International Journal on Advances in Networks and Services

2013 vol. 6 nr. 3&4

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 3 & 4, 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
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, University of Malta, Malta
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, Middle Tennessee State University, 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
Michael von Riegen, University of Hamburg, Germany

Joris Walraevens, Ghent University, Belgium
You-Chiun Wang, National Sun Yat-Sen University, Taiwan
Gary R. Weckman, Ohio University, USA
Chih-Yu Wen, National Chung Hsing University, Taichung, Taiwan
Michelle Wetterwald, EURECOM - Sophia Antipolis, France
Feng Xia, Dalian University of Technology, China
Kaiping Xue, USTC - Hefei, China
Mark Yampolskiy, Vanderbilt University, USA
Dongfang Yang, National Research Council, Canada
Qimin Yang, Harvey Mudd College, USA
Beytullah Yildiz, TOBB Economics and Technology University, Turkey
Anastasiya Yurchyshyna, University of Geneva, Switzerland
Sergey Y. Yurish, IFSA, Spain
Faramak Zandi, La Salle University, USA
Jelena Zdravkovic, Stockholm University, Sweden
Yuanyuan Zeng, Wuhan University, China
Weiliang Zhao, Macquarie University, Australia
Wenbing Zhao, Cleveland State University, USA
Zibin Zheng, The Chinese University of Hong Kong, China
Yongxin Zhu, Shanghai Jiao Tong University, China
Zuqing Zhu, University of Science and Technology of China, China
Martin Zimmermann, University of Applied Sciences Offenburg, Germany

International Journal on Advances in Networks and Services
Volume 6, Numbers 3 & 4, 2013

CONTENTS

Heterogeneous Wireless Network Selection: Load Balancing and Multicast Scenario

Svetlana Boudko
Norsk Regnesentral
Oslo, Norway
svetlana.boudko@nr.no

Wolfgang Leister
Norsk Regnesentral
Oslo, Norway
wolfgang.leister@nr.no

Stein Gjessing
University of Oslo
Norway
steing@ifi.uio.no

***Abstract*—The increasing demand for real-time multimedia streaming from mobile users makes important deployment of network selection in wireless networks. Coexistence of various wireless access networks and ability of mobile terminals to switch between them make an optimal selection of serving mobile networks for groups of mobile clients a challenging problem. Since scalability can easily become a bottleneck in large-scale networks, we study the decision-making process and selection of the data that needs to be exchanged between different network components. In this paper, we present two decentralized solutions to this problem that we compare and evaluate in the OMNet++ simulation environment.**

***Keywords*—Wireless networking; mobile network selection; decentralized algorithms.**

I. Introduction

This article extends the work presented by Boudko et al. [1, 2] and studies load balancing and multicast communication over heterogeneous mobile networks. This article is also an extention of [3, 4].

Availability of various wireless network technologies and continuous development of mobile devices and services lead to complex and highly dynamic networking and challenge resource limitations of wireless access networks. According to a recent forecast [5], monthly data consumption for wireless networks will increase over 15 times in the years between 2011 and 2016. In 2016, the demand for mobile bandwidth will exceed the average capacity by about 32 %. Despite constantly increasing demand, the range of frequencies is the same. Consequently, during peak demands when the bandwidth becomes an insufficient resource the consumer is likely to experience degradations in the form of reduced service, slow service, or even no service.

To avoid some of these negative effects of a network that is challenged by resource limitations we need to consider the resource allocation problem from a different angle, including collaboration between mobile user nodes and networks to improve the overall utilization of resources. Referring to wireless access networks, the ability to be connected to several network technologies simultaneously offers new possibilities to formulate effective strategies for network selection.

The network selection problem inspired by the "always best connected" concept was mostly focused on the definition of metrics to address the best end user quality of service for each single consumer, neglecting the impact to the other consumers in the network. Contrary to this, we use metrics that express quality of service for all users collectively and evaluate benefits for the system components from complexly applied network selection. In our problem formulation, we take into account that *1*) a large number of mobile devices can operate simultaneously inside an area with overlapping coverage of various mobile networks; *2*) some of the networks can experience degradation of service while some of them can accommodate more users; and *3*) some groups of these devices can listen to the same feeds from the same Internet locations while being connected to different access points. We consider the network selection problem to use for multi-user environments with possible multicast configurations that allows the network to perform load balancing, improve the users' overall QoS, and increase the networks' throughput.

Being originally introduced for use in the wired Internet, multicast is an efficient method for point-to-multipoint communications, which reduces drastically the traffic load when the same content is sent to a large group of users. The 3rd Generation Partnership Project (3GPP) and its successor 3GPP2 defined the *Multimedia Broadcast and Multicast Service* (MBMS) and the *Broadcast and Multicast Service* (BCMCS) [6], respectively. The Long-Term Evolution (LTE) project introduced *LTE Broadcast*, also denoted as *evolved Multimedia Broadcast Multicast Service* (eMBMS) [7]. Different types of applications like video conferencing, file distribution, live multimedia streaming, IPTV can benefit from deploying multicast networking. It is also advantageous in cases of the flash crowd phenomenon when the popularity of a certain item increases rapidly over a short period of time. The *LTE whitepaper* [8] shows that already from three to five subscribers in one cell site achieve break-even of cost between unicast and multicast.

However, the complexity of managing multicast networks makes the deployment of multicast even more challenging in wireless environments when mobility issues have to be considered. Also, the notion of a link interface for a wireless multicast channel differs from that for a wired network. Multicast management in wireless heterogeneous networking also involves mobile network selection for a group of clients in addition to construction of multicast trees like in conventional multicast protocols.

In this paper, we consider a solution for the network selection problem for heterogeneous mobile networking as a part of multicast group management. Previously, we have proposed a method that provides an optimal network selection for a given network topology, network conditions and user preferences assuming that all needed information can be collected from the network and is available for a central decision-making unit that

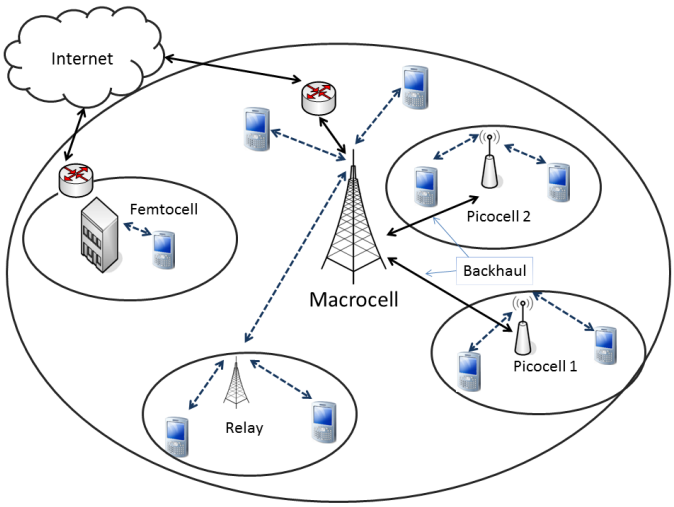


Figure 1. LTE Advanced Heterogeneous Network Architecture.

computes the assignment of mobile multicast groups of nodes to networks [1, 2]. In this paper, we refine the aforementioned method and apply it as an upper bound for evaluating methods that use only limited information shared among the decision makers.

The work proposed in this paper includes the following contributions: *1*) We propose two approaches that allow network selection in a decentralized manner with only limited information shared among the decision makers. *2*) Through extensive simulations, we study how different sets of information available to decision makers influence the performance of the system. *3*) We discuss how the evaluated approaches can be combined and propose ideas for further improvements.

The remainder of the paper is organized as follows. After presenting an overview of related work in Section II, we discuss two representative scenarios for load balancing and for multicast networking in Section III. Load balancing problem in heterogeneous wireless network is addressed as follows. We present the problem as a team-decision problem in Section IV, and outline suitable algorithms. The simulation set-up and the results of simulations are presented and analyzed in Section V Multicast transmission in heterogeneous wireless network is covered in the following sections. We present the problem formulation and outline an optimal solution to the problem in Section VI. The system components are considered in Section VII. Two decentralized solutions are presented in Section VIII. Simulation results are given in Section X. We discuss future work and conclude in Section XI.

II. Related Work

To the best of our knowledge, the research field regarding resource allocation and selection of a network in heterogeneous wireless networks from a perspective of load balancing and multicast delivery is not well explored. In what concerns network selection for mobile multicast groups, four research areas can be considered as related work *1*) handoff management; *2*) network selection in wireless networks; *3*) multicast in wireless networks; *4*) LTE-Advanced Heterogeneous Networks.

A. Handoff Management in Mobile Networks

Prediction-based techniques have been suggested in several studies [9–12] aiming to reduce handoff delays.

To represent the movement behavior of a mobile user, Paramvir et al. [9] propose a two-level user mobility model consisting of a local level and a global level. A hierarchical location prediction algorithm is proposed based on an approximate pattern-matching algorithm implemented in the global level and Kalman filtering techniques implemented in the local level.

Akyildiz and Wang [10] propose a mobility model that uses historical records and stochastic behavior of mobile users to predict their future position. The model is built upon a framework of user mobility profiles (UMP). In the proposed prediction algorithms, many factors are taken into consideration including velocity and direction of mobile users, historical records, stochastic model of cell residence time and path characteristics. The authors claim that these algorithms predict more accurately than previous schemes. However, the complexity of the algorithms make them impractical for mobile applications.

In two studies, Tseng et al. [11] and Choi et al. [12] propose using cross-layer information to perform layer-3 handoff in parallel with or prior to the layer-2 handoff. However, these schemes can lead to false alarms and cause unnecessary MIP registrations. Ray et al. [13] conclude that deciding upon the ideal choice and timing of cross-layer triggers in order to reduce layer-3 latency is still an open problem.

Vertical handoff is the handoff between the networks of different wireless technologies and has been addressed in several studies [14–20]. While horizontal handoffs are typically triggered when the received signal strength (RSS) of the serving access router drops below a certain threshold the vertical handoff can be initiated due to other reasons such as user preferences or network conditions including coverage, bandwidth, cost and power consumption. The decision process is therefore more complex for vertical handoffs than for horizontal ones.

While some authors only use RSS as an input parameter for the handoff decision process [10, 21], others combine the use of RSS with bandwidth information [22–24]. Using cost functions has been proposed earlier [14–16]. Nasser et al. [25] propose a cost function that depends of the cost of service, security, power consumption, network conditions and network performance. However, in their evaluation, all weights except the bandwidth weight are set to zero. This renders their cost function to a function of one parameter: bandwidth.

Algorithms based on fuzzy logic or artificial neural networks in combination with multiple criteria [17–19] suffer from high handover delay because of their complexity and the training process. Unfortunately, the authors of these algorithms did not provide throughput results.

Recently, some studies [26, 27], proposed solutions for group vertical handoffs in heterogeneous environments. These studies consider scenarios when many mobile users send handover requests almost at the same time. In these scenarios, the influence of multiple users is important to consider for optimal network selection. The solutions presented in both studies require a centralized approach to be adopted to implement the proposed schemes. The obvious drawback of this approach is a poor scalability of these solutions.

B. Admission control and network selection in wireless networks

Ormond and Murphy [28] propose a network selection strategy that explores a number of possible utility functions. The solution is user-centric, and an interplay between different users and networks is not considered. Ormond and Murphy conclude that the impact of multiple users operating in the same region needs to be further examined.

Gluhak et al. [29] consider the problem of selecting the optimal bearer paths for multicast services with groups of heterogeneous receivers. The proposed algorithm selects the bearer path based on different optimization goals. However, Gluhak et al. address the problem only for the ideal static multicast case without taking into account users crossing different cells. In their work, multicast membership does not change during the duration of a service, and multicast groups are not built with consideration of users' movements. In our opinion, this is not a realistic case for wireless networks.

Jang et al. [30] present a mechanism for efficient network resource usage in a mobile multicast scenario. This mechanism is developed for heterogeneous networks and implements network selection based on network and terminal characteristics and QoS. However, in the proposed mechanism, the network selection is performed purely based on terminal's preferences, the network perspective is not considered, and the solution does not optimize the utilization of network resources.

Tragos et al. [31] propose a generic admission control algorithm that allows network selection for 4G heterogeneous wireless networks. The algorithm aims to provide maximum utilization of the network, prevent overloading situations and ensure best QoS. However, implementation of the algorithm requires the presence of a centralized entity.

Khan et al. [32] present a game theoretic solution for resource allocation and call admission in wireless networks using cooperative games. The main goal is to increase the utilization of the available bandwidth and to reduce the call blocking. The solution is applicable to wireless network scenarios where networks are willing to cooperate. Kalai-Smorodinsky Bargaining Solution is used to solve the cooperative game. The authors also propose the request distribution algorithm that allows to allocate the request to several different networks and split the requested bandwidth between these networks. Simular to Tragos et al. [31], the implementation requires also a centralized entity that is responsible to handle bargins between the participating networks.

In our analysis, we recognize that several problems are not yet addressed, or where the currently available solutions need to be improved. Decentralized algorithms that rely on information only partly shared between the decision-makers need to be implemented and evaluated in multi-user scenarios. These considerations motivate us to look at distributed and computationally efficient methods of network selection in heterogeneous mobile environments.

C. Network Selection in Wireless Networks

Ormond and Murphy [28] propose a network selection approach that uses a number of possible utility functions. Their solution is user-centric and does not present any multicast scenario. An interplay between different users and networks is not considered either. Ormond and Murphy conclude that the impact of multiple users operating in the same region needs to be further examined.

Gluhak et al. [29] consider the problem of selecting the optimal bearer paths for multicast services with groups of heterogeneous receivers. The proposed algorithm selects the bearer path based on different optimization goals. However, Gluhak et al. address the problem only for the ideal static multicast case without taking into account users crossing different cells. In addition, it requires that the knowledge of the conditions in wireless networks and preferences of receivers is fully shared. In their work, multicast membership does not change during the duration of a service, and multicast groups are not built with consideration of users' movements. In our opinion, this is not a realistic case for wireless networks. Also, the proposed selection algorithm is built upon a rule according to which the receivers are partitioned into two sets: the receivers for which only one network is available versus the receivers for which several networks are available. The impact of the users inside the second group, as a result of this partitioning, is neglected.

Yang and Chen [33] propose a bandwidth-efficient multicast algorithm for heterogeneous wireless networks that is formulated as an Integer Linear Programming problem that is solved using Lagrangian relaxation [34]. The algorithm deals only with constructing optimal shortest path trees for multicast groups. In this approach, important parameters, such as cost of service or the user's velocity, are not considered.

Jang et al. [30] present a mechanism for efficient network resource usage in a mobile multicast scenario. This mechanism is developed for heterogeneous networks and implements network selection based on network and terminal characteristics and Quality of Service (QoS). However, in the proposed mechanism, the network selection is performed purely based on terminal's preferences, the network perspective is not considered, and the solution does not optimize the utilization of network resources.

Hou et al. [35] propose a cooperative multicast scheduling scheme for multimedia services in IEEE 802.16 based wireless metropolitan area networks (WMAN). The scheduling is considered for one base station that further re-sends the data to multiple subscriber stations. These are grouped into different

multicast groups and the users are assigned to the groups. The authors consider two approaches to select multicast groups for services: the random selection and the channel state aware selection. The process is controlled by the base station and limited to one network technology. No network heterogeneity is considered.

D. Multicast in Wireless Networks

The Multicast Mobility (multimob) working group [36] focuses its activity on supporting multicast in a mobile environment. The main goals of the group are to work out mechanisms for supporting multicast source mobility and mechanisms that optimize multicast traffic during a handover. The group also documents the configuration of IGMPv3/MLDv2 in mobile environments. In this sense, they extend the IGMPv3/MLDv2 protocols for implementation in the mobile domain and improve *Proxy Mobile IPv6* to handle multicast efficiently. However, they do not consider any modifications across different access networks.

The Long-Term Evolution (LTE) project introduces evolved Multimedia Broadcast Multicast Service (eMBMS) [7]. This standard covers technically the terminal, radio, core network, and user service aspects that provide a point-to-multipoint service for transmitting data from a single source to multiple recipients. The performance is improved due to higher and more flexible LTE bit rates, single frequency network (SFN) operations, and carrier configuration flexibility. The eMBMS Service Layer offers a Streaming- and a Download Delivery Method and is enhanced with video codec for higher resolutions and frame rates and forward error correction (FEC), and the radio network with procedures to ensure MBMS reception in a multifrequency LTE network. eMBMS also allows LTE network and backhaul offloads.

E. LTE-Advanced: Heterogeneous Networks

Though several improvements were introduced in the LTE-Advanced standard [37], the homogeneous networks with only macrocells deployments will not be able to cope with future mobile traffic. A step towards optimization of performance in wireless networks is done by LTE-Advanced [38] through enhancements in network topology. The LTE-Advanced proposed implementation of heterogeneous networks (HetNets) topologies that combine utilization of both macrocells and small cells, the latter including micro, pico, femtocells and relays, each having different transmit power and access rules for user devices.

1) Macrocells: Macrocell is an outdoor base station and the main base station in the cell. The transmitted power is about 45 dBm. Macro cells are connected with each other through backhaul that are usually built upon a wired infrastructure. In some cases, e.g., for rural areas, wireless links can also be used.

2) Micro and Picocells: These cells are, usually, an outdoor low cost base stations with open access and a small coverage.They are connected with the macro cell using a backhaul link. The transmitted power is about 35 dBm. The range is about two kilometers wide for microcell and about 200 meters for picocell.

3) Femtocells: Femtocells are indoor base stations either with open or limited access and low transmitted power that is less than 23 dBm. Though these cells are positioned as an alternative to micro and picocells, their coordination with the macrocell still is not fully achieved in current deployments.

4) Relays: Relay stations receive, demodulate and retransmit the signals between base stations and mobile users. They can decode the data and provide error correction. Relays are used to increase throughput and to extend coverage of cellular networks. Relays do not need wired connection to the base station; therefore, the backhaul costs can be saved.

In the HetNet network model, macrocells provide full coverage for a wide area and small cells cover some areas with extra traffic demand. It is a useful network architectural feature since the bandwidth demand is not uniform across the area and users and traffic are often concentrated in particular areas. Another important benefit of using small cells inside of a macrocell is to improve coverage in places where coverage of the macrocell is not sufficient, e.g., in cell edges. Since deploying extra macro cells in these areas results in additional interferences, the deployment of lower power picos is a better solution and can give a cost reduction. A typical LTE-Advanced HetNet scenario with a macro base station and several small cells is illustrated in Figure 1.

For the purpose of this paper, we evaluate network selection solutions considering both the LTE-Advanced HetNet network model and a general heterogeneous mobile network system with overlapping of several wireless technologies, e.g., Wi-Fi and cellular networks.

In our analysis, we recognize that the presented previous work has not addressed several important aspects related to the network selection for mobile multicast groups. We need to study how the users' movements influence the optimal selection of members for multicast groups and how the information needed for network selection is exchanged between the decision makers.

III. Scenario

A. Load Balancing Scenario

We consider a network selection scenario for a group of users in a hotspot area like a crowded city center, a public transportation node or an exhibition site where a coverage of several base stations or access points from different networks is possible. We assume a substantial overlap in coverage of these stations. The networks implement different network technologies. We also consider a situation with multiple overlapping IEEE 802.22 wireless regional area networks where self-coexistence is allowed. A representative scenario of such networking is illustrated in Figure 2. These user terminals are capable of connecting to several access networks, and vertical handoffs between different networks are technically possible. The terminals periodically receive beacon signals from base stations or access points of the available access networks that are typically broadcast once per second.

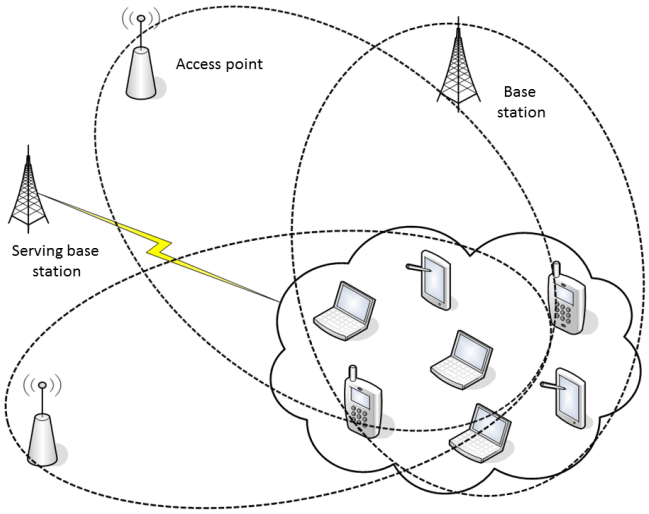


Figure 2. Network topology built upon multiple mobile networks with heterogeneous technologies to serve a group of clients.

