150.

[45] *Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems*, International Electrotechnical Commission (IEC) Std., Apr. 2010.

[46] K. Werthschulte, "Integration von heterogenen Bussystemen in die Heimautomatisierung unter Verwendung von Middleware," Ph.D. dissertation, Technical University Munich, 2003.

[47] C. Busemann, C. Kuka, U. Westermann, S. Boll, and D. Nicklas, "Scampi – sensor configuration and aggregation middleware for multi platform interchange," in *Proceedings of the 39th Annual Conference of the Society for Informatics*, Lübeck, 2009.

[48] Ing. Punzenberger COPA-DATA GmbH. (2013) HMI SCADA Software zenon by COPA-DATA. Accessed: 2013-02-25. [Online]. Available: http://www.copadata.com/en/home.html

[49] Quanmax AG. (2013) MF Security Gateway. Accessed: 2013-02-25. [Online]. Available: http://www.underground8.com/de/products/mf_security_gateway.html

[50] cTrixs GmbH. (2012) CBU cTrixs Base Unit. Accessed: 2013-03-14. [Online]. Available: http://www.ctrixs.com/systemubersicht

[51] M. Melik-Merkumians1, T. Baier, M. Steinegger, W. Lepuschitz, I. Hegny, and A. Zoitl, "Towards OPC UA as portable SOA Middleware between Control Software and External Added Value Applications," in *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies and Factory Automation (ETFA 2012)*, Krakow, Sep. 2012.

[52] T. Tietjen, D. Müller, and A. Decker, *FMEA Praxis – Das Komplettpaket für Training und Anwendung*, 3rd ed. Carl Hanser Verlag, Mar. 2011, in German.

[53] OPC Foundation. (2012) OPC – The Interoperability Standard for Industrial Automation & Other. Accessed: 2012-06-19. [Online]. Available: http://www.opcfoundation.org

[54] *ISO/IEC 14543-3:2006 Information technology – Home Electronic Systems (HES) Architecture*, International Standards Orginization (ISO) Std., 2006, Accessed: 2013-03-15. [Online]. Available: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=43364

[55] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, RFC 5280, IETF Std., May 2008.

[56] *ISO/IEC 12207-2008 - ISO/IEC/IEEE Standard for Systems and Software Engineering - Software Life Cycle Processes*, IEEE Std., Jan. 2008, Accessed: 2013-02-21. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=4475822

[57] *IEEE 829-2008 - IEEE Standard for Software and System Test Documentation*, IEEE Std., Jul. 2008, Accessed: 2013-02-21. [Online]. Available: http://ieeexplore.ieee.org/servlet/opac?punumber=4578271

[58] The European Parliament and the Council of the European Union, "Directive 2009/72/ec," *Official Journal of the European Union*, vol. L 211, pp. 55–93, Aug. 2009. [Online]. Available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2009:211:0055:0093:EN:PDF

A Study of the Performance of Cooperative Caching in Static Ad Hoc Networks

Francisco J. González-Cañete and Eduardo Casilari
Department of Electronic Technology
University of Málaga
Málaga, Spain
{fgc, ecasilari}@uma.es

***Abstract*—In this paper, we evaluate the performance of the CLIR (Cross-Layer Interception and Redirection) cooperative caching scheme for ad hoc networks in static grid ad hoc networks. This caching scheme implements a local cache in every node of the network allowing the nodes to work as request interceptors for the rest of the nodes. In addition, it also implements a redirection cache that stores information about the location of the documents in the network in order to redirect the requests to nodes that are situated closer than the servers. Finally, a piggy-backing technique is incorporated to the routing protocol with the aim of finding the documents in the network while the routes to the servers are created. By means of simulations, we evaluate the mean traffic generated in the wireless network, the delay perceived by the users, the percentage of failed searches, the mean number of retrieved documents, the local and remote cache hits and the mean percentage of redirection hits as a function of the mean time between requests, the Time To Live (TTL) of the documents, the traffic pattern and the cache sizes. We compare the performance of our proposal with another five cooperative caching schemes as well as the option of no using a caching scheme. The simulation results show that our proposal outperforms the other caching schemes in terms of the studied parameters. In addition, we compare the redirection caching scheme of our proposal to the redirection policies implemented by other caching schemes.**

Keywords-cooperative caching; grid; ad hoc network; redirection cache.

I. INTRODUCTION

The CLIR (Cross-Layer Interception and Redirection) caching scheme was evaluated in static grid Ad Hoc networks in [1]. Therefore, the aim of this work is to extend this performance evaluation by analysing new metrics: the mean number of retrieved documents, the percentage of cache hits (local and remote). This will offer a more concise comparison of the performance of CLIR to other caching schemes. Additionally, the redirection technique will be also evaluated as a metric specifically developed to measure its performance called redirection cache hit.

The aim of a caching scheme is to reduce the traffic generated in the network, as well as the delay perceived by the users and the servers' load [2]. The reduction of the traffic in a wireless network also decreases the probability of collisions and interferences, and hence, the probability of packet loss. Reducing the delay perceived by the users when they request documents improves the user experience and makes the network more attractive to be used. Finally, as a consequence of the caching mechanism, the document requests can be served by other nodes in the wireless network instead of the servers. In a very loaded network, the servers could be a bottleneck as all the requests are sent to them. The caching mechanism mitigates this effect by moderating the overload of the servers so they can reply more requests.

MANETs (Mobile Ad Hoc NETworks) [3] were proposed as a solution for deploying communication applications in places where a wired network was not available. Unfortunately, they have some limitations:

- Restricted hardware capabilities. Some light weight devices are constrained in their processing and computing capabilities.
- Limited batteries. Mobile devices operate with batteries. In order to maximize their lifetimes, the number of messages that they generate should be moderated.
- Scarce bandwidth. Wireless medium has restricted bandwidth so signaling traffic should be minimized.
- Temporary connection to external networks. The integration of MANET into external networks is guaranteed through Gateways. However, the mobility of the MANET may provoke the Gateway to be temporarily unavailable.

Although many cooperative caching schemes have been proposed for MANETs, they have not been evaluated for static ad hoc network, that is, wireless networks where the nodes do not move. The objective of this work is to evaluate the performance of different caching schemes proposed for MANETs in static grid network scenarios.

The rest of this document is organized as follows. In Section II, the related work about cooperative caching schemes for MANETs is presented. In Section III, the proposed caching scheme is described. Section IV defines the system model and shows the performance evaluation of the caching schemes. Finally, Section IV enumerates the main conclusions of this work.

II. RELATED WORK

The cooperative caching schemes for ad hoc networks can be classified into four groups: broadcast-based, information-based, role-based and direct-request. The broadcast-based caching schemes employ broadcast messages as the first choice in order to find the documents in the network. These broadcast messages can be sent to the entire network, as in the case of MobEye [4]. Other schemes such as SimpleSearch [5], follow a more restrictive approach that limits the distance of the messages to four hops. ModifiedSS [6] is an evolution of SimpleSearch that employs GPS (Global Positioning System) in order to send the requests to the direction where the servers are located. Similarly, the caching scheme proposed by Moriya in [7] sends the broadcast messages to the neighbourhood so that, if the document is not found, the request is transmitted to the server.

The information-based cooperative caching schemes employ information of the location of the documents in the network. Nodes obtain this information by analysing the messages that they forward. As examples of this category of caching schemes we can mention: DGA (Distributed Greedy Algorithm) [8], Wang [9], Cho [10] and POACH (POware Aware Caching Heuristic) [11].

Under a role-based caching scheme, each node in the wireless network has a predefined role. That is, they can be caching nodes, requesting nodes, coordinator nodes, gateway nodes, etc. The role-based caching schemes are usually applied to cluster networks. CC (Cluster Cooperative) [12] and Denko [13] are examples of this kind of caching policy.

Finally, the direct-request caching schemes directly send the requests to the server with the hope of being served by an intermediate node in the route from the requester to the server. The proposal by Gianuzzy in [14] is an example of this kind of caching schemes.

However, the groups in this classification of caching schemes are not mutually exclusive. Thus, the caching schemes COOP [15], ORION (Optimized Routing Independent Overlay Network) [16], IXP/DPIP (IndeX Push/Data Pull/Index Push) [17] and COCA (COoperative CAching) [18] are schemes that employ network information and broadcast requests. On the other hand, COACS (Cooperative and Adaptive Caching System) [19] and GROCOCA (GROup-based COoperative CAching) [20] are role-based caching schemes that also utilize information obtained from the network. In addition, CacheData, CachePath, HybridCache [21] and GroupCaching [22] are direct-request caching schemes that also employ the location information. Finally, ZC (Zone Cooperative caching) [23] and Sailhan [24] use direct requests and broadcast requests depending on some heuristic.

The CLIR cooperative caching scheme was proposed in [25]. It can be classified as a direct-request and information-based cooperative caching scheme. The main novelty of CLIR is the implementation of a cross-layer interception cache technique as well as the optimization of the redirection technique. Its performance was evaluated for MANETs and compared to other five cooperative caching schemes. The objective of this paper is to study the performance of CLIR in a static grid ad hoc network and compare this performance with other caching schemes.

Figure 1 summarizes the classification presented in this section.

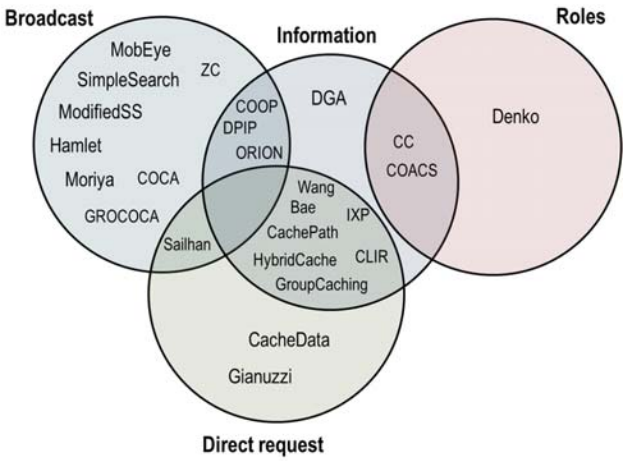


Figure 1. Cooperative caching schemes classification.

III. CACHING SCHEME

The proposed caching scheme works using a request/reply protocol very similar to HTTP (Hyper Text Transfer Protocol) as utilized in the caching schemes proposed in Section II. Consequently, the nodes request documents (information, data, etc.) to the data server. The data servers, as the HTTP servers, reply with a message containing the document requested. The data servers can be other nodes in the wireless ad hoc network or even external servers that are accessed through a Gateway.

CLIR implements a local cache in every node in the network. This local cache is managed using the LRU (Least Recently Used) replacement policy. Using this cache, every node stores the received documents. Therefore, further requests to the same document will be resolved by the local cache. This is called a local cache hit. As the requests must be forwarded hop by hop from the requester node to the server node, the intermediate nodes in the route from the source to the destination of the requests can reply directly if the requested document is stored in their local cache. This is called an interception cache hit.

When the route from the source node of the request to the destination node has not been created, CLIR utilizes the routing protocol to piggy-back the request in the routing protocol messages. By using this technique, the routing protocol is able to create the route to the destination node and search for the requested document at the same time. If any node that receives the route request message has a copy of the requested document in its local cache, it will reply using the route reply message informing that this node has a copy of the document. When the requesting node receives the route reply message, the route between both nodes has just been created so the requester will forward the request to

the node that has the copy of the document. This is called a cross-layer interception hit. This mechanism allows finding the documents in the network even if the servers are temporarily unavailable. On the other hand, it also finds the documents in nodes located closer than the servers, reducing the delay and the network load as the number of messages sent are reduced. This kind of technique requires the implementation of an on demand routing protocol as the piggybacking messages are sent on every document request if a route to the destination is not available at this moment.

CLIR also implements a redirection cache that stores information about where the documents are located in the network. In order to obtain this information, the nodes analyse the request and reply messages they forward. The redirection cache manages information about the source of the requests and the corresponding replies. It also stores the number of hops and the TTL of the documents. This TTL is employed to set the validity of the information stored in the redirection cache as the documents will be obsolete after this time. Additionally, the redirection cache also takes into consideration the mean time the documents are stored in the caches in order to avoid the redirection of a request to a node that has evicted the document from its local cache. Consequently, the redirection cache estimates the time that the documents are stored in the local caches calculating the mean time the documents are stored in its own local cache. Moreover, the expiration time assigned to the information stored in the redirection cache is the minimum between this estimated time and the TTL of the document.

The redirection cache is managed by means of two LRU lists, one for the information of those documents whose TTL is known (called *KNOWN_TTL_LIST*) and the other with the documents with an unknown TTL (called *UNKNOWN_TTL_LIST*). The TTL of a document is unknown until the reply is forwarded by the node. The memory for each structure is dynamically assigned although the memory space reserved for both structures is set to a constant. The information obtained from the messages adds or updates the data to the lists. If an entry of the *UNKNOWN_TTL_LIST* structure is updated with the TTL of a document, it will be transferred to the head of the *KNOWN_TTL_LIST*. When a new entry must be stored and there is not enough room because the reserved storage space is full, the oldest entry in the *UNKNOWN_TTL_LIST* structure is evicted. If this list is empty, the oldest entry in *KNOWN_LIST_LIST* will be deleted. An entry is also deleted if the information is obsolete because all the associated TTLs have expired.

When a node receives a request and the redirection cache contains information of a node that is closer to the original destination of the request, the request is forwarded to this closer node. When the node to which the request has been redirected receives the message, it replies with the document. This is called a redirection cache hit. In the case that the redirected node has evicted the document from its local cache, a redirection error message is sent to the redirection node in order to update the information of the redirection cache.

Finally, CLIR also implements the storage of the replied document in the node located in the middle of the route from the source and destination of the reply proposed in [9]. So, the documents can be easily disseminated along the network. In order to avoid the excessive replication of documents, this mechanism is performed if the distance between both nodes is greater than four hops.

For more details on the implementation of CLIR refers to [25].

IV. PERFORMANCE EVALUATION

In order to evaluate the performance of the proposed cooperative caching scheme we have implemented CLIR using the NS-2.33 [26] network simulator. Additionally, for comparison purposes, the cooperative caching schemes MobEye, HybridCache, COOP, DPIP and SimpleSearch have also been implemented. The no cache (NC) option has also been taken into consideration, that is, the case where no caching scheme is considered. Every point represented in the figures shown in this paper corresponds to the mean obtained value of five simulations using the same parameters but changing the seed. Depending on the simulation, the analysed variable is changed while the rest of the parameters are set to a default value. All figures include a confidence interval of 95% for each performance parameter.

A. Simulation model

Table I summarizes the main simulation parameters. We suppose that the nodes in the ad hoc network do not move. Depending on the evaluated configuration, nodes form a regular grid of 5x5, 7x7 or 9x9 nodes. Moreover, the nodes located in the corners of the simulation area, that is, in the positions (x,y)=(0,0) and (x,y)=(1000,1000), are considered to behave as Data Servers (*DS*). There are 1000 different documents stored in the *DSs*. For simulation simplicity, we have considered a numeric identification for each document although the caching scheme can be extended to manage complete URLs. In order to distribute the traffic along the network, the documents with even identification are located in one server while the documents with odd identification are stored in the other *DS*. In addition, every document has an associated TTL (Time To Live) that determines when it expires, and hence, it is considered obsolete. The expired documents stored in the local caches are deleted in order to free storage space. The TTL of the documents follows an exponential distribution with a mean time between 250 and 2000 seconds. In this way, low and high TTL variability is modeled. Moreover, the infinite TTL is also analyzed, that is, the case where the documents never expire. Additionally, we consider the size of the documents to be constant and equal to 1000 bytes.

Every node that is not a server is programmed to generate requests to the servers during the simulation time. When a request is served, another request is generated after a waiting time period. If the request is not served after a

predefined timeout, the request is sent again. The waiting time between requests follows an exponential distribution with mean values between 5 and 50 seconds (the default value is 25 seconds). If the waiting time is modified, a wide range of nodes activity can be evaluated and, consequently, the traffic load of the network. The documents that are not served before the timeout is triggered are requested again. This timeout is defined to be constant with a value of 3 seconds.

TABLE I. SIMULATION PARAMETERS

Parameter	Default values	Other utilized values
Simulation area (square meters)	1000x1000	
Number of nodes	49	25-49-81
Number of Servers	2	
Number of documents	1000	
Document size (bytes)	1000	
Timeout (s)	3	
TTL (s)	2000	250-500-1000-2000-∞
Mean time between requests (s)	25	5-10-25-50
Traffic pattern (Zipf slope)	0.8	0.4-0.6-0.8-1.0
Replacement policy	LRU	
Local Cache size (number of documents)	35	5-10-35-50
Redirection Cache size (number of registers)	35	
Simulation time (s)	20000	
Warm-up period (s)	4000	
Coverage radio (meters)	250	
MAC Protocol	802.11b	
Radio propagation model	Two Ray Ground	
Routing protocol	AODV	

The document request pattern follows a Zipf-like distribution, which has been demonstrated to properly characterize the popularity of the documents in the Internet [27]. The Zipf law asserts that the probability *P(i)* for the *i*-th most popular document to be requested is inversely proportional to its popularity ranking as shown in (1).

$$P(i) = \frac{\beta}{i^{\alpha}} \quad (1)$$

The parameter α is the slope of the log/log representation of the number of references to the documents as a function of its popularity rank (i). Values between 0.4 and 0.8 have been selected for the Zipf slope (with a default value of 0.8) in order to model low and high temporal locality.

Every node in the network implements a local cache that employs the LRU replacement policy. The default cache size allows storing 35 documents (35000 bytes). Cache sizes of 5, 10 and 50 documents have also been considered. In order to avoid cache misses due to the emptiness of the caches at the start of the simulation [28], a warm-up time has been considered using the first 20% of the simulation time. As the simulation time has been set to 20000 seconds, the warm-up time has a value of 4000 seconds. During the warm-up period, the performance metrics are not computed. Consequently, the analyzed statistics correspond to the time after the warm-up time. The redirection cache has the capacity of storing 35 registers.

As the coverage radio of the nodes is 250 meters and the simulation area is 1000x1000 m^2, the connectivity among neighbour nodes is different for each evaluated grid configuration. Figure 2 shows the connectivity for the 5x5, 7x7 and 9x9 grid configurations. As it can be observed, as the density of nodes increases, the number of neighbour nodes grows.

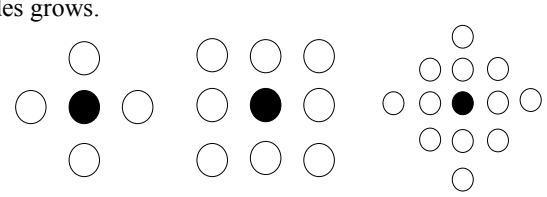


Figure 2. One hop connectivity of a node for 5x5, 7x7 and 9x9 grids.

The parameters employed in the rest of the evaluated caching schemed (HybridCache, DPIP, SimpleSearch and MobEye) are those proposed by their authors.

As performance metrics we consider:

- Traffic load: It measures the mean amount of traffic generated or forwarded by each node during the simulation. As the wireless medium is limited, the greater the generated traffic the greater the probability of interferences and collisions.
- Delay: It is defined as the mean time that a request requires to be served, that is to say, the mean time that a user will have to wait to receive the requested document.
- Timeouts: This metric defines the percentage of requests that have failed and have been requested again because the document has not been received before the timeout.
- Number of served documents: It is defined as the mean number of documents that have been retrieved to the nodes of the network during the simulation time. This metric shows the caching schemes that can serve more documents in a predefined time.
- Cache hits: They can be divided into local cache hits and remote cache hits. The local cache hit measures the percentage of requests served by the local cache. A local cache hit implies a zero delay, avoids the traffic generation in the network and reduces the servers load. The remote cache hit measures the percentage of requests served by network nodes different from the servers. A remote cache hit implies the reduction of the servers load. Depending on the cache scheme, the remote cache hit can be categorized into other classes. Taking into consideration the CLIR caching scheme, interception, cross-layer interception and redirection cache hits are considered as remote hits.
- Redirection hits: It is defined as the percentage of redirections that have been correctly resolved, i.e.,

the document has been served to the requester after a redirection. This metric measures the goodness of the redirection cache technique for those caching schemes that implement it.

The figures presented in this section correspond to the evaluation of a 7x7 grid network as the results obtained with the 5x5 and 9x9 networks are very similar. The performance evaluation will be studied as a function of the time between requests, the TTL of the documents, the Zipf slope and the local cache size. The redirection hit metric will be applied only to those caching schemes that implement the redirection technique, that is, CLIR, COOP and HybridCache.

B. Time between requests

Figure 3a represents the mean processed traffic by each node as a function of the time between requests. CLIR, DPIP and HybridCache are the caching schemes that generate the lowest traffic, followed by No Cache and SimpleSearch. MobEye generates more traffic because of the use of broadcast messages.

Figure 3b compares the mean delay of the requests and replies. CLIR is the caching scheme with the lowest delay. In fact, it is the only scheme that obtains a lower delay than the option of not using caches. SimpleSearch and MobEye employ a four request-reply messages method, and hence, they experience a greater delay and a greater traffic generation as previously observed. COOP has not been shown in this figure due to the high delay obtained. This behaviour is caused by the timeout needed to perform the direct request to the *DS* after the broadcast request has failed. DPIP also achieves a high delay due to the *DPIP_Timer* parameter that fixes a lower bound to the messages delay. Finally, HybridCache achieves a low performance for high loaded networks although this performance is improved as the traffic load is decreased. This fact is due to redirection loops caused by a wrong redirection management (Figure 3f). When time between requests increases, the information stored in the redirection table is obsolete related to the documents stored in the local caches as they are evicted from the local caches before the information can be considered obsolete. As the number of evictions in the local caches decreases the redirection cache is able to obtain more redirection hits because it only takes into account the TTL of the documents to delete the information of the redirection cache.

Figure 3c shows the mean percentage of timeouts per node. HybridCache obtains a high percentage of timeouts due to the bad redirection management as previously explained. Similarly, COOP presents the same behaviour as HybridCache because of the same reasons. Finally, the rest of the caching schemes obtain a percentage of timeouts close to zero. In fact, this should be the normal behaviour of the caching schemes as the servers are always available and it is always possible to create a route to them.

Figure 3d illustrates the mean number of documents received per node. All the compared caching schemes obtain a similar performance except COOP and, especially, HybridCache for times between requests lower than 25 seconds. For high loaded networks (5 or 10 seconds between requests), HybridCache achieves a very low performance, obtaining only a half of the documents compared to the rest of the caching schemes due to the timeouts (Figure 3c) and redirection errors (Figure 3f).

Figure 3e depicts the mean percentage of local and remote cache hits as a function of the mean time between requests. MobEye is the caching scheme that obtains more hits, although its performance decreases as the time between requests increases. However, the rest of the caching schemes show the opposite behaviour. In SimpleSearch, the percentage of remote hits is practically zero for high loaded networks. As the requests are performed very close in time, the broadcast method is not employed because the route to the servers is already created, and hence, the requests are sent directly to them.

Figure 3f represents the mean percentage of redirection hits as a function of the mean time between requests. The performance of COOP is reduced as the traffic load is decreased because of the reduction of the redirection cache hits due to incorrect redirections of requests to nodes that have removed the documents from their local caches. On the other hand, CLIR obtains a performance close to 100% for all the traffic loads. Finally, HybridCache shows a redirection cache hit close to zero except for very low loaded networks where the performance reaches 5%.

C. TTL of the documents

Figure 4a represents the mean traffic processed by each node as a function of the mean TTL of the documents. CLIR, DPIP and COOP generate less traffic than no Caching for all the studied TTLs. HybridCache is very sensitive to the TTL of the documents and, as the TTL is increased, the generated traffic also soars. This behaviour is due to the redirection cache, which only takes into account this parameter to delete the information in the redirection cache. Consequently, if a node evicts a document from its local cache, the nodes with information about the location of this document in their redirection caches will maintain incorrect data.

Figure 4b compares the mean delay as a function of the mean TTL of the documents. CLIR is the caching scheme that obtains the lowest delay. HybridCache, as shown in the previous study, is very sensitive to the TTL and the delay is highly increased as the TTL is incremented. The rest of the caching schemes obtain delays greater than the case of no Caching due to the four messages needed to obtain the document.

Figure 4c shows the evolution of the percentage of timeouts as a function of the TTL of the documents. COOP and HybridCache are the caching schemes with a percentage of timeouts greater than zero due to the previously

commented reason. In fact, the percentage of timeouts is highly increased in HybridCache for TTLs greater than 2000 seconds.

Figure 4d illustrates the mean number of documents received per node as a function of the mean TTL of the documents. As it can be observed, a similar number of documents is obtained by all the caching schemes except COOP, with a slightly lower performance, and HybridCache with a drastically reduction of the retrieved documents with a mean TTL greater than 1000 seconds. This behaviour shows that HybridCache is very sensitive to the TTL of the documents because of a bad management of the redirection cache (as shown in Figure 4f). This figure reports that the redirection hits decrease until practically zero for a mean TTL of the documents greater than 1000 seconds. The incorrect management of the redirection cache also causes an increment of the traffic (Figure 4a), a greater number of timeouts (Figure 4b) and a higher delay (Figure 4c)

Figure 4e depicts the percentage of local and remote cache hits as a function of the mean TTL of the documents. As the mean TTL of the documents increases, they are stored for more time in the local caches; hence, the probability of a cache hit is also incremented. HybridCache obtains a good performance for low TTLs although this performance is drastically reduced when the documents do not expire. CLIR achieves a performance similar to the rest of the caching schemes and it is only overcome by MobEye and DPIP for high TTLs because of the use of broadcast message. On average, SimpleSearch is the caching scheme with the lowest cache hit ratio.

Finally, Figure 4f compares the percentage of redirection cache hits as a function of the TTL of the documents. As it was previously commented, HybridCache achieves a very high redirection hit rate for small TTLs. However, as the TTL increases, this rate drastically falls and reaches a percentage of zero for TTLs greater than 2000 seconds. On the other hand, it can be observed that COOP is also sensible with the TTL of the documents, as it obtains a lower percentage of redirection hits when they become obsolete more frequently. As the mean TTL increases, the redirection cache hit of COOP is also increased. Finally, CLIR achieves a percentage of redirection hits close to 100 % for all the studied TTLs.

D. Zipf slope

Figure 5a depicts the mean traffic processed by node as a function of the Zipf slope. CLIR is the caching scheme that obtains the lowest delay for all the slopes while MobEye and SimpleSearch generate more traffic than the No Caching option due to the broadcast requests. On the other hand, HybridCache also generates more traffic than the No Caching scheme for low slopes. This behavior is due to the replacement policy implemented by HybridCache, called SxO (Size x Order) [21]. This replacement policy is very sensitive to the popularity of the documents. Consequently, a low Zipf slope causes the reduction of the local cache hits, increasing the traffic generated in the network.

Figure 5b compares the mean delay as a function of the Zipf slope. The delay obtained by COOP is not shown because it is much greater than the rest of the caching schemes. Only CLIR and HybridCache (for a slope of 1.0) obtain a lower delay than the No Caching scheme. DPIP has a delay of even three times greater than CLIR although this difference is reduced as the Zipf slope increases. CLIR is the caching scheme with the lowest delay for all the considered Zipf slopes.

Figure 5c shows the mean percentage of timeouts per node as a function of the Zipf slope. As observed in previous studies, only HybridCache, COOP and MobEye present a percentage of timeouts different to zero. The behaviour of HybridCache and COOP is due to the incorrect implemented redirection technique. Nevertheless, the percentage of timeouts of these caching schemes is decremented as the Zipf slope increases because, as the Zipf slope increases, the percentage of local and remote cache hits increases and the documents can be served before the timeout. The rest of caching schemes obtain a percentage of timeouts close to zero.

Figure 5d illustrates the mean number of documents retrieved per nodes as a function of the Zipf slope. As observed in previous the studies, HybridCache achieves the lowest results, although this value is incremented as the Zipf slope is incremented because of the rise of the cache hits (Figure 5e). The errors produced by the redirection cache mechanism (Figure 5f) causes high occurrences of timeouts (Figure 5c), and hence, the reduction of the number of obtained documents. Similarly, COOP also achieves a lower performance than the rest of caching schemes. CLIR, DPIP, SimpleSearch and MobEye obtain a similar number of documents for all the studied Zipf slopes.

Figure 5e compares the percentage of local and remote cache hits as a function of the Zipf slope. The Zipf slope indicates the probability that the more popular documents will be requested more times. Hence, if the Zipf slope increases, the percentage of local cache hits is expected to increase too. This behaviour is confirmed in this figure. MobEye, as occurred in the previous studies, is the caching scheme that achieves a higher cache hit percentage because of the use of broadcast requests. CLIR presents a performance similar to the rest of the caching schemes.

Figure 5f depicts the percentage of redirection cache hits as a function of the Zipf slope. As observed in the previous studies, HybridCache obtains a redirection cache rate close to 0 %. COOP reduces the percentage of cache hits as the Zipf slope increases. On the contrary, CLIR achieves the same performance (close to 100 %) for all the studied values of the Zipf slope.

E. Cache size

Figure 6a depicts the mean processed traffic by the nodes as a function of the local cache size. As the cache size

rises the generated traffic is decreased because the probability of a local cache hit is increased. CLIR, DPIP and COOP are the caching schemes that generate a lower traffic than the No Caching scheme for all the studied cache sizes. MobEye is the caching scheme that generates more traffic due to the use of broadcast requests. On the other hand, HybridCache only performs better than No Caching when the cache size is greater than 20 documents. Hence, HybridCache does not work correctly when using small caches due to the implemented SxO replacement policy.

Figure 6b compares the mean delay as a function of the local cache size. CLIR is the caching scheme with the lowest delay and, in this case, is the one that performs better than the No Caching scheme for all the studied cache sizes. HybridCache presents a big delay for small caches, although it is drastically reduced as the cache size increases. In addition, SimpleSearch and MobEye always obtain a bigger delay than the No Caching scheme for all the studied cache sizes due to the four messages needed to obtain a document. Finally, DPIP shows a delay close to 150 milliseconds due to the limit imposed by the *DPIP_Timer*.