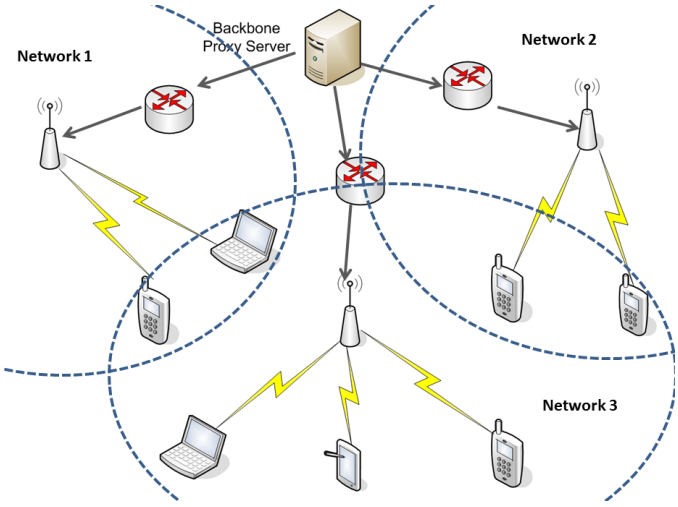


Figure 3. Multicast streaming scenario for a group of mobile clients served by several mobile networks before regrouping.

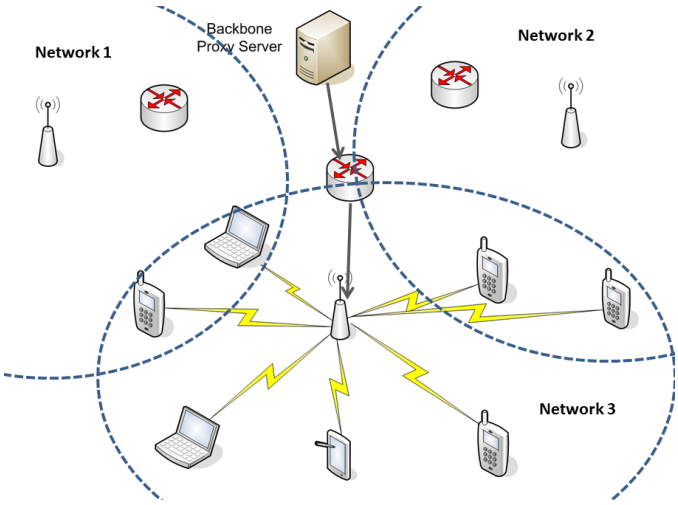


Figure 4. Multicast streaming scenario for a group of mobile clients switched to one mobile network after regrouping.

Users located in the same cell of a mobile network can experience degradation in quality due to shortage of available bandwidth. Though admission control mechanisms are designed to ensure the quality of wireless connections and to prevent network congestion, there is still a possibility that a user is admitted to the network while requiring low bandwidth, e.g., for web browsing, will require more resources for video streaming shortly after. We consider also a situation when a base station may act in a proactive way and monitor the available resources in adjacent cells. The users that are going to move to a cell that, at the moment, is not able to admit new users can be notified to perform a vertical handoff to another available network. Since users may have different preferences and request different types of service their utility functions are built upon different criteria.

To provide better load balancing between the networks, and to avoid disturbing ping-pong effects, joint coordination, and information exchange between the users and the base stations is essential; both the clients and the networks can benefit from cooperative handoffs. However, due to strict bandwidth and power limitations of mobile networks, and also due to scalability issues, a complete information exchange between mobile users and networks is not feasible. Decentralized network selection is therefore essential.

B. Multicast Scenario

To illustrate the yet unsolved challenges for optimal network selection in multicast networks, we consider a multimedia streaming scenario for a group of mobile users that concurrently receive the same content from the Internet. We assume that a backbone proxy server (BPS) is placed at the network edge. The BPS is a member of a content distribution system (CDN). This scenario is an extension of a scenario that we previously have considered to illustrate an adaptive multimedia streaming architecture to mobile nodes [39].

The BPS streams content that either is hosted on a streaming server, or re-sends the streaming content as a part of an application layer multicast. The users of this network are located in an area with a substantial overlap in coverage of several mobile networks, and are connected to different networks. One of examples of such networking is Heterogeneous networks (HetNets) in LTE-Advanced [38].

The base stations of the system have multicast capabilities, implementing, for example, Multimedia Broadcast Multicast Service [40]. A representative scenario of such networking is illustrated in Figure 3.

In our scenario, we assume that the mobile terminals are capable to connect to several access networks, and vertical handoffs between these networks are technically possible. Further, we assume that these terminals are equipped with GPS receivers, so that their location information can be transmitted to the BPS. The BPS can use this information to determine how users can be regrouped in multicast groups. Such regrouping is beneficial as it saves network resources. Hence, users

that get the same content can exploit the same wireless link because the content can be broadcasted to them. The resources in the backhaul network are also better utilized because the content is now delivered only to one mobile network instead of being spread to several networks. An example of such regrouping is depicted in Figure 4.

Technically, to facilitate such a mechanism, the user terminals will have the possibility to switch to other mobile networks after receiving certain messages from the BPS. Since users may have different preferences depending on diverse criteria, for example, power consumption, security, or network cost of service, the interplay between the users' utilities and the networks' utilities is important to consider. Network selection support for multi-access networks can be implemented at any layer of the protocol stack. There are certain tradeoffs to consider at the design stage. Cross-layer signaling can potentially be added to allow the application level to control the process, hence, to prevent breakup of ongoing sessions.

IV. Problem Formulation for Load Balancing in Heterogeneous Wireless Networks

Decentralized network selection can be formulated as a team decision problem [41, 42] where several decision variables are involved. These decisions are made by different decision makers that have access to different information but participate in a common goal.

Team decision theory is concerned with determining the optimal decisions, given a set of information for each of several decision makers, that work together to achieve a payoff. In team decision problems, these sets of information are different though often correlated for different decision makers. These optimal decisions can be either person-by-person optimal or team optimal. In person-by-person optimal cases, each person makes the decisions that optimize the individual's payoff, but not necessarily the team payoff. These cases are optimal for a particular team member, given that the decision functions for other members are fixed. In team optimal cases, the group payoff is optimized. Team optimality is a stronger condition, and is thus harder to achieve. Taking into account that person-by-person optimal strategies may result in unfair distribution of the resources, we focus our research on team optimal strategies.

A. System Model

Taking into consideration our understanding about preferences of mobile nodes and the networks, we are now ready to formalize our observations into a system model. We consider a set of networks $N = 1, 2, ..., n$ and a set of mobile terminals $M = 1, 2, ..., m$. For each terminal m_j and network n_i the following is defined. Streaming bitrate requirements of mobile nodes are denoted by r_j; $rss_{i,j}$ is the received signal strength in network i for terminal m_j while power consumption and cost of service in network n_i for terminal m_j are denoted by $p_{i,j}$ and $c_{i,j}$, respectively. The total available bandwidth of network n_i is denoted by b_i. For each terminal m_j, we define a user preference profile that is described by a tuple containing $Th^p_{i,j}$, $Th^c_{i,j}$, and $Th^{rss}_{i,j}$. These denote thresholds, or user preferences, for respectively power consumption, cost of service and received signal strength. We define a time period $\tau_{i,j}$ during which terminal m_j is served by network n_i before performing a handoff and moving to the next cell of this network.

For each mobile terminal m_j and each network n_i we define the function $x_{i,j}$ which mimics the decision taken by a mobile terminal m_j to switch to or to stay in mobile network n_i.

$$x_{i,j} = \begin{cases} 1, & \text{if } m_j \text{ has roamed to or stays in } n_i \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

We define the common goal of the set of networks and users of these networks as maximization of consumed bandwidth over a period of time, minimization of the number of handoffs and reduction of signalling between the networks and terminals. To achieve this goal, the participating networks and the terminals need to cooperate while trying to maximize their own performance. To facilitate a decentralized approach, we define two components, the network component and the mobile node component. We formulate the problem and solve it separately for these two components.

1) Network Component: For each network n_i, we define a utility function U_i as a sum of consumed bandwidth of all users of this network over the time the user is connected to it, as defined in Eq. (2).

$$\forall\{i\} : U_i = \sum_j x_{i,j} \cdot r_j \cdot \tau_{i,j} \quad (2)$$

In this sense, the networks benefit if they select the users that not only request a higher bandwidth but also intend to stay in the network for longer periods of time, which also can eliminate the ping-pong effect when the user needs to change the network again recently after the handoff. Basing our decision on research done by other research groups [43–48], we assume that a mobile network is capable of predicting the residence time of a mobile node inside the network based on mobile node's velocity, movement patterns and the local area. We realize that this problem is an ongoing research work. For the purpose of this paper, we assume that the prediction can be performed with acceptable precision.

The common goal is the maximization of the expected value of the sum of network utility functions.

$$\max \sum_i \mathbb{E}[U_i] \quad (3)$$

The utility function is constrained by the network resources. As any network of the system has a limited knowledge about the resources and decisions of the rest of the system, $x_{i,j}$ is given as its expected value.

$$\forall\{i\} : \sum_j \mathbb{E}[x_{i,j}] \cdot r_j \leq b_i \quad (4)$$

At Mobile Node m_j :
if Network Selection is triggered
search for available networks
for each available network n_i
  if ($p_{i,j} < Th^p_{i,j}$, $c_{i,j} < Th^c_{i,j}$, $rss_{i,j} < Th^{rss}_{i,j}$) then
    add the network to list of candidate networks
for each network n_i in list of candidate networks
  send requests to candidate network n_i
wait for response from candidate networks
upon reception of response from candidate network n_i
if (admitted value == true) then
  add response to response list
if (response list == null)
  stay in the current network
else
  for all responses in response list
    choose the network with the highest $\tau_{i,j}$

Figure 5. Distributed Algorithm for Network Selection, Mobile Node Component

At Candidate Network n_i :
wait for admission requests from mobile nodes
upon reception of admission requests from mobile node m_j
  using available knowledge solve Eq. (2) with constraints Eq. (4)
  return *admission* message containing
    admitted value = true/false and expected time

Figure 6. Distributed Algorithm for Mobile Node Admission, Network Component

2) Mobile Node Component: For each user we define a utility function f_j as a function of power consumption, cost of service, available bandwidth and received signal strength. We realize that the set of parameters that define user preferences can be larger than the one mentioned above and can also differ from user to user. We also realize that users might employ different utility functions but for this work we limit us to this definition.

$$\max f_j(p_{i,j},\ c_{i,j},\ r_j,\ rss_{i,j}) \tag{5}$$

Eq. (5) poses a multiparameter optimization problem that can be solved by introducing weights and normalization. Another solution is to relax the optimization problem by reducing it to a one variable optimization Eq. (6). Consequently, we introduce a set of constraints in Eq. (7), where some of parameters $z \in \{p, c, rss\}$), namely power consumption, cost of service, and received signal strength, are limited by their thresholds $Th^z_{i,j}$ defined by the user's preferences.

$$\max \sum_i x_{i,j} \cdot r_j \cdot \tau_{i,j} \tag{6}$$

$$p_{i,j} \leq Th^p_{i,j}, \quad c_{i,j} \leq Th^c_{i,j}, \quad rss_{i,j} \leq Th^{rss}_{i,j} \tag{7}$$

B. Algorithm

The system model defined in Section IV-A is used in the decentralized algorithm for network selection outlined below. We build the algorithm based on the following *a*) maximization of the total consumed bandwidth by distributing the users between the networks taking into account the networks' available bandwidth, and *b*) minimization of the number of performed handoffs between the networks.

There are two events that may trigger the execution of the network selection algorithm: *1*) based on monitoring of available resources in its cells, and the prediction of users' location information, the network informs mobile nodes that are about to move to a congested cell to switch to another available network instead; *2*) mobile terminal m_j experiences degradation of network performance detected by increased packet loss or delay on the mobile terminal.

When the network selection is triggered, the effected terminal runs the selection algorithm as shown in Figure 5. As a consequence, the network receives calls from mobile nodes it runs the algorithm shown in Figure 6. To calculate the expected values of the utility function defined by Eq. (2), this algorithm takes as input the knowledge available for this network. Depending on how the knowledge of the system is shared among the networks, we differ between two versions of the algorithm: Algorithm A and Algorithm B.

1. **Algorithm A**: Each mobile node m_j, while sending a request to the network n_i, informs the network about the requests sent to other networks. Based on this information, each network calculates the probability of mobile node m_j to choose this network if accepted.
2. **Algorithm B**: Each network n_i shares its information with exactly one more network n_k. The network n_i does not accept a mobile node m_j if the node is accepted by the network n_k and $\tau_{i,j} < \tau_{k,j}$.

Also, we consider using a combination of these two versions, refered further as Algorithm AB. In this version, in addition to the information exchanged between the networks as in Algorithm B, the nodes specify in their requests the number of all requested networks, as in Algorithm A.

C. Algorithm Evaluation

To evaluate the algorithms, we define upper and lower bounds to their operation. The upper bound is achieved by applying a centralized solution with fully shared knowledge of the conditions in all evaluated networks and is further referred as global knowledge reference. This reference can also be viewed as a modification of the algorithms [26, 27, 31] discussed in Section II-B The algorithm [31] is now extended to a multi-user scenario. Its utility function is defined in Eq. (8). The utility function is constrained by resource limitations of networks as described in Eq. (4) and preferences of mobiles nodes described in Eq. (7). This problem belongs to the class of integer linear programming problems, which is known to be NP-complete, thus problematic for real-time tractable implementation and in most cases can be used only as a reference for evaluating algorithms.

$$U = \sum_i \sum_j x_{i,j} \cdot r_j \cdot \tau_{i,j} \tag{8}$$

The lower bound corresponds to a situation when all networks base their decisions only on local knowledge (Eq. (2)) and is further referred as local knowledge reference. The local

knowledge reference can also be viewed as a modification of algorithm [28] discussed in Section II-B and applied to a multi-user scenario. In this sence, the algorithms are compared with the related work.

V. Simulations for Load Balancing in Heterogeneous Wireless Networks

The performance and functionality of the algorithms have been evaluated through multiple simulation runs. We have implemented both versions of our algorithm in the OMNet++ environment [49]. In Algorithm A, the network gets the information about how many other networks are requested by the same mobile node. This information is submitted to the network by the mobile node along with the request to join the network. As no other information is available, we assume that the probability of being assigned to any of the networks is equal for all participating networks. In Algorithm B, each network shares its information with exactly one more network. In our testing scenario, these networks do not overlap. We compare the algorithms with the global knowledge reference and the local knowledge reference.

A. Simulation Setup

For the sake of simplicity, we simulate a scenario with four wireless networks, which covers quite well the scope of the evaluation. In this scenario, a group of users from one network is about to move from one cell of the network to another cell of the same network that experiences a shortage of available bandwidth. In consequence, the cell that the users move to is not able to accommodate all these users.

For our evaluation, we run tests with 100, 200, and 300 users moving to this congested cell. Further, we divide the users into four categories in terms of requested bandwidth. To define these categories we use service class characteristics defined by Tragos et al. [31] as follows: *a*) at 64 kbps, for simple telephony and messaging *b*) at 512 kbps, for web browsing *c*) at 1024 kbps, for interactive media and *d*) at 2000 kbps, for video streaming, each category having approximately the same number of users.

Note that none of the networks have enough resources to accommodate all users alone. All four networks must be used in order to meet the requirements of all users. We run also tests when total bandwidth of all networks is not sufficient to accommodate all users. The tests are done for network conditions that result in 5%, 10%, 15%, 20%, 25%, 30% dropped calls if the global knowledge reference is applied. The time $\tau_{i,j}$ for the user j to stay in the network n_i before performing a horizontal handoff, or a cell residence time, is randomly distributed in the range $[1, 100]$ time units.

B. Simulation Results

We evaluate how mobile nodes are distributed among the networks after one iteration of the algorithm run. We calculate the number of decision errors as a number of users whose connection ends up in dropped calls due to wrong network allocation. These errors are the results of wrong assignments to networks that do not have sufficient bandwidth to accommodate the assigned users. For each group of users (100, 200, 300 users), we repeat the experiment 1000 times with different sets of $\tau_{i,j}$.

For all tests done, the top and bottom 5 % of the results are excluded from the evaluation. The results are averaged over these simulation runs and are depicted for minimum value results in Figure 7(a), for average value results in Figure 7(b), for maximum value results in Figure 7(c), for cumulative distribution function in Figure 8. The global knowledge reference is 0 for all experiments meaning that in the centralized solution, all users were assigned to the networks without any dropped calls. The results with dropped calls in the global knowledge reference are depicted in Figure 9. The figure shows the results for 200 mobile nodes. The results for 100 and 300 mobile nodes are very similar to the results for 200 nodes and, therefore, are not included in the paper.

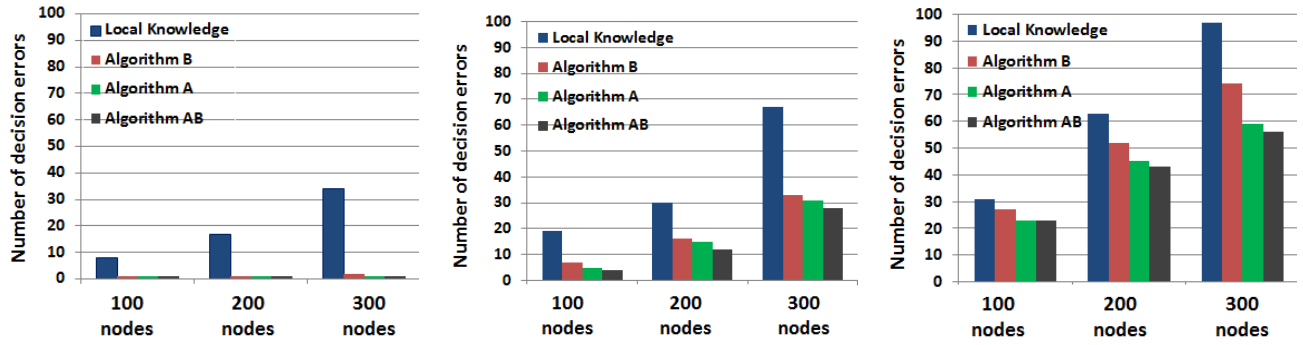
The tests show that all three proposed algorithms can distribute the users between the networks significantly better than the local knowledge reference. Algorithm AB performs better than Algorithm A and Algorithm A performs better than Algorithm B for all user groups through all tested values for dropped calls in the optimum solution.

It shows that sharing partial information about network status as in Algorithm B makes little use of extra information from just one network. It also shows that this information when used in Algorithm AB does not give any significant reduction of decision errors in comparison with Algorithm A. However, Algorithms B and AB require significantly more information to exchange between the networks than Algorithm A. It also requires more sophisticated mechanisms and protocols to be implemented in the networks, including security considerations and synchronization of the information flow. Though the information flow initiated by Algorithms AB and B is significantly less than the one initiated by the global knowledge reference, it still demands the exchange of network information across the mobile networks on a fast time scale and low-latency basis, making it quite challenging to implement the algorithms in practice for large scale networks, as the global knowledge reference.

We also evaluate the dynamic scenario. For these tests, the algorithms are run until all clients are assigned to the networks with sufficient bandwidth, also considering the arriving calls. The arrival rate of new calls is modeled with a Poisson stream. The graphs depicted in Figure 10 show the averaged results for 100, 200, and 300 users over 1000 test runs. The x-axis shows the number of iterations of the algorithm. The y-axis shows the percentage of decision errors. Clearly, Algorithms A, B and AB converge faster than the local knowledge reference. There is very little difference between Algorithms A, B and AB even though Algorithms AB and B rely on more information.

C. Signaling Overhead

We estimate signaling overhead S_o for the algorithms and the references. As signaling required to trigger network selection is the same for the references and the algorithms these



(a) Minimum value (b) Average values (c) Maximum value

Figure 7. Decision errors from the simulation using one iteration of the algorithm run after network selection is triggered. The results for 100, 200, 300 mobile nodes are based on 1000 simulation runs for each group of nodes.

messages are excluded from the estimation. For the global knowledge reference all n networks in consideration need to exchange the information about m users that get triggered network selection, and the overhead is estimated as follows (Eq. (9)).

$$S_o = n \cdot (n-1) \cdot m \tag{9}$$

To make estimations for Algorithms A, B and AB and the local reference defined respectively by Eq. (10), Eq. (11), Eq. (12), Eq. (13), we use the results of the dynamic scenario that are depicted in Figure 10.

$$S_o = 0.17 \cdot m \cdot (n-1) \tag{10}$$

$$S_o = 0.18 \cdot m \cdot (n-1) \tag{11}$$

$$S_o = 0.32 \cdot m \cdot (n-1) \tag{12}$$

$$S_o = 1.06 \cdot m \cdot (n-1) \tag{13}$$

Clearly, Algorithm A provides a significant reduction of the signaling overhead.

VI. Problem Formulation for Multicast Transmission in Heterogeneous Wireless Networks

In this section, the scenario discussed in Section III is formalized as a centralized system model, as illustrated in Figure 11. The system model for this scenario was previously presented [2]. For the sake of completeness, we revisit the model in this section. In addition, we implement some modifications to its prior definition.

A. System Model

We consider a set of networks $N = 1, 2, \ldots, n$, a set of mobile nodes $M = 1, 2, \ldots, m$ and a set of streaming contents $S = 1, 2, \ldots, s$. The contents are hosted in different BPSs. Each content s_k can be delivered to more than one mobile node m_j. Therefore, using multicast for data dissemination is beneficial. For each node m_j, content s_k and network n_i, the following is defined: available bandwidths of networks

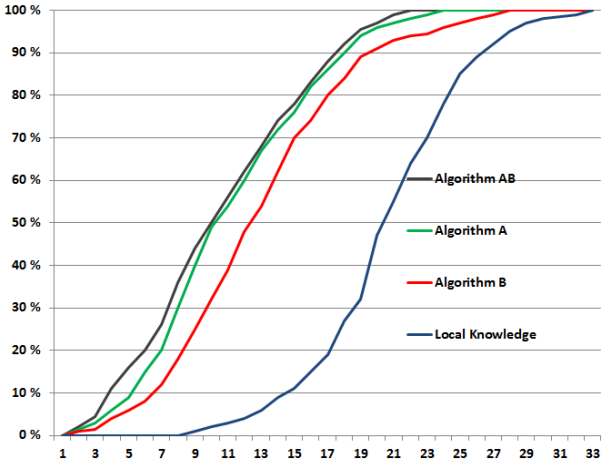


Figure 8. Cumulative distribution function for percentage of decision errors using one iteration of the algorithm run after network selection is triggered. The results are based on 1000 simulation runs for a system consisting of 200 mobile nodes.

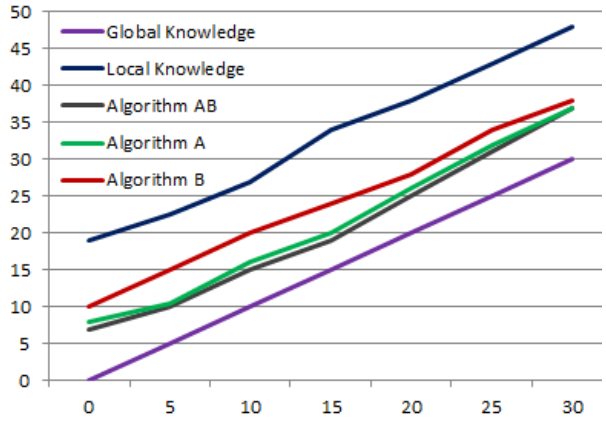


Figure 9. Dropped calls from the simulation using one algorithm run with total available bandwidth less than total required bandwidth.The results are based on 1000 simulation runs for a system consisting of 200 mobile nodes. The x-axis shows the percentage of dropped calls for the optimum (global knowledge reference). The y-axis shows the percentage of dropped calls.

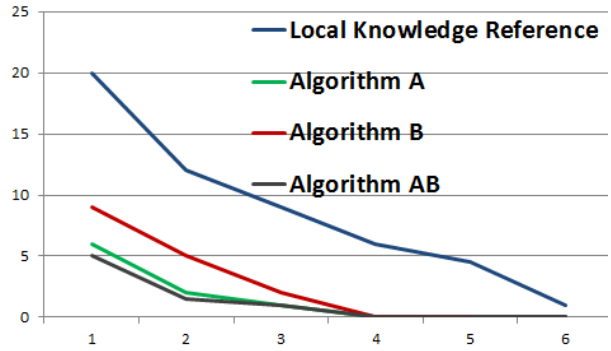


Figure 10. Percent of users received dropped calls, dynamic scenario.The x-axis shows the number of iterations of the algorithm. The y-axis shows the percentage of decision errors. The results are based on 1000 test runs.

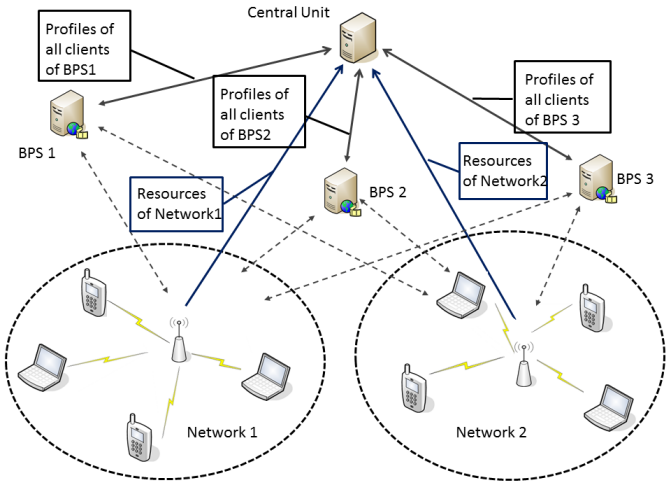


Figure 11. Centralized Approach: Mobile nodes convey their data to their BPSs, the BPSs send the data to the central unit. The central unit collects information about resource availability from the networks, performs the network selection, sends results to the BPSs. The BPSs send the results to their mobile nodes.

are denoted by b_i; streaming bitrate requirements of mobile nodes that request content s_k are denoted by r_k; $rss_{i,j}$ is the received signal strength in network n_i for node m_j, while power consumption and the cost of service in network n_i for node m_j are denoted by $p_{i,j}$ and $c_{i,j}$, respectively.

For each node m_j, we define node preferences that are described by a tuple containing Th_j^p, Th_j^c, and Th_j^{rss}. These denote thresholds for, respectively, power consumption, cost of service and received signal strength. Without loss of generality, we consider these three parameters in our work, however, this list can include other conditions. The thresholds for these parameters are determined by each node according to its own optimization policies and used as an input that constrains optimization solved locally by each mobile node for available networks. The objective function is also defined based on the node's optimization policies. Its definition is beyond the scope of our work.

$$\forall\{i,j\} : \delta(i,j) \cdot p_{i,j} \leq Th_j^p \tag{14}$$

$$\forall\{i,j\} : \delta(i,j) \cdot c_{i,j} \leq Th_j^c \tag{15}$$

$$\forall\{i,j\} : \delta(i,j) \cdot rss_{i,j} \geq Th^r ss_j \tag{16}$$

The output of this optimization is a list of mobile networks that satisfy the user's requirements. It is further referred to as a node's *network profile* captured by a function δ_j. This function is used as input for computing an optimal allocation of mobile nodes to the available networks in the model and is defined as follows.

$$\delta_j(i) = \begin{cases} 1, \text{if } n_i \text{ is selected by } m_j \\ 0, \text{otherwise} \end{cases} \tag{17}$$

We define a binary decision variable $x_{i,k}$ as follows:

$$x(i,k) = \begin{cases} 1, \text{if } n_i \text{ is allocated for } s_k \\ 0, \text{otherwise} \end{cases} \tag{18}$$

To find the best possible allocation of the requested streams to the available networks in terms of minimization of consumed bandwidth, we minimize the following objective function:

$$\min \sum_{n_i \in N} \sum_{s_k \in S} x_{i,k} \cdot r_k \tag{19}$$

The objective function is subject to the set of constraints given below.

For each mobile node m_j, we need to guarantee that it can receive the requested content from at least one network belonging to nodes profile. We need to specify that user preferences defined in their profiles are satisfied.

$$\forall\{j\} : \sum_j \delta_j(i) \cdot x_{i,k} \geq 1 \tag{20}$$

For each network, the availability of its bandwidth is checked.

$$\forall\{i\} : \sum_k x_{i,k} \cdot r_k \leq b_i \tag{21}$$

After the results for $x_{i,k}$ are computed, these are send to the nodes. If there are several networks that receive the requested content, a node can narrow its selection criteria to choose among these alternatives.

The defined problem is a typical location allocation problem that belongs to a class of integer programming problems. To solve this problem, we have taken advantage of the GNU Linear Programming Kit (GLPK) version 4.49 [50]. This is an ANSI C package that is intended for solving large-scale linear programming and mixed integer programming problems. We tested the performance of the package for solving the aforementioned problem that consisted of respectively 500 and 1000 nodes, 5 mobile networks and 10 different streaming contents

(505 and 1005 constraints respectively and 50 variables). For this test, the constraint matrix and the coefficients of the objective function were randomly generated, as in the Monte Carlo simulation. For a 2.83GHz Intel processor, the average CPU time estimates based on 1000 algorithm runs are 710 ms and 230 ms for 1000 nodes and 500 nodes respectively. Though these estimates are computer configuration-specific, they show that the problem can be solved within reasonable time for a relatively large number of nodes. Here, we assume that all necessary information is locally available for the computation. For a component, the CPU time can be precomputed and used as a threshold for deciding whether or not the optimization can be applied to a particular problem scope. On the other hand, collecting this information from different network locations can become a bottleneck for the algorithm operation.

Please note, that a node can, in fact, exploit the multipath streaming scheme and receive data from several networks concurrently. The problem is then reduced to a class of linear programming problems, which is less complex to solve using, for example, the Simplex method. For this problem, the variable $x_{i,k}$ denotes the share of the content s_k delivered to the network n_i.

VII. System Components, their Functions and Feedback Exchange for Multicast Transmission in Heterogeneous Wireless Networks

In this section, we look at architectural aspects of a system that supports the scenario considered in Section III. For this, we consider several entities and decision making components that are responsible for decisions in the system and have access to different information necessary for optimal regrouping of the mobile users. Since the decision-making process requires access to various data that originate from different network components, our system needs to implement a signaling infrastructure for the exchange of such information.

A. BPS Component

The BPS component runs on a backbone server. It either hosts the content or acts as a proxy server that re-sends the content to the user. The component maintains multimedia sessions and controls multicast groups. It receives and processes feedback from its mobile clients and the access networks these clients are connected to. Based on results of processing the data, the component can trigger the network selection. The component can also send data to other components upon their requests.

B. Mobile Network Component

This component is located inside a mobile network. It monitors network's resources including available bandwidth. It also maintains various information about clients and multicast groups of the network. The component implements the following: *1*) processes this information; *2*) sends the information to other components for further processing; and *3*) initiate network selection for multicast groups.

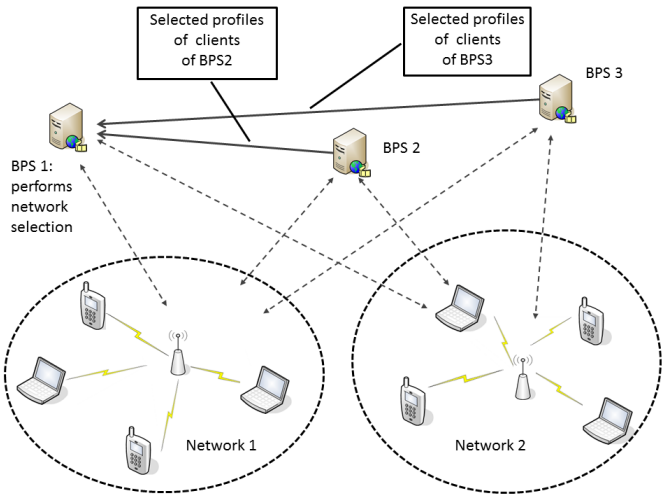


Figure 12. Decentralized Approach: BPSs convey some information to the BPS or other component that is elected to compute the selection. This component also collects information about resource availability from the networks and performs the network selection. The results are sent back to participated BPSs. The BPSs send the results to their mobile nodes.

C. Mobile Client Component

This component runs on mobile device and maintains its wireless channel state information monitoring the availability of mobile networks and received signal strength in these networks. It maintains its preferences towards these networks based on the following: *1*) power consumption in these networks; *2*) security issues; *3*) network cost of service; and *4*) channel state information.

D. Information Exchange

To achieve an optimal solution, all required information need to be exchanged between the decision components and to be communicated to some centralized unit that computes this optimal allocation, as shown in Figure 11. This centralized unit can be a predefined component of the system that other components are aware of or it can be elected on a vote-basis among the components from the mobile networks and the backbone network. The centralized approach demands the exchange of network data across the mobile networks on a fast time scale and low-latency basis. This makes it unrealistic to implement the algorithms in practice for large scale networks. It also requires the implementation of a centralized component that runs this operation and solving a computational problem of high complexity in real time.

The information overhead can be overcome by using distributed designs as illustrated in Figure 12. As an alternative to the centralized approach, we present a distributed solution in which the networks and the BPSes handle the problem completely independently from each other or in a cooperative and coordinated manner with some limited information exchange between the components. We consider the following problems:

1) How much information of any given BPS/network/user data needs to be exchanged?

2) From the architectural design, how to disseminate this information mostly efficiently?

In our work, we focus on the usage of the application specific message (APP) of the RTP/RTCP protocol suite [51] to convey client-related information to the respective mobile networks and BPS components. These protocols are designed for multicast architectures with multicast channels specified for data transmission from the sender to the receivers. In the considered model, only BPS and network components receive feedback from the clients, meaning that the client-to-client feedback exchange is not required. In addition, these feedback reports do not consume much bandwidth and are sent only when the client-related information is changed. Therefore, to deliver feedback reports, unicast transmission is used.

VIII. Decentralized Approach for Multicast Transmission in Heterogeneous Wireless Networks

In this section we consider two decentralized approaches that solve the previously defined network selection problem. Since not all knowledge for the network selection is available on these nodes the algorithms make their decisions based on the currently available knowledge. We consider the solutions (*1*) when all backbone proxies in the system perform the network selection independently from each other, further referred to as a *BPS solution*, and (*2*) when an access network performs the network selection for a set of multicast groups, further referred to as a *mobile network (MN) solution.*

Both approaches rely on the information acquired from mobile nodes regarding their network profiles. To maintain its network profile, the mobile node component periodically monitors the availability of mobile networks. As the node does not need to keep all its interfaces active all the time and the power consumption in idle mode is less than under receiving of data, this operation is not expected to drastically increase the battery use.

To disseminate the information, we make use of the application specific message (APP) of the RTP/RTCP protocol [51]. After a node's network profile changes, for example, a new network becomes available, this information is packed into the APP message. It requires that this protocol is implemented for communication and all needed modifications that allow interpreting of the message are made. This way, the information can also becomes available for the access network. If the RTP/RTCP protocol is not used, any application layer messaging can be implemented to convey feedback from nodes to other components for further processing.

A. BPS Solution

In the BPS solution, all backbone proxies in the system perform the network selection for their clients independently from each other, that is no cooperation or information exchange is performed between different proxies. We consider two versions of the solution: *a*) the network selection is run for all multicast groups of the BPS and *b*) the network selection involves only multicast groups that receive the same content.

At Backbone Proxy Server
read APP messages from RTP/RTCP stream
maintain nodes' network profiles
if network profile is changed
  if threshold $\leq$ number of nodes
    do selection for all nodes
  else
    do selection for one content
foreach node that changes network
  instruct node to change network: send *switch* message

(a) Backbone Proxy Server Component

At Mobile Node m_j
monitor available networks
if new network is available
  compute new *network profile*
  send APP message with *network profile* to BPS
wait for response from BPS
upon reception *switch* message from BPS
  switch to new mobile network

(b) Mobile Node Component

Figure 13. Network Selection Algorithm for BPS

These two versions perform the same operations and differ only in scale of involved nodes.

The network selection algorithm is depicted in Figure 13. It is initiated by the backbone proxy component when network profiles of nodes receiving any of the proxy's multicast streaming sessions change. The changes include: a new node joins the session, one of the nodes leaves the session, or a network profile of any node is updated, e.g., a new network becomes available. To check whether a reconfiguration of multicast groups is needed, the BPS solves the optimization problem defined in Section VI. We use the CPU threshold, also discussed in Section VI, to determine which of the two above versions of the algorithm to apply.

We solve the problem for all multicast groups of the BPS if the threshold allows for it. Otherwise, the problem is solved for one content only. We shall evaluate both possible algorithms separately in Section X. That way, the effect of reducing the problem's input data can be shown better.

B. Mobile Network Solution

In this solution, we consider an access network that initiates and performs the network selection for a set of multicast groups. The network maintains nodes' network profiles based on information extracted from the RTP/RTCP stream. The network selection operation is triggered by the network component when the network's available bandwidth goes below a predefined threshold. For each multicast group, the network defines a set of networks that is a conjunction of nodes' network profiles that comprise the group. The groups with high cardinality of such sets are selected. The network requests the BPSs that host the content received by the selected groups about network profiles of other nodes from other networks receiving the same content. The network solves the optimization problem defined in Section VI for the nodes detected by the above selection operation. The threshold is adjusted accordingly, if the result of the optimization exceeds the

At Mobile Network n_i
read APP messages from RTP/RTCP stream
maintain nodes' network profiles
monitor available bandwidth
if available bandwidth $\geq$ threshold
    compute conjunctions for each multicast group
    order groups by cardinality of conjunction sets
    select candidates for optimization
    request network profiles from involved BPSs
upon replies from servers
    do selection for defined scope of problem
foreach node m_j that changes network
  if $m_j \in n_i$
      instruct node to change network: send *switch* message
  else
      send response to server

(a) Mobile Network Component

At BPS
upon request from mobile network
    select network profiles for requested contents
    send network profiles to mobile network
upon response from mobile network
  foreach node that changes network
    advise node to change network

(b) BPS Component

At Mobile Node m_j
monitor available networks
if new network is available
  compute new *network profile*
  send APP message with *network profile* to BPS
wait for response from BPS
upon reception *switch* message from BPS
  switch to new mobile network
wait for response from mobile network
upon reception *switch* message from mobile network
  switch to new mobile network

(c) Mobile Node Component

Figure 14. Network Selection Algorithm for Mobile Network

current threshold of the bandwidth. In this case, the network component periodically checks the bandwidth to detect if the threshold can be reduced to its previous value.

The selection algorithm depicted in Figure 14 requires the implementation of infrastructure that supports interactivity between the involved components and signaling mechanisms that invoke exchange of information about users' profiles and network conditions and initiate network selection.

IX. LTE-Advanced HetNet Approach

In this section, we consider a solution for a network system discussed in Section II-E that is depicted in Figure 1. Usually in such systems, small cells are connected to a macrocell via the wired backhaul infrastructure, thus, wireless resources are not used for the exchange of control messages and the exchange of the channel information and multicast decisions can be performed within reasonable time period. More importantly, these networks combine a high transmit power and long range base station with several lower power short range stations. Handoff of users moving with high velocity to short range small cells will require a new handoff soon when the user leaves this cell. Therefore, from the point of view of network capacity and avoidance of the ping-pong effect, it is important to consider the mobile terminal velocity for network selection. Here, we assume that a mobile network is capable to predict the residence time of a mobile node inside a cell of the network based on terminal velocity, the local area, movement patterns, and other statistical information. We base our decision making process on research done by other authors [43–45]. While the prediction of the residence time of mobile nodes is ongoing work, we consider this beyond our scope. For the purpose of this paper, we assume that the prediction can be performed with acceptable precision.

X. Simulations for Multicast Transmission in Heterogeneous Wireless Networks

In this section, we evaluate the algorithms described in Section VIII. We present the simulation setup, then the evaluation metrics, and finally discuss the simulation results.

A. Simulation Setup

Because full-scale field experiments for several wireless networks and several hundred users are problematic and expensive to carry out, we used simulations to evaluate the validity of the proposed solutions. The performance and functionality of the system is analyzed through multiple simulation runs. We evaluate two types of network systems: a general heterogeneous wireless network system and an LTE-Advanced HetNet network.

1) Setup for Users and Streaming Content: For the simulations, we consider a scenario with five backbone streaming servers each having five different streaming contents. Video streaming content is used for the evaluation. In terms of required bandwidth, we divide the requested content into five categories: 500 kbps, 800 kbps, 1200 kbps, 1800 kbps, and 2400 kbps. These rates are recommended bit rates for live streaming for the Adobe Media Server [52]. Further, we consider that mobile users are randomly assigned to the servers and their content. The users arrive at one user per time unit, and they stay in the system for 200, 300 or 400 time units. This time period is randomly selected for each user. Except for the initial stage of 200 time units, there are always at least 200 users in the system. The initial stage is excluded from the evaluation.