Figure 6c presents the mean percentage of timeouts as a function of the local cache size. As observed in previous studies, only HybridCache, MobEye and COOP show a percentage of timeouts different to zero. This percentage is reduced, especially in HybridCache, as the cache size increases because the probability of local and remote cache also augments.

Figure 6d shows the mean number of documents retrieved per node as a function of the cache size. HybridCache is very sensitive to the local cache size as it obtains the fewest documents with small caches. However, this number of retrieved documents is incremented as the local cache size is augmented. This behaviour is caused by the SxO replacement policy as it does not work appropriately for small caches as it obtains poor cache hits (Figure 6e). On the other hand, COOP also obtains fewer documents than the rest of caching schemes. Finally, we can remark that the number of documents retrieved in the simulation time by CLIR, DPIP and SimpleSearch is not dependent on the cache size.

Figure 6e illustrates the mean percentage of cache hits as a function of the cache size. The greater the cache size is, the higher the percentage of cache hits is expected to obtain as can be observed in the figure. MobEye is the caching scheme with the higher cache hit percentage, especially the remote cache hit because of the broadcast requests. CLIR achieves, as well as MobEye and SimpleSearch, the best local cache hits.

Figure 6f compares the percentage of redirection hits as a function of the cache size. The shown behaviour is similar to that of previous studies as HybridCache obtain a performance close to 0%. COOP achieves about 85% while CLIR obtains a performance between 95% and 100%. However, there is a great variability with small cache sizes due to the great probability of replacements.

V. CONCLUSIONS

In this paper, we have evaluated the performance of the CLIR caching scheme applied to static grid ad hoc networks. This evaluation has been performed using the following metrics: mean traffic processed by the node, the delay perceived to obtain the requested documents, the mean percentage of timeouts, the mean number of retrieved documents per node, the local and remote cache hits and the percentage of redirection cache hits. We have evaluated the influence of the traffic load in the network, the TTL of the documents, the traffic pattern (Zipf slope) and the local cache size. In addition, we have compared the performance of CLIR to the caching schemes HybridCache, COOP, DPIP, SimpleSearch and MobEye. Finally, the performance of CLIR has also been compared to the case of an ad hoc network that does not implement any caching scheme.

From the set of developed simulations we can conclude that MobEye, COOP and HybridCache are not suitable for static ad hoc networks. We base this conclusion on the fact that they obtain a mean percentage of timeouts different to zero. This behaviour is not acceptable in this kind of networks where the servers are always available because the wireless nodes do not move and hence, they are always available. On the other hand, they also retrieve fewer documents than the rest of the caching schemes. Taking into account the rest of caching schemes (DPIP, SimpleSearch and CLIR), CLIR always obtains the lowest traffic generation as well as the lowest delay for all the studied situations. In addition, CLIR always presents a better performance than the No Caching Scheme for all the studied parameters and, hence, we can assert that it is suitable for this kind of networks.

Finally, CLIR presents a percentage of local cache hits similar to the rest of caching schemes and it obtains the best percentage of redirection cache hits, with a performance close to 100%. This result demonstrates that the management of the redirection cache is efficient.

As a future work we propose to evaluate the influence of the employed routing protocol on the caching scheme performance. As we have developed the piggy-backing method of CLIR using OADV, different behaviours are expected using other routing protocols.

ACKNOWLEDGEMENTS

We would like to thank Adela Isabel Fernandez-Anta for revising the syntax and grammar of this article. This study was partially supported by the National Project No. TEC2009-13763-C02-01.

REFERENCES

[1] F. J. González-Cañete and E. Casilari, “Evaluation of a Cooperative Caching Scheme for Grid Ad Hoc Networks“, Proc. 6th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2012), 2012, pp. 97-103.

[2] D. Wessels, Web Caching: Reducing Network Traffic. O’Reilly, 2001.

[3] P. Kuppusamy and K. Thirunavukkarasu, B.Kalaavathi, “A Review of Cooperative Caching Strategies in Mobile Ad Hoc

Networks", International Journal of Computer Applications, vol. 29, no. 11, 2011, pp. 22-26.

[4] G. Dodero and V. Gianuzzi, "Saving Energy and Reducing Latency in MANET File Access", Proc. 26th International Conference on Distributed Computing Systems Workshops (ICDCSW'06), 2006, pp. 16-20.

[5] S. Lim, W.C. Lee, G. Cao and C.R. Das, "A novel caching scheme for improving Internet-based mobile ad hoc networks performance", Ad Hoc Networks, vol. 4, no. 2, 2006, pp. 225-239.

[6] S. Lim, W.C. Lee, G. Cao and C.R. Das, "Cache invalidation strategies for Internet-based mobile ad hoc networks", Computer Communications, vol. 30, no. 8, 2007, pp. 1854-1869.

[7] T. Moriya and H. Aida, "Cache Data Access System in Ad Hoc Networks", Proc. 57th IEEE Semiannual Vehicular Technology Conference (VTC 2003), April 2006, vol. 2, pp. 1228-1232.

[8] B. Tang, H. Gupta and S.R. Das, "Benefit-Based Data Caching in Ad Hoc Networks", IEEE Transactions on Mobile Computing, vol. 7, no. 3, 2008, pp. 289-304.

[9] Y.H. Wang, J. Chen, C.F. Chao and C.C. Chuang, "A Distributed Data Caching Framework for Mobile Ad Hoc Networks", Proc. 2006 International conference on Wireless communications and mobile computing, 2006, pp. 1357-1362.

[10] J. Cho, S. Oh, J. Kim, K.H. Lee and J. Lee, "Neighbor Caching in Multi-Hop Wireless Ad Hoc Networks", IEEE Communications Letters, vol. 7, no. 11, 2003, pp. 525-527.

[11] P. Nuggehalli, V. Srinivasan and C.F. Chiasserini, "Energy-Efficient Caching Strategies in Ad Hoc Wireless Networks", Proc. 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2003), June 2003, pp. 25-34.

[12] N. Chand, R.C. Joshi and M. Misra, "Cooperative Caching in Mobile Ad Hoc Networks Based on Clusters", International Journal on Wireless Personal Communications, no. 43, 2007, pp. 41-63.

[13] M.K. Denko, "Cooperative Data Caching and Prefetching in Wireless Ad Hoc Networks", International Journal of Business Data Communications and Networking, vol. 3, no. 1, 2007, pp. 1-15.

[14] V. Gianuzzi, "File Distribution and Caching in MANET", Technical Report DISI-TR-03-03, DISI Tech University of Genova (Italy), 2003.

[15] Y. Du and S. Gupta, "COOP – A cooperative caching service in MANETs", Proc. Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services (ICAS-ICNS 2005), October 2005, pp. 58-63.

[16] A. Klemm, C. Lindemann and P.D. Waldhorst, "A Special-Purpose Peer-to-Peer Sharing System for Mobile Ad Hoc Networks", Proc. IEEE Semiannual Vehicular Technology Conference (VTC 2003), October 2003, pp. 2758-2763.

[17] G. Chiu and C. Young, "Exploiting In-Zone Broadcast for Cache Sharing in Mobile Ad Hoc Networks", IEEE Transactions on Mobile Computing, vol. 8, no. 3, 2009, pp 384-397.

[18] C.Y. Chow, H.V. Leong and A. Chan, "Peer-to-Peer Cooperative Caching in Mobile Environments", Proc. 24th International Conference on Distributed Computing Systems Workshops (ICDCSW'04), March 2004, pp. 528-533.

[19] H. Artail, H. Safa, K. Mershad, Z. Abou-Atme and N. Sulieman, "COACS: A Cooperative and Adaptive Caching Systems for MANETs", IEEE Transactions on Mobile Computing, vol. 7, no. 8, 2008, pp. 961-977.

[20] C.Y. Chow and H.V. Leong, A. Chan, Group-based Cooperative Cache Management for Mobile Clients in a Mobile Environment, Proceedings of the 33rd International Conference on Parallel Processing (ICPP'04), 2004, pp. 83-90.

[21] L. Yin and G. Cao, "Supporting Cooperative Caching in Ad Hoc Networks", IEEE Transaction on Mobile Computing, vol. 5, no. 1, 2006, pp. 77- 89.

[22] Y. Ting and Y. Chang, "A Novel Cooperative Caching Scheme for Wireless Ad Hoc Networks: GroupCaching", Proc. International Conference on Networking, Architecture and Storage (NAS 2007), 2007, pp. 62-68.

[23] N. Chand, R.C. Joshi and M. Misra, "Efficient Cooperative Caching in Ad Hoc Networks", Proc. 1st International Conference on Communication System Software and Middleware (Comsware'06), January 2006.

[24] F. Sailhan and V. Issarny, "Cooperative Caching in ad hoc Networks", Proc. 4th ACM International Conference on Mobile Data Management (MDM'2003), January 2003, pp. 13-28.

[25] F.J. González-Cañete, E. Casilari and A. Triviño-Cabrera, "A cross layer interception and redirection cooperative caching scheme for MANETs", EURASIP Journal on Wireless Communications and Networking 2012, 2012:63, doi:10.1186/1687-1499-2012-63.

[26] http://www.isi.edu/nsnam/ns/ [retrieved: February, 2013]

[27] L.A. Adamic and B.A. Huberman, "Zipf's law and the Internet", Glottometrics, vol. 3, 2002, pp. 143-150.

[28] S.G. Dykes, K.A. Robbins and C.L. Jeffery, "Uncacheable Documents and Cold Starts in Web Proxy Cache Simulations", Technical Report CS-2001-01, Texas University (EEUU), 2000.

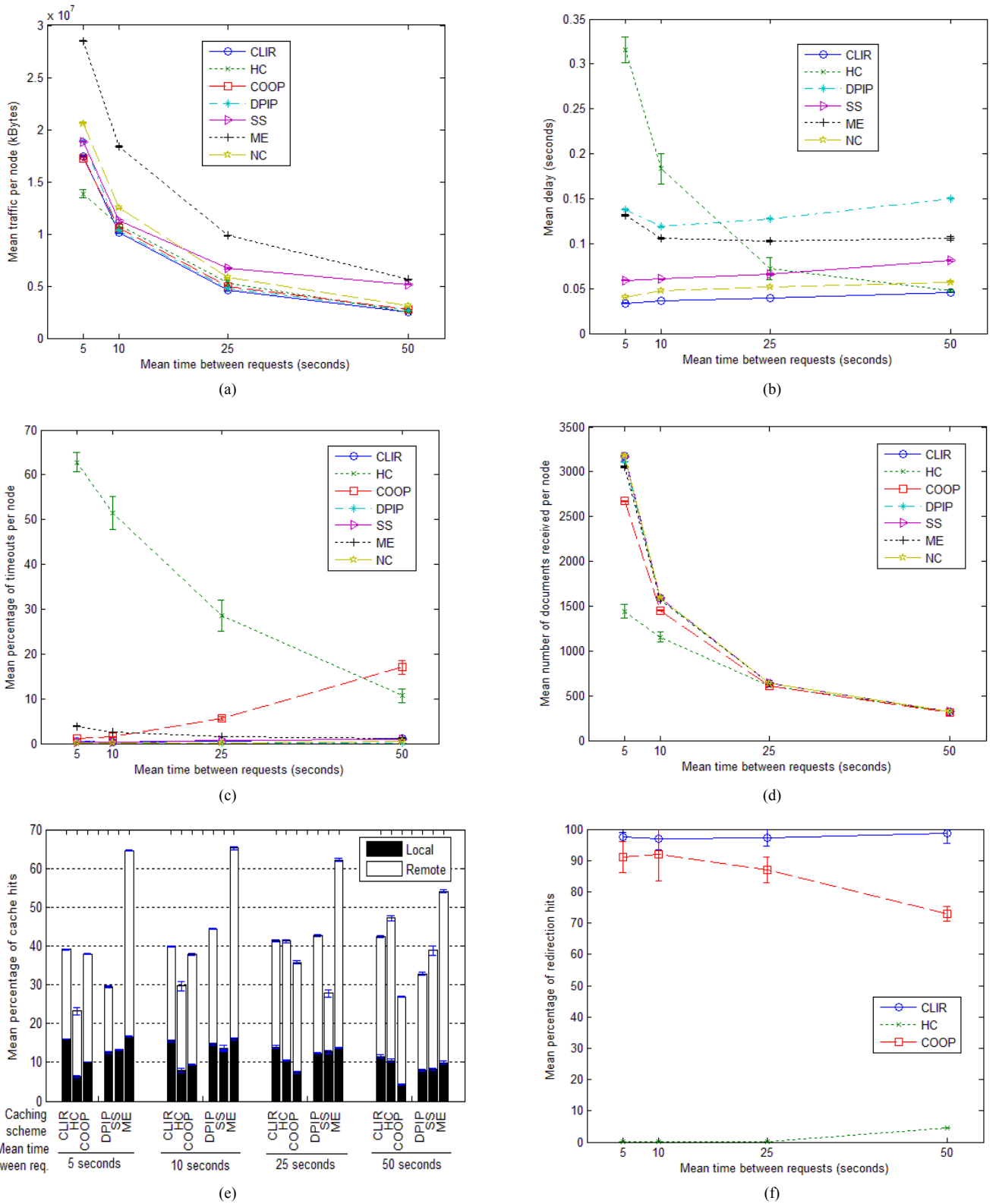


Figure 3. Mean traffic processed by node (a), delay (b), percentage of timeouts (c), documents received (d), cache hits (e) and redirection cache hits (f) as a function of the mean time between requests.

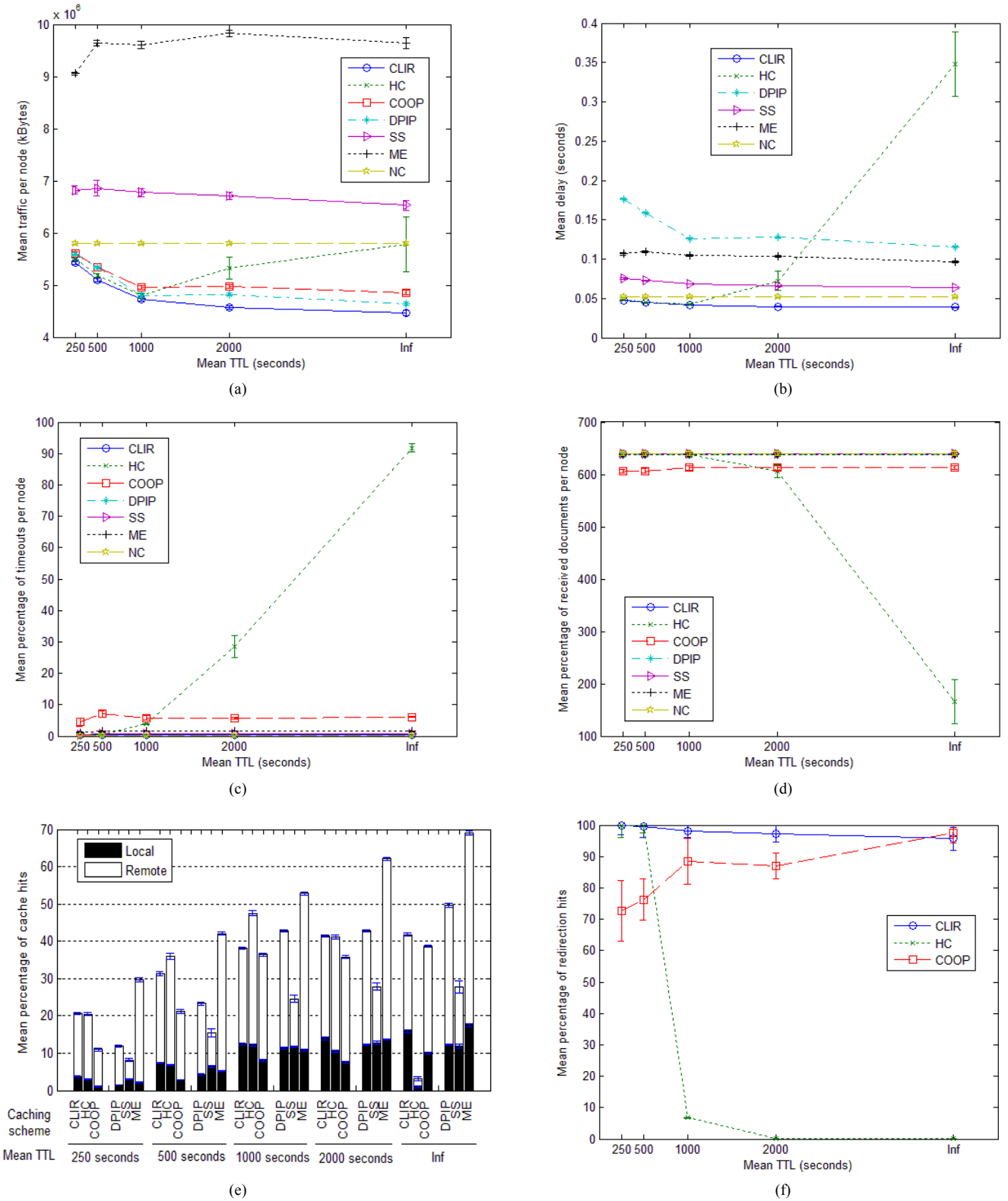


Figure 4. Mean traffic processed by node (a), delay (b), percentage of timeouts (c), documents received (d), cache hits (e) and redirection cache hits (f) as a function of the mean TTL of the documents.

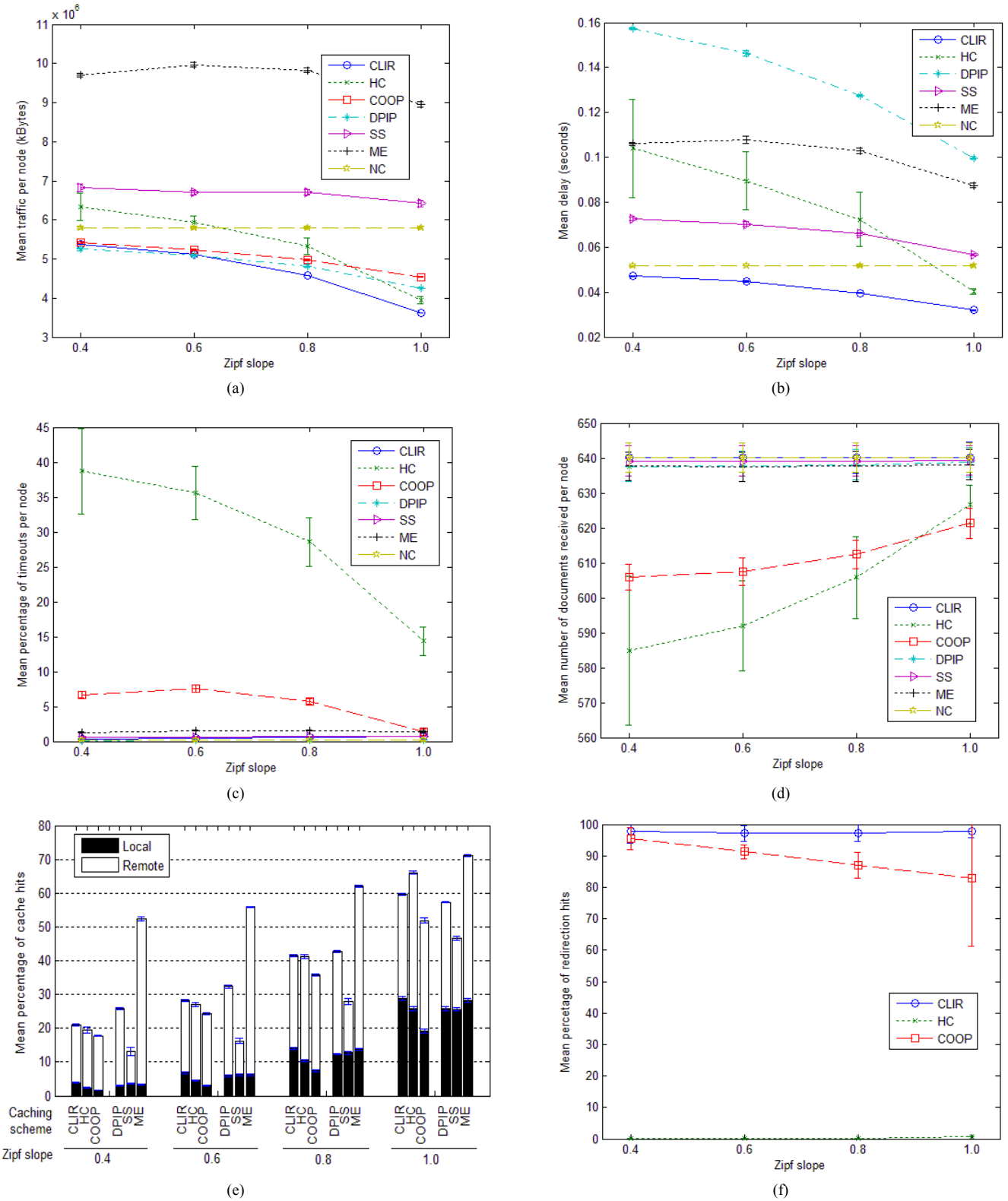


Figure 5. Mean traffic processed by node (a), delay (b), percentage of timeouts (c), documents received (d), cache hits (e) and redirection cache hits (f) as a function of the Zipf slope.

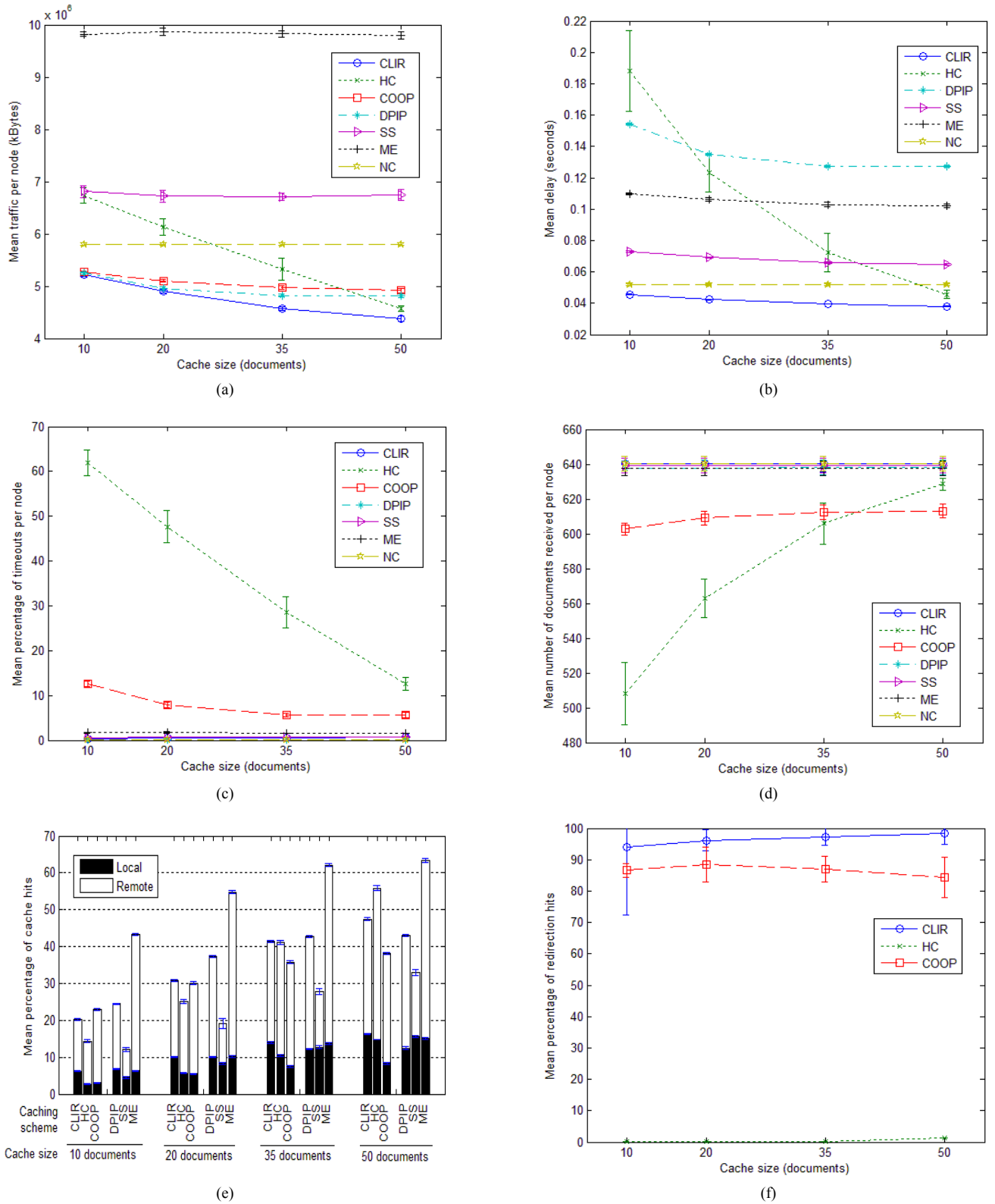


Figure 6. Mean traffic processed by node (a), delay (b), percentage of timeouts (c), documents received (d), cache hits (e) and redirection cache hits (f) as a function of the cache size.

Optimized Resource Management using Linear Programming in Integrated Heterogeneous Networks

Umar Toseef *†, Yasir Zaki‡, Andreas Timm-Giel*, and Carmelita Görg†
*Institute of Communication Networks, Hamburg University of Technology, Hamburg, Germany
Email: {umar.toseef, timm-giel}@tuhh.de
†TZI ComNets, University of Bremen, Bremen, Germany, Email: {umr, cg}@comnets.uni-bremen.de
‡Computer Science Department, New York University, Abu Dhabi, UAE, Email: yz48@nyu.edu

***Abstract*—There have been tremendous advances over the past decades when it comes to wireless access technologies. Nowadays, several wireless access technologies are available everywhere. Even mobile devices have evolved to support multiple access technologies (e.g., 3G, 4G or WiFi) in providing the best possible access to the Internet. However, all of these devices can communicate using only one access technology at a time. It is foreseen that an integration of these access technologies to offer users a network access through multiple simultaneous connections would be beneficial for both end users and the mobile network operators. This paper investigates how to tackle the simultaneous usage of multiple wireless access technologies in the downlink. For this purpose, a practical example of heterogeneous network is considered where 3GPP LTE and non-3GPP WLAN access technologies are integrated together. Furthermore, a novel decision mechanism is proposed, that focuses on optimizing the network resource management based on a mathematical formulation of the system. The mathematical model is implemented using the Linear Programming techniques. The paper demonstrates the gains that are achieved from using such innovative decision mechanism as well as the benefits that arise from the simultaneous usage of multiple wireless heterogeneous accesses.**

Keywords*— *LTE and WLAN interworking, User QoE optimization, Linear programming, Access link modeling, Heuristic methods

I. Introduction

The Long Term Evolution (LTE) of the Universal Mobile Telecommunication System (UMTS) is one of the latest milestones achieved in an advancing series of mobile telecommunication systems by the Third Mobile Generation Partnership Project (3GPP). LTE is well positioned today, and is already meeting the requirements of future mobile networks. On the other hand, the technology of handheld mobile devices has also made significant advancements in recent years. This has made mobile broadband subscriptions to increase rapidly worldwide. Every year, hundreds of millions of users are subscribing for mobile broadband services. This is because a number of broadband applications have been redesigned to substantially enhance user experience by taking advantage of mobility support and large data rates of new access technologies. Such applications include social-networking (e.g., Facebook, Google+, Twitter etc.), multi-player gaming, content sharing (e.g., Youtube, Cloud Storage etc.), WebTV, video telephony, search engines etc. The traffic data generated by the rapidly increasing broadband subscribers due to use of the aforementioned applications is manifold higher in volume compared to pure voice traffic. The existing 3GPP mobile communication networks (e.g., HSPA and LTE) are already facing difficulties to meet this high demand for wireless data. This has made users and operators to rely onto Wireless Local Area Networks (WLAN) based on IEEE 802.11 set of standards. The modern WLANs are capable of offering very high data rates but provide a small coverage area and limited mobility support. Therefore, they are more suitable to areas with highly dense demand for high data rate wireless access with limited mobility support. On the other hand, 3GPP networks are designed to provide ubiquitous coverage through mobility support and therefore better suited to areas with moderately dense demand for wireless access with high mobility. In this way, WLAN and 3GPP networks can complement each other in making high-speed Internet access a reality for a large population. This work discusses how the integration of these two technology types can be realized, what benefits are possible for the users and operators from this integration, and what are the challenges involved in the resource management of these heterogeneous networks. This work also proposes several mechanisms for efficient resource management of heterogeneous networks and evaluates their performance with the help of simulation based studies.

Fig. 1 shows how 3GPP and non-3GPP access technologies can be integrated to build heterogeneous networks according to the 3GPP standards [2]. This architecture called as, System Architecture Evolution (SAE), also allows mobile users to roam between the two access technology types. For this purpose, a seamless mobility is achieved by employing Proxy Mobile IPv6 (network based mobility) or Dual Stack Mobile IPv6 (host based mobility) [2]. The 3GPP SAE architecture, however, has certain limitations when it comes to supporting multi-homed users. This implies that a user can be associated with only one of the available access networks but cannot connect to more than one network simultaneously.

This work is an extension of [1] which investigates how the multihoming support can be realized in 3GPP SAE architecture. In addition, it discusses how the network operators can make an optimum use of the aggregated bandwidth resources and network diversity in a multihoming scenario through traffic flow management. The rest of the paper is organized

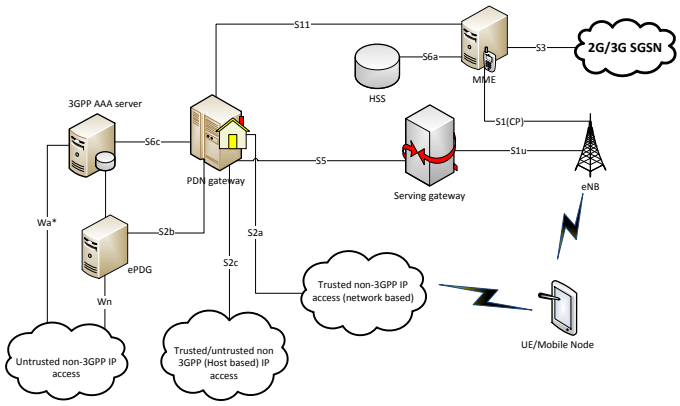


Fig. 1. 3GPP proposed SAE architecture for the integration of non-3GPP and 3GPP access technologies

as follows: related work has been discussed in Section II, Section III describes how the current 3GPP SAE architecture can be extended to provide user multihoming support. Section IV describes the importance of flow management function in a heterogeneous network, and Section V explains the Linear Programming technique to do optimized flow management operation. Section VI provides proof of concepts through the discussion of simulation results of the investigated realistic scenarios. Finally, in Section VII a heuristic based resource management algorithm is devised that offers near-optimum performance without high computational complexity.