2) Modeling of Movements: We realise that usage of real world traces evaluates performance only for these particular scenarios. Therefore, we chose to use random generated sintetic data since these data allow more comprehensive performance evaluation by using a large number of variations. Several parameters for modeling of movements are important for our evaluation. For the simulation of movements of mobile nodes, we looked at different studies concerning mobility models for the wireless communications [53, 54]. In the random waypoint model, the location of mobile nodes, their velocity and direction of the movement are chosen randomly and independently of other nodes. We captured the randomness

At Backbone Proxy Server
for each request { $\delta_j(i)$, s_k } received from mobile node m_j
  define set of mobile networks N_k receiving s_k and satisfying $\delta_j(i)$
  if $N_k \neq \emptyset$
    return N_k to node m_j
  else return *NULL*

(a) Backbone Proxy Server Component

At Mobile Node m_j
compute $\delta_j(i)$
send request { $\delta_j(i)$, s_k } to Backbone Proxy Server
wait for response from Backbone Proxy Server
upon *reply* from Backbone Proxy Server
  if reply $\neq$ *NULL*
    select network from *reply*
  else select network from network profile

(b) Mobile Node Component

Figure 15. Algorithm for Computing Lower Bound Reference

of these parameters by random time during which any mobile network in consideration is available to a mobile user.

In the simulation, we also have a number of users who do not move, e.g., people in public places like internet café or train stations. For these users, the subset of the available networks is the same during the whole simulation run. We also distinguish between single users moving alone and groups of users moving together using, e.g., public transport. For users belonging to the same group, the availability of the networks changes likewise while the streaming content can differ. Since we do not have any observations for realistic distribution of these different types of users, we model them as roughly equally distributed, varying from 25% to 40% of users in each group.

3) General Heterogeneous Wireless Network Setup: In this setup, we consider a scenario with four wireless networks. The networks only cover parts of the area in consideration. Therefore, only a selection of them is available for each user at a given time. We have implemented different scenarios for network availability. In all these scenarios, each user has access to at least one network continuously during the whole session of an experiment run. Some users have access to all networks during the whole session. We vary the percentage of these users in different scenarios. For the rest of the users and networks in each scenario, the users can access these networks during some period of the session randomly chosen from the session duration.

4) LTE Heterogeneous Network Scenario: We evaluate an LTE Advanced Heterogeneous Network scenario, the so called macro-pico scenario [38], with several picocells deployed inside a macrocell as illustrated in Figure 1. We consider four picocells deployed inside one macrocell. All users can access the macrocell and some of users have access to one of picocells. The access areas of the picocells do not overlap. The picocells that the users can access during the session are randomly chosen for each user. Accordingly, the periods when picocells are available for the users are randomly chosen from the session duration.

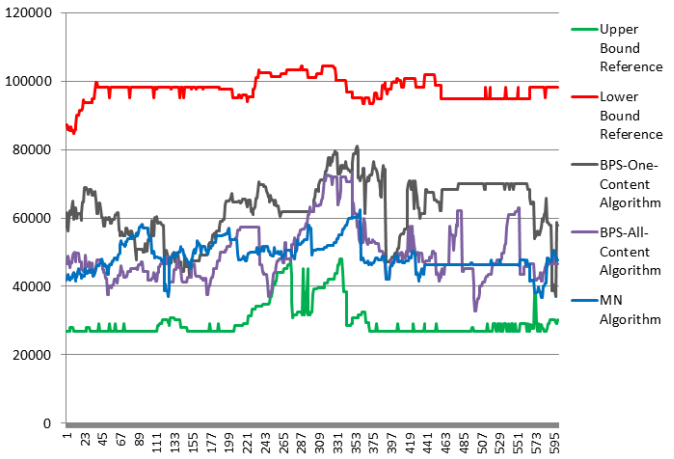


Figure 16. Total Bandwidth Consumption for simulations with background traffic changes applied. The x-axis shows time units. The y-axis shows consumed bandwidth in kbits. The results are an average of 500 simulation runs. The duration of one simulation run is 600 time units.

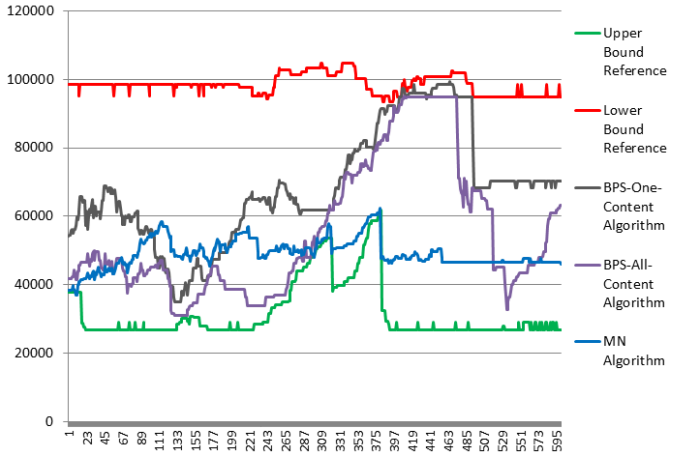


Figure 17. Total Bandwidth Consumption for simulations with applied background traffic changes and insertion of a flashcrowd. The x-axis shows time units. The y-axis shows consumed bandwidth in kbits. The results are an average of 500 simulation runs. The duration of one simulation run is 600 time units.

B. Evaluation Metrics

To evaluate the algorithms, we define upper and lower bounds to their operation. By the *upper bound*, we mean the theoretical best possible allocation of nodes to multicast groups that can be achieved in terms of resource utilization. It is established by applying a centralized solution with fully shared knowledge of the conditions in all evaluated networks and preference profiles of all nodes. It is defined in Section VI-A and is further referred to as an *upper bound reference*.

The lower bound corresponds to an algorithm depicted in Figure 15. Upon a request from a node, which specifies the requested content s_k and the node's network profile $\delta_j(i)$, the server assigns the node to a multicast group, if any that satisfies the node's network profile exists. If no such

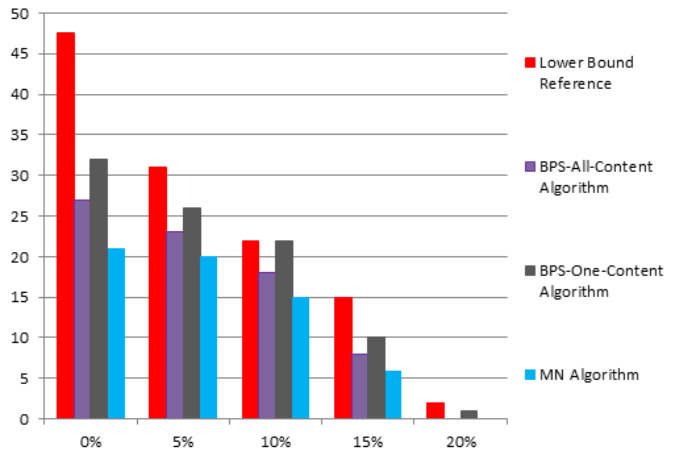


Figure 18. Dropped Connections. The x-axis shows the percentage of bandwidth surplus compared to the Upper Bound Reference. The y-axis shows the number of dropped connections for each algorithm. The number of dropped connections for the Upper Bound Reference is 0 for all algorithm runs. The results are an average of 2000 simulation runs.

group exists, the server opens streaming to a network that meets the user's preferences. This simple algorithm uses a low number of operations and input information to establish multicast groups. We decided to use this algorithm instead of a trivial unicast scenario to provide a fairer comparison of the proposed solutions. In our evaluation, it is referred to as a *lower bound reference*. The lower bound reference is also used as a initialization algorithm for the mobile network solution. Note that the MN algorithm is triggered only when the available resources in a network are dropped below a certain threshold. Thus, some initialization is required and the MN solution evaluated in this section is a hybrid approach.

We evaluate the performance of the algorithms using two performance metrics: total bandwidth consumption and a number of dropped calls, as follows.

Total bandwidth consumption is a direct measure of the bandwidth usage of all multicast groups in the system. We measure the total bandwidth consumption for two different bandwidth variation profiles for the available bandwidth of the access networks. These bandwidth variations model the background traffic for each network.

1) Changes in bandwidth are applied to all access networks, and their values are normally distributed in the range $[-0.1, 0.1]$ of the currently available bandwidth. The results for this test are depicted in Figure 16.
2) In addition to the bandwidth fluctuations formulated above, we simulate a flashcrowd scenario with an number of nodes arriving within short-time intervals. For all access networks, we inserted from 30 to 50 additional nodes once for each session. The simulation results are depicted in Figure 17.

Note that none of the networks have enough capacity to accommodate all multicast groups alone even if the nodes are optimally allocated to their networks. All four networks must be used in order to meet the requirements of all users.

The number of dropped connections is measured for all multicast groups in the system for the following five condition cases: *1*) The total available bandwidth of the system equals to the bandwidth utilized by the upper bound reference. *2*) The total available bandwidth exceeds the amount of bandwidth utilized by the upper bound reference by 5%; *3*) by 10% ; *4*) by 15%; *5*) by 20%, respectively. To compute the setup for this type of tests, we applied a relaxation to the upper bound reference. The optimization problem is relaxed by removing the network bandwidth constraint depicted in Eq. (21). The available bandwidths of the evaluated networks are then calculated from the optimization results. For these tests, bandwidth fluctuations have not been applied. Also, once assigned, the availability of the networks does not change within a test run, meaning that the algorithms have been evaluated statically. The results for this test are depicted in Figure 18.

C. Performance Results

The simulation results are drawn from the average of 500 simulation runs to evaluate the total bandwidth consumption, and of 2000 simulation runs to evaluate the number of dropped connections that are discussed in Section X-B. The performance metrics to evaluate the total bandwidth consumption are collected for 600 time units. We excluded results for which the optimum solution has not been found, i.e., when no optimum existed. For the evaluation of dropped connections, we also excluded the top and bottom 5% of the results of all performed tests.

When evaluated in terms of consumed bandwidth, the tests show that applying any of the proposed solutions can save up to 50% of available bandwidth if compared to the lower bound reference. As expected, the all-content version of the BPS algorithm gives better results than the one-content version though it requires longer processing time and, what is more important, more signaling and reconfigurations for mobile nodes. The trade-off between these two versions can be studied. The MN algorithm behaved very close to the BPS algorithm most of the time.

For the flashcrowd scenario, the MN algorithm was able to handle better the insertion of new nodes than the BPS algorithms. This can be explained by the fact that the MN algorithm is applied across several cutting planes of the total solution space while the BPS algorithm is applied across one cutting plane. In other words, the BPS algorithm relies only on the information from one server while the MN algorithm takes information from several servers as an input. Certainly, the MN algorithm requires exchange of significantly more information across the system. To disseminate this information, we need to implement an appropriate protocol and develop mechanisms that allow the components, namely the networks and the BPSs, to cooperate with each other and exchange information about their users and network conditions, which also involves certain security considerations. Contrary to the MN algorithm, the operation of the BPS algorithm can thoroughly rely on information received from the RTP/RTCP feedback messages.

Evaluations of dropped connections show the same tendency as the evaluation of consumed bandwidth. Both the MN solution and the BPS solution give a good reduction of dropped connections, up to 50%, if compared to the lower bound reference. Compared to the BPS solution, the MN algorithm gives roughly from 10% to 20% less dropped connections.

As expected, the BPS algorithm for all streams performs better than the algorithm for one stream, giving roughly 10% to 20% difference in evaluation. At the same time, the BPS algorithm for one stream still gives good reduction in dropped connections and consumed bandwidth compared to the lower bound reference. Therefore, this version can be applied if solving the all-streams version is not computationally reasonable.

XI. Conclusion

The paper studied the problem of load balancing and forming mobile multicast groups in heterogeneous network environments. An efficient decentralized network selection solution is important for future mobile networks, since it improves utilization of the network resources and QoS of users and reduces signalling overheads. We study how the solution results depend on the information available for the decision making. The problem is considered for a multi-stream multi-server scenario. The candidate networks are selected for multicast groups based on their mobile nodes' preferences and available resources of the networks.

For load balancing scenario, the solution provides a substantial improvement in reduction of decision errors and signalling overhead in comparison to the work specified in Section II. The simulation results of our algorithms show that blocked calls can be reduced with approximately 60-50 % compared to the local knowledge reference. The test results do not differ much for 100, 200 and 300 users, and we expect that these results can be extended to the general case.

All three evaluated algorithms deliver similar results in terms of number of blocked calls. The implementation of Algorithms AB and B requires development of mechanisms for synchronizing information about the network conditions and careful security considerations when information from one network is available to other networks. Operation of Algorithms AB and B requires significantly higher signalling between the networks and the users. We therefore conclude that Algorithm A is to be preferred over Algorithms AB and B.

For multicast scenario, we proposed two solutions that establish multicast groups and assign them to networks based on incomplete information of the whole system. The operation is also performed by different components of the system with limited cooperation between the components.

Compared to the work specified in Section II, our main achievement is decentralization of the network selection for multicast groups, consideration of the impact of several multicast groups and incompleteness of information. An efficient decentralized network selection solution for multicast is important for future mobile networks, since it improves utilization of the network resources and QoS of users and reduces signaling overheads.

We studied how the solution results depend on the information sets available for the decision making. Evaluating dropped connections shows that both algorithms provide a substantial improvement in reduction of dropped connections compared the the lower bound reference. According to our findings, the MN algorithm performs better than the BPS algorithm.

In terms of consumed bandwidth, both solutions deliver similar results for monotonous variations in available bandwidth and arrivals of nodes. For the tests with insertions of extra users, the MN solution performs better than the BPS solution. However, the operation of the MN solution requires complex signaling across several mobile networks and BPSs. In addition, it requires implementation and deployment of mechanisms and communication protocols that provide cooperation between the involved components. The disadvantage of using the BPS algorithm is the necessity of the network reconfiguration of mobile nodes each time the network profile of a node changes. Therefore, a mechanism that is similar to monitoring bandwidth threshold in the MN algorithm can be considered as a next improvement.

As a further step, we intend to investigate how the system can benefit from joint operation of these solutions and limited feedback signaling. We need to implement mechanisms that detect which of two solutions is preferable for certain events. Since, the BPS solution deliver good results for the most of the cases, the operation of the MN solution is going to be triggered only under predefined circumstances. Thus, we avoid unnecessary messaging between the components. We also intend to perform more expanded tests by extending the simulation scenarios by, for example, taking down one of the networks during the simulation.

Finally, we mention that the centralized approach still can be applied for some scenarios and network configurations like small cell networks deployed by the same provider. We intend to investigate better the conditions for applying this approach and consider also implementation of partially centralized solutions.

Acknowledgment

The work described in this paper has been conducted as a part of the ADIMUS (Adaptive Internet Multimedia Streaming) project, which is funded by the NORDUnet-3 programme.

References

[1] S. Boudko, W. Leister, and S. Gjessing, "Multicast group management for users of heterogeneous wireless networks," in CONTENT 2012: The Fourth International Conference on Creative Content Technologies. International Academy, Research and Industry Association (IARIA), 2012, pp. 24–27.

[2] ——, "Optimal network selection for mobile multicast groups," in ICSNC 2012 The Seventh International Conference on Systems and Networks Communications. In-

ternational Academy, Research and Industry Association (IARIA), 2012, pp. 224–227.

[3] ——, "Team decision approach for decentralized network selection of mobile clients," in Proceedings of 2012 5th Joint IFIP Wireless and Mobile Networking Conference. IEEE Computer Society, 2012, pp. 88–94.

[4] S. Boudko and W. Leister, "Network selection for multicast groups in heterogeneous wireless environments," in MoMM '13: Proc. 11th Int'l Conf. on Advances in Mobile Computing and Multimedia. ACM, 2013, pp. 167–176.

[5] I. Gillott et al., "The potential for LTE broadcast/eMBMS," iGR, white paper, January 2013.

[6] 3GPP, "Multimedia Broadcast/Multicast Service (MBMS); Stage 1," 3rd Generation Partnership Project (3GPP), TS 22.146, Jun. 2008, [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/22146.htm, accessed September 6, 2013.

[7] ——, "LTE; evolved universal terrestrial radio access (E-UTRA); long term evolution (LTE) physical layer; general description," ETSI, technical specification 3GPP TS 36.201, 2009, version 8.3.0 Release 8.

[8] Ericsson AB, Qualcomm Technologies, Inc., and Qualcomm Labs, Inc., "LTE broadcast," white paper, February 2013, [Online]. Available: http://www.ericsson.com/res/docs/whitepapers/wp-lte-broadcast.pdf, accessed September 6, 2013.

[9] T. L. Paramvir, T. Liu, P. Bahl, S. Member, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless atm networks," IEEE J. Sel. Areas Commun., vol. 16, 1998, pp. 922–936.

[10] I. F. Akyildiz and W. Wang, "The predictive user mobility profile framework for wireless multimedia networks," IEEE/ACM Trans. Netw, vol. 12, 2004, pp. 1021–1035.

[11] C.-C. Tseng, L.-H. Yen, H.-H. Chang, and K.-C. Hsu, "Topology-aided cross-layer fast handoff designs for IEEE 802.11/mobile IP environments," Communications Magazine, IEEE, vol. 43, no. 12, Dec. 2005, pp. 156–163.

[12] Y.-H. Choi, J. Park, Y.-U. Chung, and H. Lee, "Cross-layer handover optimization using linear regression model," in ICOIN 2008. Int'l Conf. on Information Networking, Jan. 2008, pp. 1–4.

[13] S. Ray, K. Pawlikowski, and H. Sirisena, "Handover in mobile wimax networks: The state of art and research issues," IEEE Communications Surveys Tutorials, vol. 12, no. 3, 2010, pp. 376–399.

[14] Q. Song and A. Jamalipour, "A network selection mechanism for next generation networks," in IEEE Int'l Conf. on Communications, ICC, vol. 2, May 2005, pp. 1418–1422.

[15] A. Hasswa, N. Nasser, and H. Hassanein, "Tramcar: A context-aware cross-layer architecture for next generation heterogeneous wireless networks," in IEEE Int'l Conf. on Communications, ICC, vol. 1, Jun. 2006, pp. 240–245.

[16] F. Zhu and J. McNair, "Optimizations for vertical handoff decision algorithms," in IEEE Wireless Communications and Networking Conference, WCNC2004, vol. 2, Mar. 2004, pp. 867– 872.

[17] P. M. L. Chan, Y. F. Hu, and R. E. Sheriff, "Implementation of fuzzy multiple objective decision making algorithm in a heterogeneous mobile environment," in IEEE Wireless Communications and Networking Conference, WCNC2002, vol. 1, Mar. 2002, pp. 332–336.

[18] L. Xia, L. Jiang, and C. He, "A novel fuzzy logic vertical handoff algorithm with aid of differential prediction and pre-decision method," in IEEE Int'l Conf. on Communications, ICC, Jun. 2007, pp. 5665–5670.

[19] N. Nasser, S. Guizani, and E. Al-Masri, "Middleware vertical handoff manager: A neural network-based solution," in IEEE Int'l Conf. on Communications, ICC, Jun. 2007, pp. 5671–5676.

[20] G. Karetsos, E. Tragos, and G. Tsiropoulos, "A holistic approach to minimizing handover latency in heterogeneous wireless networking environments," Telecommunication Systems, 2011, pp. 1–14.

[21] A. H. Zahran, B. Liang, and A. Saleh, "Signal threshold adaptation for vertical handoff in heterogeneous wireless networks," Mob. Netw. Appl., vol. 11, Aug. 2006, pp. 625–640.

[22] C. W. Lee, L. M. Chen, M. C. Chen, and Y. S. Sun, "A framework of handoffs in wireless overlay networks based on mobile IPv6," IEEE J. Sel. Areas Commun., vol. 23, no. 11, Nov. 2005, pp. 2118–2128.

[23] K. Yang, I. Gondal, B. Qiu, and L. Dooley, "Combined SINR based vertical handoff algorithm for next generation heterogeneous wireless networks," in IEEE Global Telecommunications Conference, GLOBECOM '07, Nov. 2007, pp. 4483–4487.

[24] C. Chi, X. Cai, R. Hao, and F. Liu, "Modeling and analysis of handover algorithms," in IEEE Global Telecommunications Conference, GLOBECOM '07, Nov. 2007, pp. 4473–4477.

[25] N. Nasser, A. Hasswa, and H. Hassanein, "Handoffs in fourth generation heterogeneous networks," IEEE Communications Magazine, vol. 44, no. 10, Oct. 2006, pp. 96–103.

[26] G. Zhang and F. Liu, "An auction approach to group handover with mobility prediction in heterogeneous vehicular networks," in ITS Telecommunications (ITST), 2011 11th International Conference on, aug. 2011, pp. 584–589.

[27] L. Sun, H. Tian, and P. Zhang, "Decision-making models for group vertical handover in vehicular communications," Telecommunication Systems, vol. 50, 2012, pp. 257–266.

[28] O. Ormond and J. Murphy, "Utility-based intelligent network selection," in IEEE Int'l Conf. on Communications, ICC, 2006.

[29] A. Gluhak, K. Chew, K. Moessner, and R. Tafazolli, "Multicast bearer selection in heterogeneous wireless networks," in IEEE Int'l Conf. on Communications, ICC,

vol. 2, May 2005, pp. 1372–1377.

[30] I.-S. Jang, W.-T. Kim, J.-M. Park, and Y.-J. Park, "Mobile multicast mechanism based mih for efficient network resource usage in heterogeneous networks," in Proc. of the 12th Int'l Conf. on Advanced Communication Technology, ser. ICACT'10, 2010, pp. 850–854.

[31] E. Tragos, G. Tsiropoulos, G. Karetsos, and S. Kyriazakos, "Admission control for QoS support in heterogeneous 4G wireless networks," Network, IEEE, vol. 22, no. 3, 2008, pp. 30–37.

[32] M. A. Khan, A. C. Toker, F. Sivrikaya, and S. Albayrak, "Cooperation-based resource allocation and call admission for wireless network operators," Telecommunication Systems, vol. 51, 2012, pp. 29–41.

[33] D.-N. Yang and M.-S. Chen, "Efficient resource allocation for wireless multicast," IEEE Transactions on Mobile Computing, vol. 7, no. 4, Apr. 2008, pp. 387–400.

[34] M. L. Fisher, "The lagrangian relaxation method for solving integer programming problems," Manage. Sci., vol. 50, no. 12 Supplement, Dec. 2004, pp. 1861–1871.

[35] F. Hou, L. Cai, P.-H. Ho, X. Shen, and J. Zhang, "A cooperative multicast scheduling scheme for multimedia services in ieee 802.16 networks," IEEE Transactions on Wireless Communications, vol. 8, no. 3, 2009, pp. 1508–1519.

[36] Multicast Mobility Working Group, "Charter for Working Group," 2010, [Online]. Available: http://datatracker.ietf.org/wg/multimob/charter/, accessed July 30, 2013.

[37] 3GPP, "LTE; evolved universal terrestrial radio access (E-UTRA) and evolved universal terrestrial radio access network (E-UTRAN); overall description; stage 2," ETSI, technical specification 3GPP TS 36.300, 2013, version 11.6.0 Release 11.

[38] ——, "Evolved universal terrestrial radio access (E-UTRA); mobility enhancements in heterogeneous networks," ETSI, technical specification 3GPP TS 36.839, 2013, version 11.1.0 Release 11.

[39] W. Leister, T. Sutinen, S. Boudko, I. Marsh, C. Griwodz, and P. Halvorsen, "An architecture for adaptive multimedia streaming to mobile nodes," in MoMM '08: Proc. 6th Int'l Conf. on Advances in Mobile Computing and Multimedia. ACM, 2008, pp. 313–316.

[40] G. Xylomenos, V. Vogkas, and G. Thanos, "The multimedia broadcast/multicast service," Wireless Communications and Mobile Computing, vol. 8, no. 2, 2008, pp. 255–265.

[41] R. Radner, "Team decision problems," Ann. Math. Statist., vol. 33, no. 3, 1962.

[42] Y.-C. Ho, "Team decision theory and information structures," Proceedings of the IEEE, vol. 68, no. 6, june 1980, pp. 644–654.

[43] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for multidimensional PCS networks," IEEE/ACM Trans. Netw., vol. 11, no. 5, Oct. 2003, pp. 718–732.

[44] I. Akyildiz, J. S. Ho, and Y.-B. Lin, "Movement-based location update and selective paging for PCS networks," IEEE/ACM Trans. Netw, vol. 4, no. 4, 1996.

[45] G. Yavas, D. Katsaros, O. Ulusoy, and Y. Manolopoulos, "A data mining approach for location prediction in mobile environments," Data Knowl. Eng., vol. 54, August 2005, pp. 121–146.

[46] B. Jabbari, Y. Zhou, and F. Hillier, "A decomposable random walk model for mobility in wireless communications," Telecommunication Systems, vol. 16, 2001, pp. 523–537.

[47] M. Canales, J. Gállego, Á. Hernández, and A. Valdovinos, "An adaptive location management scheme for mobile broadband cellular systems," Telecommunication Systems, 2011, pp. 1–17.

[48] A. Ulvan, R. Bestak, and M. Ulvan, "Handover procedure and decision strategy in lte-based femtocell network," Telecommunication Systems, 2011, pp. 1–16.

[49] G. Pongor, "Omnet: Objective modular network testbed," in MASCOTS '93: Proc. Int'l Workshop on Modeling, Analysis, and Simulation on Computer and Telecommunication Systems. Society for Computer Simulation, 1993, pp. 323–326.

[50] A. Makhorin, "GLPK (GNU Linear Programming Kit)," Free Software Foundation, 2010–2012, [Online]. Available: http://www.gnu.org/software/glpk/, accessed September 6, 2013.

[51] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," RFC 3550 (Standard), Jul. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt [Accessed: 10. Dec 2013].

[52] A. Kapoor, "Recommended bit rates for live streaming," January 12, 2009, [Online]. Available: http://www.adobe.com/devnet/adobe-media-server/articles/dynstream_on_demand.html, accessed July 25, 2013.

[53] C. Bettstetter, H. Hartenstein, and X. Pérez-Costa, "Stochastic properties of the random waypoint mobility model," Wirel. Netw., vol. 10, no. 5, Sep. 2004, pp. 555–567.

[54] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile ad hoc Networking: Research, Trends and Applications, vol. 2, 2002, pp. 483–502.

Formal Modeling of Temporal Interaction Aspects in Multi-Agent Systems

Djamila Boukredera
Laboratoire des mathématiques appliquées
Université Abderrahmane Mira
Béjaia, Algérie
boukredera@hotmail.com

Ramdane Maamri
LIRE Laboratory
Université Mentouri
Constantine, Algérie
rmaamri@yahoo.fr

***Abstract*—Multi-agent interaction protocols play a crucial role in multi-agent systems (MAS) development. They are used to manage and to control interactions among several autonomous agents in a MAS. Their formal specification, as well as their verification, constitute an essential task for the design of MAS applications. Several approaches have been proposed to formally represent agent interaction protocols, but there still lacks a formalism for representing temporal interaction constraints. This time dimension is an essential parameter in the protocol modeling seeing that most real world applications they support are time-sensitive. This paper proposes to use Timed Colored Petri Nets (TCPN) to model correctly and formally this temporal issue often defined as interaction duration and message deadlines. We then take the well-known Contract Net protocol (CNP) as an example to show that interaction protocols with time constraints can be modeled naturally and efficiently with this formalism. Finally, thanks to simulation techniques and state space analysis we will prove that the most important keys namely model correctness, deadline respect, absence of deadlocks and livelocks, absence of dead code, agent terminal states consistency, concurrency and validity are met.**

Keywords-Interaction protocols, Contract net protocol, Multi-agent systems, Timed Colored Petri Nets.

I. INTRODUCTION

Agent Interaction protocols (AIPs) represent an essential component of the dynamical model of a MAS. It is now recognized that interaction is the most important characteristic of complex systems. Based on many interacting agent components, such systems are generally time-sensitive and are known to be more complex to specify, to verify and to validate. However, the main step in designing an AIP is certainly the formal specification phase, which is crucial since it conditions the protocol design success. This paper is an extension version of the conference paper [1] and aims at providing a greater insight into the formal approach proposed to model the temporal aspects of any agent interaction protocol. Several formal models were proposed in the literature [2]–[8], but few works tackled the modeling of temporal interaction aspects, that are specified by FIPA. However, in current real life applications, time is of great importance and must be taken into account in all the design steps.

This paper addresses this issue and proposes to extend the AIPs with time constraints. We propose to use Timed Colored Petri Nets (TCPN) to formally model the two temporal constraints:

- Deadlines: it is a time constraint for message exchange. They denote the time limit by which a message must be sent. Once the deadline expires, the manager starts the evaluation of the received proposals. All proposals, which arrive after the due time will be considered to be invalid and consequently ignored.
- Duration: it is the interaction activity time period. It represents the time elapsed between the sending of a request message and the reception of the response. Duration includes two periods: transmission time and response time (task duration).

We adopt TCPN models because, besides their simplicity, they are particularly suitable in the modeling, simulating and analyzing of timed concurrent systems and, moreover, they use appropriate and powerful tools to generate interactive simulations of the modeled systems and apply a wide range of formal analysis alternatives. Our work contributes to the formal specification as well as the verification of the temporal interaction aspects in MAS. We then demonstrate the efficiency of our approach on the well-known CNP example, and prove that the key properties are satisfied. This contribution can be enumerated as follows: firstly, we present and we implement the proposed model using CPN Tools. We analyze it by means of the simulation and the state space techniques for various values of the protocol parameters namely the deadline and the number of participants. Secondly, we prove that the above mentioned key properties of the protocol are satisfied.

The rest of this paper is structured as follows: Section II describes the temporal interaction constraints. Section III briefly introduces the modeling methodology and the support tools. Section IV presents the CNP. In Section V, we detail the structure and the operation of the extended CNP. Simulation and state space analysis of our model are given in Section VI. Lastly, Section VII concludes the paper and gives some perspectives.

II. Modeling temporal aspects of interaction

Temporal constraints are time related relationships that must be reflected in a MAS modeling. Such constraints can be within the specification of either the internal behavior (vertical constraints) or the external behavior (horizontal constraints) of interacting agents. Most real-life applications are time-sensitive and may require that the timing constraints must be satisfied for correct operation and acceptable outputs. This is why it is important to take into account this temporal dimension in a MAS design.

In this paper, we will consider the two temporal interaction aspects specified by FIPA [9]: duration constraint and deadline constraint. The first one is the interaction activity time period, which includes two periods: transmission time and response time. Figure 1 illustrates the AUML [10] representation of the duration constraint. In our model, we have assumed that the transmission time t1 is infinitesimal and can consequently be ignored. On the other hand, the response time represents an activity duration and hence random functions are proposed to estimate it.

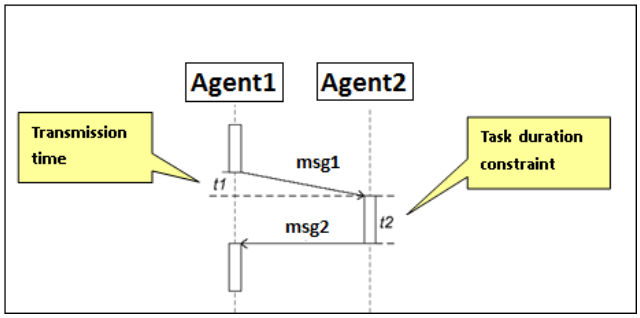


Figure 1: AUML representation of the interaction duration constraint

The second temporal aspect, deadlines, is a time limit for the message exchange. It may refer to a particular point in time by which a task must be accomplished or a time limit by which a message must be submitted. In most protocols, the moderator needs a deadline to decide whether a participant fails to reply or to meet a commitment and then to end an unachieved interaction. In this work, we model a deadline by means of timeout mechanism. In doing so, the moderator sets a wait time constraint (timeout) to receive replies from participants who must respond within this time limit, otherwise the response will be ignored. The response time value is declared as a random function depending on the specified deadline, which represent a key parameter of the timed model. Figure 2 illustrates the AUML modeling of the timeout mechanism. This time constraint indicates an alternative path when the deadline is reached. The alternative is therefore time-sensitive and this is graphically symbolized by the hourglass in the corner of the rectangle.

Figure 2 shows an agent *Seller* sending a proposal (*offer product*) to one or several receivers (*Buyers*) who have to answer by an offer before the expiration of deadline (equal to 100 units). Beyond this time limit, any answer from *Buyers* will be ignored and the *Seller* agent announces the identity of the winner buyer (*product sold (who)*). In this example, the *Seller* agent processes each bid received in the due time then determines and announces to the buyers the new top bid, if any. This process iterates until the deadline is reached.

Notice that AUML is one among the most used formalisms to represent agents' interactions [11]–[14]. However, AUML diagrams only offer a semi-formal specification of these interactions and their time constraints. This weakness can lead to several incoherencies in the description of MAS's behavior. That is why we prefer to adopt a more formal approach to specify agents' interaction, which obviously offers several advantages. Especially, it allows us to create more precise and rigorous specifications that can directly support verification and validation processes, and for which computer based support is available.

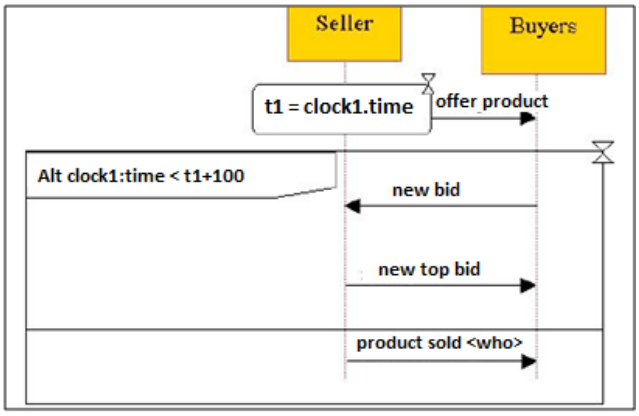


Figure 2: Representing the deadline by a timeout in AUML

III. Modeling methodologies and tools

When designing a software system we often have to deal with two central issues: (1) correctness and (2) performance. Verifying the correctness of the system means proving that it performs correctly all its specified functions and meets all the required key properties. The performance is a quantitative measure of how well a system works, it determines the usefulness of the system. The performance is often characterized by performance measures like: response times, waiting times, maximum capacity, etc.

To evaluate the correctness and performance of a complex system, we need powerful analysis methods and tools. Several formal specification techniques based on different theories exist in the literature, each of them has a preferred domain. In particular, in the field of interaction protocol systems, Petri nets have already proven to be extremely useful for description and analysis of such systems. They

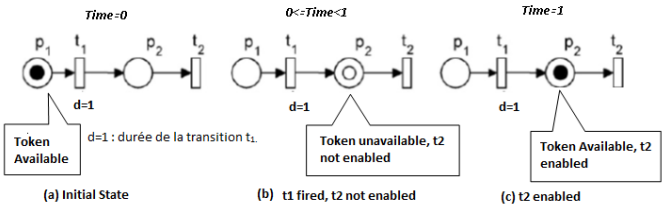


Figure 3: Timed Petri Nets with Holding Durations

follow an elaborated mathematical syntax and provide a clear, intuitive and demonstrative graphical representation of the model thus facilitating its simulation and analysis, which is a basic strength compared to other verification methods.

Since this work aims to assure the functional correctness of the proposed extended CNP, we adopt to use an approach based on Petri nets as suitable methodology and CPN Tools [15] as an adequate supporting tool suite, not only because of the maturity of both its theory and the related tool set CPN Tools [15] that this work rely on, but also because it allows to express and assess sophisticated temporal properties. In doing so, we assume a global clock.

A. Timed CPN

Petri nets are powerful tools for modeling, simulating, analyzing, supervising and debugging many complex distributed concurrent systems. TCPN, however are particularly suitable for real time systems because they allow the developer to produce a precise specification of the temporal behavior of such systems. The concept of time was not explicitly provided in the original definition of Petri nets. As described in [16], we distinguish three basic ways of representing time in CPN: Firing Durations (*FD*), Holding Duration (*HD*) and Enabling Duration (*ED*).

The principle of the *FD* is when a transition with a time delay becomes enabled, it removes immediately the tokens from the input places but does not create tokens in the output places until the firing duration has elapsed. However, if Holding Durations are included then the net semantics are changed. The principle of *HD* is based on two notions: the availability and the unavailability of the tokens. Available tokens can enable a transition whereas unavailable ones can not. In this case, when a transition, which is assigned a duration fires, removing and creating tokens are done instantaneously. However, the created tokens are unavailable and consequently can not enable any new transition until they have been in their output places for the time specified by the duration of the transition, which creates them. Figure 3 graphically illustrates the principle of HD.

In fact, *FD* and *HD* represent the same way of representing time. The only difference is that in *HD* the tokens are held by places whereas in *FD*, they are held by the transitions.

Besides, the *ED* leads to a different temporal behavior of the system. With *ED*, the firing of the transitions is done immediately; that is, removing and creating tokens are done instantaneously and the temporal duration is modeled by forcing the concerned transitions to be enabled for a specified period of time before they can fire. The main difference between *ED* and *HD* appears when there is conflicts in the petri nets, for more details the reader can refer to [16]. Choosing one of these three techniques depends strongly on the system to be modeled and its specifications. We should note, however, that it is natural to use *HD* technique in modeling most processes as transitions represent operation events which, once start, do not stop until they end. It is exactly the case of the system we are modeling. When a transition, which is assigned an *HD* duration, fires, removing and creating tokens are done instantaneously. However, the created tokens are not available to enable new transitions until they have been in their output place for the time specified by the transition, which created them. For more details concerning these three techniques of time modeling, the reader can refer to [16]. CPN versions, which use HD technique define implicitly the notion of tokens's unavailability by attaching to these tokens a timing attribute called a timestamp.

B. Formal definition of TCPN with Holding Durations

To represent tokens with timestamps we adopt the notation given by [17], [18]. Each token carries a timestamp preceded by the @ symbol. For instance, 2 tokens with timestamp equal to 10 are noted 2@10. The timestamp specifies the time at which the token is ready to be removed by an occurring transition. Timestamps are values belonging to a Time Set TS, which is equal to the set of non negative integers N+. The timed markings are represented as collection of timestamps and are multi-sets on TS: TS_{MS}. The formal definition of TCPN using holding durations is as follows: TCPN = (Σ, f, M_0) where Σ is a colored PN as described in [17]:

- Σ= (S, P, T, A, N, C, G, E) where:
  - S is a finite set of non-empty types, called color sets.
  - P is a finite set of places.
  - T is a finite set of transitions.
  - A is a finite set of arcs such that: P $\cap$ T = P $\cap$ A = T $\cap$ A = $\emptyset$.
  - N is a node function. It is defined from A into P $\times$ T $\cup$ T $\times$ P.
  - C is a colour function. It is defined from P into S.
  - G is a guard function. It is defined from T into expressions such that: $\forall$ t$\in$ T: [Type(G(t)) = Bool $\wedge$ Type(Var(G(t))) $\subseteq$ S].
  - E is an arc expression function. It is defined from A into expressions. The arc expression associates

with every arc an expression, which will be used to verify or create new token-values. Every arc expression should evaluate to a set of tokens (a multi-set over the different types allowed by the place). E contains input expressions as well as outgoing actions.

- **f**: T →TS represents the transition function, which assigns to each transition t ∈ T a non negative determinist duration
- **M**: P → TS_{MS} is the timed marking, M_0 represents the initial marking of TCPN.

To determine whether tokens are available or unavailable, we define functions over the marking set M. So, For a marking M and the given model time (global clock), we have:
m: $P \times M \times TS \rightarrow$ N, which defines the number of available tokens and n: $P \times M \times TS \rightarrow$ N, which defines the number of unavailable tokens for each place of the TCPN model at a given instant k, where k and the model time belong to TS. There are several computer tools, which perform automatic validation and verification of Petri net models. Nevertheless, only CPN Tools permits, besides time representation, the modeling of high level petri nets particularly colored and hierarchical ones.

C. CPN Tools

CPN Tools [15] developed at the University of Aarhus is a strength tool for constructing, editing, simulating and analyzing CPN models. Using CPN Tools, it is possible to perform investigation of modeled system design and behavior using simulation, to verify properties by means of state space methods and model checking, and to conduct simulation-based performance analysis. CPN Tools proposes very powerful class of Petri nets for models' description namely hierarchical timed colored Petri nets, which we have chosen to use for our modeling. The language description is a combination of Petri net graph and programming language CPN ML (Markup Language). Notice that the functionality of the tool can be extended with user-defined Standard ML functions.

In the following, we will consider the CNP as an example to illustrate our proposition.