II. Related Work

A number of research studies can be found in literature making use of cross-layer techniques and soft handover to optimize handover cost in terms of packet delay and loss in heterogeneous networks. For example, Song and Jamalipour [3] describe an intelligent scheme of vertical handover decisions in selecting the best handover target from several candidate heterogeneous networks. Several other proposals have been made to improve the performance of cellular and 802.11 networks. Song et. al. [4] has discussed admission control schemes to improve the performance of integrated networks. Fei and Vikram [5] propose a service differentiated admission control scheme based on semi-Markov chain that although very accurate but has high computational complexity. Similarly, Zhai et. al. [6] has shown that by controlling the collision probability with the help of input traffic rate of users, the maximum throughput can be achieved by keeping 802.11 network in a non-saturated state. Other studies have focused on developing solutions for load balancing in the integrated heterogeneous network environment. Such a proposal can be found in [8], where policy based load balancing framework has been presented to effectively utilize the aggregated resources of loosely coupled cellular/WLAN network. In this work, we explore the practical limits of the achievable performance in a heterogeneous network scenario by going down to the MAC layer functionalities of the involved access technologies. The goal is to maximize the spectral efficiency of the network bandwidth resources and fulfill the application QoS requirements at the same time. In contrast to other studies, we provide an analytical solution to the problem that adapts to the time varying channel conditions of the user and dynamically decides the best network paths for user traffic flows in order to achieve system wide optimized performance and improved user QoE. The focus of this work is, however, restricted to the downlink transmission of access technologies.

III. Network Simulation Model

This work follows the proposal of the 3GPP specifications in the integration of 3GPP access technology (namely, LTE) and trusted non-3GPP access technology (namely, legacy WLAN 802.11a) where host based mobility solutions, i.e., Dual Stack Mobile IPv6 is considered. For this purpose, a simulation network model has been implemented using the OPNET [9] network simulator. This includes the detailed implementation of LTE network entities following the 3GPP specifications. In this simulation model, home agent (HA) function is located at the Packet Data Network (PDN) gateway. The remote server acts as a correspondent node (CN) from where mobile users access application services like VoIP, video, HTTP and FTP (see Fig. 4).

OPNET model library implements the basic MIPv6 functionality. This implies that a mobile node may have several care-of addresses but only one, called the primary care-of address, can be registered with its home agent and the correspondent nodes. In order to achieve multi-homing, this basic support has been extended according to the IETF RFC for multiple care-of address (MCoA) registration [10]. This enables the user to register the care-of addresses from all of its active network interfaces with its home agent. This work assumes that the user never attaches to its home network, and both LTE and WLAN networks are seen as foreign networks by the user. Therefore, a user configures one IPv6 care-of address when it is in the coverage of LTE and still another care-of address is obtained when WLAN access is available.

Though MCoA extension enables a user to register up to two care-of addresses with its home agent, user cannot communicate over the two network interfaces simultaneously. This is because MCoA recommends using only that single care-of address, which has been most recently registered/refreshed. This calls for the need of another MIPv6 extension namely Flow Binding Support [11] that permits UE to bind one or more traffic flows to a care-of address. A traffic flow, in this extension, is defined as a set of IP packets matching a traffic selector [12]. Traffic selector helps identify the flow to which a particular packet belongs through the matching of the source and destination IP addresses, transport protocol number, the source and destination port numbers and other fields in IP and higher-layer headers. The Traffic selector information is carried as a sub option inside the new mobility option "Flow Identification Mobility Option" introduced by the flow binding support extension. A comprehensive description of this heterogeneous network simulator can be found in [13].

It should be noted that our focus is only on the downlink access for LTE and WLAN. This implies that no uplink transmissions are performed for WLAN during the whole simulation time. Instead uplink traffic (e.g., TCP ACK packet etc.) is transmitted by the user through the LTE uplink access.

Fig. 2 presents the resource management architecture to be used in conjunction with the heterogeneous network simulator. This is an open and flexible architectural framework, where the resource management task is performed in 3 steps: (1) information collection; (2) decision making; (3) decision enforcement [14]. These steps are handled by three functional entities:

- Information Management Entity (IE) is in charge of gathering the information required by the decision making, e.g., link quality, power limitation, load and congestion of the networks. IE also pre-processes and filters the gathered information before it is delivered to the other entities.
- Decision Making Entity (DE) is the most intelligent part of this system architecture. It makes use of the information available from the IEs to take a decision in accordance with pre-defined policies. Examples of such decisions are association to a certain access network, vertical handover hints, change in a service treatment, grant or deny user access to a service/network etc. A decision making entity residing in the network is denoted with DE_n and that in the user terminal by DE_u.
- Execution and Enforcement Entity (EE) finally executes or performs the decision made by DE. In this work, DE_n entity takes all resource management decisions. These decisions are executed locally using the O_{DE} interface to EE entity. The decisions to be executed at user terminal are propagated via the O_{DD} interface to the DE_u that enforce them using the local EE entity.

The algorithms and policies used by the DE_n in making decisions will be discussed in more detail in the next sections of this chapter. Typical examples of these decisions are: when a particular user terminal should attach or de-attach to WLAN access network and how much traffic should be directed to each network path for uplink and downlink communication of a multihomed user. The decisions related to the association with the WLAN access network are executed at the user terminal via the DE_u entity. The decisions regarding traffic distribution have to be executed at home agent as well as at user terminal.

IV. Flow Management

In the developed simulation environment, a user can communicate simultaneously through 3GPP access technology (i.e., LTE) as well as through non-3GPP access networks (i.e., WLAN). The question still remains how a network operator or a user can make an efficient use of the two network paths from two access technologies. For this purpose a flow management function is introduced at the home agent. In other words, it specifies how DE_n entity should function. The flow management function makes use of the MIPv6 extensions and allows controlling the user data rate on each network path. In general, there are two options of managing traffic flows for a multi-homed user. The first option is to carry one complete application traffic flow over one path of choice, this is known as "traffic flow switching". The second option is to divide the traffic flow into several smaller sub-flows where each sub-flow is carried over one network path. This will be called "traffic flow splitting".

If flow management is performed properly a considerable improvement in network capacity and user satisfaction can be achieved. That is why the decision engine (i.e., DE_n) of the flow management function, which controls the user data rate over the network paths is of core importance. In order to attain the goals of optimized network performance, the decision engine needs to know the precise information of the available network resources and the user demands. Once this information is available, DE_n with the help of the proposed mathematical techniques, can optimally assign network resources to the users fulfilling their demands while making use of all available network paths.

Each wireless access network usually has a fixed amount of network resources (e.g., spectrum bandwidth), and the network performance itself depends on the fact with what (spectral) efficiency these resources are utilized. In order to achieve higher data rates, a network resource management function should select those users more often who can attain high spectral efficiency. This work adopts the term "network path cost" to denote the required network resources per unit data rate. For a user, its network path cost can be accessed through cross layer information from the MAC layer of the corresponding access technologies. In the following subsections, it is shown how the network path cost can be computed for users in WLAN and LTE networks.

A. Network path cost for WLAN

Most modern wireless LAN access networks follow IEEE 802.11 standards, marketed under the name of Wi-Fi. In the infrastructure mode of 802.11 typically a number of stations are associated to an access point (AP) (which is normally a router) that serves as a bridge to a wired infrastructure network. 802.11 MAC uses one of the following three techniques to provide channel access control mechanisms.

1) Point coordination function (PCF): resides on the access point to coordinate the channel access for all associated stations through polling. A polled station can communicate with the access point in a contention free manner within a time slot. PCF is not part of the Wi-Fi Alliance interoperability and therefore is rarely found implemented on a portable device.
2) Distributed coordination function (DCF): is a random access scheme based on the Carrier Sense Multiple Access with Collision Avoidance protocol (CSMA/CA) with a binary exponential back-off algorithm. The DCF has two operating modes: the basic channel access mode and the RTS/CTS (Request-to-Send/Clear-To-Send) mode. DCF does not provide a contention free medium access and

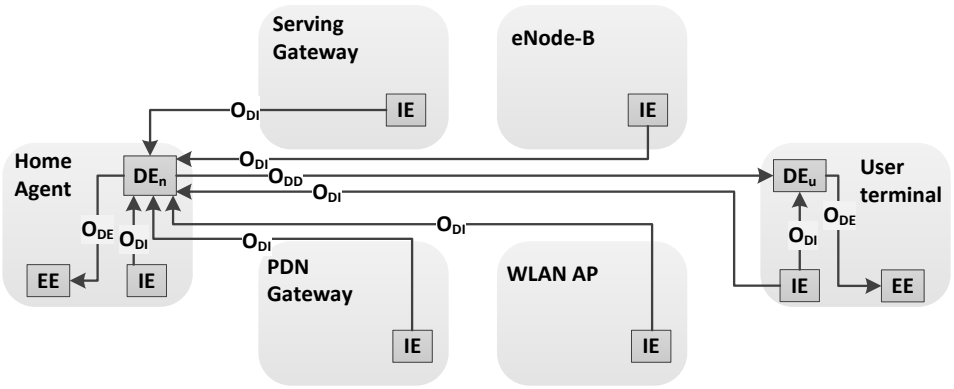


Fig. 2. Resource management architecture for the heterogenous network simulator.

therefore collisions can occur during the transmission if other stations also start transmitting at the same time. DCF is the de-facto default setting for Wi-Fi hardware.

3) Hybrid coordination function (HCF): has been designed to provide a differentiated medium access. Though HFC does not provide service guarantees, it establishes a probabilistic priority mechanism to allocate bandwidth based on traffic categories. HCF was introduced for the 802.11e standard, but it is hard to find complaint hardware. In recent time the 802.11n standard has incorporated HCF, which though becoming increasingly popular, is still available for a limited number of portable devices.

From the above description it can be deduced that a dominant percentage of today's portable Wi-Fi capable devices operate in the DCF mode of 802.11a/b/g. Three flavors of 802.11, i.e., a,b&g, follow very similar procedures in medium access mechanism therefore in this work we focus only on one of the flavors, i.e. 802.11a. The readers are encouraged to refer to [15]- [17] for further details on 802.11 specifications and performance.

Now, in order to explain the network path cost computations for WLAN, consider a scenario where a WLAN access network consists of a station associated with an access point. Assume that the station is just receiving a downlink traffic flow from the access point and does not transmit anything in the uplink. In this way, there is no contention for medium access. The transmission of one data frame with RTS/CTS enabled takes T_S seconds including the exchange of control frames such as RTS/CTS, SIFS (Short Interframe Space), DIFS (DCF Interframe Space), and ACK frames, where

$$T_S = T_{backoff} + T_{DIFS} + T_{RTS} + T_{CTS} + T_{data} + 3 \cdot T_{SIFS},$$

$$T_{backoff} = \frac{W_{min} - 1}{2} \cdot T_{slotTime},$$

$$T_{DIFS} = T_{SIFS} + 2 \cdot T_{slotTime}$$

All components of T_S except T_{data} can be found in the 802.11 standards (see Table II and I). The value of T_{data} can be computed based on the PHY data rate of transmission, i.e., $T_{data} = \frac{\sigma}{\varphi}$, where σ is the data frame size in kbit, and φ is the PHY transmission data rate in [kbit/sec]. Accordingly, the maximum downlink capacity η can be estimated as follows: $\eta = \frac{\sigma}{T_S}$ [kbit/sec].

TABLE I
MAC/PHYSICAL LAYER PARAMETERS OF 802.11A.

SIFS	SlotTime	RTS	CTS	ACK	W_{min}	W_{max}
16 μs	9 μs	160 bit	116 bit	116 bit	16	1024

TABLE II
DURATION OF CONTROL FRAMES IN 802.11A FOR DIFFERENT PHYSICAL LAYER DATA.

Data Rate (Mbps)	Modulation	Bits per Sym.	RTS		CTS/ACK	
			Sym.	μs	Sym.	μs
6	BPSK	24	7	28	5	20
9	BPSK	36	5	20	4	16
12	QPSK	48	4	16	3	12
18	QPSK	72	3	12	2	8
24	16-QAM	96	2	8	2	8
36	16-QAM	144	2	8	1	4
48	64-QAM	192	1	4	1	4
54	64-QAM	216	1	4	1	4

It is clear that 802.11 MAC follows the Time Division Multiple Access (TDMA) like scheme, where users share the wireless access medium for short periods of time. Considering resource allocation time intervals of 1 second, a user needs an exclusive medium access for a γ fraction of that interval to achieve a unitary data rate of 1 kbit/sec. This way, the γ that is expressed in units of $[\frac{sec}{kbit/sec}]$ represents the network path cost. Its value directly depends on T_S, which is the delay experienced in transmitting one data packet of average size σ [bit] operating at PHY data rate φ [kbit/sec]. That is, $\gamma = \frac{T_S}{\sigma}$.

B. Network path cost for LTE

In contrast to 802.11, LTE performs a managed scheduling of available bandwidth resources. The smallest unit of bandwidth resource is referred as a physical resource block (PRB) in the LTE specification. Based on the allocated frequency spectrum size, LTE has a certain number of PRBs. The LTE MAC scheduler residing at the eNodeB schedules these PRBs using a 1ms transmission time interval (TTI). The LTE MAC

scheduler has a very complex way of assigning resources to the associated users. Without digging into the details of the MAC scheduler operation, we focus on the last stage of the resource assignment procedure in a certain TTI. Upon reaching this stage, the MAC scheduler already builded up a list of users which will be transmitting/receiving data in that TTI. For each user entry in the list, there is a corresponding value of the allocated number of PRBs, as well as the channel dependent Modulation and Coding Scheme (MCS) index. The MCS index is then used to lookup the Transport Block Size Index (TBS index). With the help of TBS index and number of allocated PRBs, the Transport Block Size (TBS) is obtained from a table defined in the 3GPP specifications [21]. This is a two dimensional table where each row lists TBS sizes corresponding to the number of PRBs for a particular TBS index. The obtained TBS value defines the size of the MAC frame transmitted to the user in that TTI. In this way, the user received throughput at the MAC layer in a certain TTI can be estimated if the TBS value for that user is known.

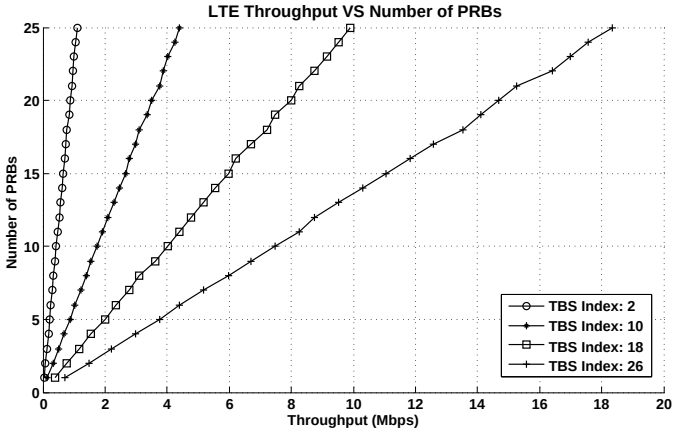


Fig. 3. Relationship of LTE air interface throughput and number of PRBs for different TBS index values [21]. Each curve represents one TBS index.

Fig. 3 shows that for a particular TBS index, the LTE throughput value has almost a linear relationship with the used number of PRBs. If described mathematically, this relationship can be used to determine the required number p of PRBs/TTI to achieve a certain data rate X [kbit/sec] for a user having TBS index i. That is

$$p = \alpha_i \cdot X + \beta_i$$

α_i is the slope of a straight line (as shown in 3) described in units of PRBs/kbps. β_i is the intercept of straight line at the y-axis and is expressed in units of number of PRBs. Both α and β together determine the network path cost of a user's LTE access link.

V. Optimized network resource utilization

When the network path costs for a multi-homed user are known, the problem of optimal resource utilization can be solved using mathematical techniques. In this work, we prefer Integer Linear Programming (ILP) to solve this problem. This choice has been made due to several reasons. For example, ILP guarantees an optimum solution for a correctly formulated problem, the problem formulated in ILP can be extended or restricted by introducing appropriate constraints as well as it saves additional implementation work by making use of already available Linear Programming solvers.

TABLE III
Mathematical model for the optimized resource utilization in algebraic form

Given

U	a set of users
α_j	Data rate dependent part of the LTE link cost in PRBs per kbps for user j, for each $j \in U$
β_j	Data rate independent part of the LTE link cost in PRBs for user j, for each $j \in U$
γ_j	Cost of WLAN link in seconds per kbps for user j, for each $j \in U$
δ_j	Minimum data rate (kbps) demand of a traffic flow destined to user j, for each $j \in U$
Δ_j	Maximum data rate (kbps) allocation for a traffic flow destined to user j, for each $j \in U$
Ω	Number of available PRBs for the LTE access network

Defined variables

X_j	Size of sub-flow in kbps sent over the LTE access link to user j, for each $j \in U$
Y_j	Size of sub-flow in kbps sent over WLAN access link to user j, for each $j \in U$
Z_j	Auxiliary binary variable; its value for a user j is either 1 if $X_j > 0$ or 0 otherwise, for each $j \in U$

Maximize

$\sum_{j \in U} X_j + Y_j$

Subject to

1. $\sum_{j \in U} \alpha_j \cdot X_j + \beta_j \cdot Z_j \leq \Omega$
2. $\sum_{j \in U} \gamma_j \cdot Y_j \leq 1$
3. $\delta_j \leq X_j + Y_j \leq \Delta_j$ for each $j \in U$
4. $Z_j \leq X_j \cdot 10^{20}$ for each $j \in U$
5. $Z_j \geq X_j / \Delta_j$ for each $j \in U$
6. $0 \leq X_j \leq \Delta_j$ for each $j \in U$
7. $0 \leq Y_j \leq \Delta_j$ for each $j \in U$
8. $Z_j \in \{0, 1\}$ for each $j \in U$

Table. III shows the formulation of the problem in algebraic form. The model defines U as the set of multi-homed users. Each element of this set has a number of input parameters, e.g., network path costs for LTE (α,β) and WLAN network (γ) according to the user channel conditions in the corresponding network. The maximum and minimum range of user data rate demands (δ, Δ) based on the individual user application. The amount of available network resources in LTE (Ω) and WLAN (which is 1 second) are also considered as input parameters. The output parameters for each user in set U include the assigned data rate over the LTE network and the WLAN network paths (X,Y). It is obvious that the goal of this model is to achieve the highest possible spectral efficiency from the two network access technologies. The higher the spectral efficiency, the higher the network throughput. Hence,

the objective is to maximize the user data rate over the two network paths, i.e., X and Y for every multi-homed user.

The model imposes eight constraints, which are listed at the bottom of Table III. The first two constraints ensure that the available network resources should not be exceeded when allocating the data rates for users. The third constraint dictates that the user data rate allocation should lie in the specified range. The 4th and 5th constraint determine the value of variable Z based on the X value. If there is a need, a user is allowed to receive its whole demanded data rate over a single network path as shown in constraint number 6 and 7. Constraint 8 is set in order to emphasis that Z is a binary variable, which has value either 0 or 1.

It is assumed here that each user is running only one application. For a constant bit rate application, e.g., VoIP or video the minimum data rate is set equal to the maximum data rate in the model input parameters. For TCP based flows, these two values can be set according to the network operator's policy. It should be noted that the problem has been formulated in a way that it guarantees the minimum data rate for all users and then assigns an additional data rate up to the maximum data rate while optimizing the spectral efficiency of the access networks.

In the investigated scenario, the LTE coverage is available in the whole area of user movement while WLAN coverage is limited in a circular area of 100 meter radius around a hotspot. This implies the users always have LTE access available and WLAN coverage is only found in the vicinity of the hotspot (see Fig. 4). During the resource assignment process, the flow management function classifies users into the following three categories (i) users with LTE access only and running VoIP or video applications (ii) users with LTE and WLAN access running any type of application (iii) users running FTP or HTTP applications with LTE access only. Users in the first category must be assigned the required minimum data rate through LTE as there is no other access available for them. Users in the second category are multi-homed users whose data rate will be decided by the aforementioned mathematical model. For users belonging to the third category, they must get their traffic through the LTE path; however, it is not clear how much data rate should be allocated to them in order to achieve the optimized resource allocation objective. This issue is resolved by using the following work around: the users are assigned a WLAN network path cost greater than unity and they are put into the second category. The WLAN network cost greater than unity will refrain the LP solver to assign any data rate for these users over the WLAN path while the data rate for the LTE path will be decided based on the global objective of the optimized resource allocation.

The resource assignment process by the flow management function is carried out periodically every 100ms[1] in order to adapt to any changes in the user channel conditions. For this purpose user channel condition parameters are obtained through cross layer information from the base stations of the two access technologies. With the help of these parameters, costs for each user network path is computed and fed to the above described mathematical model as the input arguments accompanied with the user data rate demands. As described earlier, the mathematical model is formulated using Linear Programming and solved using the C application programming interface (API) of ILOG CPLEX from IBM [22], which has been integrated inside the OPNET simulator by the authors. The output of this process consists of user data rates on each network path. These decided data rates are then implemented for each user through a traffic shaping function residing at the home agent.

[1]Section VI-A analyzes the selection of this time period

VI. Simulation Results

In this section, the performance of the proposed scheme for optimized resource allocation is evaluated with the help of a simulation scenario. Fig. 4 shows an overview of the scenario in OPNET. The system is populated with 12 users generating a rich traffic mixture of: Voice over IP (VoIP), downlink File Transfer Protocol (FTP), Hyper Text Transfer Protocol (HTTP), video conference (i.e., Skype video call), and video streaming. The users move within one LTE eNodeB cell, and within this cell one wireless access point is present. Table IV shows the parameter configuration for this scenario.

The network performance achieved by the Linear Programming approach will be compared with the other two approaches discussed in [26], i.e., "3GPP-HO" and "Channel Aware". In the "3GPP-HO" approach user multihoming is not supported; instead the policy is to serve a user preferably over WLAN access network in the overlapped coverage of WLAN & LTE access networks. In contrast to this, the "Channel Aware" approach makes use of multihoming and flow management to serve users efficiently. In this approach the capacity of each of the user access links is precisely estimated and all available bandwidth resources are bundled together in achieving the best user Quality of Experience (QoE). Now with the help of the Linear Programming approach, data rates are assigned to the users in a way that network capacity is maximized as well as the minimum data rate demands are met for all users. For this purpose, the DE_n employs the resource allocation model shown in Table III. At each decision instant, the model is solved using updated parameters of user channel conditions and QoS demands and the resulted data rates are then imposed on the user access links by the EE entities.

Fig. 5 shows the performance of the FTP downlink application in terms of IP throughput and file download time. It is evident that the "Linear Programming" approach achieves the highest performance among the three competing approaches. The optimized resource allocation strategy of the "Linear Programming" approach helps increasing user QoE experience by 25% in terms of file download time compared to "3GPP-HO" approach. The "Linear Programming" approach also outperforms the "Channel Aware" approach by reducing the file download time up to 13%.

Similar conclusions can also be drawn for the HTTP application whose performance has been shown in Fig. 6. It

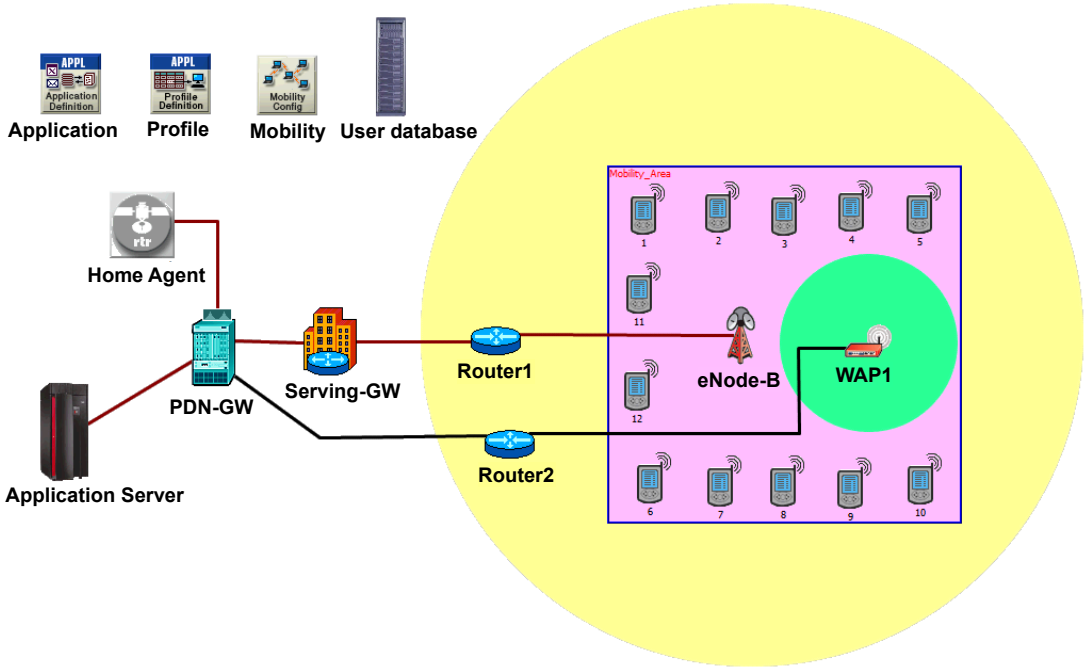


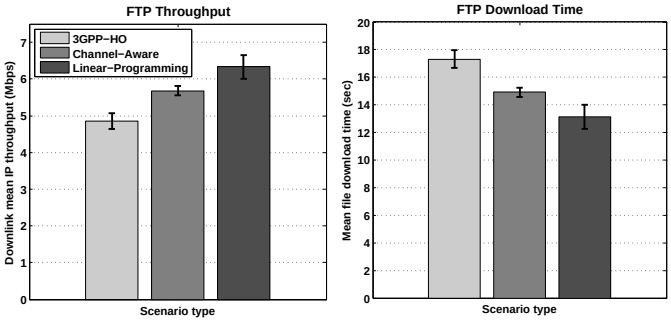
Fig. 4. Simulation scenario setup in the OPNET simulator. The large circular area shows the coverage of LTE and the smaller circular area represents the WLAN network coverage. The user movement is restricted to the rectangular area.

TABLE IV
SIMULATION CONFIGURATIONS FOR EVALUATION OF THE DOWNLINK FLOW MANAGEMENT SCHEME USING LINEAR PROGRAMMING.

Parameter	Configurations
Total number of PRBs	25 PRBs (5 MHz specturm)
Mobility model	Random Direction (RD) with 6 Km/h
Number of users	2 VoIP calls, 1 video streaming, 3 Skype video calls, 2 HTTP and 4 FTP downlink users
LTE channel model	Macroscopic pathloss model [23], Correlated Slow Fading.
LTE MAC scheduler	Time domain: Optimized Service Aware [24], Frequency domain: Iterative RR approach [25]
WLAN access technology	802.11a, RTS/CTS enabled, coverage $\approx$ 100 m, operation in non-overlapping channels
Transport network	1Gbps Ethernet links, no link congestion
VoIP traffic model	G.722.2 wideband codec, 23.05 kbps data rate and 50frame/s
Skype video model	MPEG-4 codec, 512 kbps, 30frame/s, 640x480 resolution, play-out delay: 250ms
Streaming video model	MPEG-4 codec, 1 Mbps, 30frame/s, 720x480 resolution, play-out delay: 250ms
HTTP traffic model	100 bytes html page with 5 objects each of 100Kbytes, page reading time: 12s
FTP traffic model	FTP File size: constant 10MByte, as soon as one file download finishes, the next FTP file starts immediately.
TCP configurations	TCP new Reno, Receiver buffer: 1Mbyte, Window scaling: enabled, Maximum segment size: 1300Byte, TCP reorder timer: 50ms
DE_n decision interval	Every 100ms
Data rate demands $[\delta, \Delta]$	[200 kbps, 25 Mbps] for FTP and HTTP users
Simulation run time	1000 seconds, 10 random seeds, 95% Confidence interval

can be noticed that the HTTP application could attain much less IP throughput compared to the FTP application. This is due to small sized embedded objects of web pages. The download of these objects finishes before the TCP connection could achieve the maximum possible throughput. Owing to this fact, even the "Linear Programming" approach could not significantly improve user QoE over the "Channel Aware" approach. However, a substantial gain is observed compared to the default "3GPP-HO" approach. A large variation in web page download can also be noticed for the "3GPP-HO" approach, the reason of which is as follows. If an HTTP user is found in WLAN coverage, "3GPP-HO" approach serves it solely over the WLAN access link. Now if the WLAN access network is not heavily loaded because there are no FTP users at that instant, then the HTTP users get high throughput and finish page download fast. In other situations, they have to share bandwidth resources with the demanding FTP users and hence page download time elongates.