IV. The Contract Net Protocol

CNP, originally proposed by Smith [19], is one of the most popular interaction protocols used in diverse negotiation contexts. Developed to resolve decentralized task allocation, the CNP represents a distributed negotiation model based on the notion of call for bids. In this protocol, agents can dynamically take two roles: manager or contractor (initiator or participant according to FIPA terminology [9]).

In CNP as illustrated by the AUML diagram of Figure 4, a manager and participants interact with one another to find a solution for a problem through a four-stage negotiation process. The manager initiates the negotiation process by

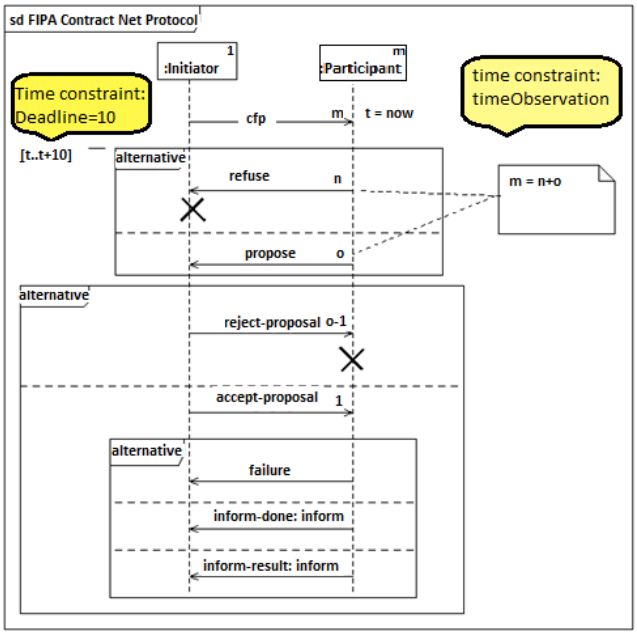


Figure 4: AUML diagram of the contract net protocol

issuing a Call For Proposals (CFP) announcing the task specification to a number of potential participants. The CFP includes a deadline by which the participants must respond with bids. Participants evaluate the CFP and decide whether to answer with a refusal message or with a proposal to perform the task. Once the deadline expires, the manager evaluates all the received proposals (in due time) and, in turn, awards the contract to the most appropriate participant, which becomes a contractor. The manager ignores any proposal that arrives beyond the deadline. The contractor performs the task and sends to the manager an informing message, which can be an error one in the case of a failure. Consequently, the negotiation process includes several scenarios depending on whether the bid process ends with or without a contract, and as the execution of the task ends with or without a success. Therefore, the manager and the participants can reach various states during this process. We suggest to represent the internal behavioral of both types of agents by means of AUML2 statesharts diagrams [10]. These diagrams define the different states that will be later used in the TCPN model of the protocol. Figure 5 (a) and Figure 5 (b) illustrate respectively the internal behavior of the manager and the participant agents. Table I summarizes the various states and their semantics.

V. TCPN model of the Contract Net Protocol

When modeling a protocol, there are several design requirements and key characteristics that this protocol should satisfy. Authors in [6] have summarized these issues in 5 factors: state set, role set, rule set, action set and message set. By analogy with our case study, Table I describes

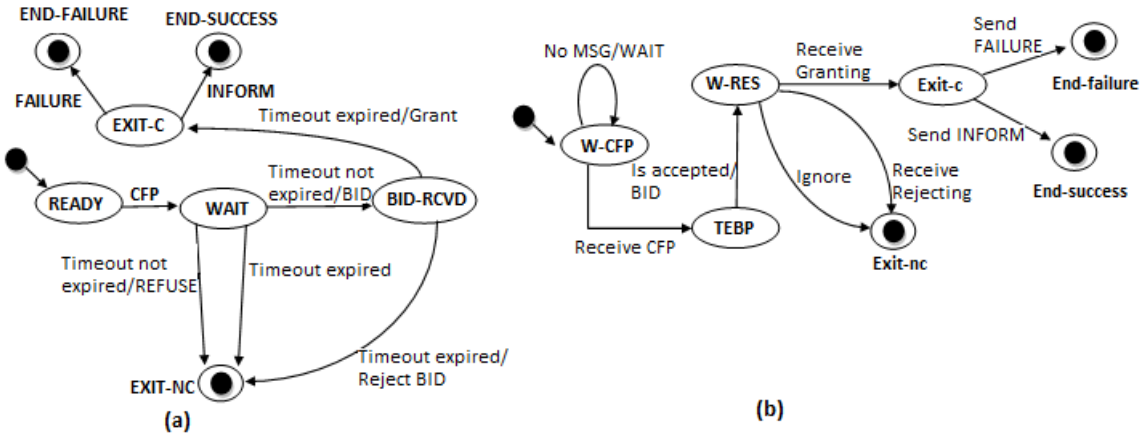


Figure 5: Internal behavior of the manager and the participant agents

Table I: Representation of states

Manager (Initiator)	Participant
READY (READY to send a CFP)	W-CFP (Waiting for CFP)
WAIT (Waiting for bids or for time-out)	TEBP (Task evaluation and bid preparation)
BID-RCVD (Bid received)	W-RES (Waiting for result)
EXIT-NC (EXIT with no contract)	Exit-nc (exit with no contract)
EXIT-C (EXIT with contract)	Exit-c (exit with contract)
END-SUCCES (END of negotiation with SUCCESS)	End-success (end of task execution with success)
END-FAILURE (END of negotiation with FAILURE)	End-failure (end of task execution with failure)

Table II: Representation of messages in the TCPN model

Messages issued by the manager	Messages issued by the participant
CFP (Call For Proposals)	BID (BID)
GB (Grant Bid)	REFUSE (REFUSE CFP)
RB (Reject Bid)	FAILURE (task Execution FAILURE)
CB (Cancel Bid)	INF-DONE (INForm-Done)
	INF-RES (INForm-RESults)

the various states that negotiation process should reach and Table II defines messages exchanged between the manager and the participants. This section highlights our contribution and presents how Contract Net Protocol extended with the temporal aspects described in Section II can be modeled as TCPN using CPN Tools. When creating the model, we have assumed some assumptions such as the reliability of the communication channel, and that participants have to reply to the CFP. Moreover, when modeling the interaction following the contracting phase, we should not take into consideration task duration, given that this work focuses on temporal interaction aspects. The manager starts evaluating bids after deadline expiration and lastly, the details of messages exchanged are excluded for a sake of abstraction.

A. Declarations

Being inspired by [2], our TCPN model is readable and has a compact structure. For each type of agents, we use a single place, which would store all its possible states.

```
▼Declarations
  ▶Standard declarations
  ▼(*----PARTICIPANTS--*)
    ▼val MaxParts=1;
    ▼colset PART=index B with 1..MaxParts;
    ▼var parts:PART;
  ▼(*----States----*)
    ▼colset STinit=with READY|WAIT|BID_RCVD|EXIT_NC|EXIT_c|END_SUCCESS|END_FAILURE;
    ▼colset STpart=with W_CFP|TEBP|W_RES|exit_nc|exit_c|end_success|end_failure;
    ▼colset PART_STinit=product PART*STinit;
    ▼colset PART_STpart=product PART*STpart;
  ▼(*-----MESSAGES---*)
    ▼colset MESinit=with CFP|GB|RB|CB;
    ▼colset MESpart=with BID|REFUSE|FAILURE|INF_DONE|INF_RES;
    ▼colset PART_MESinit=product PART*MESinit ;
    ▼colset PART_MESpart=product PART*MESpart timed;
    ▼colset InfRes = subset MESpart with [INF_DONE,INF_RES];
    ▼colset ResPart=subset MESpart with[BID,REFUSE];
    ▼var Res:ResPart;
    ▼var resp:MESpart;
    ▶var Inform
  ▼(*-----TIMEOUT--*)
    ▼colset IN=unit with i timed;
    ▼val deadline=1;
    ▼colset INTrange = int with 0..(2*deadline-1);
    ▼colset OUT=unit with out;
    ▼colset LATE=unit with late;
    ▼var random: INTrange;
  ▼(----GRANT----*)
    ▼colset GR1=with gr1;
```

Figure 6: Declarations for the TCPN model of the CNP

Similarly, we distinguish two places, which represent a reliable channel for both directions of the communication. Figure 6, taken directly from CPN Tools, shows all the declarations used in the model.

B. Model structure

Figure 7 shows the TCPN diagram of CNP. The manager with the timeout mechanism is modeled on the left, the participants on the right. They communicate via a reliable not ordered channel represented by the two places INIT2PART and PART2INIT. The place INIT2PART only contains messages issued by the manager to the participants. Respectively, PART2INIT only contains messages of the participants to the manager. In this model, the timed messages carry timestamps indicating when they should be available. Initially, the manager is in the state READY with respect to all the participants. Whereas, all the participants

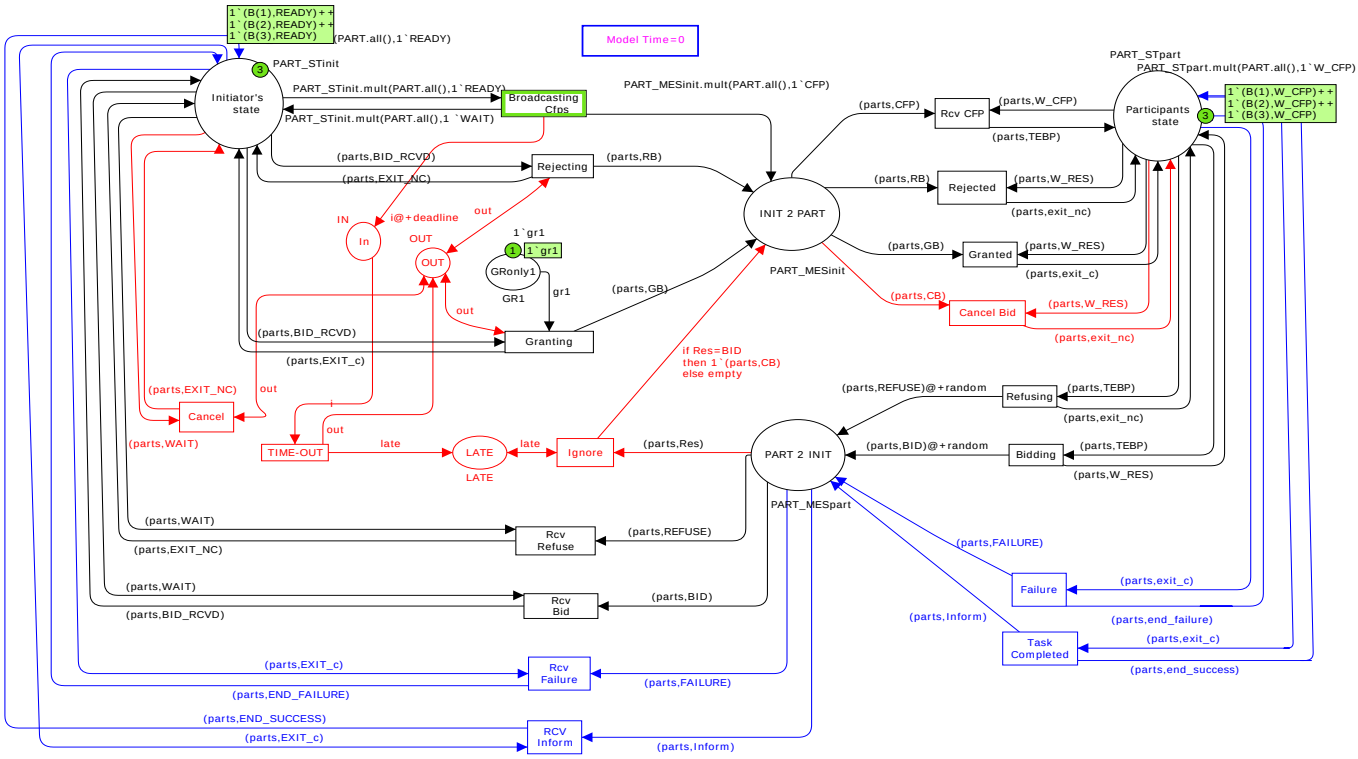


Figure 7: TCPN diagram of the contract net protocol

are in the state W_CFP. The place GRonly1 contains one token GR1 and all the other places are initially empty.

C. Operation of the model

Initially, the time model is equal to 0. The manager (in the state READY with respect to all the participants) is ready to send a CFP to each of the participants (which are in the state W_CFP). The manager initiates the negotiation process by issuing the CFP to all the participants. The transition *Broadcasting Cfps* is then fired and the manager changes state to WAIT with respect to all these participants. At the same moment, the timer is armed with the deadline via creating a timed token in the place In. This token carries a timestamp equal to deadline and, therefore, does not enable the transition *TIME-OUT* because it is unavailable for this duration. The transition *Rcv CFP* is consequently enabled and can be concurrently fired by all the participants. Once a participant removes its CFP from the place INIT2PART it changes state to TEBP and starts evaluating the given task, based on its capabilities and available resource. Then, it decides to make an offer or to refuse the request. In the first case, the participant have to prepare the bid that satisfies the criteria specified in the CFP. As mentioned above, and since they does not affect the operation of the model, all message details are omitted for an abstraction concern. At this point, both transitions *Refusing* and *Bidding* are enabled and the choice of the transition to fire is non deterministic. In the case where the transition *Bidding* fires then the participant changes state to W-RES (waiting for decision about its submitted bid). On the other hand, if the transition *Refusing* has rather occurred then the participant changes state to exit-nc (end of negotiation, with this participant, without a contract). The occurrence of either transition creates a timed token in the place PART2INIT, which represents the reply message. The timestamp of this token is randomly calculated by the predefined function *random* based on the interval [0..2*deadline-1]. In doing so, we assume that the response time of the participant cannot exceed 2 times the given deadline. It should be noted that any message gets through the place PART2INIT in due time updates the state of the manager. In this case, once the time allocated to the timed message expires the transition *Rcv Bid* or *Rcv Refuse* is enabled according to the message BID or REFUSE respectively. It is then that the manager changes state to BID-RCVD or EXIT-NC (with respect to this particular participant) depending on the occurrence of the *Rcv Bid* or *Rcv Refuse* respectively. This process of updating the manager state with respect to any participant reply arrived in the due time continuous until the deadline expires, i.e., the token in the place In becomes available and the transition *TIME-OUT* is, therefore, enabled and fired creating two tokens: one in the place OUT, which could enable both transitions *Rejecting* and *Granting*, and an other in the place LATE, which could enable the transition *Ignore* if any reply

gets through the channel after the deadline. At this point, we distinguish 3 scenarios:

- ***Scenario 1***: *All the participants reply within the due time*. In this case, the transitions *Ignore* and *Cancel* would never be enabled. The manager starts evaluating the bids received (if any) and according to its negotiation strategy decides to accept or to reject a given offer. Note that the manager could grant any of the bids or reject all the bids. If any bid was received then both transitions *Rejecting* and *Granting* are concurrently enabled. In the case where the manager opts to accept a given bid, then the transition *Granting* is fired and a message GB (Grant Bid) is sent to the concerned participant. The token *gr1* is removed from the place GRonly1, signifying that only one bid could be granted. All the other received bids would be, therefore, rejected and a RB (Reject Bid) message must be sent to the corresponding participants via the transition *Rejecting*. In this situation, the manager would be in the state EXIT-C (end of negotiation with contract) with respect to the participant of the granted bid and in the state EXIT-NC with respect to the rest of the participants.
  Another possibility is that the manager could reject all the received bids leading to an end of negotiation without contract. In this case, the transition *Granting* would not occur and the manager would be in the state EXIT-NC with respect to all the participants. At this point, the transition *Rejected* or *Granted* is enabled depending on the message RB or GB respectively. The fire of the transition *Rejected* causes the participant to change state to exit-nc, while the occurrence of the transition *Granted* causes the participant to change state to exit-c. At this latter case, the negotiation ends with a contract and we propose to model the following bilateral interaction between the manager and the winning participant. That is, once the participant performs the task, it would complete it either with a success or a failure. We model this process of task completion non-deterministically. Thus, both the transitions *Failure* and *Task Completed* would be enabled and concurrently fired. On occurrence of the transition *Failure*, the participant change state to end-failure and sends a FAILURE message to the manager. However, if the transition *Task Completed* occurs then the participant change state to end-success and sends an inform message (which could be INF-DONE or INF-RES) to the manager. Once the message reaches the manager, the transition *Rcv Failure* or *Rcv Inform* would be enabled depending on the message FAILURE or Inform respectively. Firing the transition *Rcv Failure* causes the manager to change state to END-FAILURE (end of the negotiation with a failure), while the occurrence of the transition *Rcv Inform* causes the manager to change state to END-SUCCESS (end of the negotiation with success). It should be noted that the task duration has not been modeled, this is because this work focuses on representing temporal interaction aspects and not the real time task management. This would be the subject of a future work.
- ***Scenario 2***: *Some replies get through the channel after the deadline*. In this case and once the transition *TIME-OUT* occurs, two concurrent processes could be conducted by the manager: evaluation of the bids received (if any) and cancelation of any CFP that have not yet received a response. The first process operates in a similar way as mentioned in scenario1 where the negotiation could end either with none contract or with a contract awarded to one participant, which would complete the execution of the task with a success or a failure. In the second process, however, the transition *Cancel* is chosen and fired, implying that the manager would not wait any more the late replies and, consequently, it changes state to exit-nc with respect to those late participants. In the other hand, the occurrence of the transition *TIME-OUT* puts a token in the place LATE, which would enable the transition *Ignore* (ignore all late replies) every time a late message in the place PART2INIT becomes available. In the case the late message is a Bid, then the transition, whose its guard evaluates to true, fires and sends a CB (Cancel Bid) message to that participant. This is causes the enabling of the transition *Cancel Bid*, which once occurred, updates the state of the corresponding participant to exit-nc. In doing so, the bidders do not risk to wait indefinitely for a decision about their submitted bids. Moreover, we assure that at the end of the negotiation, the manager and the participants would be in consistent terminal states. Note that if the late message is REFUSE then the corresponding participant is already in the state exit-nc and the transition *Ignore* is, thus, a sink transition.
- ***Scenario 3:*** *All the replies get through the channel after the deadline*. In this particular case and once the transition *TIME-OUT* occurs, only the transition cancel is enabled. It operates in the same way as mentioned above and causes the manager to change state to EXIT-NC with respect to all the participants. This is the case where the negotiation process ends without a contract because of a deadline overrun (by all the participants). The late messages would be consumed by the the transition *Ignore* as soon as they become available, allowing, thus, the net cleaning.

VI. VERIFICATION OF THE MODEL

Verification is a method to exhaustively examine a design and check to make sure certain predefined key properties are met. There are several software tools to automate this task,

however, CPN Tools [15] is currently the most used tool for high level Petri nets particularly for the timed colored ones (TCPN). This tool helps us to assess the correctness of the model.

A. Simulation

Using CPN simulator, we have conducted several automatic and interactive simulations, which help us to identify and resolve several omissions and errors in the design. In simulation runs carried out the protocol terminated correctly and the agents were in the desired and coherent states. Interactive simulation also shows that the characteristics such as concurrency and validity are satisfied. This makes it likely that the protocol works correctly but it cannot guarantee that simulation covers all possible executions. That is why simulation cannot be used to verify other functional and performance properties such as the absence of deadlocks and others. However, state space analysis techniques allow us to verify if the system satisfies these behavioral properties.

B. State space analysis

With regard to untimed CPN models, calculating timed state space is a non trivial task and can be quite difficult and time consuming. This is because the reachability graph is too large and can be infinite even if the state space of the corresponding untimed CPN model is finite. This is due to the fact that several timed markings including global clock and timestamps can be different even if the corresponding untimed markings are identical. That is why we have to use some CPN ML (CPN Meta Language) queries to verify some properties.

Model Correctness. In this section, we verify the absence of deadlocks and the consistency in beliefs between the manager and the participants. Table III presents the state space analysis results. It shows the properties of the state space obtained by varying the parameter *MaxParts* (Maximum number of Participants) from 1 to 4 and the parameter deadline from 1 to 5. The analyzing of the property DeadMarking allows us to verify the model correctness. Each dead marking corresponds to a terminal state of the negotiation protocol. All dead markings are obtained after the deadline expiration, i.e., from t=*d* to t=2**d*-1 (proposed estimation for the participants response time), for each discrete value of t belonging to this interval. For any value of *MaxParts*, one of the dead markings corresponds to an end of negotiation without a contract. In this marking, all the participants are in the state exit_nc and the manager in the state EXIT_NC with respect to all the participants. This is illustrated by the marking 14 in Figure 8. The description of this node shows that the place GRonly1 has still the token GR1 implying that none bid had been granted. The place In is empty, signifying that the deadline has expired and the timeout has fired. This particular dead marking is acceptable because the manager may reject all the bids or may not receive any bid in the due time. This worst case can be reached by 7 paths describing the pessimistic scenarios that can occur before and after the deadline. Figure 9 shows some paths examples, which lead to this particular case where the negotiation ends with no contract awarded. For example, the path (1,2,3,7,13,15,21,14) of Figure 9a corresponds to the scenario where the participant has issued an offer at t = 0, but it was rejected by the manager; the path (1,2,3,5,10,14) of Figure 9b corresponds to the situation where the participant has refused the CFP. We note that if a participant's response arrives at t = *d* then the choice between firing the timeout or receiving the response is non-deterministic: the transitions *TIME-OUT* and *Rcv_Bid* (or *Rcv_Refuse*) are concurrent, which leads to two different paths in the reachability graph. This is the case, for example, of the two paths (1,2,3,6,12,15, ...) and (1,2,3,6,11,18,22,14) in Figure 9c where the node 6, which corresponds to the reception of an offer at t=*d* is followed either by the node 12 (offer reception by the manager), or by the node 11 (timeout firing and hence cancelation of the offer and the end of the negotiation without contract). This case confirms that the concurrency property is satisfied in the model. Among the rest of the dead markings, we distinguish those calculated at t=*d* and those obtained at t>*d*:

At t = *d* and for any values of *MaxParts*: besides the particular dead marking mentioned above, the dead markings calculated at this time corresponds to the end of negotiations where a contract has been awarded to one participant (i=1..*MaxParts*) while the rest of negotiation with the rest of participants has ended without a contract. Therefore, P_i changes state to exit_c, performs the task, which can ends by a success or a failure. P_i can, then, be in the state end_success or end_failure respectively. At the same time, the manager, which was in the state EXIT_C with respect to P_i (and EXIT_NC with respect to the rest of the participants) changes to END_SUCCESS or END_FAILURE with regard to P_i. All the other participants P_j (j≠i) are in the state exit_nc. Thus, we can deduce that at t=*d* and for any value of *MaxParts* we have:

$$NumberDeadmarkings = (2*MaxParts + 1)$$

The rest of the dead markings is calculated at t>*d*, which correspond to scenarios after the fire of the timeout where at least one participant is not in the due time. Two cases can be distinguished: a particular case of a single participant (*MaxParts*=1) and a general case of several participants (*MaxParts* > 1):

t > *d* and *MaxParts* = 1: this is particular because the single participant may miss the deadline and, consequently, changes state to exit_nc because of the canceling of its late response. The manager is in the state EXIT_NC with respect

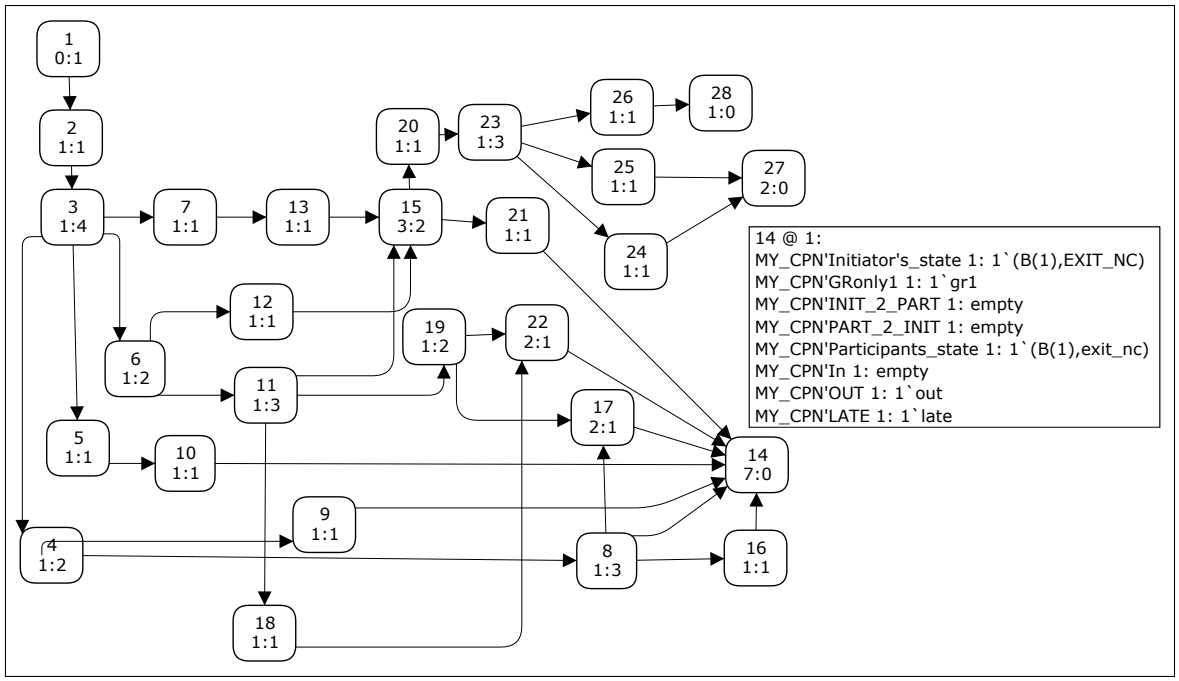


Figure 8: State space for (*MaxParts* = 1 et d = 1)

to this participant. This corresponds to the end of negotiation without a contract caused by the deadline overrun. This dead marking is reached for any discrete value of t where d<t>=2*d-1, i.e., (d-1) times and thus we deduce:

$$NumberDeadmarkings = 2 * MaxParts + d \quad (1)$$

which is equal in this case to (2+d).

t > *d* and *MaxParts* > 1: all the dead markings calculated after the timeout and for each discrete value in the interval (d..2*d-1) are similar to those obtained at t=d. The only difference is that the global clock values and the timestamps of the tokens differ. Thus, these are equivalent timed markings. Consequently, we obtain (d-1) times the same number of dead markings, i.e., (d-1)* (2**MaxParts* +1) and, therefore, we deduce:

$$NumberDeadmarkings = (2 * MaxParts + 1) * d \quad (2)$$

All these dead markings are desired terminal states of the protocol. This discussion justifies that the protocol works correctly and the beliefs between the manager and the participants are consistent. Also, it should be noted that if for a given marking two or more transitions are enabled, then the choice of the transition to fire is non-deterministic. This means that our system satisfies concurrency and non-determinism, which are key characteristics. About the communication channel, we note that at the end of negotiations, the places PART2INIT and INIT2PART are empty, signifying that there is no unprocessed messages in the network, proving, hence, that the property of cleaning the network from late messages is satisfied.

Absence of livelocks and correct termination. Table III shows that the size of the state space increases exponentially with the number of participants and the value of the deadline. This is illustrated by the graph of the Figure 10. The large number of nodes and particularly of dead markings is essentially caused by the increasing value of the deadline. The reason for this is that the timing information makes more markings distinguishable and contributes to the presence of more nodes in the state space leading to several equivalent timed markings. To verify that all the dead markings for all the values of *MaxParts* specified in Table III form a home space, we have used the CPN ML function HomeSpace (ListDeadMarkings()), which evaluates to true. This confirms that there is no livelocks and the system will always terminate correctly. Table III also shows that, for all values of *MaxParts* examined, the number of nodes and arcs in the SCC graph always remains the same as that of the state space, this implies that there is no cyclic behavior in the system, which is expected. From Table III, we conclude that there is no live transitions because of the presence of dead markings.

Absence of dead code. A dead code corresponds to a dead transition. According to Table III, there is no dead transitions in the system for all values of *MaxParts* examined, this implies that all the specified actions are executed.

Channel bound. Table III shows that the communication

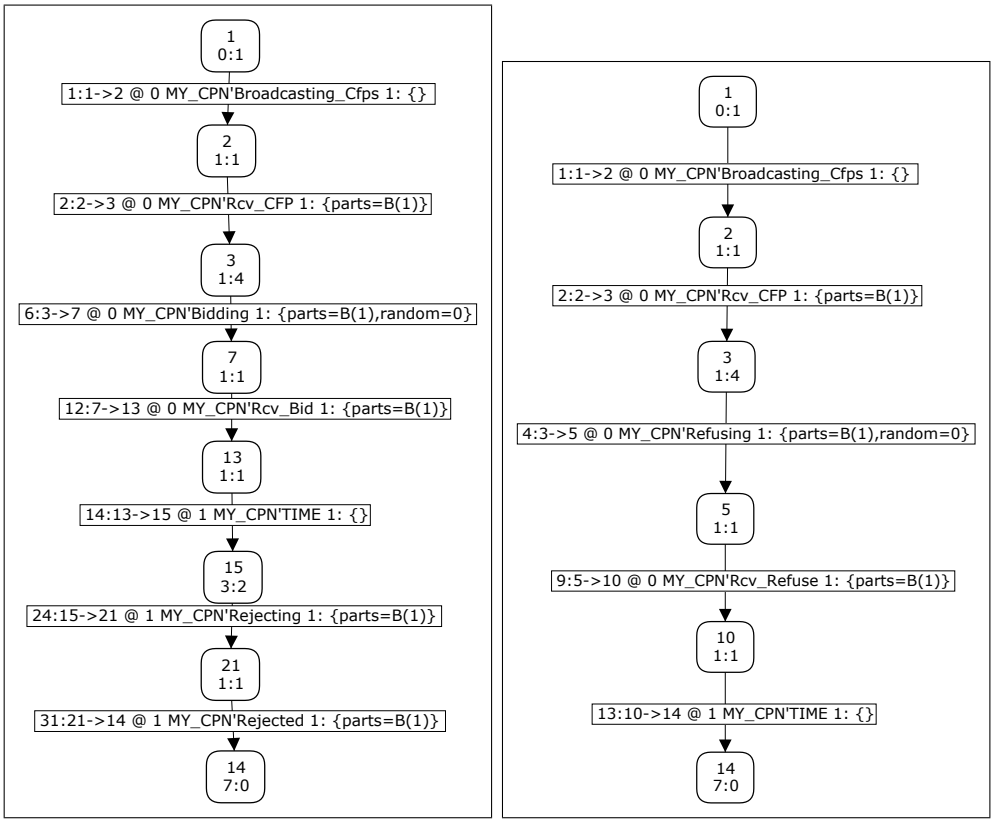


Figure 9: Examples of paths leading to the worst case (*MaxParts* = 1 et *d* = 1)

Table III: State space analysis results as a function of the parameters *MaxParts* and deadline (*d*)

Properties	*MaxParts*=1					*MaxParts*=2					*MaxParts*=3					*MaxParts*=4	
	d=1	*d*=2	*d*=3	*d*=4	*d*=5	*d*=1	*d*=2	*d*=3	*d*=4	*d*=5	*d*=1	*d*=2	*d*=3	d=4	*d*=5	*d*=1	*d*=2
State Space Nodes	28	40	52	64	76	317	605	989	1469	2045	3669	9165	18645	33216	54164	42337	140513
State Space Arcs	38	53	68	83	98	801	1357	2081	2973	4033	14113	30143	55863	93817	146549	221393	619193
Time (seconde)	00	00	00	00	00	00	00	00	01	02	07	33	161	404	831	1298	16119
SCC nodes	28	40	52	64	76	317	605	989	1469	2045	3669	9165	18645	33216	54164	42337	140513
SCC Arcs	38	53	68	83	98	801	1357	2081	2973	4033	14113	30143	55863	93817	146549	221393	619193
Dead Markings	3	4	5	6	7	5	10	15	20	25	7	14	21	28	35	9	18
HomeSpace	true	true	true	true	true	true	true	true	true	true	true	true	true	true	true	true	true
Dead Transition Instances	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
Live Transition Instances	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
Channel bound	1	1	1	1	1	2	2	2	2	2	3	3	3	3	3	4	4

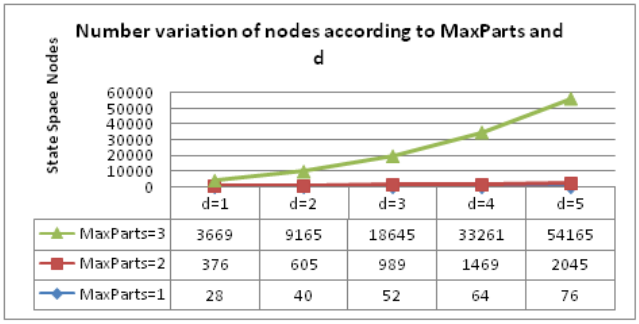


Figure 10: Number variation of the reachability graph nodes according to *Maxparts* and the deadline

channel is bounded by the MaxParts value examined, this confirms that the manager issues a single message to each of the participants and then *MaxParts* messages. Similarly, each participant issues, at a given moment, one message to the manager justifying the limit of MaxParts responses.

VII. Conclusion and Perspectives

In this paper, we have proposed to extend the AIPs with temporal aspects. We have taken the CNP as an example to illustrate our approach. A TCPN model of the contract net protocol was presented and thanks to the simulation and the state space analysis techniques we have proved that the proposed model satisfies some key properties for different values of both parameters *MaxParts* and deadline. We have also proved the beliefs consistency between the manager and the participants and that the protocol works and ends correctly. The properties namely concurrency, absence of livelocks and absence of dead code were verified too. Furthermore, we have shown how the number of dead markings (terminal states) is related to both *MaxParts* and deadline parameters. The channel bound is, however, related to only the *MaxParts* parameter.

As perspectives, we would like to use advanced state space reduction methods [20], [21] like equivalence classes to alleviate the impact of the state explosion problem, which is most accentuated for timed models. In doing so, we would verify the model for wider values of the protocol parameters. We would also like to model real time contract net [22] where, besides interaction aspects, time constraints related to task execution would be considered. These extensions would concern more complex versions of CNP. On the other hand, we would like to take into account the commitment violation by modeling a fault tolerant AIP [23] so that the sender provides a fault tolerant behavior if ever the receiver crashes during task performing or fails to meet a commitment.

References

[1] D. Boukredera, R. Maamri, and S. Aknine, *A Timed Colored Petri-Net-based Modeling for Contract Net Protocol with Temporal Aspects*, in Proceedings of the Seventh International Multi-Conference on Computing in the Global Information Technology (ICCGI 2012), pp 40-45, Venice (Italy), June 24-29, 2012.

[2] J. Billington and A. Gupta, *Effectiveness of Coloured Petri Nets for Modelling and Analysing the Contract Net Protocol*, Proc. Eighth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark, 2007, pp. 49-65 (ISSN 0105 8517).

[3] S. Aknine, S. Pinson, and M. F. Shakun, *An Extended Multi-Agent Negotiation Protocol, Autonomous Agents and Multi-Agent Systems* 8(1), pp. 5-45 (2004).

[4] J. Shujuan, Q. Tian, and Y. Liang, *A Petri-Net-Based Modeling Framework for Automated Negotiation Protocols in Electronic Commerce*, Lecture Notes in Computer Science, 2009, Volume 4078/2009, pp. 324-336.

[5] J. Billington, A. K. Gupta, and G. E. Gallasch, *Modelling and Analysing the Contract Net Protocol - Extension Using Coloured Petri Nets*, Lecture notes in computer science, 2008, NUMB 5048, pp. 169-184, Springer-Verlag.

[6] F. S. Hsieh, *Automated Negotiation Based on Contract Net and Petri Net*, Lecture Notes in Computer Science, vol. 3590, pp. 148-157, 2005. (SCI).

[7] W. L. Yeung, *Behavioral modeling and verification of multi-agent systems for manufacturing control*, Expert Systems with Applications, 38(11):13555-13562, 2011.

[8] L. Changyou and W. Haiyan, *An Improved Contract Net Protocol Based on Concurrent Trading Mechanism*, iscid, vol. 2, pp. 318-321, 2011 Fourth International Symposium on Computational Intelligence and Design, 2011.

[9] *FIPA*, Foundation for intelligent physical agents 2003. *FIPA Modeling Area: Temporal Constraints*. Retrieved May 10, 2012, from *http://www.fipa.org*.

[10] *Agent Unified modeling language, AUML*. Retrieved May 15, 2012, from *http://www.AUML.org*.

[11] L. Cabac, *Modeling Agent Interaction with AUML Diagrams and Petri Nets*, Diplomarbeit, University of Hamburg, Department of Computer Science, Vogt-Kolln Str. 30, 22527 Hamburg, Germany, 2003.

[12] M .P. Huget and J. Odell, *Representing Agent Interaction Protocols with Agent UML*, In Proceedings of the Third International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004), New York, USA, July 2004.

[13] M. P. Huget, *Extending Agent UML Protocol Diagrams*, In Proceedings of Agent Oriented Software Engineering (AOSE-02), Fausto Giunchiglia and James Odell and Gerhard Weiss (eds.), Bologne, Italie, July 2002.

[14] L. Kahloul, K. Barkaoui et Z. Sahnoun, *Using AUML to derive formal modelling agents interactions* , In 3rd ACS/IEEE Int'l. Conf on Computer Systems and Applications, pp. 109-116, IEEE Computer Society Press, 2005. (ref. CEDRIC 1384).

[15] *CPN tools homepage*. Retrieved May 10, 2012, from *cpntools.org/*.

[16] F. D. J. Bowden, *A brief survey and synthesis of the roles of time in Petri nets*, Mathematical and Computer Modelling 31 (2000) pp. 55-68.

[17] K. Jensen and L. M. Kristensen, *Coloured Petri Nets - Modelling and Validation of Concurrent Systems*, Springer, July 2009.

[18] W. M. P van der Aalst, *Interval timed coloured petri nets and their analysis*, In Application and Theory of Petri Nets 1993, Proc. 14th International Conference, volume 691, pp. 453-472, Chicago, (USA), 1993. Springer-Verlag, Lecture Notes in Computer Science.

[19] R. G. Smith, *The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver*, IEEE Trans. Computers 29(12): 1104-1113 (1980).

[20] B. Berthomieu, *La méthode des Classes d' Etats pour l'Analyse des Réseaux Temporels - Mise en Oeuvre, Extension à la multi-sensibilisation, Modélisation des Systèmes Réactifs*, In Proc. of MSR'2001, pp. 254-263, Toulouse, France, 2001. Hermes Sciences.

[21] M. A. Piera and G. Music, *Coloured Petri net scheduling models: Timed state space exploration shortages*, Mathematics and computer in simulation 82, (2011), pp. 428-441, Elsevier.

[22] L. Qiaoyun, L. Jiandong, D. Dawei, and K. Lishan, *An extension of contract net protocol with real time constraints*. Wuhan University Journal of Natural Sciences. Wuhan University Journals Press. ISSN:1007-1202 (Print) 1993-4998, Volume 1, Number 2 / juin 1996, pp. 156-162.

[23] N. Dragoni, M. Gaspari, and D. Guidi, *An ACL for specifying Fault-Tolerant protocols*, AI*IA 2005: Advances in Artificial Intelligence 9th Congress of the Italian Association for Artificial Intelligence, Italy 2005 pp. 237-248.