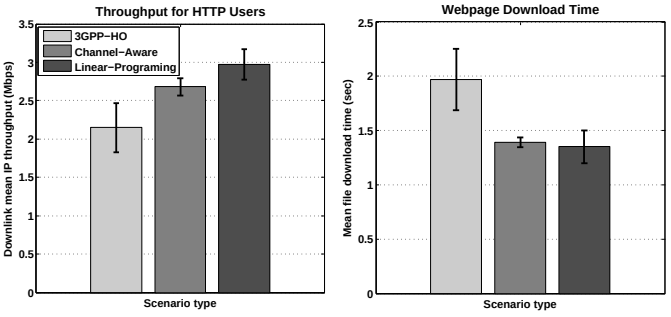
The performance gain of "Channel Aware" and "Linear Programming" over "3GPP-HO" approach can be attributed to manifold factors. For example, both of them are capable of aggregating bandwidth resources from multiple access links, they can accurately estimate the capacity of an access link and utilize it accordingly, they can periodically reevaluate their



(a) (b)

Fig. 5. FTP downlink performance comparison for "3GPP-HO", "Channel Aware", and "Linear Programming" approaches.

assessment about the network conditions and the capacity of user access links, etc. In addition, "Linear Programming" is capable of performing optimized resource allocations. Among all of them, the ability to estimate user access link capacity is the feature that helps these approaches to establish a definite superiority over the default "3GPP-HO" approach. Without access link capacity estimation, the buffers at the air interface could have very large occupancy. On the one hand, employing large buffers leads to long queuing delays, which adversely affects realtime applications in particular. On the other hand, keeping buffer capacity small causes numerous packet drops that degrades, especially, the performance of TCP based applications. By exploiting the knowledge of user access link capacity, "Channel Aware" and "Linear Programming" approaches just send the exact sufficient amount of data to the air interface schedulers that avoids both the large queuing delays and packet drops. In addition, it also minimizes the risk of losing the buffered packets at the access point during the instants of link failure or vertical handover.



(a) (b)

Fig. 6. HTTP downlink performance comparison for "3GPP-HO", "Channel Aware", and "Linear Programming" approaches.

The user QoE for VoIP application has been depicted in Fig. 7. The Box plot shows the sample values of Mean Opinion Score (MOS)[2] [28] computed for a user employed wideband codec. A Box plot graphically depicts the groups of numerical data through their five number summary, i.e., (1) minimum, (2) maximum, (3) median (or second quartile), (4) the first quartile, and (5) the third quartile. The bottom and top of the box are the first and third quartiles, respectively. The band near the middle of the box is the median. The whiskers represent the maximum and minimum of all the data values. Moreover, any data not included between the whiskers is plotted as an outlier with a cross '+' sign.

It can be seen in Fig. 7 that both "Channel Aware" and "Linear Programming" deliver excellent performance by keeping MOS values at the maximum level for most of the time. However, the "3GPP-HO" approach fails to achieve a matched performance. Though the median value lies close to the MOS score 4.0, the other values show quite lower score due to long queuing delays at WLAN access point. Even some of the outliers fall below MOS score 2.2, which could be very annoying for the users. The reason for the "3GPP-HO" approach to sometimes achieve a very high MOS score (i.e., above 4.0) lies in the fact that when the VoIP users are being served over the LTE access network, they are provided with the guaranteed QoS service. The problem arises only when these users are handed over to the WLAN access network.

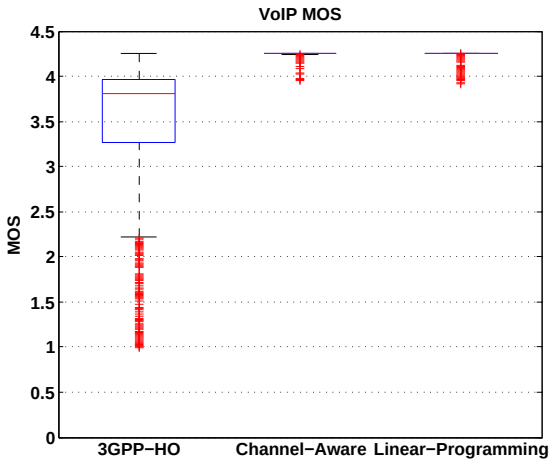
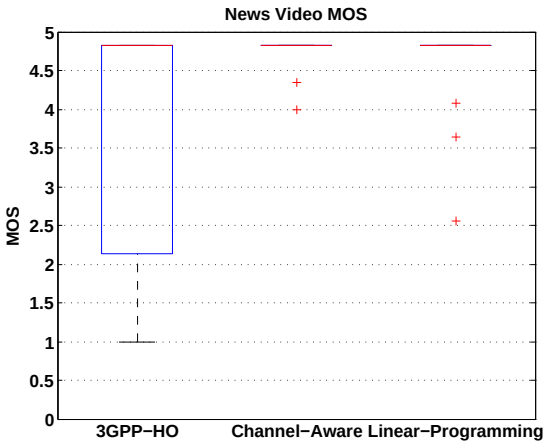


Fig. 7. Downlink performance comparison of VoIP application for "3GPP-HO", "Channel Aware", and "Linear Programming" approaches.

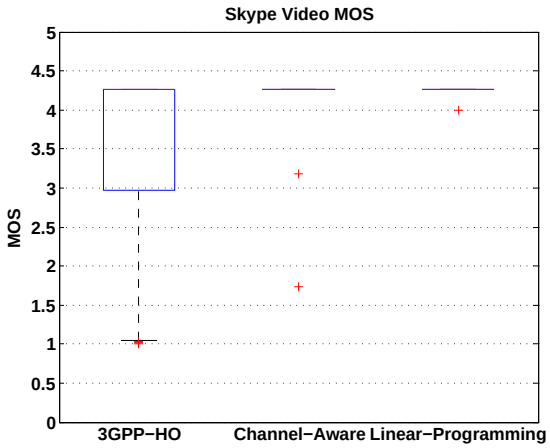
The "Linear Programming" approach also offers excellent user QoE for video applications. This can be confirmed by referring to Fig. 8, which shows the Box plot of user experienced MOS score for their video applications, i.e., video conference and video streaming. Almost all video quality evaluations result in the best MOS score for video applications for both "Channel Aware" and "Linear Programming" approaches. However, "3GPP-HO" fails again to offer an acceptable performance for video application users. Its performance pattern is similar to that of the VoIP application, i.e., the median value stays at the best MOS score while the 3^{rd} & 4^{th} quartiles show the suboptimal performance. As already explained for VoIP case, this phenomenon can be understood with the help of end-to-end packet delay plots shown in Fig. 9. Considering the

[2]MOS gives a numerical indication of the perceived user QoE of realtime applications. MOS is expressed on a scale from 1 to 5, 1 being the worst and 5 the best.

play-out delay limit of 250ms, any packet arriving later than this limit is assumed as lost by the video quality evaluation mechanism. Such packet losses in turn lead to performance degradation. It is evident from Fig. 9 that a large number of packets experience more than 250ms delay for the case where "3GPP-HO" approach is employed. It is mainly the large MAC queue occupancy at the WLAN access point that is the main reason behind these delays. During the times when video users are served over LTE packet end-to-end delays remain under control due to QoS aware scheduling employed in the LTE MAC scheduler. In these situations, the users are satisfied with the service as indicated by the best MOS score. However, during the time when users receive their video application traffic over the WLAN access link, the chances are higher that they have to encounter a congested network.



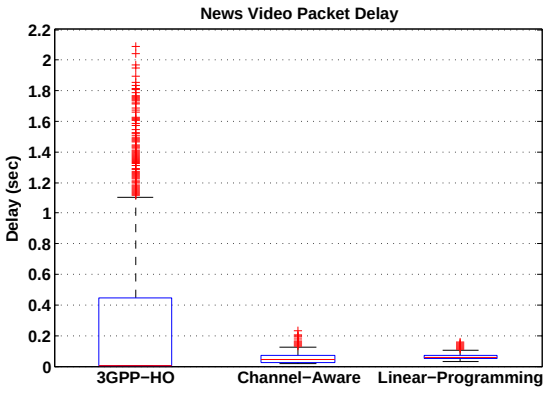
(a)



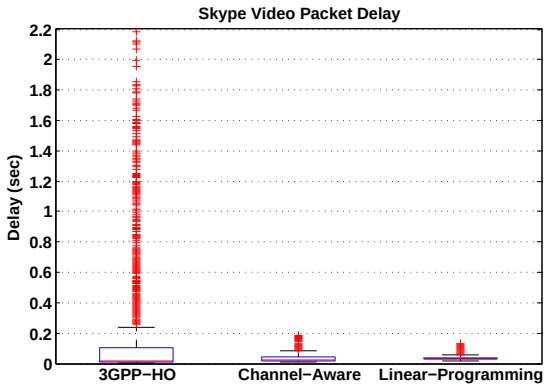
(b)

Fig. 8. Downlink performance comparison of video applications for "3GPP-HO", "Channel Aware", and "Linear Programming" approaches.

Now that the performance of all applications has been observed, it can be inferred that both the "Linear Programming" and "Channel Aware" approaches provide similar performance for realtime applications. However, the "Linear Programming" approach excels when it comes to non-realtime applications like FTP, HTTP etc. This is because realtime applications



(a)



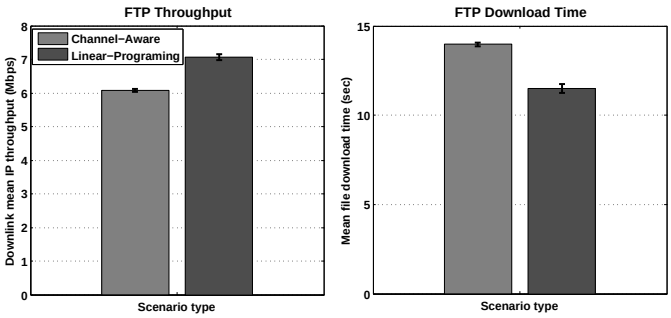
(b)

Fig. 9. Packet delay comparison of video applications for "3GPP-HO", "Channel Aware", and "Linear Programming" approaches.

have stringent QoS requirements, which have to be fulfilled at all costs in order to keep users satisfied. Therefore, both approaches preferably deliver the data rate demands of the realtime services. However, the "Linear Programming" approach, with the help of optimized resource allocation techniques, manages to offer these data rates by consuming lower network resources. This way, larger network resources are made available to non-realtime application users in order to enhance their QoE as well as to increase network capacity.

The discussion on downlink communication is concluded by comparing the performance of "Channel Aware", and "Linear Programming" approaches in another scenario where only FTP users exist within an area of complete LTE and WLAN coverage overlap. Each of these seven FTP users download 10Mbyte files continuously, i.e., as soon as one file download ends, a new file download is started. Fig. 10 shows the FTP application throughput and file download time as experienced by the users. In this particular scenario, the "Linear Programming" approach manages to achieve 16% higher throughput compared to the "Channel Aware" approach. This is slightly higher than 13%, which was observed in the case of the previous scenario with mixed traffic. The reason behind this improved performance is the lower 'minimum data rate' requirement of FTP users compared to video users. Owing to the fact that the 'minimum data rate' requirements must be fulfilled, the users with bad

channel conditions consume lots of resources in achieving that data rate. On the other hand, if this requirement is less, fewer network resources will be consumed even when a user is suffering from bad channel conditions.



(a) (b)

Fig. 10. FTP downlink performance comparison between the "Channel Aware" and the "Linear Programming" approaches.

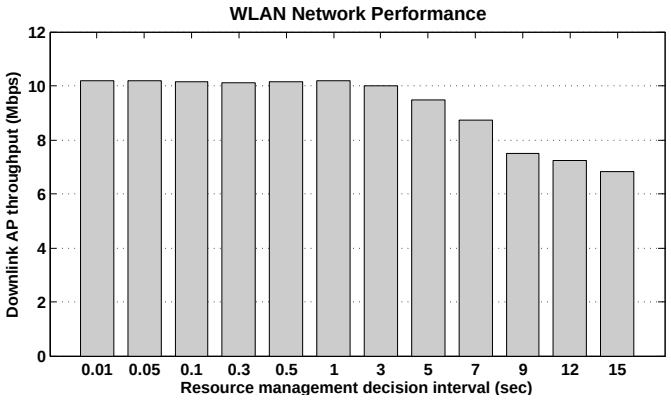


Fig. 11. Downlink throughput variations of WLAN access point for different values of the DE_n decision interval. The "Linear Programming" approach has been used for resource management decisions.

A. Sensitivity Analysis of DE_n Decision Intervals

It has been explained that the decision making entity (DE_n) of the flow management overlay architecture that resides in the network, is responsible for the resource management decisions. These decisions are based on the network information (e.g., user channel conditions, application QoS demands, traffic load, congestion, etc.) supplied by the information management entities (IE) installed at different monitoring points across the network and at the UE. Owing to the dynamic load conditions of the networks and variable channel conditions of mobile users, the information provided by IEs has a short validity period after which it must be refreshed. In this way, the resource management decisions made by DE_n at a certain time instant remain no longer the optimal decisions as soon as the IE supplied information on which these decisions were based becomes obsolete. Therefore, the IEs must send the fresh information to the DE_n periodically to prevent the aforementioned situation. As soon as the DE_n receives the updated information, it revises its resource management decisions and enforces them to achieve an optimal network performance over time. There can be two reasons that the DE_n receives a delayed information from IE entities set up across the network. First, there exists congestion in the network due to which it takes longer for the information data to reach the DE_n. Second, an operator wants to cut down the signalling traffic load generated by that information element by reducing the frequency of updates. In such situations the question is how long is the validity period of the IEs provided information and what could be the consequences if resource management decisions are not updated in due time?

The above questions can be answered with the help of the simulation results shown in Figs. 11, 12, and 13. For this purpose the same simulation scenario, which has been discussed earlier in this section and whose configurations has been listed in Table IV is employed. In this simulation study, the DE_n decision interval is varied from 10ms to 15s and the "Linear Programming" approach is used to make the resource management decisions. It is clear that the optimal resource management decisions should provide an optimum network capacity for both WLAN and LTE networks. It can be seen in the Fig. 11 that the WLAN access point throughput, which represents that network's capacity, remains at the optimum point until the DE_n updates the resource management decisions at least every 1 second. Any further delays would cause the system throughput to reduce. This is due to the user movements (at 6 km/h speed) because of which their channel conditions vary and hence their PHY data rates change. When these variations are not tracked by the DE_n due to lack of fresh information the optimal resource management decisions cannot be carried out. For example, if a user's PHY data rate has increased from 24 Mbps to 36 Mbps during the elongated decision interval, his throughput will not be upgraded by the DE_n until the information about this change reaches DE_n and it revises the resource management decisions. Similarly a high traffic volume will be continuously sent to a user whose PHY data rate has decreased from 36 Mbps to 24 Mbps during the decision interval. This will cause that user to experience large packet delays due the fact that some of the data are being buffered at the access point due to PHY data rate downgrade. Due to such events the WLAN network performance degrades. It can be noticed from the Fig. 11 that increasing the DE_n decision interval to 15s causes approximately 30% degradation in the network capacity.

Fig. 12 shows a similar behavior for the LTE network, which may undergo cell throughput degradations due to the elongated DE_n decision intervals. The cell throughput can reduce up to 9% compared to its optimal value if a decision interval of 15s is considered. However, it can be observed that the performance of the LTE network is less sensitive to DE_n decision intervals compared to WLAN network. For example, a noticeable capacity degradation is seen for the LTE network for a 5s decision interval while such a behavior was observed for the WLAN network at a 3s decision interval. The reason for this phenomenon lies in the fact that WLAN

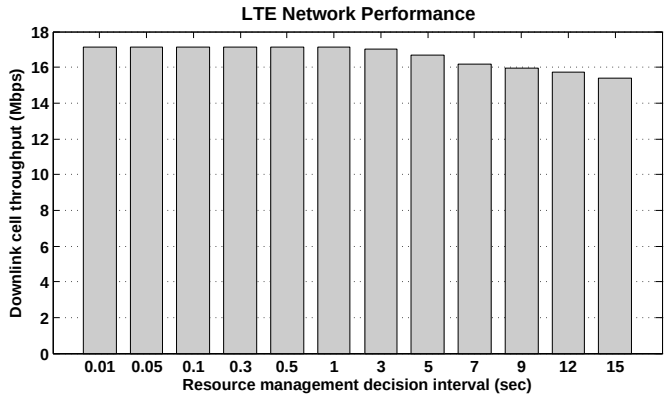


Fig. 12. LTE downlink cell throughput variations for different values of the DE_n decision interval. The "Linear Programming" approach has been used for resource management decisions.

has a smaller coverage area and the user PHY data rates decrease sharply when commuting away from the access point. Therefore, the information about user PHY data rate becomes stale relatively faster and, in turn, affects the optimality of the resource management decisions.

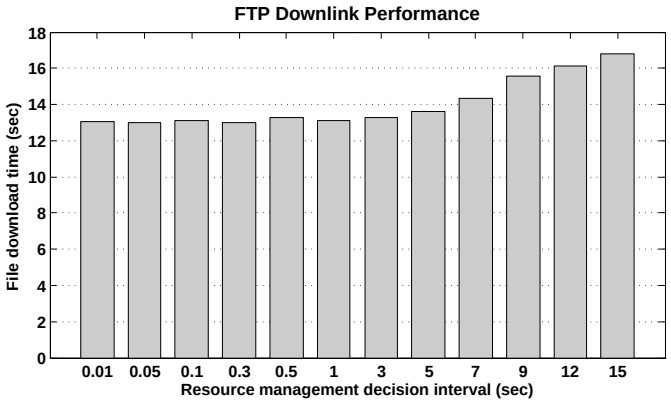


Fig. 13. Mean file download time experienced by FTP users. The figure shows how the FTP performance is affected by different values of DE_n decision interval. The "Linear Programming" approach has been used for resource management decisions.

It has been seen that when the resource management decisions are not optimal, capacities of access networks are reduced. A natural consequence of this will be deteriorations in the perceived user QoE. An example of this is illustrated in Fig. 13, which shows the mean file download time for FTP users. It can be observed that the users have to wait longer for file download completion if DE_n fails to make optimal resource management decisions. Actually, this is because of the reduced network capacities that the users can no longer achieve high throughput and hence suffer from QoE degradations. The simulation results show that file download time can increase up to 24% compared to the optimal value, if DE_n makes resource management decisions every 15s.

The above discussion implies that a decision interval of at most 1 second should be employed in order to achieve an optimal network performance. However, this value is specific to the simulation scenario being discussed and may not hold for other scenarios. For example, in the current scenario the users are moving with a speed of 6km/h following the random direction mobility model. If this configuration is changed or some additional dynamic background traffic load is added to the network, a rerun of this sensitivity analysis will be needed.

VII. Heuristic Algorithms

The solution of the resource allocation problem obtained through mathematical modeling using Linear Programming provides an upper limit on the achievable network capacity. A common practice in this regard is to consider that maximum achievable performance as a target and then devise some heuristic algorithms, which try to attain a similar performance. The rationale behind this practice is the involved high complexity of Linear Programming problem. The high complexity requires substantial computing power and time to solve these problems. This makes the use of Linear Programming unsuitable for realtime optimization tasks in most of the cases. In this section, first of all the complexity of the proposed Linear Programming approaches is discussed and then two heuristic algorithms are developed for downlink and uplink communication scenarios. The effectiveness of the suggested algorithms is also evaluated by comparing it with the corresponding Linear Programming approaches.

A customary way of analyzing the complexity of a Linear Programming problem is through the number of involved variables and constraints. Fig. 14 depicts the complexity of the Linear Programming problem for downlink communication. The two curves indicate that the number of variables and constraints increase linearly with the number of active users in the network. Moreover, even for a large number of users (e.g., 100) the Linear Programming problem seems to have fairly small computational complexity. This is because only few hundreds of variable and constraints are involved at that user count. This fact is also verified by examining the wall-clock time required to solve these Linear Programming problems on a laboratory server computer [3]. The machine was able to solve any of such problems in less than 10ms of wall-clock time. The observation is based on an analysis involving 20,000 random problems with active number of users varying from 3 to 100.

TABLE V
An example problem of resource allocation in downlink communication.

User	Normalized network path cost per kbps		Data rate demand [kbps]	
	WLAN	LTE	Minimum	Maximum
UE1	6×10^{-5}	4×10^{-5}	10^3	10^3
UE2	9×10^{-5}	5×10^{-5}	10^3	10^3

Before the development of the heuristic algorithm, an understanding of the resource allocation problem is developed

[3]Microsoft Windows Server 2008 R2 Enterprise 64bit, Intel©Xeon CPU @ 2.67GHz, 48GB RAM

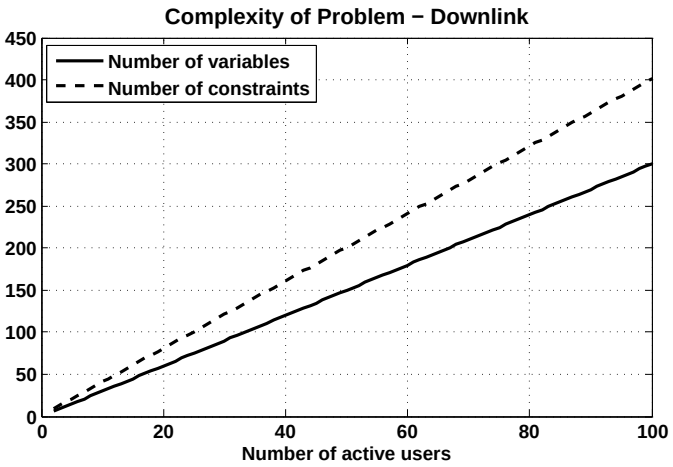


Fig. 14. The complexity of Linear Programming problem for downlink communication described in Table III.

with the help of a simple example presented in Table V. In this example, there are two multihomed users who require a fixed data rate of 1 Mbps to run a realtime service, e.g., video streaming. The normalized network path cost on each access link of the user is also mentioned in the table. The normalized cost represents the fraction of total access network resources to offer a user with 1 Kbps data rate over that access network. The normalized costs help to directly compare the resource consumption of WLAN and LTE access networks for a given amount of data rate, e.g., it can be seen that UE1 has less path cost for the LTE access link compared to its WLAN access link.

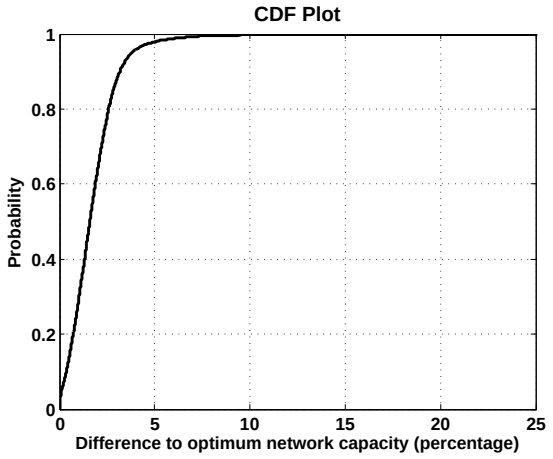
The most suitable strategy to allocate resources in such a situation is through the greedy approach. This implies that users should be served over that particular access link that costs less network resources. It can be noticed from the table that both users have less normalized cost for LTE access links compared to that of WLAN access links. Therefore, according to the greedy strategy both users should be served over their LTE access link. Serving them with their minimum data rate over the LTE access network will consume 4×10^{-2} and 5×10^{-2} fraction of resources, respectively. In other words, it will require a total of 9% of the available LTE resources.

This strategy of the greedy approach is the main driver behind the heuristic algorithm developed for resource allocation in downlink communication as depicted in Fig. 16. The algorithm takes the network path costs and user data rate demands as inputs. It traverses through the list of multihomed users and serve them with the minimum data demands over their less expensive access links. If it happens that the available network resources are already assigned, then the rest of the users are served over the other access network. In case, the network resources of both access networks are consumed without satisfying the minimum data rate demands of all users, the algorithm returns an error message. The error message indicates that the provided problem is infeasible and there is no solution to the problem.

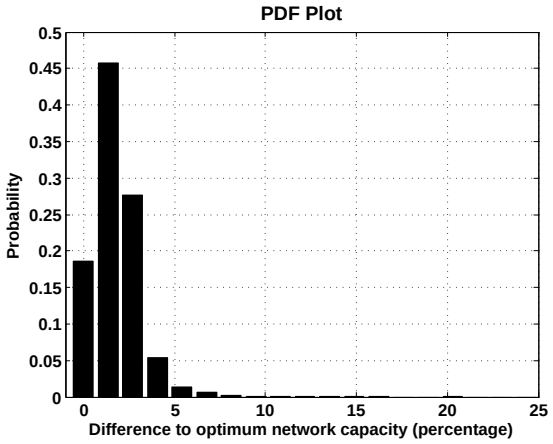
After fulfilling the minimum data rate demands of all users, the left over network resources should be assigned to the users whose maximum data rate demand is greater than their minimum data rate demand. Typically, they are the FTP/HTTP users. Though the same greedy approach can also be employed here once again, it has to be slightly modified. This is because satisfying the maximum data rate demand of 'each' user is not compulsory. Therefore, only those users should be served who can achieve greater data rates with the available network resources. For this purpose, a list is prepared where the network path cost of each user for 'both' of its access links is added. The size of this list is twice the number of users. Sorting this list in ascending order, users are served in the same order in which their access link costs appear in the list. This procedure of serving users up to their maximum data rate demand is performed in subprocess (A) in Fig. 16. A flow chart of subprocess (A) has been shown in Fig. 17. At the end, the heuristic algorithm returns the user data rate assignments over each of their access links.

In order to evaluate the performance of the heuristic algorithm described in Figs. 16 and 17, its results are compared with that of the "Linear Programming" approach. The evaluation is made more comprehensive and thorough by solving 20,000 random problems of resource allocation using both the heuristic and "Linear Programming" approaches. The problems are generated automatically with the help of a script that considers a large range of active users from 3 to 100. A probability of 50% is used to determine if a user in a random problem should be using the realtime service (i.e., minimum and maximum data rate demands are same) or the non-realtime service (i.e., maximum data rate demand is greater than minimum data rate demand). Fig. 15 summarizes the outcome of this evaluation process. The figure shows a CDF and PDF curves of values representing how large the total network capacity is achieved using "Linear Programming" approach compared to that obtained by the heuristic algorithm in random resource allocation problems. The CDF curve depicts that in 90% of the problems, the heuristic approach achieved a network capacity, which was at most 3% less than the optimum achievable capacity computed by the "Linear Programming" approach. Considering the simple complexity of the heuristic approach it is a great performance.

A question can be raised at this point; why the simple greedy approach cannot achieve the same performance as shown by "Linear Programming" approach. This can be explained with the help of an example shown in Table VI. It is a slightly modified version of the example presented in Table V where the maximum data rate demands of users have been raised to 23.75 Mbps. The resource allocation problem in this example is solved using the developed heuristic algorithm as follows. First of all, both users are served with their minimum data rate demands (i.e., 1 Mbps) over LTE access network due to minimum involved resource consumption. This costs 9% of LTE resources. As there are still 91% of LTE and 100% of WLAN access network resources available, a sorted list of network path costs is prepared to utilize the remaining



(a)



(b)

Fig. 15. The performance of the proposed heuristic algorithm for downlink communication. The CDF and PDF curves show the difference of the achieved network capacity using the heuristic algorithm compared to the optimum value obtained using "Linear Programming" approach.

resources. The cost 4×10^{-5} of LTE access link from UE1 comes at the top, therefore 91% of LTE access network resources are allocated to UE1, which translates to a data rate of 22.75 Mbps. This way UE1 is assigned with a total data rate of 23.75 Mbps considering also 1 Mbps data rate allocation in the first step. The next lowest access link cost is of UE2 for its LTE access link (i.e., 5×10^{-5}), however, there are no resources left on the LTE access network. Therefore, no action is taken for UE2 this time. The next lowest cost would be 6×10^{-5} of UE1 for its WLAN access link, but this user has already been served up to its maximum data rate demand. Hence no additional resource can be assigned to UE1. The last entry in the sorted list of cost would be 9×10^{-5} of UE2 over its WLAN access link. 100% of the WLAN access network resources are assigned to this user, which amount to a data rate of 11.1 Mbps. This way, UE2 gets a total data rate allocation of 12.2 Mbps considering also 1 Mbps data rate allocation in the first step. Hence, the total network capacity amounts to 22.75+12.2=34.95 Mbps, in this case.

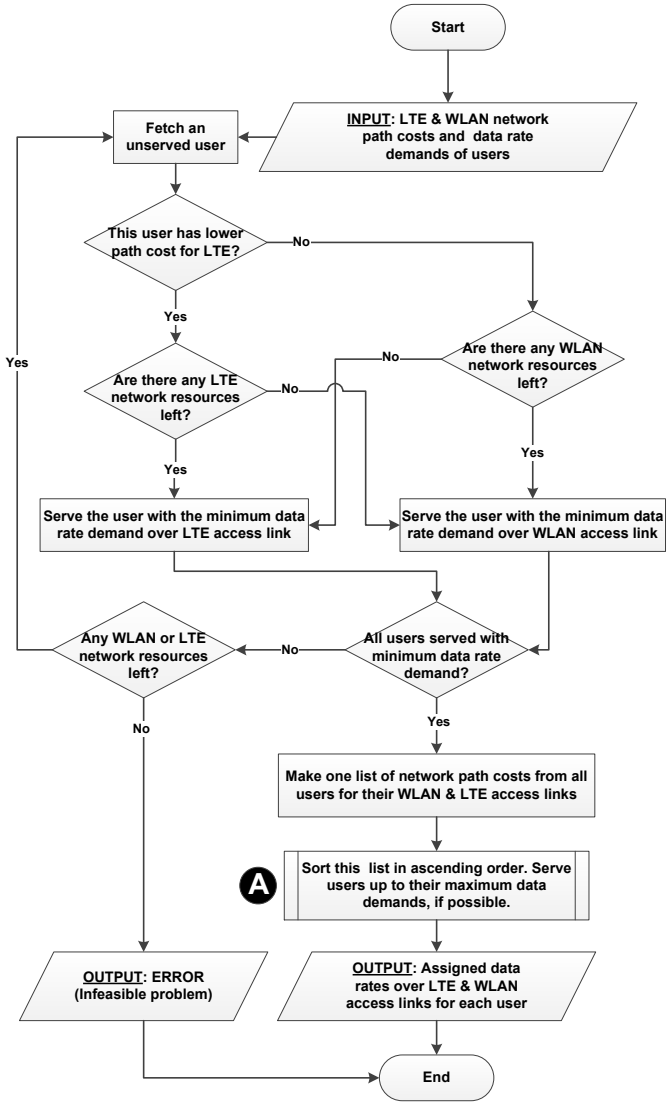


Fig. 16. Flow chart of the heuristic algorithm to solve the resource allocation problem in downlink communication.

TABLE VI
ANOTHER EXAMPLE PROBLEM OF RESOURCE ALLOCATION IN DOWNLINK COMMUNICATION.

User	Normalized network path cost per kbps		Data rate demand [kbps]	
	WLAN	LTE	Minimum	Maximum
UE1	6×10^{-5}	4×10^{-5}	10^3	23.75×10^3
UE2	9×10^{-5}	5×10^{-5}	10^3	23.75×10^3

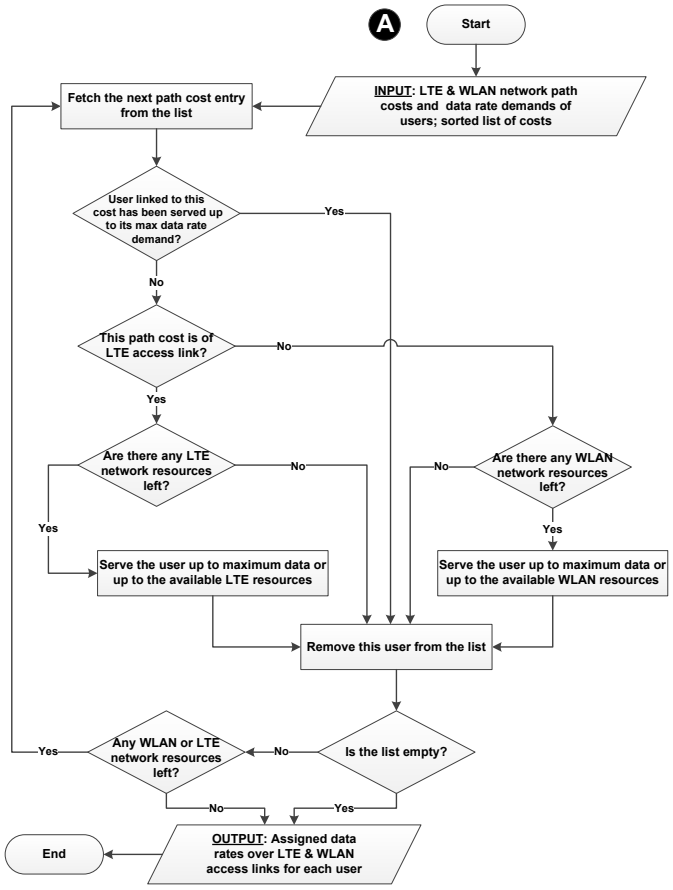
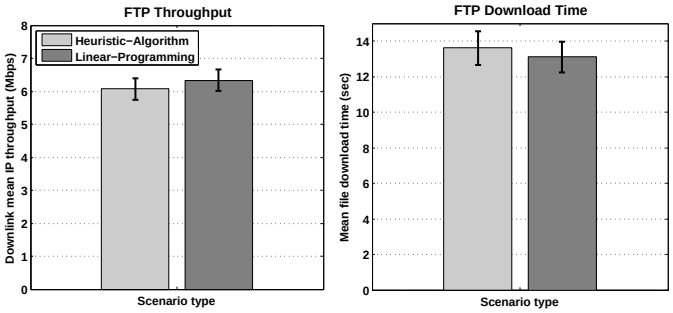


Fig. 17. Flow chart of the subprocess (A) in Fig. 16.

Solving the same problem using the "Linear Programming" approach serves UE1 completely over WLAN access network despite the fact that it has lower cost for LTE access link. This is because assigning all LTE resources to UE1 means that UE2 will have to be served over its WLAN access link that has the highest path cost. This would be a bad move that could decrease the over spectral efficiency of the network. Therefore, the "Linear Programming" approach takes an intelligent decision of serving UE1 over WLAN access network and keep LTE resources for UE2. Following this strategy, UE1 is served completely over WLAN access network with data rate of 16.67 Mbps and UE1 over the LTE access network with data rate of 20 Mbps. This way, total network capacity amounts to 36.67 Mbps which is 4.9% higher than that attained by using the heuristic approach.

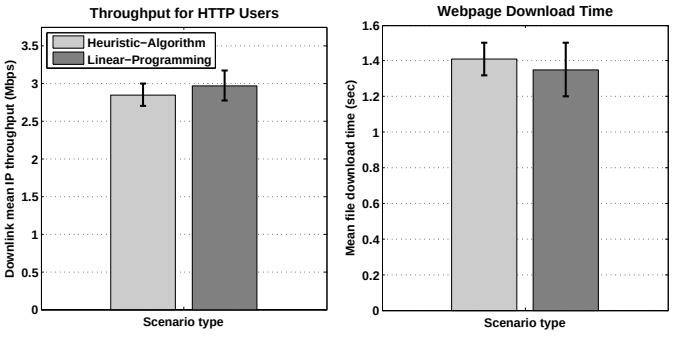
A sophisticated heuristic algorithm that mimics the "Linear Programming" approach in conceiving the effects of resource allocation of a user on the achievable spectral efficiency of the other users will be overly complex. This is because as the user count increases, each resource allocation will have to get feedback from many of the users in a recursive way. Based on this feedback, the algorithm would have to decide whether performing this resource allocation could degrade the achievable spectral efficiency of other users. Above all, devising such an advanced scheme would not offer a significant performance gain and would be against the idea of developing a simple alternative approach.

The performance of the developed heuristic approach is evaluated using the simulation scenario discussed in Section VI. This way, the results of the heuristic approach can be compared with that of the "Linear Programming" approach. Fig. 18 compares the performance of the FTP downlink application for two competing approaches. It is expected that the heuristic approach might not be able to deliver a performance matching to that of "Linear Programming". The FTP downlink throughput as well as file download time verify this expected behavior. However, the performance degradation is not significant. A comparison of numerical values reveals that the loss of performance is as low as 4%. A very similar observation is also made for HTTP application performance. In this case users encountered just 3% degradation in their QoE for webpage downloads. Moreover, the absolute values of increase in download times are in the range of milliseconds. Such a slight increase in download time remains unnoticed for human users.



(a) (b)

Fig. 18. FTP downlink performance comparison for "Heuristic Algorithm" and "Linear Programming" approaches.



(a) (b)

Fig. 19. HTTP downlink performance comparison for "'Heuristic Algorithm" and "Linear Programming" approaches.

Though non-realtime applications suffer slightly due to the use of the approach based on heuristic algorithm, the performance of realtime applications essentially remains unaltered. The reason behind this phenomenon has already been

discussed. That is, the foremost target of the heuristic approach is to satisfy the minimum data rate demands of all users. Owing to the fact that realtime applications require a fixed amount data rate, their minimum data rate demands are always fulfilled. Only after allocating the minimum required data rates to all users, the heuristic approach distributes the left-over resources among the non-realtime users. Therefore, if the resources are not utilized optimally, there will be fewer resources left to serve the TCP users with the data rates surplus to their minimum data rate demands.

The simulation results of realtime applications (i.e., VoIP and video) has not been shown here in order to avoid unnecessary repetitions.

VIII. Conclusion

This work highlights the importance of multihoming support in the integrated heterogeneous wireless networks of 3GPP and non-3GPP access technologies. The existing 3GPP specifications for integration of two types of the access technologies are extended to realize multihoming support for the users. Following the proposed extensions, a network simulation model is developed, where LTE and WLAN co-exist. This work also focuses on the problem of optimum resource utilization in such heterogeneous networks, where the users and network operators can take advantage of multihoming support. The problem of optimum network resource allocation is mathematically modeled using the Linear Programming technique. The proof of concept is provided through the simulation results. With the help of simulation results it is shown that the proposed scheme of resource allocation brings twofold benefits when compared to the 3GPP proposal. On the one hand, it significantly improves the network capacity and on the other hand it fulfills the user application QoS demands, which otherwise cannot be satisfied from QoS unaware non-3GPP access technologies. In addition to the Linear Programming based solution, this work also proposes a heuristic based method for network resource management. This method not only exhibits less computational complexity but also accomplishes a performance gain close to that attained by mathematical optimization techniques. This makes it feasible for use in real world network equipment.

References

[1] U. Toseef, Y. Zaki, A. Timm-Giel, and C. Görg, Optimized Flow Management using Linear Programming in Integrated Heterogeneous Networks, The Seventh International Conference on Systems and Networks Communications, Lisbon, Portugal, November 2012.

[2] 3GPP Technical Report TS 23.402, Architecture enhancements for non-3GPP accesses, 3rd Generation Partnership Project, v10.6.0, December 2011.

[3] Q. Song and A. Jamalipour, Network Selection in an Integrated Wireless LAN and UMTS Environment using Mathematical Modeling and Computing Techniques, IEEE Wireless Commun., June 2005.

[4] W. Song, H. Jiang, and W. Zhuang, Performance analysis of the WLAN-first scheme in cellular/WLAN interworking, IEEE Trans. Wireless Commun., vol. 6, May 2007.

[5] F. Yu and V. Krishnamurthy, Optimal Joint Session Admission Control in Integrated WLAN and CDMA Cellular Networks with Vertical Handoff, IEEE Transaction on Mobile Computing, vol. 6, January 2007.

[6] H. Zhai, X. Chen, and Y. Fang, How Well Can the IEEE 802.11 Wireless LAN Support Quality of Service?, IEEE Trans. Wireless Commun., vol. 4, 2005.

[7] W. Song, H. Jiang, and W. Zhuang, "Call admission control for integrated voice/data services in cellular/WLAN interworking", IEEE ICC06, vol. 12, June 2006.

[8] S. Lincke-Salecket, Load shared integrated networks, Personal Mobile Communications Conference, 2003.

[9] OPNET website, http://www.opnet.com, as accessed in June 2013.

[10] R. Wakikawa, V. Devarapalli, G. Tsirtsis, T. Ernst, and K. Nagami, Multiple care-of addresses registration (RFC 5648), 2009.

[11] G. Tsirtsis, H. Soliman, G. Giaretta, and K. Kuladinithi, Flow bindings in mobile IPv6 and NEMO basic support (RFC 6089), 2010.

[12] G. Tsirtsis, G. Giaretta, H. Soliman, and N. Montavont, Traffic selectors for flow bindings (RFC 6088), 2011.

[13] U. Toseef, Y. Zaki, A. Timm-Giel, C. Görg., Development of Simulation Environment for Multi-homed Devices in Integrated 3GPP and non-3GPP Networks, The 10th MobiWAC conference, Paphos, 2012.

[14] SAIL consortium, D.C.1: Architectural Concepts of Connectivity Services, July 2011.

[15] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications ISO/IEC 8802-11:1999(E); ANSI/IEEE Std 802.11.

[16] G. Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 3, pp. 535-547, March 2000.

[17] R. Litjens, F. Roijers, J. L. van den Berg, R. J. Boucherie, and M. Fleuren, Performance Analysis of wireless LANs: an Integrated Packet/Flow Level Approach, ITC Conference, Berlin, Germany, August 2003.

[18] J. Klaue, B. Rathke, and A. Wolisz, EvalVid - A Framework for Video Transmission and Quality Evaluation, 13th International Conference on Modeling Techniques and Tools for Computer Performance Evaluation, pp. 255-272, Illinois, USA, September 2003.

[19] Recommendation ITU-T G.722.2, "Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB)", Approved in July 2003.

[20] U. Toseef, M. Li, A. Balazs, X. Li, A. Timm-Giel, C. Görg, Investigating the Impacts of IP Transport Impairments on VoIP service in LTE Networks, 16th VDE/ITG Fachtagung Mobilkommunikation, 2011

[21] 3GPP Technical Report TS 36.213, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, v10.2.0, June 2011.

[22] IBM CPLEX Optimizer, http://www.ibm.com, as accessed in June 2013

[23] 3GPP Technical Report TS 25.814, Physical layer aspects for E-UTRA, 3rd Generation Partnership Project, v.7.1.0, September 2006.

[24] N. Zahariev, Y. Zaki, T. Weerawardane, C. Görg, and A. Timm-Giel. Optimized service aware lte mac scheduler with comparison against other well known schedulers. In 10th International Conference on Wired/Wireless Internet Communications, WWIC 2012, June 2012.

[25] S. N. K. Marwat, T. Weerawardane, Y. Zaki, C. Görg, and A. Timm-Giel, Design and Performance Analysis of Bandwidth and QoS Aware LTE Uplink Scheduler in Heterogeneous Traffic Environment, 8th International Wireless Communications and Mobile Computing Conference, Limassol, Cyprus, August 2012.

[26] U. Toseef, Y. Zaki, L. Zhao, A. Timm-Giel, and C. Görg, QoS Aware Multi-homing in Integrated 3GPP and non-3GPP Future Networks, The 7th International Conference on Systems and Networks Communications, Lisbon, Portugal, November 2012.

[27] Y. Zaki, T. Weerawardane, C. Görg, and A. Timm-Giel, Multi-QoS-Aware Fair Scheduling for LTE, VTC Spring, 2011.

[28] Recommendation P.800, Methods for subjective determination of transmission quality, Approved in August 1996.

Statistical and Simulation Analysis of Photonic Packet Switching Networks - The Emerging Functions Concept

Antonio de Campos Sachs, Ricardo Luis de Azevedo da Rocha, Fernando Frota Redígolo, and Tereza Cristina Melo de Brito Carvalho

Departamento de Engenharia de Computação e Sistemas Digitais (PCS)
Escola Politécnica da Universidade de São Paulo (EPUSP)
São Paulo, Brazil
antoniosachs@larc.usp.br; luis.rocha@poli.usp.br; fernando@larc.usp.br; carvalho@larc.usp.br

***Abstract*—A transparent Optical Packet Switching network designed with Emerging Functions Concept is analyzed as a complex system showing desirable characteristics that emerges from the bottom-up organization. The objective is to describe the Emerging Functions Concept applied to different topologies. The bottom-up organization is obtained from simple rules executed by individual nodes. It is possible to create those simple rules, or fundamental individual functions executed by individual nodes, in order to potentiate scalability, robustness, and other desirable characteristics referred to as Emerging Functions. The scalability is enabled after the avoidance of all long distance signalizations. The robustness emerges from next neighbor signalizations and from the mesh topology with a large number of alternative paths. All emerging functions are better observed for networks with more nodes. The concept is described for the Manhattan Street Network to show the emergence of scalability and robustness. It is also applied to the National Science Foundation Network in order to generalize the procedure showing its applicability to nonsymmetrical and more real topologies. Failure effect segregation is shown for 256 nodes Manhattan Street Network.**

Keywords-complex network; emerging function; scalability; robustness; nondeterministic physical layer

I. INTRODUCTION

One important constraining factor for the scalability of the number of nodes in a network is the long time necessary for signalization between two distant nodes. The current approach, which treats the Optical Packet Switching (OPS) network as a complex system and the network nodes as autonomous entities, utilizes the Emerging Function Concept (EFC) described in the EMERGING-2012 international conference [1]. That approach avoids long distance signalization. A packet is sent from source to destination without any previous path determination. The routing procedure needs to use the shortest path table previously calculated at the moment of the network initialization. From that shortest path table, each node knows the address of the output port that corresponds to the shortest path connecting itself to any other node. Each packet carrying the destination address can find the path from source to destination from node to node in a multi hop schema using the output port corresponding to the shortest path or the alternative port in those cases in which the preferred one is not available.

The use of a simple switching device without optical buffers that forwards the arriving packet without delay to the preferential output port or to the alternative one is a procedure referred to as "hot potato routing" [2]. That operation can be performed by using an optical sample removed before a FDL (Fiber Delay Line). This sample can be converted into electrical media enabling the logical treatment that is performed by conventional electronic circuitry. The optical switch can easily be constructed based on SOA (Semiconductor Optical Amplifiers) devices [3]. The optical switch can be positioned to address the packet to the correct output port before the arrival of the packet that is traveling through the FDL. Such operations have been adopted since the precursor projects KEOPS [4] and DAVID [5].

There is an option to avoid the conversion from optical to electrical media that consists in the utilization of new photonic devices that can do all the jobs, including logical operations. With those photonic devices the switching operation can be performed in a fully optical process [6]. The network described herein works for any technology employed for reading the address and forwarding the packet to the output port. Whatever the technology used inside the node, the network can operate as a complex system, with a bottom-up organization. Each node has the autonomy to carry on the switching operation, performing its work exclusively with information locally obtained.

The use of a large number of nodes with a large number of alternative paths, provided by the mesh topology, is known to be important for the network survivability. Since the beginning of the digital telecommunication technology Baran [2] worked with mesh topology and got very strong robustness for a network with a large number of nodes. The survivability of a complex network is associated to the intrinsic robustness of complex systems. Carlson and Doyle [7] claim that all complex systems are intrinsically robust for the most frequent daily events; however they are very fragile due to rare and unexpected environment events. Barabási [8] claims “hubs make the network robust against accidental failures but vulnerable to coordinated attacks”. All agree that complexity is intrinsically related to robustness.

In addition to robustness, or acting together with the robustness, the self-organization is another property associated to a complex system [9]. With the self-organization ability the network gains autonomy and can

reduces the external management effort. That characteristic is referred to as Autonomic Network Management (ANM). A survey into ANM is presented in [10]. The present work contributes to the ANM effort with the creation of a pure physical layer mechanism, that provides traffic self-distribution, scalability, self-protection and restoration.

Self-organization or bottom-up organization is a convenient and necessary approach to deal with large size networks [11]. Not restricted to the future, the complexity is already a reality at the optical metropolitan networks that is characterized by large number of nodes necessary for the capillarity, accessibility and high capacity. The scalability of those networks is limited by the utilization of a strictly deterministic approach. The present proposal brings the non-determinist behavior to the physical layer and calls attention to the convenience of using the complexity approach in all network layers.

Next, Section II describes aspects related to the EFC and how it can be applied to a network. Section III describes the network architecture and the operations responsible for the packet insertion without collision and for the protection and restoration properties. Section IV presents a generalization that shows the applicability of the EFC to virtually any topology. Section V describes the theory adopted for the statistical model and aspects of calculation performance. Section VI describes the simulation performed to validate the results achieved by the statistical model. Section VII presents discussions about the failure distribution effect for a 256-node study case. Section VIII presents the final conclusion and future work.

II. Emerging Function Concept Applied to a Network

The term Emerging Function is used in a number of different areas, such as physics, chemistry and biology. Although there is no single formal definition for the term, two main definitions can be inferred:

- A function that is not regularly present in a system and appears or is activated automatically in an emergency situation;
- A function that is always present in a system (it characterizes the system) and emerges from simple operations executed by its individual parts.

An emerging function is associated to the whole system and not to its individual parts although its emergence is the result of small changes in the normal operations (first definition) or of regular operations of individual parts of the system (second definition).

A system based on emerging functions can be classified as a bottom-up organized system or, equivalently, a self-organized system [9] and it is associated with a complex system composed by a large number of individual units following simple operation rules. It is difficult to deal with such complex systems, with a large number of elements, just employing a classical and reversible treatment that calculates all the possible events in all the system components. The models considering the probability of transition from one state to the next describing the system evolution seem to be a more feasible strategy. That is also the same strategy found in the chaos theory [12], where the final results cannot be derived from the initial conditions because there is high sensitivity to tiny fluctuations in the initial conditions.

The network routing layer (OSI network layer 3) does not control the routing function herein. It emerges from simple fundamental functions executed by each node individually. There is no high level entity responsible for the operation of the switches or for the path followed by each packet in the network. Instead, the node operation is based on the local situation and on the packet header information: each packet is sent to the preferred output port, or sent to the available port if the preferred one is busy. This operation rule, by itself, turns the network self-organized or bottom-up organized, and provides autonomic network operation. Therefore, it is possible to consider "routing" as a function emerging from individual node operations or, in other words, that routing is an Emerging Function.

Traffic distribution, which can be considered the set of all routes, is also an Emerging Function. As the shortest path is not always the one chosen, the traffic distribution obtained is better than the one obtained using only the shortest path.

The access to the network is made only if there is a time interval to accept the new packet without collision. This is possible because of a fiber delay line (FDL) positioned before any input port. Collision avoidance can also be interpreted as an Emerging Function, since it is not executed by any higher protocol layer, but it shows up after the careful local insertion procedure.

Protection is an important network function that can be enabled by means of the insertion of an extra individual node operation function based on a backward signalization sent to all the input ports. The output ports integrity can be checked through the signalization received from the next node. Protection can also be considered as an Emerging Function.

III. Network Architecture and Operation

The network architecture is based on the "Hot Potato Heuristic Routing Doctrine" [2] made up by network nodes executing simple well-defined rules. A set of Emerging Functions arise from those simple rules. The network complexity is related to its size and the number of nodes. Each node, in contrast, is idealized to be simple. The first simplification is the omission of optical buffers. Without optical buffers, it is necessary to use symmetrical nodes in order to avoid packet losses. In symmetrical nodes, with the same number of input and output ports, there is always a free output port for any arriving packet. The simplest possible symmetrical node to be connected in a mesh topology is a 2x2 switching node with two inputs and two outputs as shown in Fig. 1. Without optical buffers, the optical packet is immediately forwarded to the output port and the front part of the packet starts traveling through the next link while the back part of the packet is still arriving to the node input port. An eventual packet P2, arriving when the node is occupied, is immediately forwarded to the available output.

The Manhattan-Street Network (MSN) [13] was chosen as the main topology for the scalability analysis, but any other mesh topology can be considered. This particular

choice facilitates the calculations for increasing the number of nodes without changing the network symmetry. As an example of applicability to any topology the EFC was also applied to the National Science Foundation Network (NFSNet) [14].

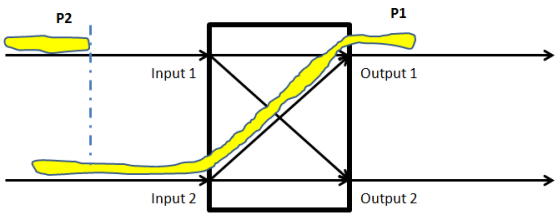


Figure 1. Simple optical switching node with two inputs and two output without optical buffer.

A. *Packet insertion without collision*

Each packet goes through a Delay Line Fiber (DLF) before arriving to the optical switch. An optical splitter takes a sample of the optical signal before the DLF. That sample is analyzed by a logical circuitry that accounts for reading the packet header, consulting a previously stored routing table and sending a signal to the optical switch control. The optical switch is prepared for dealing with that particular packet before its arrival. In addition to providing a secure time interval for the switching positioning the DLF has a second important function that consists in a sight of the near future that permits the insertion of new packets without collision. Fig. 2 illustrates the DLF functionalities. A small time interval, represented by t_1 in Fig. 2, is necessary for the header reading, logical procedure and switch positioning. That time interval is very small. One hundred nanoseconds should be enough for t_1. It is required a longer time interval for introducing a new packet in the network. The remaining time t_0-t_1, where t_0 represents the total time for the light to travel inside the DLF, defines the maximum packet size that can be inserted into the network without collision. For example, in a 10 Gb/s optical link, one single bit spends 0.1 ns in propagation and 12 thousand bits (1500 bytes) would require 1.2 µs. Taking t_0-t_1 equal to 1.2 µs and t_1=100 ns this results in t_0=1.3 µs. Thus the DLF, in this case, should be 260 meters long. A longer DLF can be adopted for jumbo packets or for a set of packets put together into a burst in an Optical Burst Switching (OBS) technology.

Concerning the example in Fig. 2, after time interval t_1, the node already knows which output port the packet is going to use. Given that P_3 arrived before P_1, it had priority and chose the port first. In that case, P_3 is going to use output 2 and P_1 is going to use output 1. In that situation, there is enough time for the insertion of a new packet P_2 with size $t_2 < (t_0-t_1)$. P_2 can be inserted directly into the output 1 link and can be completely inserted before the arrival of P_1.

Occasionally, packets need to be stored for a long period of time before finding an appropriate time interval to be inserted. In this case, several packets can be lost at the full buffer condition. The buffer has a finite size and need to discard packets that arrives in that full condition. In order to minimize such losses, the suggestion is to use the FILO (First In Last Out) buffer strategy that will avoid small packets disposal while a long packet is waiting for a long and rare free time interval to be inserted.

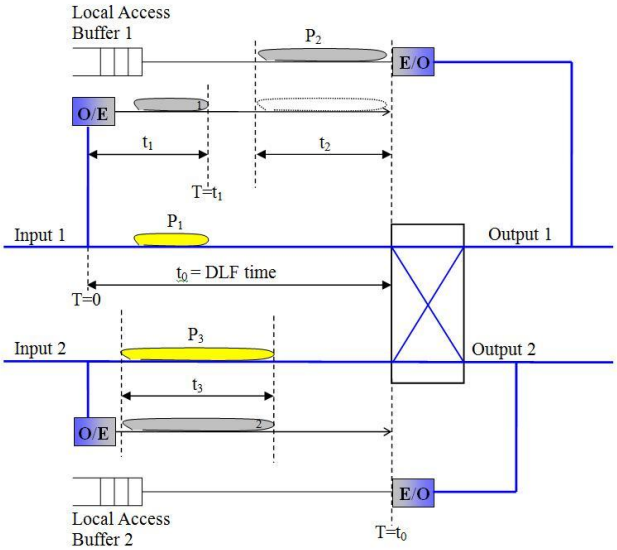


Figure 2. Delay Line Fiber (DLF) architecture.

B. *Protection emerging function*

The implementation of the protection emerging function requires the differentiation between the two output ports in order to define different link sub-domains. The idea is to deactivate only one sub-domain in the case of link failure. Fig. 3 is a MSN with 36 nodes showing clockwise and counterclockwise sub-domains as first described in [15]. Each node belongs to two sub-domains and each sub-domain contains four nodes. It is desirable to have a small number of nodes in each sub-domain because an entire sub-domain needs to be deactivated to deal with the failure.

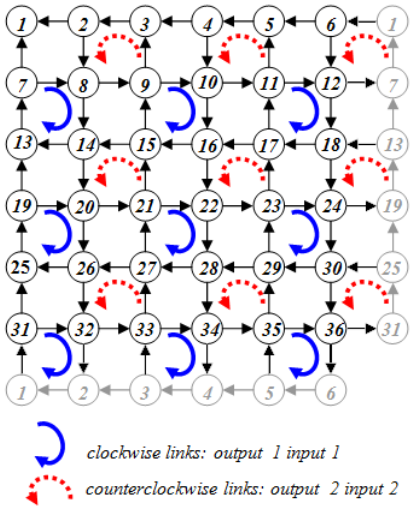
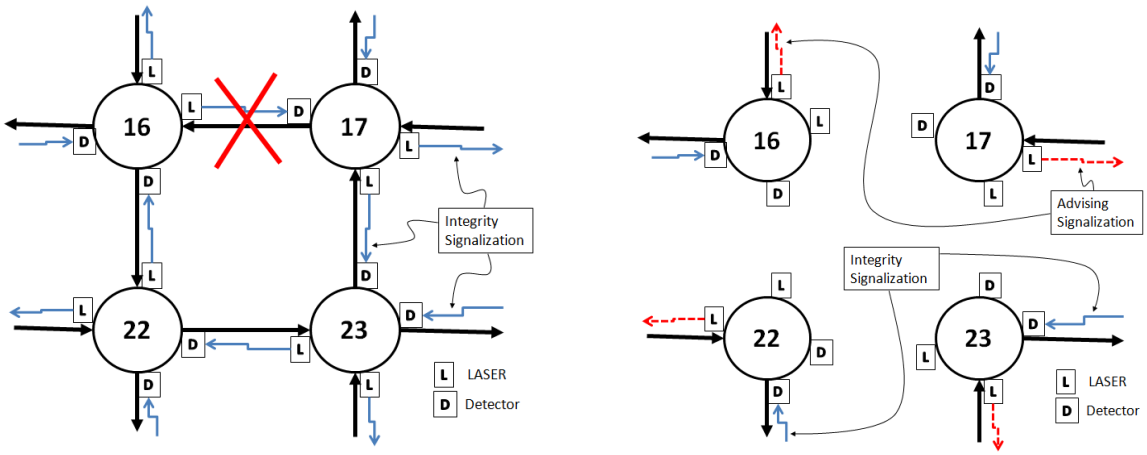


Figure 3. MSN organized with clockwise and counterclockwise link sub-domains [15].



(a) Before failure (b) After failure

Figure 4. (a)Before failure, all nodes send a backward (opposed to the packet direction) Integrity Signalization. (b)After failure the Integrity Signalization is turned off for all links belonging to the failed sub-domain. The signalizations addressed to the four next neighbors that can access the failed sub-domain are changed to a second type of signalization called Advising Signalization represented by dashed red line arrow (see text for more details).

After organizing the network links in small sub-domains, it is possible to create the protection function by including an operation rule for all network nodes. This rule is constituted by a continuous optical signal sent backward from all the nodes. That signal is denominated Integrity Signalization. It is received by the backward next neighbor indicating that the corresponding output is properly connected. The Integrity Signalization can be implemented by supplying each node with two lasers and two photo-detectors as shown in Fig. 4.

Nodes 16, 22, 23 and 17 are connected by four links in a counterclockwise sub-domain as shown in Fig. 4 as a zoom of the same nodes in Fig. 3. Each node has two lasers sending a continuous optical signal backwards (opposite the packet directions), and two detectors placed to receive the laser signals from the downstream neighbors. In the case of one link interruption, the detector that first stops receiving the signal turns off the laser corresponding to the same sub-domain. In Fig. 4, for example, a failure occurred in the link that connects node 17 to node 16. After the failure, node 17 does not receive the integrity signalization and stops sending packets through that link and also turns off the laser corresponding to node 23 that belongs to the same sub-domain as node 16. Node 23, in turn, after stopping receiving the integrity signalization from node 17, turns off its laser addressed to node 22. This one does the same and the final result is that all the four links in the ring are forced to be interrupted. The four lasers are turned off and, without the Integrity Signalization, no packet can be sent through those links.

The integrity signalization is enough for the protection schema operation, but a second type of signalization has been implemented aiming at better network performance. That second type of signalization consists in sending a different type of signal to the nodes outside the failed sub-domain to inform that the next node belongs to a sub-domain in failure, although the link still works properly. The implementation of that advising signalization can be performed by a square wave light signal replacing the continuous light signalization or, alternatively, the continuous laser can be used with half optical power to differentiate from the full optical power of the regular link integrity signalization. The implementation of both signalization types, indeed, can also be implemented by the utilization of smart photonic devices transmitting digital signalization and processing the signal in a fully optical process.

In the case of Fig. 4 (right side figure), the failure, at the same time, causes four links to stop the integrity signalization and also to start the advising signalization outside the failed sub-domain. From Fig. 3, it is possible to recognize that the nodes receiving the advising signalization are nodes 18, 10, 21 and 29.

The action of the node receiving the advising signalization is to deflect all packets to the other output port (to the port that is receiving the integrity signalization), with the exception of the packets addressed to the node that is sending the advising signalization. This is the only way a failed sub-domain node can receive a packet. That deflection corresponds to an adaption in the local preferential port table. All the nodes at a failed sub-domain, nevertheless, continue to work with only one input and one output port. In that situation, they continue to be symmetrical (one input and one output) and can transfer all the packets arriving from the input to the always available output port. All the other nodes, far from the failure, have no information about the failure and their procedure remains the same, including the use of the same preferential output port matrix.

Both modes of signalization were implemented in the calculations, and the results are shown in Fig. 5 for 16 nodes (N=16) and for 256 nodes (N=256). The caption ending in "F" refers to the first level protection schema, characterized by not using the advising signalization type. The caption ending in "F2" refers to the second level protection schema that includes the advising signalization type. For the 16-node

case, the second level protection schema (N=16F2) shows an interference of the failure remarkably smaller than that observed at the first level protection schema (N=16F). The advising procedure reduces the mean number of hops.

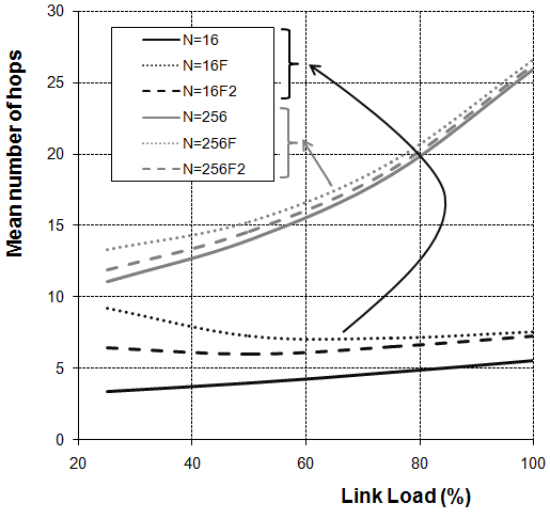


Figure 5. Mean number of hops degradation after failure for two types of signalizations.

One additional feature obtained with the second type of signalization is the correction of a strange behavior that occurs for low charge condition. In that region (Link Load < 50%) the failure causes a large enhancement in the number of hops and it is quite odd to see the number of hops decreasing for higher load condition (curve N=16F in Fig. 5). That behavior can be explained by the fact that in the low load condition, the packets take the preferential output port more often as compared to the large load condition. Consequently, the packets are more often forced to proceed through the failed region. At the large load condition, the packets are naturally dispersed and the failure does not cause too much degradation to the number of hops. The failure is more efficiently avoided with the second signalization, minimizing this effect. All the unnecessary trial through the failed region is avoided.

C. *Restoration emerging function*

After failure, the use of the same preferential output port matrix causes performance degradations. The network, however, can operate under acceptable condition, while waiting for physical reparation. After reparation, in order to get an automatic restoration, it is necessary to implement additional fundamental functions, to be executed by all nodes, only in the emergency case. First, the nodes involved in the failure need to change their states from “normal” to “emergency”, and, in that new state, it is prepared for the restoration. Second, a node in the “emergency” state must restart its backward integrity signal to reset the system after physical reparation of the failure.

Based on these fundamental functions, once the link is repaired, the integrity signal is propagated backwards, restoring the sub-domain. Again, the functions executed by individual nodes are responsible for the emergence of the global function restoration. That means that restoration is also considered an Emerging Function. It is important to stress that those functionalities can be easily implemented in practice.

For long emergency state time, it may be necessary to consider a network reset to get a new preferential output port matrix for the new topology (topology with failure and disabled sub-domain). In this case, it is convenient to store the old preferential output port matrix or to reset the network again after failure reparation. The restoration function should recover the original matrix as soon as it receives the integrity signal from another node and a new signalization should inform the entire network to reset or to recover the original topology matrix. That operation, including signalization for the entire network (twice), cannot be considered as Emerging Function because it is not carried out only by localized operations. It takes too much time and causes scalability limitations.

IV. GENERALIZATION

Optical network topologies, in real world, are not like the MSN but something more like the NFSNet (National Science Foundation Network) [14] that is shown in Fig. 6. The NFSNet has bidirectional links and each connection represents two optical fibers, one input and one output. Most of the nodes have three inputs and three outputs. Nodes 4 and 10 have four bidirectional connections and nodes 5 and 12 have only 2 inputs and 2 outputs. This network can be transformed into an equivalent representation that exhibits only 2x2 switches as described in sub-section A or it can be treated considering nodes 3x3 and 4x4 according to the discussions presented in sub-section B.

A. *Representation using only 2x2 nodes*

The same procedure employed for the MSN can be adopted for NFSNet by reconstructing all 3x3 and 4x4 nodes, present in the NSFNet, as a set of unidirectional 2x2 nodes. Each three bidirectional connection node can be represented, as shown in Fig. 7a, by three unidirectional 2x2 nodes. Generalizing this idea, each node with k bidirectional connections can be replaced by k unidirectional 2x2 nodes. After reconstruction 42 nodes containing only 2x2 optical switches as shown in Fig. 7b can represent the NFSNet .

The definition of link sub-domains, applied to MSN for developing the protection emerging function, can also be applied to NFSNet. The main idea is to choose two types of sub-domains such that each 2x2 node has one input and one output connected to the first type and the other pair input/output connected to the second type. Another idea to be reassigned is related to the size of each link sub-domain. Each sub-domain needs to be small because it is going to be deactivated in case of failure. The smaller the sub-domains, the better the network performance after failure. After those considerations, the protection can be applied to NFSNet using the same backward signalization adopted for the MSN, and the sub-domain types can be considered separating long-distances and short-distances links.

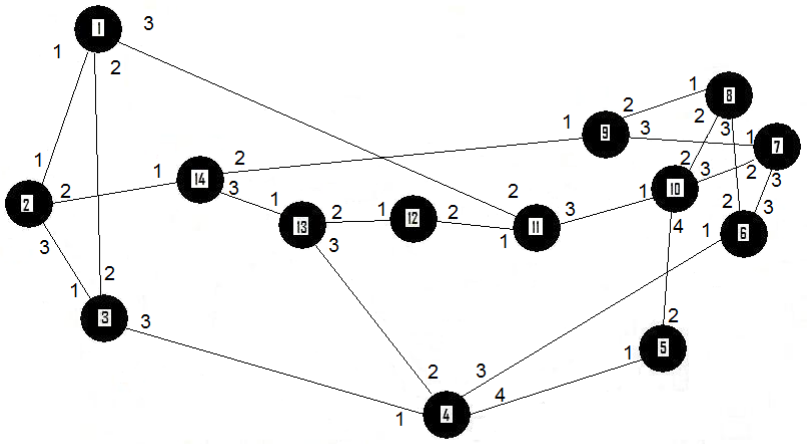
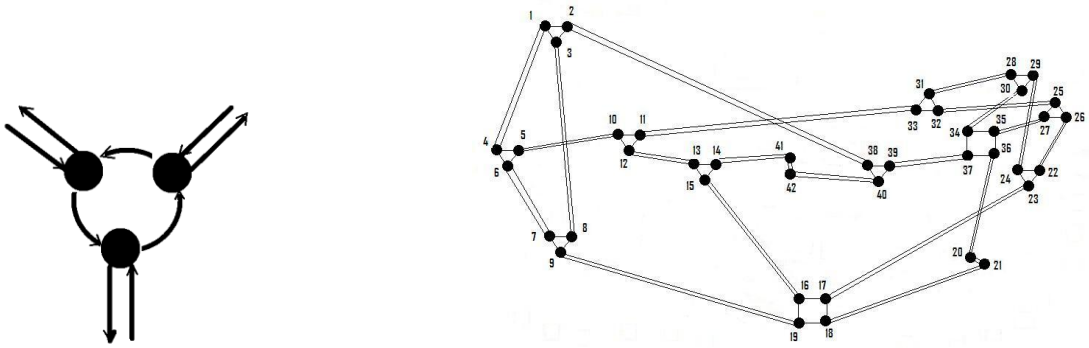


Figure 6. NFSNet with 14 nodes. All links are bidirectional.



(a) (b)

Figure 7. Three 2x2 nodes replacing one node with three bidirectional connections (a) and the NFSNet designed with 42 nodes (b). A simple optical switch with two input ports and two output ports represents each node.

As illustrated in Fig. 7a, all nodes have a pair input/output for the long distance links and a second pair for the short distance links. The separation of groups of sub-domains to be disconnected in case of failure considers that all long distances links belong to one type of sub-domain and all short distance links constitute the second type of sub-domain. Each node belongs to two types of sub-domain and all the long distance sub-domains have only two links, while the short distance sub-domain can have 2, 3 or 4 links depending on the case.

After the transformation of the original NFSNet (Fig. 6), by using the alternative shown in Fig. 7a, it results in a network with 42 nodes as shown in Fig. 7b.

B. Generalization for NxN node type

It is also possible to consider that all 2x2 nodes physically located in the same place can, almost instantly, have the same information. With the knowledge of the arriving packet and its destination, the network performance seemed to be better, but the calculations may be harder to be implemented because it is necessary to find out at least two better output port options to send each packet. Considering the original NFSNet topology (Fig.6), with 2x2, 3x3 and 4x4 nodes, it is necessary to find out a second preferential output port for all 3x3 and 4x4 nodes. It may not be important to deal with the third preferential port because in most cases it is the last available port and in the 4x4 nodes, the third option can be considered to be the path going back to the same place it came from. If the smallest possible path (first option) has one hop, the second option can have two hops and considering the third option to be returning to try again, it has only three hops. In some cases the way back (return to try again) is the second option.

After those considerations, it is possible to construct a table of preferred output port just for the first and second preferential output ports. Table I shows an example obtained from the NFSNet represented in Fig. 6. Columns represent the actual position of a packet, lines represent the final destination and the numbers represent the first or the second options output port. The number of each output port can be confronted with the numbers presented in Fig. 6. For example, from node number 1 (first column) to node number 3 (third line), the first option is port number 2 (Table Ia) and the second option is port number 1 (Table Ib).

TABLE I

First option (a) and second option (b) for sending packets from any of the 14 nodes in the NFSNet shown in Fig. 6. Columns 1 to 14 represent the actual position of a packet, lines 1 to 14 represent the final destination and the numbers inside the table are the first or second options output port numbers.

0	1	2	1	1	1	2	2	1	1	2	2	3	1
1	0	1	1	1	1	1	1	1	1	2	2	1	1
2	3	0	1	1	1	3	3	1	4	2	2	3	1
2	3	3	0	1	1	3	3	3	4	2	1	3	3
2	3	3	4	0	1	2	2	3	4	3	2	3	3
2	3	3	3	1	0	3	3	3	3	3	1	3	2
3	2	3	3	2	3	0	3	3	3	3	2	3	2
3	2	3	3	2	2	3	0	2	2	3	2	3	2
1	2	1	3	2	3	1	1	0	3	3	1	1	2
3	1	2	4	2	3	2	2	3	0	3	2	3	2
3	1	2	1	2	3	2	2	3	1	0	2	2	1
3	1	2	2	1	1	2	2	1	1	1	0	2	3
1	2	3	2	1	1	3	3	1	4	1	1	0	3
1	2	1	2	1	1	1	1	1	3	2	1	1	0

(a) First option

0	3	1	2	2	3	3	3	3	4	1	1	2	3
2	0	2	2	2	3	3	3	2	4	1	1	3	3
1	1	0	2	2	3	1	1	3	1	3	1	1	3
1	2	1	0	2	3	2	2	1	3	3	2	2	1
3	1	2	2	0	3	3	3	2	3	2	1	2	2
3	2	1	2	2	0	1	2	2	2	2	2	2	3
1	1	2	4	1	2	0	1	2	2	1	1	1	3
1	1	2	4	1	3	1	0	3	3	1	1	1	3
3	1	3	2	1	2	3	3	0	2	2	2	3	3
2	3	3	3	1	2	3	3	2	0	1	1	2	1
1	3	2	4	1	1	3	1	1	2	0	1	3	3
1	2	3	1	2	3	3	3	3	4	3	0	1	1
2	3	1	4	2	3	1	1	3	1	2	2	0	1
0	0	3	1	2	3	3	3	2	4	1	2	2	0

(b) Second option

Results for the mean number of hops as a function of the network load are shown in Fig. 8. The case of NFSNet with 14 nodes and 42 links as compared to a MSN with 16 nodes and 32 links is shoen. The NFSNet is better (uses less number of hops) because of the larger number of links available.

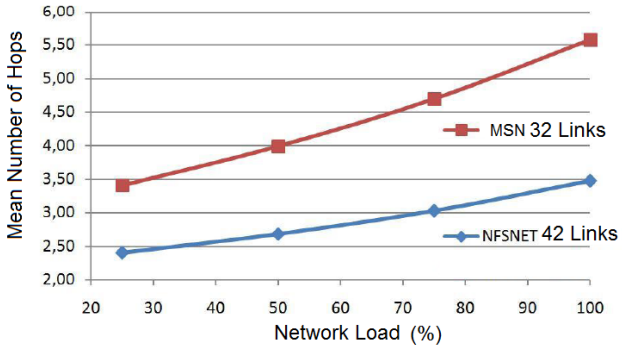


Figure 8. Mean number of hops of NFSNet 14 nodes and 42 links compared to MSN with 16 nodes and 32 links.

V. CALCULATIONS

To deal with scalability, the number of nodes can be higher than practical calculations can support. It is impractical to implement calculations for an arbitrarily large number of nodes. In order to minimize the time and memory used, the connection matrix "c" and the preferential output port matrix "pp", were calculated separately. Data were saved in files that could be interpreted by the main program. The algorithm employed to obtain the MSN connection matrix "c" is shown in sub-section A. The shortest path calculation is presented in sub-section B. The algorithm description for the mean number of hops calculation is presented in sub-section C. The model validation carried out by comparison with the simulation model is presented in sub-section C. One important result, the segregation of the failure effect, is presented in sub-section D.

A. Algorithm for MSN Connection Matrix

The connection matrix "c" for a MSN with N nodes is a matrix NxN where the columns represent all the N network nodes. Each column has only two non-zero element in the position corresponding to the two nodes that can be reached directly. Instead of using number "one" to represent an adjacent element as in an ordinary adjacency matrix, here two different values are used to represent two different sub-domains.

It is not necessary to find out a general formula for all MSN types. The goal is just to obtain a sequence of MSN types, each one with an increased number of nodes. That calculation was obtained by constraining the number of nodes "N" to those that are the square of an even number "n". With that restriction ($N=n^2$, n even), an algorithm to obtain the connection matrix for an arbitrarily large number of nodes was implemented. Following Fig. 3 for link sub-domain identification, the algorithm constructs a matrix that attributes the value "one" to the output port connected to the counterclockwise link sub-domain (red in Fig. 3) and attributes the value "two" to the output port connected to the clockwise link sub-domain (blue in Fig. 3). The code lines for the algorithm are presented below in a syntax used for the scilab simulation platform [16].

```
//OUTPUT PORT 1:COUNTERCLOCKWISE (RED)
 for i=1:2:n-1; //ODD LINES
  for j=2:2:n-1; //EVEN COLUMNS
  c((i-1)*n+j,(i-1)*n+j+1)=1; c(i*n+j,(i-1)*n+j)=1;
  c(i*n+j+1,i*n+j)=1; c((i-1)*n+j+1,i*n+j+1)=1;
  end;
  c(i*n,i*n-n+1)=1; c(i*n+n,i*n)=1;
  c(i*n+1,i*n+n)=1; c(i*n-n+1,i*n+1)=1;
 end;
```

```
//OUTPUT PORT 2:CLOCKWISE (BLUE)
  for i=2:2:n-1; //EVEN LINES
    for j=1:2:n-1; //ODD COLUMNS
    c((i-1)*n+j+1,(i-1)*n+j)=2; c(i*n+j+1,(i-1)*n+j+1)=2;
    c(i*n+j,i*n+j+1)=2; c((i-1)*n+j,i*n+j)=2;
    end;
  end;
  for j=1:2:n-1;
    c(j,j+1)=2; c(N-n+j,j)=2;
    c(N-n+j+1,N-n+j)=2; c(j+1,N-n+j+1)=2;
  end;
```

B. *Shortest Path Calculation*

The shortest path to reach the destination is calculated once for a non-failed MSN topology. As the packet can be deflected to any output port, it must be able to find out the destination shortest path from any place in the network and not only from the origin. The packet is informed about the shortest path through a preferential port matrix "*pp*" with dimensions *NxN*, where *N* is the total number of nodes. Each column of the "*pp*" matrix represents the actual position of a packet. The lines represent the final destination and the matrix elements are numbers indicating the best option: number 1 for output 1 or number 2 for output 2 or number 3 to indicate the "don't care" condition, in which there is a shortest path starting from both outputs.

The preferential port matrix "*pp*" is constructed from connection matrix "*c*". This is done column by column, in a nondeterministic procedure [17] just at the beginning of the algorithm. The procedure is nondeterministic because it is necessary to calculate the smallest path starting from all possible packet positions. From column number 1, which represents node number 1, a tree is established and the starting level is called Level Zero. Based on the Manhattan-Street network with *N* = 36, presented in Fig. 3, the calculation procedure considers the evolution of the tree presented in Fig. 9. Level 1 is formed by the two branches going down from node 1 to node 6 and from node 1 to node 31. Those are the two nodes directly reachable from node 1. The "*pp*" matrix lines 6 and 31 can now be filled with the preferential output port, respectively number 2 and number 1. Those numbers can be obtained from the already known "*c*" matrix for connections 1 to 6 and 1 to 31. The first connection (output port number 2) will be used by all nodes found on the left side and the second (output port number 1) will be the preferential port for all nodes found on the right side of the tree.

Following this procedure, the number of elements in each level would increase exponentially. That is a strong limitation for the number of nodes scalability. To fix that scalability problem, the adaptive tree mechanism was deployed [17]. In that mechanism, the tree is adapted at each level and all the nodes that were already used at earlier levels are removed. This is exemplified in Fig. 9, where node 1 was removed twice in Level 4 because it was already used in Level zero at the beginning. Several branches are removed in Level 5. For example, node 3 in Level 4 is connected to nodes number 2 and 33 (the connections can be identified in Fig. 3). Those nodes were already reached at Level 3. That adaptive procedure reduces the computational complexity from exponential to polynomial growth as a function of the total number of nodes.

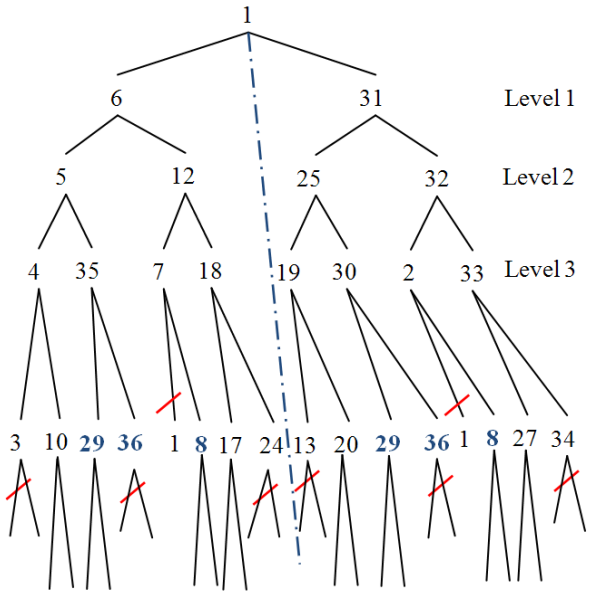


Figure 9. Adaptive tree for shortest path calculation.

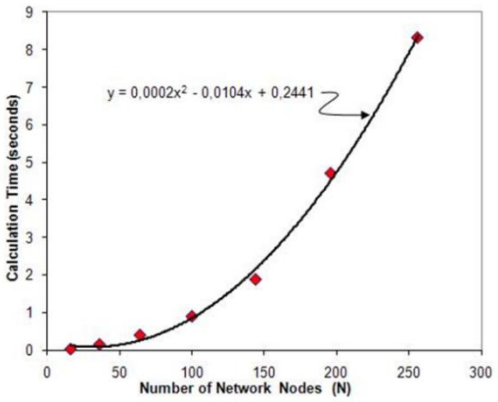


Figure 10. Calculation time to determine preferential output port.

In addition, the repetition inside the same level needs an extra verification. It is verified if they belong to a different half of the tree, which means that the destination can be reached with the same number of hops from any output port from node number 1. This is known as the "don't care" situation and the "*pp*" matrix element is set to be equal to 3.

The procedure continues until column one in the "*pp*" matrix is completed. And then the procedure is repeated for all the other columns, which represent all the possible origin nodes. The resulting time for "*pp*" calculation shows a quadratic polynomial growth with respect to the number of nodes as can be seen in Fig. 10, which was obtained by performing all calculations in a Pentium 4 personal computer with 3GHz CPU and 2GB memory.

There are other algorithms that could be used to find out the shortest path. Dijkstra's algorithm, most commonly used,

would need to be modified in order to consider the "don't care" cases and not only the traditional shortest path first (spf) found. The following algorithm based on the adaptive tree (Fig. 9) is comparable to a modified Dijkstra's algorithm and can meet all needs for this case, including the registration of the don't care situations. The lines of the code presented here allow calculating preferential port matrix "*pp*" from any number of nodes *N* and for a MSN connection matrix "*c*" previously calculated as described in sub-section A. The code syntax is also adapted to the scilab platform [16].

```
for j=1:N; fff=find(c(:,j)); ppp(fff(1),j)=0; ppp(fff(2),j)=0; end;
 for ii=1:N;
[w]=find(c(:,ii),2); per=[w(1) w(2)]; nivel=1;
while sum(ppp(:,ii))>0; nivel=nivel+1;
  perpro=[];
  k1=sum(size(per))-1;
  for i=1:k1;  [w]=find(c(:,per(i)),2);
    for j=1:2;
      ori=ii; des=w(j); pai=per(i);
      if ppp(des,ori)==1;
        if pp(pai,ori)<>0;
          pp(des,ori)=pp(pai,ori);
          ppp(des,ori)=0;
          pniv(des,ori)=nivel;
          perpro=[perpro w(j)];
        end;
      elseif
pp(des,ori)<>pp(pai,ori)&pniv(des,ori)==nivel&pp(pai,ori)<>0;
        pp(des,ori)=3; end;
    end;
  end;
  per=perpro;
end; end;
```

C. Mean number of hops calculation

The network performance is measured by the mean number of hops $\overline{H}$ a packet completes traveling from origin to destination. The main program, used to calculate the main number of hops is based on the evolution of a vector *P(x)* with *N* dimensions. The *x* variable is the discrete position for the packet *(x = 1, 2, ..., N)*. Each vector represents the probability of finding a hypothetical packet in each node. That is called probability distribution vector. The mathematical treatment for the evolution of a probability distribution through time corresponds to the application of an operator *"U"* to probability vector $P_t(x)$ at any instant of time "*t*" to obtain probability vector $P_{t+1}(x)$ at the instant of time "t+1" after a discrete time interval. The unitary increment of time corresponds to one hop from one node to the following in the packet traveling from source to destination.

$$P_{t+1}(x) = UP_t(x) \quad (1)$$

Operator *U* is analogous to the "Perron-Frobenius operator" employed in the chaos theory for calculating the time evolution of a probability distribution [12]. An analogy can be constructed with the chaos theory, in which the idea of trajectory is abandoned and replaced by the evolution of a probability distribution. In this work, the idea of a path that a packet should follow from its origin to its destination is replaced by the probability distribution vector with a time evolution described by (1). Acampora and Shah [18] considered a similar statistical procedure to describe the behavior of a store-and-forward routing as a comparison with hot-potato routing. Due to the fact that the probability to go directly from one node to the other is zero for almost all nodes, except for those two directly connected, most of the elements of operator *U* are zero. Each column has only two non-zero elements. The preferential output port has probability *Ppp* and the alternative port, corresponding to the deflection port, has probability *Pd* given by:

$$Pd = 1 - Ppp \quad (2)$$

A packet is sent to the preferential port in three cases:

a) There is no other packet in the competitor link that could arrive before it.
b) There is another packet that could arrive before it, but that has a local final address and is going to be removed before competition.
c) There is another packet arriving before it that is not a local packet, but it has a different preferential output port.

Link occupation probability *Poc* defines the probability of the first case to be 1– *Poc*. Given that case, a) is not true, the local packet probability *Plp* defines the second case probability term as $Poc*Plp$. Finally, given that case a) and case b) do not apply, considering *Pop* as the probability of the competitor packet to have a different preferential port (another port), the third term is defined as $Poc*(1-Plp)*Pop$. Hence, the final probability of a packet to go through preferential port *Ppp* is given by:

$$Ppp = 1 - Poc + Poc*Plp + Poc*(1-Plp)*Pop \quad (3)$$

In (3), *Poc* is the occupation probability associated to the link load. A fully loaded link (not considering the FDL length) is assumed to corresponds to *Poc*=1. The probability of a packet preference pointing to another port *Pop* is assumed to be 50% and *Pop*=0.5 in all cases. Local packet probability *Plp* is evaluated to be $1/\overline{H}$, with $\overline{H}$ calculated as a preliminary mean number of hops obtained with a first guess value $Plp=1/(N-1)$.

The $Plp = 1/\overline{H}$ hypothesis is originated by the fact that every packet, at any time, is positioned in some place in its own path that starts on the origin and finish on destination. The mean number of hops in all possible paths is $\overline{H}$ and the packet is considered to be a local packet only in the last of those $\overline{H}$ hops. That means that $1/\overline{H}$ is the probability of a packet to be positioned at the last hop of its path from origin to destination.

Without failure, the MSN architecture belongs to a symmetry group called automorphism [13]. In this group, it is impossible to differentiate any node from the other concerning its position in the network. The mean number of

hops is the same regardless of the position of the final address node. Yet, introducing a failure, the symmetry is broken and the mean number of hops may assume different values for different final destinations. In this case, it is necessary to calculate the mean number of hops for all the possible destinations and to adopt the arithmetic mean of those values as the final network mean number of hops.

One more consideration should be made about the "don't care" nodes. They are already identified and signalized by number 3 in the preferential port matrix "*pp*". In that case, the model considers that the packet plays no role in the decision of the preferable output port. The position of the switch may be adjusted to the preferred output port of the packet eventually arriving in the competitor link. This case corresponds to considering *Ppp=Pd=50%* in all "don't care" situations.

The number of hops is obtained recursively by (1) starting with $P_1(x)$, which represents the probability to reach the destination with one hop, to calculate $P_2(x)$, which represents the probability to reach the destination with two hops. That procedure is repeated k times while the total probability is less than 100%, with an arbitrary criteria chosen to be $\Delta P=10^{-6}$. Further reduction of that criterion interferes only with the calculation time and no change is observed in the results for $\Delta P=10^{-5}$.

The mean number of hops for each destination x is calculated by the equation:

$$\overline{H} = \sum_{1}^{k} tP_t(x) \tag{5}$$

With the condition:

$$1-\Delta P < \sum_{1}^{k} P_t(x) \le 1 \tag{6}$$

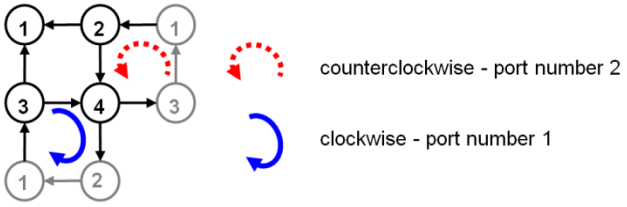


Figure 11. MSN with four nodes (N=4) showing link sub-domains.

D. Trivial case calculation example

As an example, consider the network with four nodes shown in Fig.11. Suppose that the link load is very near zero (without any charge). Then, *Poc=0*, *Ppp=1* and *Pd=0* are the values used. In order to find out the mean number of hops it suffices to calculate the mean number of hops to get to node number 1. Calculations for all the other destinations will yield the same value because of the MSN automorphism.

The initial hypothesis is that a generically chosen packet is not in node number 1 but is addressed to node number 1. This condition is represented by initial probability vector $P_0(x)$ given in (7).

$$P_0(x) = \begin{bmatrix} 0 \\ 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \tag{7}$$

Connection matrix "*c*" is given by (8):

$$c = \begin{bmatrix} 0 & 1 & 2 & 0 \\ 2 & 0 & 0 & 1 \\ 1 & 0 & 0 & 2 \\ 0 & 2 & 1 & 0 \end{bmatrix} \tag{8}$$

Preferential port matrix "*pp*" is given by (9).

$$pp = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 3 & 1 \\ 1 & 3 & 0 & 2 \\ 3 & 2 & 1 & 0 \end{bmatrix} \tag{9}$$

After one hop, probability vector $P_1(x)$ is obtained by (1) as shown in (10).

$$P_1(x) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} = \begin{bmatrix} 2/3 \\ 1/6 \\ 1/6 \\ 0 \end{bmatrix} \tag{10}$$

Operator "*U*" in (10) has a null first column because the packet in node 1 is going to be removed (has arrived to destination) and cannot go anywhere. After two hops, the probability vector $P_2(x)$ yields:

$$P_2(x) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2/3 \\ 1/6 \\ 1/6 \\ 0 \end{bmatrix} = \begin{bmatrix} 1/3 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

Condition (6) is achieved because $P_1(1)+P_2(1)=1$ and the mean number of hops given by (5) results in $1.P_1(1)+2.P_2(1)=2/3+2/3=4/3$. This value can be confirmed by analytical formulae deduced in [13] for MSN and shown in (12) valid for the cases in which N is an even number and the shortest path is always used. The trivial example presented here uses N=4 and n=2. In this case, $n/2$=1, so ($n/2$) odd number case formula applies. The ($n/2$) even number case formula (12) can be used for the empty link (*Link Load*=0%) and both N and $n/2$ must be even numbers.

$$\begin{cases} \overline{H} = \dfrac{(N/2)(n+2)-4}{N-1} \rightarrow (n/2)\ even \\ \overline{H} = \dfrac{(N/2)(n+2)-2n-2}{N-1} \rightarrow (n/2)\ odd \end{cases} \quad (12)$$

E. Results and discussions

The results for the mean numbers of hops are shown in Fig. 12 for the MSN with *N* nodes (*N=64*, *N=144 and N=256*). All the results were calculated for both cases: with and without failure.

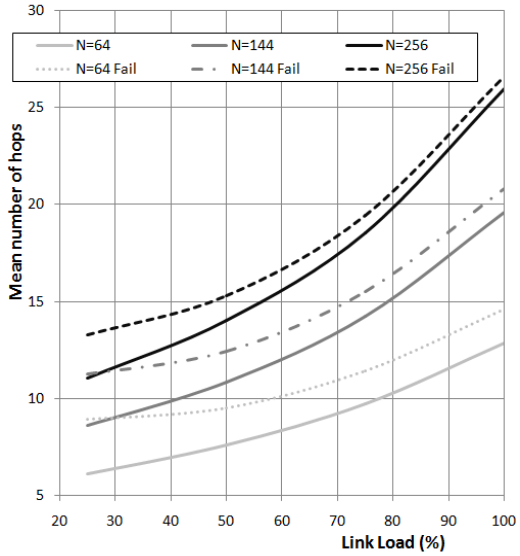


Figure 12. Mean number of hops versus Link Load condition.

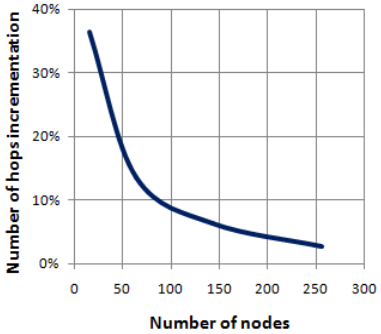


Figure 13. Relative Mean Number of hops degradation after failure as a function of the number of nodes.

Figure 12 shows that the degradation of the mean number of hops due to the failure is smaller for 100% link load condition when compared to the degradation for non-full load condition. Also, that degradation is smaller for higher number of nodes than for small networks. Fig. 13 shows a plot of the relative mean number of hops degradation at that full load condition (*Link Load* = 100%) as a function of the number of nodes. It confirms that larger number of nodes results in better robustness of the complex system. That conclusion was previously discussed for calculations up to 144 nodes [15].

VI. SIMULATION MODEL

The time domain simulation model (TDSM) was developed over the OMNeT++ platform [19]. The simulation model considers all the nodes sending packets to all the others and following the same rules used in the analytical model. Every packet arriving to one 2x2 node is addressed to the better output port, unless the node is already occupied with a competitor packet. In this case, the packet is sent to the available output port. The destination and the exact instant of packet generation are randomly chosen. Each link load condition is governed by the packet size. A packet with half the link size is used to simulate the 50% link load condition. Each packet that reaches the destination stimulates the insertion of a new one, addressed to a new randomly chosen destination. That procedure ensures the maintenance of the link load condition all along the simulation time. The simulation considers a 40Gbps bit rate and one kilometer link length. The delay line fiber length is considered to be equal to the link length, the same hypothesis employed in the analytical model.

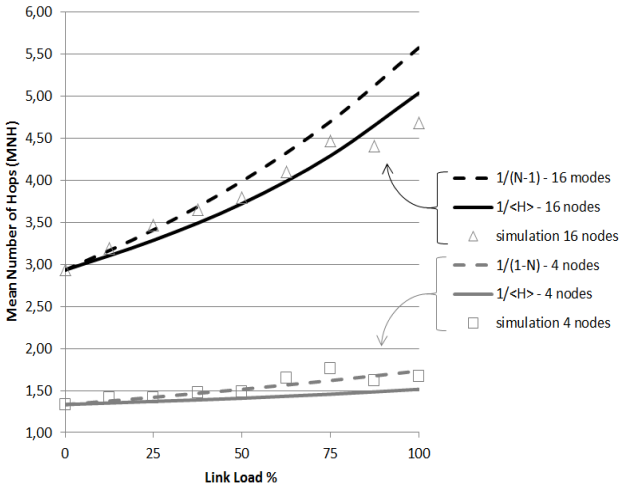


Figure 14. Analytical and simulation models.

Figure 14 shows the simulation results compared to the analytical model for two hypotheses used for the evaluation of the local packet probability *Plp*. The agreement between models is better for hypothesis $Plp = 1/\overline{H}$ as compared to the hypothesis of the first guess *Plp* = 1/(*N*-1). In fact, that first guess hypothesis is very close to simulation results for small number of nodes but tends to decrease faster than $Plp = 1/\overline{H}$, producing wrong results for a higher number of nodes. The simulation time is far higher than the analytical calculation time, limiting its utilization for scalability issues. The simulation model was important to validate the statistical analytical model and to determine that the first guess used for the *Plp* probability was not correct for large number of nodes.

VII. FAILURE EFFECT DISTRIBUTION MAP

The last calculation performed was the failure distribution effect. In case of failure, the symmetry is broken and the mean number of hops is no longer the same for any destination. Then, it is necessary to calculate the mean number of hops executed by an arbitrary packet addressed to all the 256 nodes. The overall mean value was considered to be the arithmetic mean of those previously calculated values. Considering the full load traffic condition (100% link load), the map in Fig. 15 shows an important distribution feature. The map shows nodes 1 to 16 in the first line and 16 nodes per line up to node number 256. The failure occurs in a link belonging to the clockwise sub-domain connecting nodes 89, 90, 106 and 105. Most of the destination nodes are not perturbed by the failure and remain with the same average number of hops (ANH) they had before failure (ANH<26). The ANH increases only for the destinations near the failure. Outside the contour lines, the ANH is less than 26. Crossing one contour line, the ANH is less than 27. Increased by one unit after crossing each contour line, the ANH will be less than 34, near failure, after crossing 8 contour lines.

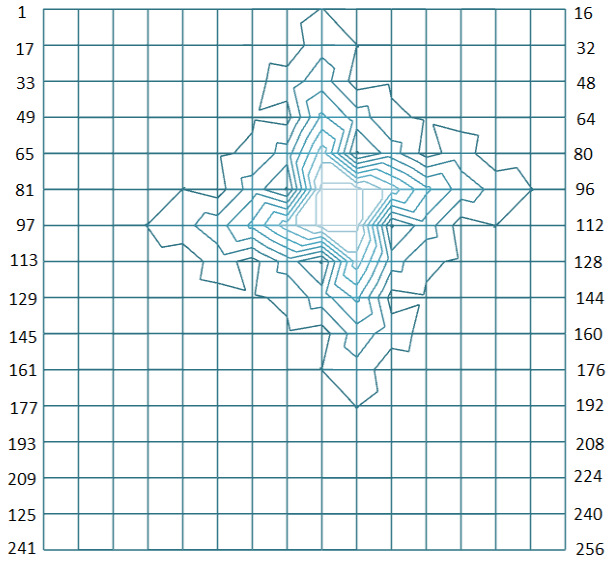


Figure 15. Failure segregation. Each contour line corresponds to one more hop from source to destination. Outside the contour lines, the Average Number of Hops (ANH) is less than 26. Inside all the lines, the ANH is less than 34.

VIII. CONCLUSION AND FUTURE WORK

The approach used in this work allowed treating a large number of nodes network as a complex system, working as a bottom-up organization system. The approach was fully analyzed with both a statistical analytical model and a simulation model. It was possible to investigate the scalability, the protection and the restoration as physical layer emerging functions. Protection and restoration are achieved by local signalizations that modify only the node operations around the failure. No signalization needs to be transmitted over a long distance regardless of the size of the network and that behavior is responsible for the scalability. A map with the number of hops after failure illustrates that the network performance degradation occurs only around the failure. The segregation of the failure effects represents a new feature that could be observed due to the new approach. Generalization was performed for the National Science Foundation Network (NFSnet) working as a complex system in a bottom-up type of organization. The approach used herein can be considered as an Autonomic Network Architecture (ANA) [20] extension to the physical layer. Future work will provide more details about the complex behavior through the use of the same statistical analysis for larger networks (more than 256 nodes). With more nodes, new Emerging Functions can be revealed because they can be emphasized for larger number of nodes. Several new features can be proposed or investigated. Burst switching (packets larger than the link length) traffic distribution behavior, delay and delay variation are the same candidates to be analyzed as Emerging Functions. Considering the case of long emergency state time and the EFC, one possible future work is to apply a machine learning technique to automatically rebuild the routing tables. This can be done from the initial failure point recursively until reaching all nodes; the signalization is the trigger of this procedure. In this case, the adaptive tree is changed without being commanded to restart. The bottom-up organization and the complex system treatment at the physical layer allow this good performance and robustness for network.

ACKNOWLEDGMENT

The authors thank FAPESP – The State of São Paulo Research Foundation – sponsor of the KyaTera Project. The authors also thank Lucas Pauli Simões for the contributions to the simulation results. The second author also wants to thank FAPESP for the support through grant 2011/17096-5.

REFERENCES

[1] A. Sachs, R. Rocha, F. Redígulo, and T. Carvalho, "Emerging function concept applied to photonic packet switching network", EMERGING 2012: The Fourth International Conference on Emerging Network Intelligence, IARIA, Barcelona, Spain, September 23-28, 2012, pp.22-26. ISBN: 978-1-61208-239-4.

[2] P. Baran, "On distributed communications netwoks", IEEE Transactions on Commnications Systems, CS-l2, 1964, pp.19.

[3] A. C. Sachs, "Self-organized network architecture deployed by the utilization of optical packet switching technology" (in Portuguese). 2011. Doctoral Theses (Digital Systems) - Escola Politécnica, Universidade de São Paulo, São Paulo, 2011. Available at: <http://www.teses.usp.br/teses/disponiveis/3/3141/tde-05082011-152444/>. Retrieved: July, 2012.

[4] C. Guillemot, M. Renaud, P. Gambini, C. Janz, I. Andonovic, R. Bauknecht, B. Bostica, M. Burzio, F. Callegati, M. Casoni, D. Chiaroni, F. Clerot, S. L. Danielsen, F. Dorgeuille, A. Dupas, A. Franzen, P. B. Hansen, D. K. Hunter, A. Kloch, R. Krahenbuhl, B. Lavigne, A. Le Corre, C. Raffaelli, M. Schilling, J. C. Simon, and L. Zucchelli, "Transparent optical packet switching: the european ACTS KEOPS Project

approach", J of Lightwave Technology, vol. 16, pp. 2117-2134, 1998.

[5] L. Dittmann, C. Develder, D. Chiaroni, F. Neri, F. Callegati, W. Koerber, A. Stavdas, M. Renaud,, A. Rafel, J. Solé-Pareta, W. Cerroni, N. Leligou, Lars Dembeck, B. Mortensen, M. Pickavet, N. Le Sauze, M. Mahony, B. Berde, and G. Eilenberger, "The european IST Project DAVID: a viable approach towards optical packet switching", JSAC Special Issue on High-Performance Optical/Electronic Switches/routers for High-Speed Internet II. IEEE Journal on Selected Areas in Communications, vol. 21, pp. 1026 – 1040, 2003.

[6] C. Stamatiadis, M. Bougioukos, A. Maziotis, P. Bakopoulos, L. Stampoulidis and H. Avramopoulos, "All-optical contention resolution using a single optical flipflop and two stage all-optical wavelength conversion", paper OThN5 Proceedings of OSA / OFC/NFOEC 2010. Available at: <http://www.photonics.ntua.gr/PCRL_web_site/OFC_10_OThN5.pdf>. Retrieved: July, 2012.

[7] J. M. Carlson and J. Doyle, "Complexity and robustness", Proceedings of the National Academy of Sciences - PNAS, February 19, vol. 99, suppl. 1, 2002, pp. 2538–2545.

[8] A. L. Barabási, "The architecture of complexity", IEEE Control Systems Magazine, vol. 27, 2007, pp. 33-42.

[9] D. L. Turcotte and J. B. Rundle, "Self-organized complexity in the physical, biological, and social sciences", in Proceedings of the National Academy of Sciences – PNAS, February 19, vol. 99, suppl. 1, 2002, pp. 2463–2465.

[10] Z. Movahedi, M. Ayari, R. Langar, and G. Pujolle, "A survey of autonomic network architectures and evaluation criteria", IEEE Comm. Surveys & Tutorials, vol. 14, n. 2, 2012, pp.464-490.

[11] A. L. Barabási and R. Albert, "Emergence of scaling in random networks", Science, vol. 286, Oct. 1999, pp. 509–512.

[12] I. Prigogine, Le leggi del caos, Roma-Bari, Editori Laterza, 1993.

[13] A.G.Greenberg and J.Goodman, "Sharp approximate models of adaptative routing in mesh networks", Telegraffic Analysis Computer Performance Evaluation. Elsevier Science -North Holland, 1986, pp. 255-269.

[14] NSFNET: A Partnership for High-Speed Networking Final Report 1987-1995. Available at: http://www.merit.edu/networkresearch/projecthistory/nsfnet/pdf/nsfnet_report.pdf. Retrieved: set. 2008.

[15] A. Sachs, C. M. B. Lopes, and T. C. M. B. Carvalho, "Protection schema for optical packet switching network with large number of nodes", Microwave and Optoelectronics Conference (IMOC) 2009 SBMO/IEEE MTTS-International, 3-6 Nov 2009, pp.47-50.

[16] Scilab, free and open source software for numerical computation. Available att: http://www.scilab.org. Retrieved: Feb, 2013.

[17] H. Pistori, J. J. Neto, and M. C. Pereira, "Adaptive non-deterministic decision trees: general formulation and case study". INFOCOMP Journal of Computer Science, Lavras, MG, 2006. Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.60.1885&rep=rep1&type=pdf>. Retrieved: July, 2012.

[18] A. S. Acampora and S. I. A. Shah, "Multihop lightwave network: a comparison of store-and forward and hot potato routing", IEEE Transactions on Communications, vol. 40, 1992, pp. 1082-1090.

[19] OMNeT++, discrete event simulation environment free for academic and non-profit use. Available at http://www.omnetpp.org. Retrieved: July, 2012.

[20] M. Sifalakis, A. Louca, A. Mauthe, L. Peluso, and T. Zseby, “A functional composition framework for autonomic network architectures”, Autonomic Network Architecture (ANA) Project, which is sponsored by the EU-IST initiative on Future and Emerging Technologies, 2006.

Detecting Pedestrian Flows on a Mobile Ad Hoc Network and Issues with Trends and Feasible Applications

Ryo Nishide and Hideyuki Takada
Faculty of Information Science and Engineering
Ritsumeikan University
Kusatsu, Japan
nishider@fc.ritsumei.ac.jp, htakada@cs.ritsumei.ac.jp

***Abstract*—Due to the rapid development of mobile and ad hoc communication technology, research on extracting social contexts including the movement and density of pedestrians has also emerged in recent years. This study explores methods to extract pedestrian flows in a distributive manner from Bluetooth detection logs. Bluetooth devices are widely installed in mobile equipment such as laptops, tablet PCs, cell phones and PDAs, which pedestrians carry with them in daily life. The results of experiments have revealed that detection logs implicitly record traces of surrounding pedestrian flows, which might provide possibilities to analyze and distinguish pedestrian flow patterns in various situations. Moreover, the paper discusses the data management system on a serverless ad hoc network, including methods for interpolating detections in order to pick up missing devices and configurating a range query designated to gather data within the geographical range. While examining problems to be faced with technological changes, several possible scenarios are also presented for feasible applications.**

Keywords-distributed database; Bluetooth; social context; mobile devices; ad hoc network.

I. INTRODUCTION

This study is an extension of a previous paper [1] dealing with the methods and issues in extracting pedestrian flows in a distributive manner by examining the detectable signals from mobile devices.

With the increase of urban population and the expansion of social activities, we cannot avoid sharing the same public spaces with other people when traveling or in day-to-day life. On many occasions, it will be one of the major concerns for people whether the area is crowded or less-crowded, and sometimes it is necessary to know what is actually going on in such places, including the changing flow of pedestrians. Many location-based services have appeared on the market, thanks to the enhancement of computational ability and wireless communication technology in mobile devices. These include Bluetooth, WiFi and GPS technology. These advancements have paved the way to methods for detecting pedestrian flows or social contexts using high performance mobile devices [2], [3].

Our research employs methods to extract the density and flows of pedestrians from Bluetooth detection logs, while considering the data management scheme on a mobile ad hoc network [4]. This ad hoc network can be generated from connections between mobile devices to work as a distributed database, which can manage and update the detection log data, or modify the log data by accessing geometrically adjacent devices to check for missing detections. The policy of this work is to avoid initial preparations, such as installing a large number of expensive immovable sensors and high performance computational equipment in physical space, in order to minimize cost, time and effort. In this research, we focus on extracting pedestrian flows in physical world, while the specific services to utilize the detection results are left for future work.

We attempt to grasp social contexts such as changes of pedestrian flows and density by detecting the surrounding electronic equipment. Recent hand-held electronic equipment such as cell phones, smart phones, PDAs and laptops contain wireless devices such as WiFi and Bluetooth, which pedestrians carry with them in their daily lives. If these devices surrounding the user are detected and logged continuously, it may be possible to detect not only the density of crowd, but also the changes of pedestrian movements.

We have conducted a preliminary investigation to examine the statistics of detectable types of terminal (mobile phone, PC, etc.) at various locations [5]. Comparing two wireless technologies, WiFi and Bluetooth, WiFi was detected from many types of electronic equipment either carried by pedestrians or fixed in the environment. Therefore, it seems difficult to discriminate the type of equipment by WiFi, and in particular whether they are carried by the pedestrians or not. On the other hand, most of the detected Bluetooth radios were from mobile devices. In this paper, we focus on Bluetooth devices installed in equipment to be carried by users in order to examine the flows and movements of pedestrians.

Related research and comparable studies are reviewed in Section II. Section III explains our method of extracting pedestrian flows using Bluetooth detection logs. Based on the results of experiments, the density and movements of pedestrians are examined in different situations by the analysis of detection patterns in Section IV. Section V discusses the distributive and autonomous data management

scheme, and the interpolation of missing detections. Section VI describes issues arising from technological changes and then suggests possible applications of the proposed method in related fields. Section VII reviews the research with conclusion.

II. Related Work

The development of mobile equipment and ad hoc communication has led to several attempts to analyze social context. O' Neill et al. [6] and Nicolai et al. [7] examined the correlation between Bluetooth detections and pedestrian movement by deploying stationary Bluetooth sensors in the environment and analyzing the logs. Eagle et al. [8] have shown methods to analyze social patterns of users' daily activities. These studies show that scanning for Bluetooth (or other wireless devices) and analyzing detection logs give us the possibilty to extract the flow of pedestrians or discover relationships in the community. However, not every Bluetooth device can be guaranteed to be detected depending upon the performance of the device and its physical environment. Thus, their methods may not be able to cope when too much incoming data is generated in crowded environments.

To cope with such problems, Kim et al. [9] examined the detection pattern of Bluetooth device logs, and employed a clustering algorithm and Gaussian blur to remove noise caused by inquiry fault of undetected Bluetooth devices. They inferred the transition time of events from multiple device detections. However, inquiry faults for devices cannot be detected individually. As there are many complicated situations in the physical world, this method may not be enough to cope with all situations. Weppner et al. [10] estimated crowd density through collaboration with multiple devices to improve the accuracy of detections. Users carried multiple devices for Bluetooth scanning, which might be awkward or inconvenient.

Other related work includes Bulut et al. [11], which exploits friendship-based features of a mobile social network to perform efficient routing. Friendship is defined from the traces of surrounding personal wireless devices, and the closeness relationship is analyzed from frequency and duration of the connectivity between devices. Our work aims to extract users' activities and the situation occurring around them, while Bulut examines the social relationship between a user and the surrounding pedestrians.

Our research is designed to extract social context by scanning Bluetooth devices in the surrounding environment, with consideration given to the user's location and the communication range of Bluetooth devices. The method is proposed to work autonomously and distributively with the users' devices on an ad hoc network, avoiding such troubles as installing fixed sensors or carrying multiple devices. It also enables to deal with inquiry faults by performing computation collaboratively with nearby devices.

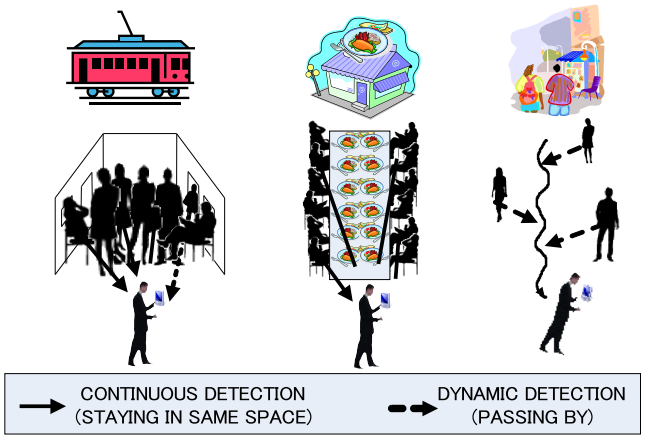


Figure 1. Detections for different places and situations

III. Detection of Pedestrian Flows

A. Concept of our Research

Our objective is to extract social contexts such as changes of pedestrian flows or the crowdedness of pedestrians from the traces of detected devices carried with them in their daily activities. These devices include cell phone, smart phone, PDA, portable games, tablet PC and laptop equipped with wireless devices such as WiFi and Bluetooth. In fact, the detection pattern differs depending upon the situations of the surrounding pedestrians (Figure 1). Thus, by analyzing the detection patterns, it might be possible to infer the social contexts or trends and changes of surrounding situations. We avoid extracting the personal information of pedestrians, such as location and name, since recording this kind of information might violate the privacy of pedestrians. Instead, we examine the detection patterns (e.g., numbers and changes of simultaneous or continuous detections) of devices carried by pedestrians surrounding the user.

B. Features of Bluetooth Devices

During the manufacturing process of a Bluetooth device, it is assigned a unique ID in the form of a 48-bit MAC address. This address is called a Bluetooth Device Address (BDA) and is used for communicating with other devices by exchanging BDAs for identification. Thus, BDAs are sent constantly without requiring authentication to build connections with other Bluetooth devices. We target class 2 Bluetooth devices embedded in hand-held mobile equipment as cell phones, laptops, PDAs, etc., having a communication range of approximately 10 meters. The protocol for the Bluetooth inquiry first receives the BDA of surrounding Bluetooth devices, and then requests the names of these devices. A combination of BDAs and timestamps are stored in the log file for each fixed time interval.

Figure 2 shows an example of the pedestrian's Bluetooth Device, which has entered the reachable communication

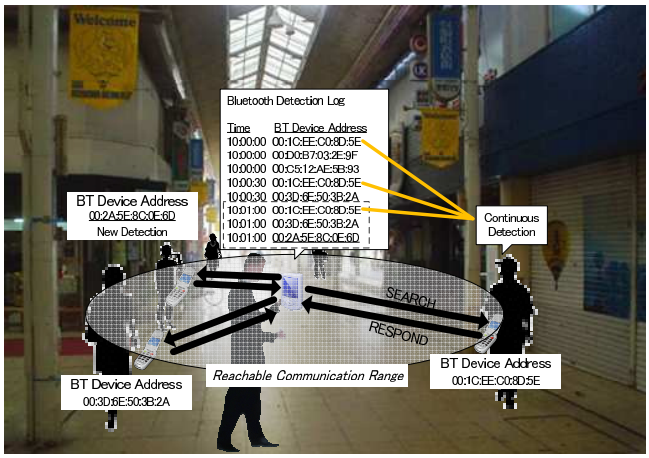


Figure 2. Detection of pedestrian flows

range of user's device. The user's device continuously inquires for nearby pedestrian devices, and logs the time and BDA of devices which respond. From the log, different types of detection patterns can be verified, such as continuously detected, newly detected, undetected or disappeared, and so on. These are the key to determining the dynamic flow of pedestrians in the physical world.

The model of Bluetooth device can be determined by inspecting the 24-bit prefix of the BDA. In other words, every device which has the same BDA prefix has the same manufacturer, which enables to identify what kind of device it is. For example, a BDA with prefix 0022F3 identifies a cell phone from a certain manufacturer, the device with prefix 001BDC a certain desktop computer, the device with prefix 001D4F a certain laptop computer, and so on. We can identify the device models without retrieving the device name and classify whether the device is cell phone, smart phone, laptop, PDA, iPad, and so on.

C. Trends for Detectabe Types of Bluetooth Devices

Here we discuss some points about the detectability of Bluetooth devices arising from a preliminary investigation examining the detectable types of terminal.

Bluetooth devices can only be detected when the Bluetooth function is turned on with Discovery mode enabled (a configuration option to enable the surrounding terminals to discover the user's terminal). Fortunately, many mobile phones were detected easily, probably because several models with Bluetooth functions were sold in Discovery mode as a default setup. Moreover, it seems that there are several cases in which inexperienced mobile phone users unintentionally accept the mobile phone application's request to turn on the Bluetooth device without knowing what it is. For example, a chat application for Softbank mobile phones has been popular for several years in Japan [12] allowing users to communicate with other users within

Table I
CHARACTERISTICS OF PEDESTRIAN FLOW BY SITUATIONS

	User		Surrounding ppl		Sharing Space
	Stay	Move	Stay	Move	
Town	△	◯	△	◯	△ or ×
Conference room	◯	×	◯	×	◯
Cafeteria	◯	×	◯	◯	◯ or ×
Train Moving	×	◯	×	◯	◯
Train Stopping	◯	×	◯	◯	◯ or ×

notations: (◯)many; (△)some; (×)few/none

the Bluetooth device detection range. This application asks users to enable the Bluetooth function, but does not ask them to disable it when no longer needed. Another example is users who purchase cell phone and then try to use all the functions of the phone while reading the instruction manual, inadvertently turning on the Bluetooth function on without paying attention. In either case, the result is that many users walk around town with their mobile phone's Bluetooth device turned on and discoverable, enabling the method described here for detecting those devices.

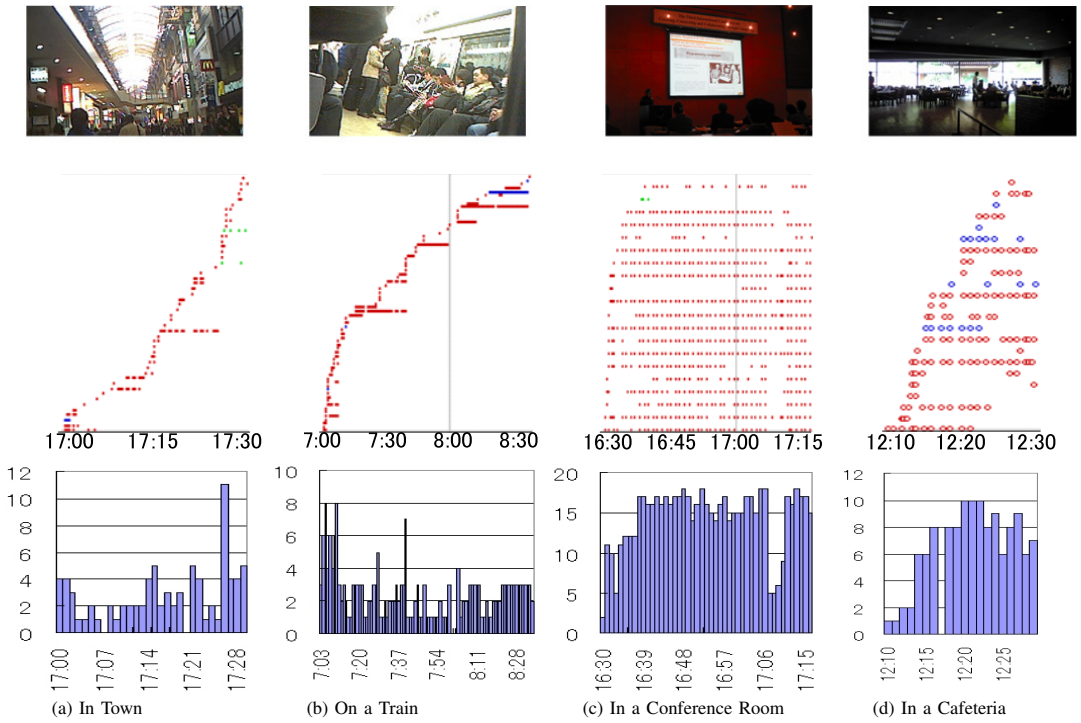
We have chosen Bluetooth devices as our detection target in order to extract the flows and movements of pedestrians, because most Bluetooth devices are installed in equipment to be carried by users. The method we have proposed can be performed using only the mobile device carried by the user, without installing additional equipment such as mounting fixed sensors or video cameras in the environment.

IV. VERIFICATION OF DETECTION PATTERNS

The authors have done several investigations to observe surrounding Bluetooth devices in various situations, such as: (i) normal daily routine for commuting to and working at a university, (ii) special events such as conferences and school festivals, and (iii) other off-campus activities such as tourism, shopping, public festivals and new year celebrations. To collect data, we used a HP iPAQ 112 Classic Handheld PDA set to record BDA with a timeout interval of 6 seconds and a 30-second inquiry signal cycle.

Table I shows Bluetooth detection logs in different situations, and their characteristics for specific movements of the user and the surrounding people. Four different cases have been examined in this paper, namely strolling in town, traveling by train, attending a conference, and taking lunch at a cafeteria.

The results of examination of detection logs are summarized in Figure 3. The upper half of Figure 3 shows the detection pattern of Bluetooth devices, with the time-line expressed on the horizontal axis and the device ID assigned in chronological order of the incoming BDA on the vertical axis. The mobile phones are shown in red, and PCs and devices other than mobile phones in green, and unidentified devices in blue. The lower half of Figure 3 shows the number of detected devices, with the time-line expressed on the horizontal axis and the quantity of BDA on the vertical axis.



(a) In Town (b) On a Train (c) In a Conference Room (d) In a Cafeteria

Figure 3. Detection pattern of BDA (upper), detected number of BDA (lower)

(a) Strolling in Town: Figure 3(a) shows the changes of multiple detection logs encountered while strolling in town. The number of BDAs is not constant as the number of passers-by is always changing. Even if the pedestrians are walking in the same direction, their devices disappeared occasionally probably because their directions coincided only for a while or their walking speed was different. On the other hand, the same BDA was continuously identified in some places while the examiner was lingering in a store.

(b) Transporting by Train: Figure 3(b) shows the detection in a train during rush hour. From the log, we can identify characteristics such as: (i) devices were continuously detected from passengers in the same carriage; (ii) many incoming and outgoing devices were detected when changing trains; and (iii) a large number of people got on/off the train at major stations. The passenger's devices can be constantly detected while the train is moving. However, due to the limited size and shape of the carriage, the detection has been low even in rush hour. This observation shows that it is necessary to identify the situation from detection patterns by integrating the analysis from different points of view, as it cannot be inferred merely by the quantity of devices.

(c) Attending a Conference: Figure 3(c) shows that many BDAs were detected continuously in the same room. As most of the participants were staying in the room during the conference, the number of BDAs was almost constant (14 to 18 devices), except during the coffee break. As the room was wide enough to hold many people, the quantity of detections remained high.

(d) Taking lunch at a Cafeteria: Figure 3(d) shows that many devices have been detected during lunch time, as customers enter, take lunch and leave the cafeteria one after another. Some devices are detected continuously with long duration, and others are divided into several times with short duration, because two types of situation are mixed together: people sitting and eating lunch, and people walking around to look for seats or friends.

These results show that pedestrian flow can be inferred by analyzing the detection logs as follows:

- **The number of logged BDA detections:** crowdedness of people (requiring reference to the scale of space)
- **Time length of BDA detection:** people staying in same space or duration of the event
- **Appearance/Disappearance in BDA detection:** people staying, entering, leaving, or passing by

The detection logs show that there are several undetected devices even among those staying in the same space. Therefore, a method to interpolate the missing detections is also explored in the following sections.

V. Construction of the Ad hoc Network

Another issue of concern is the management of the log of pedestrian flow obtained from each mobile device. It is not efficient to collect and manage the entire data sent from mobile devices on a server. In this section, we describe a peer-to-peer (P2P) mechanism necessary to manage data and

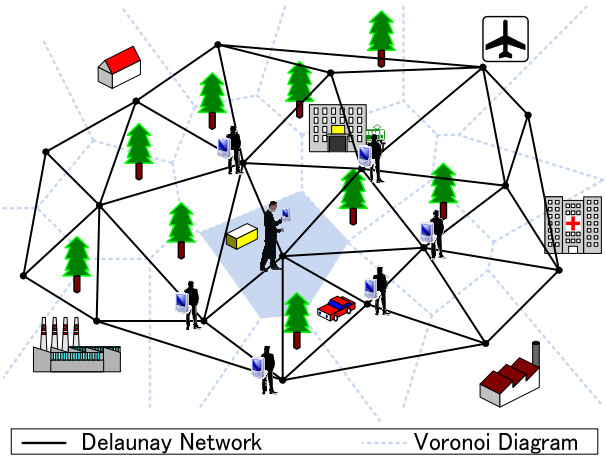


Figure 4. Delaunay network and Voronoi diagram

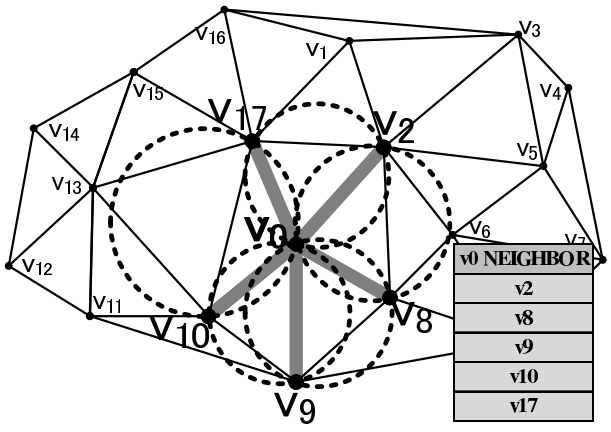


Figure 5. Selection of nodes to connect

perform computation between mobile devices cooperatively, and provide a method to perform location-based range queries for retrieving the data managed on mobile devices.

A. Scheme for Distributive Data Management

To build this mechanism we propose an ad hoc network between the mobile devices to manage the data and communicate with other devices. In this network, each device builds connections directly with other devices without communicating to a base station. In generating connections, it is important to employ an efficient scheme to choose mobile devices to connect with, considering their location and limited communicable distance. Note that not all surrounding pedestrians with mobile devices are viable for generating connections on the ad hoc network. Some of their devices might be cell phones or other devices with limited or no computational capability.

We propose a peer-to-peer Delaunay network, which is a geometry-based network whose topology is defined by the geometric adjacency of mobile devices (see Figure 4) [13], [14]. These devices are connected in a geometrical structure called a Delaunay Diagram, which is well-known in computational geometry. It has the following features: (i) each device connects to nearby devices based on geographical distance, (ii) the degree of connection for each device is low (approximately six), (iii) the network can deal with join/leave of a device only affecting the surrounding devices to reconstruct and update the connection, and (iv) the data on distant devices is retrievable through multi-hop communication.

A P2P Delaunay Network lets us construct an environment in which the mobile devices are connected to each other autonomously and distributively. It is not necessary to prepare a server in order to maintain the system or manage pedestrian flow data on it. Moreover, it also provides possibilities to perform collaborative computation or processing to work with set of mobile devices nearby. Delaunay Networks are effective for accessing data in geometrically adjacent devices, which can expose missing detections by comparing the detection logs of nearby devices. In this paper, we refer to the geographical location of each mobile device as a node.

A Voronoi Diagram is the dual of Delaunay Diagram and is generally used to determine the governing regions for each node. Let $V = \{v_1, v_2, \ldots, v_n\}$ be a set of nodes distributed in a plane. The Voronoi diagram for V is a partition of the plane into n Voronoi regions, each region associated with each point v_i of V. For example, the shaded region in Figure 4 is a Voronoi region of a node with a man in the center. Each node v_i has its own Voronoi region, and the entire n Voronoi regions cover the entire plane, of which any particular point is managed by one of the nodes of V. The edges of the Voronoi Diagram are generated by connecting the perpendicular lines stretched out from the midpoint of the edges of the Delaunay Diagram. In this work, we use a Voronoi Diagram to determine the nodes to which point or range queries are sent for access to distant nodes.

B. Network Construction

In the present investigation, we apply a method proposed previously [13] to generate a P2P Delaunay Network with mobile devices. We assume that each mobile device only has the location information of other devices, but not the knowledge of how the other devices are connected. Thus, each mobile device must choose the appropriate mobile devices to connect to, using their locations to generate a P2P Delaunay Network.

To build connections of a P2P Delaunay Network under such conditions, each node draws an inscribed circle with two other nodes on a plane, with the Delaunay Network property that no other nodes shall be enclosed within the circle. Figure 5 shows an example of v_0 determining the nodes to generate connections to on a plane. Inscribed circles

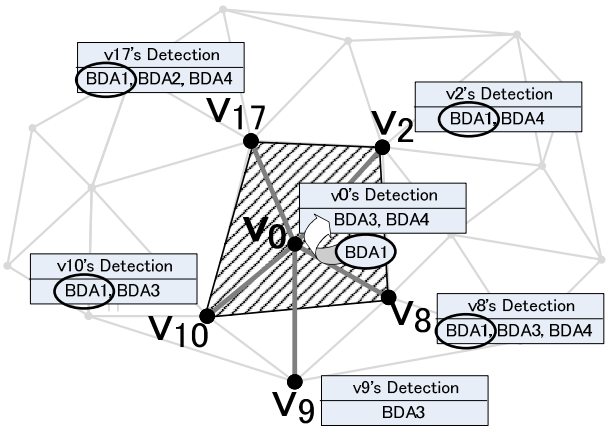


Figure 6. Interpolation of BDA data (BDA1)

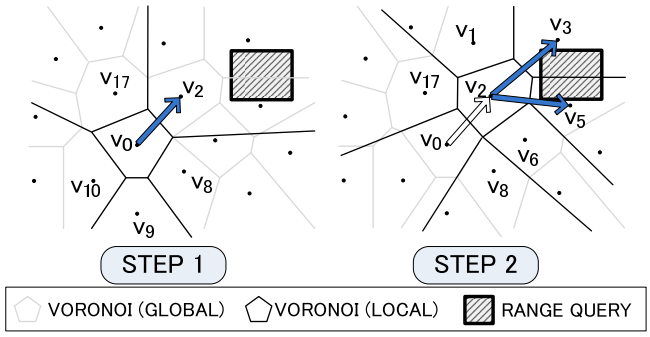


Figure 7. Geometric-based routing method for range query

are generated connecting three nodes each, namely (v_0, v_i, v_j) $\{0 \leq i \leq 17, 0 \leq j \leq 17, i \neq j\}$, which any of these circles has no nodes in the internal. The nodes (v_2, v_8, v_9, v_{10}, v_{17}) are assigned as the neighbors of v_0 to generate connections with. If the rest of the nodes (v_1 - v_{17}) perform the same processes, a Delaunay Diagram can be generated. The detailed algorithm for generating and maintaining connections are discussed in the previous work [13]. Delaunay Network can be used not only to generate or maintain connections with adjacent nodes on a plane, but also to perform collaborative computation with adjacent nodes as described in the following section.

C. Interpolation of Missing Detection

We have described methods to extract and manage the Bluetooth detection logs on an ad hoc network. However, there are false-negative cases in which some devices within the communication range may not be detected. That is, too much BDA data arrives at once in a crowded place, and the device cannot handle it all within the limited time interval while scanning for the surrounding Bluetooth devices.

To deal with such problems, we consider methods to check the detection logs of adjacent nodes on Delaunay network, and interpolate the BDA data which is definitely within the communication range of Bluetooth device. Initially, each node sends a copy of its own detection logs to adjacent nodes, and receives their copy of detection logs. Then, it extracts the BDA data which is not detected from its device, but detected from other adjacent nodes' devices. These BDA data will be the target data to perform interpolation, and the location of these adjacent nodes will be the criterion to determine whether or not to perform interpolation.

We validate only the BDA data owned by more than three adjacent nodes to perform interpolation. That is, a polygon is drawn using the location of adjacent nodes with the target BDA data as vertices. If the location of its own node is within the polygon, then the target BDA will be interpolated. We have chosen a polygonal shape to determine the interpolation because it is obvious that the entire polygonal region is covered from the communication range of the Bluetooth device. The purpose of this interpolation method is to deal with missing detection, and the deformation of communication range caused by walls, buildings, and other obstacles is beyond our focus.

Figure 6 shows the interpolation process using the same Delaunay Network as Figure 4. Node v_0 has five adjacent neighbor nodes, namely v_2, v_8, v_9, v_{10}, v_{17}, and has the copy of their BDA detection logs. Among the BDA on detection logs, BDA1 is the only one that v_0 does not have, but more than three adjacent nodes (v_2, v_8, v_{10}, v_{17}) do have it. Using these nodes as vertices, a polygon is drawn starting from the upper node in clockwise direction. Finally, BDA1 can be determined to be included in v_0's detection data, as it is allocated within the polygon area.

D. Location-based Range Query

When users would like to know the pedestrian flows or the situation of a physical area, they designate a particular location or range and ask what is going on in that location. Therefore, the location should be used as a key for searching and gathering data on the ad hoc network.

In order to send a query or data to a destination, each node v_i generates a Voronoi diagram using v_i and the neighbor nodes of v_i. Here, we define *global Voronoi diagram* as a Voronoi diagram drawn with all the nodes on a plane, and *local Voronoi diagram* as a Voronoi diagram drawn only with a given local node and its neighbor nodes. Using its local Voronoi diagram, each node v_i selects the neighbor node to send the query, and performs multihop communication between nodes. The shape and locational data of the query range is sent together along with the query.

Figure 7 shows the process of sending a query to a particular range on a plane. First of all, v_0 determines the nodes from its neighbor nodes v_2, v_8, v_9, v_{10}, v_{17} to send the query. Among these nodes, v_2 is the only node selected

to send the query, as its Voronoi region of v_2's local Voronoi diagram covers the entire query range. Next, v_2 determines the nodes from its neighbor nodes v_1, v_3, v_5, v_6, v_8, v_0, v_{17} to send the query. For this situation, v_3 and v_5 cover the query range and so the query is sent to both of the nodes. This process is performed recursively until the query has reached the destination.

Using this method, we can generate either a point query search, by designating a particular point on a plane, or a range query search which can request data within a designated range. Thus, the global view of the location of nodes is not required to send data to a destination.

VI. Issues with Trends and Applications

A. Considerations for Technological Changes

We have continuously attempted to collect the Bluetooth detection logs in order to observe the changes in various environments for a couple of years. We have also noticed that the technological advancements and changes in consumers' lifestyle have considerably affected the results of detection. In general, it is noticeable that the detection rate by Bluetooth device logs has gradually been reducing even during these past couple of years. In particular, it has become difficult to collect Bluetooth device logs in college environments such as school buses, campus cafeteria or restaurants, lecture rooms, and so on. It can be assumed that students and the younger generations are using smart phones instead of feature phones, and that these smart phones are probably programmed to turn off the Bluetooth connection if not used during a certain amount of time.

We have observed, at the same time, that the detection rate by Bluetooth devices is changing among age groups when we compared the data with those detected in the train, train stations, stores, or while strolling in towns. As these places are likely to be occupied or visited by mixed age groups, including commuters, holiday goers, travelers, children, students, elderly groups, and many others, the changes of Bluetooth detection rate are rather gradual, though not increasing, compared with the detection on campus or while boarding the school bus where the majority of the passengers are young students.

On the other hand, the number of WiFi detections has been increasing rapidly especially in campus environment, probably because several mobile WiFi terminals have appeared on market, or some smart phones are equipped with WiFi connection technology which works as an access point. Therefore, we are also going to collect WiFi logs along with Bluetooth detection logs, and are planning to examine WiFi logs to see whether the device can be classified between static devices that are fixed at a single location, or dynamic devices that move along with people. If both Bluetooth and WiFi log data are detected and examined together, it might be possible to perform more accurate analysis.

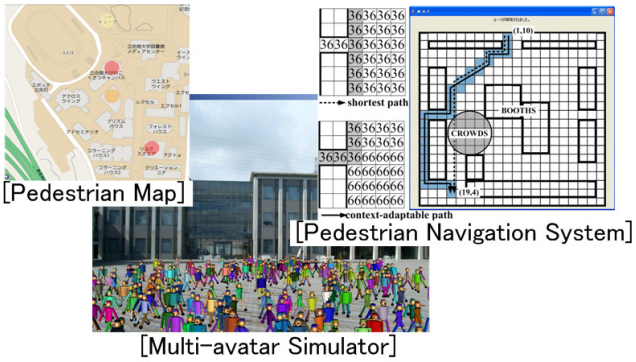


Figure 8. Applications to deploy the extracted pedestrian flows

Table II shows the changes in detection logs from the initial experimental stage in 2010 to the most recent data in 2013. The average number of detected logs by minute are calculated for 10 weekday mornings when the user himself is commuting. The results indicate that WiFi detection has been increasing in general even in the train where people with various age groups and different occupations are on board, and that it is more evident with the campus bus in which most of the passengers are students. Therefore, it seems important to detect and analyze using both Bluetooth and WiFi logs in future research.

It is also noted that there may be other reasons to prioritize differently for the detection devices which can be used for the experiment, since the technological advancement as well as lifestyles and cultures are different depending upon particular regions and localities.

B. Possible Applications for Practical Scenarios

It is necessary to consider efficient applications to deploy our method for detecting pedestrian flows. Some of the possible applications are mentioned here as suggestions for further research.

1) Map Visible of Pedestrian Flows: One of the systems to consider for the feasible applications is a map to visualize the flow and situation of pedestrians or the location of congested area as shown in Figure 8 upper left. Pedestrian flows are dynamic and likely to change frequently, therefore it is necessary to gather and process the information as soon as possible. Our method of detecting and analyzing Bluetooth devices may be applicable as it can collect the data seamlessly in real-time, and pass it to the application at the same time. Moreover, it is desirable for users to input time and place so that they can view the past or present location of congested area or pedestrian flows, in order to estimate the future pedestrian flows based on the calculated patterns from the same daily routine of pedestrian activities.

Visualization method of pedestrian flows might also be an important factor for users to understand the situation. The crowded area can be expressed as a circle, and locations

Table II
NUMBER OF BLUETOOTH VS WIFI DETECTED IN THE TRAIN AND BUS

Train, Year 2010

Date	Time(min.)	Bluetooth	BT per min.	WiFi	WiFi per min.
Day 1	71	275	3.87	116	1.63
Day 2	72	182	2.53	207	2.88
Day 3	81	370	4.57	223	2.75
Day 4	88	532	6.05	150	1.70
Day 5	82	399	4.87	170	2.07
Day 6	70	132	1.88	236	3.37
Day 7	64	389	6.08	122	1.91
Day 8	64	344	5.38	77	1.20
Day 9	77	372	4.83	127	1.65
Day 10	83	466	5.61	181	2.18
Avg/Total			4.57/6.70(68%)		2.14/6.70(32%)

Bus, Year 2010

Date	Time(min.)	Bluetooth	BT per min.	WiFi	WiFi per min.
Day 1	29	38	1.31	116	4.00
Day 2	13	58	4.46	37	2.85
Day 3	13	27	2.08	84	6.46
Day 4	12	0	0.00	31	2.58
Day 5	13	28	2.15	87	6.69
Day 6	13	29	2.23	81	6.23
Day 7	14	0	0.00	38	2.71
Day 8	21	125	5.95	85	4.05
Day 9	18	111	6.17	88	4.89
Day 10	12	4	0.33	92	7.67
Avg/Total			2.47/7.28(34%)		4.81/7.28(66%)

Train, Year 2013

Date	Time(min.)	Bluetooth	BT per min.	WiFi	WiFi per min.
Day 1	50	71	1.42	386	7.72
Day 2	53	297	5.60	208	3.92
Day 3	54	181	3.35	314	5.81
Day 4	53	129	2.43	132	2.49
Day 5	55	107	1.94	318	5.78
Day 6	54	139	2.57	194	3.59
Day 7	57	142	2.49	350	6.14
Day 8	58	108	1.86	425	7.33
Day 9	54	58	1.07	371	6.87
Day 10	57	18	0.32	481	8.44
Avg/Total			2.30/8.12(28%)		5.81/8.12(72%)

Bus, Year 2013

Date	Time(min.)	Bluetooth	BT per min.	WiFi	WiFi per min.
Day 1	13	33	2.54	93	7.15
Day 2	13	0	0.00	74	5.69
Day 3	15	3	0.20	100	6.67
Day 4	15	41	2.73	96	6.40
Day 5	17	3	0.18	99	5.82
Day 6	12	0	0.00	103	8.58
Day 7	16	8	0.05	159	9.94
Day 8	14	0	0.00	120	3.57
Day 9	25	0	0.00	134	5.36
Day 10	15	4	0.27	63	4.20
Avg/Total			0.64/7.48(9%)		6.83/7.48(91%)

with many Bluetooth detections may be filled with dark colors together with numbers expressing the congestion rate. However, the circle may not be enough if there are many sensing devices which cause the creation of multiple overlapping circles or numbers. As an alternative, expressing it with coloring the Voronoi diagrams, instead of circles, might be very helpful in that case. Coloring circles and Voronoi diagrams have both merits of their own. Coloring circles may express the location of sensing devices accurately in vacant areas and Voronoi diagrams may express the detection results from many sensing devices. Therefore, it might be effective to use the advantageous aspects from both methods and choose the preferable ones depending on the situation. The graphical user interface for expressing congestion and frequent changes of pedestrian activities and movements are in our plans for future work.

2) Context-adaptable Pedestrian Navigation System: We have also been working on the context-adaptable pedestrian navigation system shown in the upper right of Figure 8, which is a navigation system to provide users with the preferable route considering the user objectives and conditions of each area, such as with roof (to avoid rain) or no roof, lighted or dark, with stairs or elevators, narrow path or wide road, safe or dangerous facilities, and crowded or less crowded area [15]. Most of these situations are static, except for the crowdedness which changes according to the pedestrian flows.

In relation to this topic, we have previously proposed an algorithm to perform path search considering the user's preference and spatial context. The graph-based approach was employed using metadata to represent logical space built over the real space, and a scoring scheme was applied by partitioning target region into homogeneous cells. Path search was performed by accumulating negative or positive scores to each of the cells according to the user's intention or situation, and the path with the lowest scores to destination was provided. However, not much discussion has been held with the method to detect the situation in space.

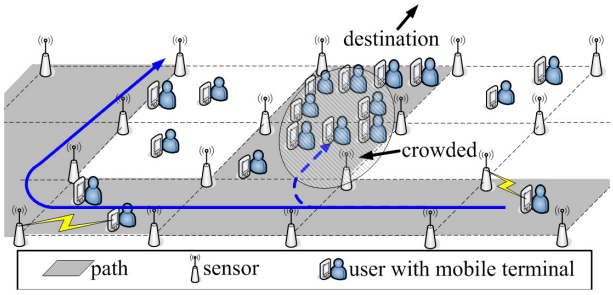


Figure 9. Immobile wireless sensors to detect pedestrian flows

To detect pedestrian flows, we had previously thought to prepare many fixed immobile devices on site, such as stepped-on sensors, wireless sensors, video cameras, etc., as shown in Figure 9. This approach is inefficient, however, as it is expensive, requires a massive number of sensors, takes time and effort to install sensors, and cannot be deployed in all desirable locations. It requires abundant initial preparations to build such systems. Our method to extract pedestrian flows from Bluetooth detection logs may be helpful to solve these problems.

3) Multi-avatar Simulation on Virtual Collaborative Space: We have worked on constructing an autonomous dynamic multi-avatar simulation in a 3D virtual collaborative space (lower part of Figure 8), in which the avatars move according to their behavioral patterns defined beforehand [16]. If the avatar's location and congestion information are mapped to 3D virtual collaborative space, we may be able to perform simulations based on the real world situation. Though our Bluetooth detection method cannot provide the exact number of pedestrians, if we can estimate the number of pedestrians in the real as a ratio to the number of detected pedestrians, we might be able to produce an atmosphere similar to the real world with dynamic pedestrians walking around in the virtual environment. Such simulation is useful in order to understand intuitively the situation occurring at the physical site.

4) Pedestrian Flow Analysis in Airport Environment: Our work might also be a possible extention to Pestana's work which proposes an approach to provide security and safety concerning mobile objects in an airport environment [17]. In the airport environment, there are many gates, restaurants, gift shops, etc., and passengers from various countries congregate in these locations. However, the passengers' behavior patterns might differ depending on their cultural backgrounds. For example, a particular restaurant might be preferred by people from the same or neighboring countries, or some people from the same region or areas might prefer buying many souvenirs at gift shops. Thus, collecting and analyzing the detection logs might contribute to recommending the passengers the preferable shops and restaurants depending on their nationalities, or sometimes suggesting to them uncrowded cafeterias or benches to take a rest.

Moreover, as the departure time of the flight and the departure gates are the same for every day's routine, the analysis of detection logs may clarify the daily routine of pedestrian's behavioral patterns. For example, if United Airlines uses the gate number 15 at 12 pm every day, passengers to the United States may gather at the gate around that time. Most of those passengers might take a lunch at McDonalds fast-food restaurant, even if there are various restaurants other than McDonalds. Such analysis based on the application of our method may contribute to a recommendation system or navigation system, which will provide comfortable, safe, and preferable navigation in the airport.

VII. Conclusion and Future Work

We have shown possibilities for inferring pedestrian flows by examining the detection patterns of surrounding Bluetooth devices, and proposed methods for generating mobile ad hoc networks and managing the detection data on the network, adhering to our policy to avoid initial preparations to install cameras or sensors on the environment, or manage data on a single server. For deployment in actual environments, energy consumption is an important issue that has to be considered, as the battery for HP iPAQ 112 Classic Handheld PDA used for our experiment lasts for approximately 4 hours. In addition, privacy issues are another concern because such personal data as user name and location should not be exposed to others.

For future work, we plan to perform detailed analysis of Bluetooth device logs, examine the applicability of other sensory data including WiFi, and provide location-based applications using social contexts such as pedestrian flows. We also plan to continue further study on Delaunay networks, explore efficient ways of managing social context data and log files, and evaluate our methods to interpolate missing data caused by inquiry faults.

In addition, from the observations presented in Section VI-A, we have found that Bluetooth devices are becoming harder to detect year by year. Therefore, analysis simply by Bluetooth device detection may not be enough to extract the flow of pedestrians. On the other hand, WiFi is being detected increasingly every year from portable devices, which should be another target to perform analysis on. Moreover, there may be many variations of wireless technologies coming out in the future, and thus, it is important to keep up with new technologies and trends to extend this research in the future.

Acknowledgment

This work was partially supported by MEXT/JSPS KAKENHI Grant Number 23650033, and The Telecommunications Advancement Foundation. We would also acknowledge

Prof. Yasuyuki Kono of Kwansei Gakuin University, Prof. Satoshi Nakamura of Kyoto University and Dr. Ian Piumarta of Viewpoints Research Institute for their valuable comments and suggestions.

REFERENCES

[1] R. Nishide and H. Takada, "Methods and Issues in Detecting Pedestrian Flows on a Mobile Adhoc Network," The Second International Conference on Mobile Services, Resources, and Users (MOBILITY 2012), pp.76–79, 2012.

[2] P. Lukowicz, A. Pentland, and A. Ferscha, "From Context Awareness to Socially Aware Computing," IEEE Pervasive Computing, 11 (1), pp. 32–41, 2012.

[3] A. Campbell, N. Lane, E. Miluzzo, R. Peterson, H. Lu, X. Zheng et al., "The Rise of People-Centric Sensing," IEEE Internet Computing, 12 (4), pp. 12–21, 2008.

[4] R. Nishide and H. Takada, "Exploring Efficient Methods to Extract Pedestrian Flows on a Mobile Adhoc Network," The Sixth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2012), pp. 29–34, 2012.

[5] R. Nishide, T. Ushikoshi, S. Nakamura, and Y. Kono, "Detecting Social Contexts from Bluetooth Device Logs," 11th International Conference on Ubiquitous Computing (UbiComp 2009) Supplemental Proceedings, pp. 228–230, 2009.

[6] E. O'Neill, V. Kostakos, T. Kindberg, A.F. Schiek, A. Penn, D. Fraser and T. Jones. "Instrumenting the city: Developing methods for observing and understanding the digital cityscape," 8th International Conference on Ubiquitous Computing (Ubicomp 2006), pp. 315–332, 2006.

[7] T. Nicolai and H. Kenn. "About the relationship between people and discoverable bluetooth devices in urban environments," 4th international conference on mobile technology, applications, and systems (Mobility '07), pp. 72–78, 2007.

[8] N. Eagle and A. Pentland. "Reality mining: sensing complex social systems," Personal and Ubiquitous Computing, 10, pp. 255–268, 2006.

[9] D. Kim and D.-K. Cho, BlueSense: Detecting individuals, locations, and regular activities from Bluetooth Signals, http://www.cs.ucla.edu/∼dhjkim/files/pdf/cs219_BlueSense.pdf (Retrieved August 23, 2012)

[10] J. Weppner and P. Lukowicz, "Collaborative Crowd Density Estimation with Mobile Phones," Second International Workshop on Sensing Applications on Mobile Phones (PhoneSense 2011) at ACM SenSys, pp.26–30, 2011.

[11] E. Bulut and B. K. Szymanski, "Exploiting Friendship Relations for Efficient Routing in Mobile Social Networks," IEEE Transactions on Parallel and Distributed Systems, 23 (12), pp.2254–2265, 2012.

[12] Chika-game / Chika-Chat | Softbank (in Japanese), http://mb.softbank.jp/mb/service/3g/communication/chika/ (Retrieved August 23, 2012)

[13] M. Ohnishi, R. Nishide, and S. Ueshima, "Incremental Construction of Delaunay Overlaid Network for Virtual Collaborative Space," The Third International Conference on Creating, Connecting and Collaborating through Computing (C5'05), IEEE CS Press, pp. 77–84, 2005.

[14] Y. Sun, Q. Jiang, and M. Singhal, "An Edge-Constrained Localized Delaunay Graph for Geographic Routing in Mobile Ad Hoc and Sensor Networks," IEEE Transactions on Mobile Computing, 9(4), pp. 479–490, 2010.

[15] M. Kawabata, R. Nishide, M. Ueda, and S. Ueshima, "The Context-adaptable Pedestrian Navigation System and Usability in Practical Settings," IEEE Pacific Rim Conference on Communications, Computers and Signal processing (PACRIM2005), Vol.1 No.05CH37690, pp.368–371, 2005.

[16] R. Nishide, I. Kimura, M. Ohnishi, and S. Ueshima, "Modeling of Dynamic Behavior of Multi-Avatars in Virtual Collaborative Space," IEEE Pacific Rim Conference on Communications, Computers and Signal processing (PACRIM2003), Vol.1 No.03CH37490, pp.25–28, 2003.

[17] G. Pestana, A. Casaca, P. Reis, S. Heuchler, and J. Metter, "Management of Mobile Objects in an Airport Environment," The Second International Conference on Mobile Services, Resources, and Users (MOBILITY 2012), pp. 55–59, 2012.

www.iariajournals.org

International Journal On Advances in Intelligent Systems

ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, ENERGY, COLLA, IMMM, INTELLI, SMART, DATA ANALYTICS

issn: 1942-2679

International Journal On Advances in Internet Technology

ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING, MOBILITY, WEB

issn: 1942-2652

International Journal On Advances in Life Sciences

eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO, SOTICS, GLOBAL HEALTH

issn: 1942-2660

International Journal On Advances in Networks and Services

ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION, VEHICULAR, INNOV

issn: 1942-2644

International Journal On Advances in Security

ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS

issn: 1942-2636

International Journal On Advances in Software

ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, IMMM, MOBILITY, VEHICULAR, DATA ANALYTICS

issn: 1942-2628

International Journal On Advances in Systems and Measurements

ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL, INFOCOMP

issn: 1942-261x

International Journal On Advances in Telecommunications

AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA, COCORA, PESARO, INNOV

issn: 1942-2601