On the Real-Time Evaluation of Two-Level BTD Scheme for Energy Conservation in the Presence of Delay Sensitive Transmissions and Intermittent Connectivity in Wireless Devices

Constandinos X. Mavromoustakis and Christos D. Dimitriou
Department of Computer Science,
University of Nicosia
46 Makedonitissas Avenue, P.O.Box 24005
1700 Nicosia, Cyprus
mavromoustakis.c@unic.ac.cy; dimitriou.cd@st.unic.ac.cy

George Mastorakis
Department of Applied Informatics and Multimedia
Technological Educational Institute of Crete
Heraklion, Crete, Greece
gmastorakis@staff.teicrete.gr

***Abstract*—This work elaborates on the real-time implementation and comparative evaluation of an Energy-efficient scheme for sharing resources using the MICA2dot wireless nodes/motes. The proposed scheme allows the nodes to sleep adaptively according to the volume of incoming traffic, offering Energy Conservation (EC) to the moving nodes. Nodes that are exchanging delay sensitive/constrained resources apply the one-level Backward Traffic Difference (BTD) scheme or the *two-level BTD*, according to delay transmission and capacity criteria, in order to enable nodes to sleep, based on their activity and their admitted traffic. The incoming traffic impacts the Sleep-time duration of the node by using traffic's backward difference in order to define an adaptive Sleep-time duration for each node. The proposed scheme is being evaluated through real-time implementation by using MICA2dot wireless motes, which are exchanging resources in a Mobile Peer-to-Peer manner using certain motion pattern. Performance evaluation and the extracted results validate the scheme's efficiency for minimizing the Energy Consumption in real-time. In addition, comparative performance evaluations with other similar schemes show the efficiency of the proposed research approach. The framework of this paper maximizes further the efficiency and reliability of the resource exchange process of the nodes, while it minimizes the Energy consumption.**

Keywords- energy conservation scheme; lifespan extensibility metrics; one-level BTD scheme; selective two-level BTD scheme; traffic-oriented energy conservation; traffic volume and capacity metrics

I. INTRODUCTION

In recent years, the number of wireless network deployments for real-time applications, including individual and global industrial applications, has rapidly increased. As a variety of device-dependent applications were born, the necessity for developing a scheme for conserving energy becomes even more timely. Wireless nodes communicate over error-prone wireless channels with limited battery power, vulnerable reliability and through deployed energy-hungry applications. These characteristics of wireless nodes make the design of resource exchange schemes challenging. However, with mobility many inherent problems follow such as the resource scarceness, finite energy for the wireless nodes and low connectivity whereas, mobile nodes demand high levels of responsiveness that in turn, demand resource intensive computing resources. Wireless devices in order to conserve energy switch their states between Sleep mode, Wake mode and idle mode. This is reflected to the responsiveness of the underlying applications and processes hosted by these devices, which are reduced significantly. These devices, while being in the process of sharing resources, face temporary and unannounced loss of network connectivity as they move whereas, they are usually engaged in rather short connection sessions since they need to discover other hosts in an ad-hoc manner. Due to wireless resources' scarceness, in most cases the requested resources claimed by these devices, may not be available. Therefore, a mechanism that faces the intermittent connectivity problem and enables the devices to react to frequent changes in the environment, while it enables energy conservation in regards to the requested traversed traffic, is of great need. This mechanism will positively affect the end-to-end reliability, facing the unavailability and the scarceness of wireless resources.

This work elaborates on the capabilities of the backward estimation model for extracting the time-oriented differential traffic, in contrast to the nodal characteristics of the wireless device in time. The proposed work exploits the model proposed in [1] and utilizes the resource availability and capacity characteristics in a reflected model for offering Energy Conservation and minimization of scarcity of the requested resources. The proposed scheme uses the cached mechanism (as in [1]) for guaranteeing the requested resources which are delay sensitive, whereas wireless nodes are subject to sudden failures. The proposed mechanism extends the introduced Backward Traffic Difference (BTD) scheme [1], by adding a second level of traffic difference in the proposed framework-namely two-level BTD. The designed model guarantees the end-to-end availability of requested resources while it reduces significantly the Energy Consumption and maintains the requested scheduled transfers, in a mobility-enabled and cluster-based communication. Furthermore, since each node has different capacity measures and undefined remaining energy, this work adopts a differential

dissimilar assignment(s) of sleep-wake schedule estimation, based on the traffic difference through time and the relative capacity and the associated traffic characteristics. The proposed model has been applied in real-time devices and the conducted experiments using various capacity and traffic-aware metrics, were carried-out for the energy conservation and the evaluation of the proposed model. The BTD scheme initially evaluates the data volume/traffic and according to the delay bound/limitations, the model adds a second level of traffic difference. Real-time experiments show that different types of traffic can be supported, where the adaptability and the robustness that is exhibited is mitigated according to the proposed scheme's Sleep-time estimations and assignments supporting delay sensitive data transfers.

The structure of this work is as follows: Section II describes the related work done and the need in adopting a Traffic-based scheme, and then Section III follows by presenting the proposed Backward Traffic Difference estimation for Energy Conservation as in [1]. The proposed framework makes progress beyond the current state of the art by supporting delay bounded/sensitive data transfers in collaboration with the promiscuous caching recoverability mechanism in the case of intermittent connectivity. Section IV presents the real time performance evaluation results focusing on the behavioral characteristics of the scheme and the Backward Traffic Difference along with the system's response, followed by Section V with the conclusions and foundations, as well as potential future directions.

II. Related work and Motivation

Multimedia or delay-sensitive applications can only be implemented with guaranteed QoS and QoE support, in wired environments whereas, rarely mobile devices can guarantee the communication in an end-to-end reliable manner. This is primarily the reason that the number of applications beyond file sharing is kept on a low implementation level [2], despite the penetrative character of Peer-to-Peer systems nowadays. The type of application hosted on wireless devices typically relies its presence on the energy that the device hosts. Recent research has addressed the Mobile Peer-to-Peer (MP2P) connectivity from different perspectives. Work done in [3] has introduced a middleware support for client-server architectures in nomadic environments, where in an organized way the terminals take into consideration the group-oriented characteristics. In these environments, the associated RPC-based middleware mechanisms have been enhanced with queuing or buffering capabilities in order to cope with intermittent connections. Examples of these implementations are introduced in Mobile DCE in researches [4], [5], [6] and [7] including diffusion policies and resources' processing [4] and [5] and manipulation as well as different replication procedures [6] and [7].

Moreover, many recent high-quality design and validation measurement studies in [8], [9] have convincingly demonstrated the impact of traffic on the end-to-end connectivity (like the work in [10]) and thus the impact on the Sleep-time duration and the EC. The realistic traffic in real-time communication networks and multimedia systems, including wired local-area networks, wide-area networks, wireless and mobile networks, exhibits noticeable burstiness over a number of time scales [11] and [12]. This fractal-like behavior of network traffic can be much better modeled by using statistically self-similar or Long Range Dependent (LRD) processes. These processes can be further improved in terms of estimations, taking into consideration different theoretical properties from those of the conventional Short Range Dependent (SRD) processes. There are many Sleep-time scheduling strategies that model each node's transition between *ON* and *OFF* states. Existing scheduling strategies for wireless networks could be classified into three categories: the coordinated sleeping [13], [14], where nodes adjust their sleeping schedule, the random sleeping [15] and [16], where there is no certain adjustment mechanism between the nodes in the sleeping schedule with all the pros and cons as expressed in [17], and on-demand adaptive mechanisms [18], where nodes enter into Sleep-state depending on the environment requirements whereas, an out-band signaling is used to notify a specific node to go to sleep in an on-demand manner.

In addition to the existing architectures, a number of researches have attempted the association of different parameters with communication mechanisms, in order to reduce the energy consumption. These researches have been introduced in [1], [8], [9] and [19] where different traffic-based manipulations are modeled, in order to overcome the over-exposure and over-activity of nodes. These traffic-aware mechanisms can be classified into two categories: active and passive schemes. Active techniques conserve energy by performing energy conscious operations, such as traffic and data volume transmission scheduling by using a directional antenna [20], and energy-aware routing [21]. On the other hand, passive techniques conserve energy by scheduling the interfaces of the devices to the sleep mode when a node is not currently taking part in communication activity [22] and host different adaptive methodologies like the Adaptive-Traffic enabled methodologies [9], [10]. The latter takes into consideration the traffic pattern that a node is experiencing as incoming and outgoing traffic. Authors in [17] consider the association of EC problem with different parameterized aspects of the traffic (like traffic prioritization) and enable a mechanism that tunes the interfaces' scheduler to sprawl in the sleep state according to the activity of the traffic of a certain node in the end-to-end path. Authors in [18] aim to minimize energy consumption in Wireless Sensor Networks (WSNs) through a 2-tier asynchronous scheduling scheme for delay constrained connectivity in wireless devices that are asymmetrical in terms of capacity and battery lifetime.

Within the context of providing an energy-efficient traffic manipulation, a fertile ground has been the idea of the development of new heuristic approaches, by associating different traffic-aware (transmission-aware) parameters with communication mechanisms for reducing

the Energy Consumption. In this context, the main goal of this work is to further minimize the energy consumed by the wireless nodes by applying further traffic association and stationarity measures to the estimated sleep-time duration of the node. This work aims at prolonging further the network lifetime by minimizing the energy consumption. This is performed through the incoming traffic that traverses each one of the nodes, taking into consideration the repetition pattern of the traffic. In addition, with the work done in [1], the proposed scheme estimates the Backward Difference by using a second level of traffic difference estimation, for extracting the time duration for which the sensor mote (the node) is allowed to Sleep during the next time slot *T*. This mechanism, in order to enable further recoverability and availability of the requested resources, proposes an efficient way to cache the packets destined for the node with turned-off interfaces (sleep state) onto intermediate nodes and enables, through the Backward Traffic Difference estimation, the next Sleep-time duration of the recipient node to be adjusted accordingly. The model has been applied to MICA2 sensor nodes hosting a TinyOS operating system which has been programmed to tune the wireless interfaces of the motes using the Nested C (NesC) language. The designed model and the real-time conducted experimental results, show that the proposed scheme guarantees the end-to-end availability of requested resources, while it reduces significantly the Energy Consumption and maintains the requested scheduled transfers, in a mobility-enabled cluster-based communication.

III. TWO-LEVEL BACKWARD DIFFERENCE TRAFFIC ESTIMATION FOR ENERGY CONSERVATION FOR MOBILE PEER-TO-PEER OPPORTUNISTIC RESOURCE SHARING

Sleep/wakeup schemes can be classified into three main categories namely: on demand, asynchronous and the methodologies based on the scheduled rendezvous. On-demand schemes assume that destination nodes can be awakened somehow just before receiving data. As traffic-aware policy requires an active scheme to be applied and reflective solutions to be adopted, this work uses the time-based incoming traffic to minimize the energy consumption and the relative trade-offs while prolonging the systems' lifetime. The scheduled rendezvous for assigning the independent sleep/wakeup slots requires that nodes in the system are synchronized and neighboring nodes wake up and communicate at the time that at least 1-hop neighbor is awake and informed. In this work the input nodal traffic is being considered and manipulated according to the BTD. This manipulation is the basis for providing using a feedback model, the sleep-time duration estimation to nodes. Wireless nodes have to be self-aware in terms of power and processing as well as in terms of accurate participation in the transmission activity. There are many techniques such as the dynamic caching-oriented methods. The present work utilizes a hybridized version of the proposed adaptive dynamic caching [9], which is considered to behave satisfactorily and enables simplicity in real time implementation [10]. On the contrary with [10] [19], in this work a different real-time mobility scenario is modeled and hosted in the scheme, which enables an adaptive tuning of the Sleep-time duration according to the activity of the traffic on each node.

The following section presents the estimations performed on each node in order to evaluate the next Sleep-time duration according to the node's incoming activity by using the BTD and the second level of BTD estimation to extract the time duration for which the node is allowed to Sleep during the next time slot *T*.

A. Backward Difference Traffic Estimation for Energy Conservation for delay sensitive transmissions

Taking into account the fact that opportunistically connected nodes are dynamically changing their operational characteristics, when a source needs to send requested packets or stream of packets (file) to a destination where the destination node(s) may have moved or is/are set in the Sleep-state, then the requested information will be missed and lost. This implies that, in a non-static multi-hop environment, there is a need to model the activity slots that a node experiences in contrast to the requested resources in the end-to-end path such that the resources can be efficiently shared among users whereas, any redundant transmissions and lost packets/streams are avoided.

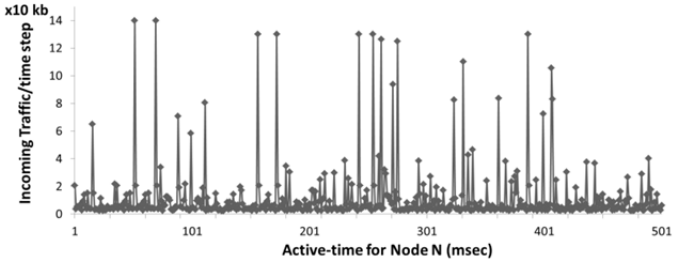


Figure 1. Real-time Incoming traffic that a node experiences with the associated traffic capacity and activity duration of the node with mean $E(A_{t_f})$.

As appropriate mechanisms are required to guide the activity periods of each one of the nodes, the traversed traffic can be the parameterized input. In this way, it evaluates the next Sleep-time or the Active periods, and considers the terminal's transmission and reception durations. This can be achieved through the BTD estimation. The proactive scheduling may increase the network lifetime, contrarily with periodic Sleep-Wake schedules, as it enables dissimilar active-time. The nodes are set in the active state for a period of time according to the incoming traffic. The activity period(s) of a node is primarily dependent on the nature and the spikes of the incoming traffic destined for this node [6]. If the transmissions are performed on a periodic basis then the nodes' lifetime can be forecasted and according to a model can be predicted and estimated [7]. In this framework, this work introduces the one-level BTD and the second level of BTD estimation, in order to associate the traversed traffic

of a node with the previous moments and, in real-time, reduce the redundant Activity-periods of the node in order to conserve energy. Figure 1 shows the incoming traffic that a node experiences in real-time with the associated traffic capacity and activity duration of the node. The traffic can be seen as a renewal process [7] that has aggregation characteristics [9] from different sources.

This work primarily assigns a dissimilar sleep and wake time for each node, based on traffic that is destined for each node which is cached onto 1-hop intermediate node(s), during the Sleep-time of the destination node. Figure 2 shows that in a pre-scheduled periodic basis, nodes can be in the Sleep-state. Likewise, the packets that are destined for the certain node can be cached for a specified amount of time (as long as the Node (i) is in the Sleep-state) in the 1-hop neighbor node (Node(i-1)) in order to be recoverable when node enters the Active state.

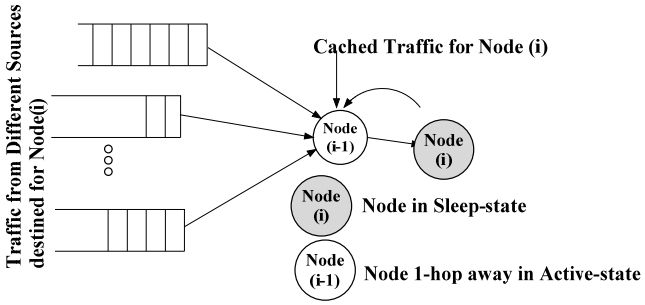


Figure 2. A schematic diagram of the caching mechanism addressed in this work.

The 1-hop neighbor node (*Node(i-1)*) is selected to cache the packets destined for the node with turned off interfaces (sleep state). The principle illustrated in Figure 2 denotes that when incoming traffic is in action for a specific node, then the node remains active for prolonged time. As a showcase, this work takes the specifications of the IEEE 802.11x that are recommending the duration of the forwarding mechanism that takes place in a non-power saving mode lays in the interval 1 nsec < τ <1 psec. This means that every ~0.125μsec (8 times in a msec) the communication triggering action between nodes may result a problematic end-to-end accuracy. Adaptive Dynamic Caching [2] takes place and enables the packets to be "cached" in the 1-hop neighboring nodes. Correspondingly, if node is no-longer available due to sleep-state in order to conserve energy (in the interval slot T=0.125μsec), then the packets are cached into an intermediate node with adequate capacity equals to: $C_{t_f,k(s)}(t) > C_{t_f,i}(t)$, where $C_{t_f} > \alpha \cdot C_i$; where α_i is the capacity adaptation degree based on the time duration of the capacity that is reserved on node N of C_k ; where $C_{t_f,k(s)}(t)$ is the needed capacity where i is the destination node and k is the buffering node (a hop before the destination via different paths).

As this scheme is entirely based on the aggregated self-similarity nature of the incoming traffic with reference to a certain node, there should be an evaluation scheme in order to enable the node to Sleep, less or more according to the previous activity moments. This means that as more as the cached traffic is, there is an increase in the sleep-time duration of the next moment for the destination node. This is indicated in the following scheme that takes into account the Self-Similarity to estimate the potential spikes of the Sleep-time duration. The Sleep-time in turn accordingly decreases or increases, based on the active traffic destined for *Node(i)* while being in the Sleep-state.

1) Backward Difference Traffic Moments and Sleep-time duration estimation

In [1], authors expanded the traffic-oriented Sleep and Wake durations by using single moment Backward Traffic Difference and exploiting the silent periods to estimate an increase of the sleep-time duration and conserve energy. Further to the work done in [1], this work evaluates the second level of BTD by using a statistical mean in the evaluation of the duration of the next sleep-time of the node. Let $C(t)$ be the capacity of the traffic that is destined for the Node i in the time slot (duration) t, and $C_{N_i(t)}$ is the traffic capacity that is cached onto *Node (i-1)* for time t. Then, the one-level Backward Difference of the traffic is evaluated by estimating the difference of the traffic while the *Node(i)* is set in the Sleep-state for a period, as follows:

$$\begin{aligned} \nabla C_{N_i(1)} &= T_2(\tau) - T_1(\tau - 1) \\ \nabla C_{N_i(2)} &= T_3(\tau - 1) - T_2(\tau - 2) \\ &\vdots \\ \nabla C_{N_i(n+1)} &= T_n(\tau - (n-1)) - T_2(\tau - (n-2)) \end{aligned} \quad (1)$$

$\nabla C_{N_i(1)}$ denotes the first moment traffic/capacity difference that is destined for *Node(i)* and it is cached onto Node (i-1) for time τ, $T_2(\tau) - T_1(\tau - 1)$ is the estimated traffic difference while packets are being cached onto (i-1) hop for recoverability. Equation (1) depicts the BTD estimation for one-level comparisons which means that the moments are only being estimated for one-level ($T_2(\tau) - T_1(\tau - 1)$). The traffic difference is estimated so that the next Sleep-time duration can be directly affected according to the following:

$$\delta(C(T)) = C_{total} - C_1, \forall C_{total} > C_1, T \in \{\tau - 1, \tau\} \quad (2)$$

In addition, the traffic that is destined for Node (i), urges the Node to remain active for $\frac{\delta(C(T))}{C_{total}} \cdot T_{prev} > 0$.

According to [9], the Long-Range Dependence of self-similar incoming traffic can be measured using the probability density function of the Pareto distribution and the corresponding mean value, whereas, the load generated by one source is mean size of a packet train divided over mean size of packet train and mean size of inter-train gap

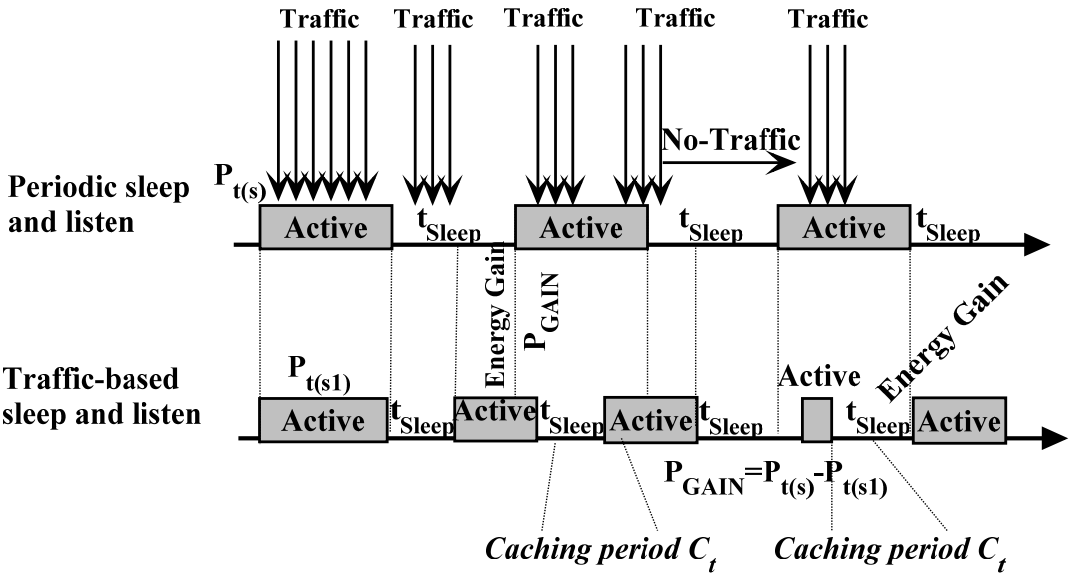


Figure 3. *ON* and *OFF* periodic durations of a Node with the associated cached periods.

or it is the mean size of *ON* period over mean size of *ON* and *OFF* periods as follows:

$$L_i = \frac{ON_i}{ON_i + OFF_i} \quad (3)$$

When a node admits traffic, the traffic flow t_f can be modeled as a stochastic process [17] and denoted in a cumulative arrival form as $A_{t_f} = \{A_{t_f}(T)\}_{T \in N}$, where $A_{t_f}(T)$ represents the cumulative amount of traffic arrivals in the time space [0..T]. Then, the $A_{t_f}(s,T) = A_{t_f}(T) - A_{t_f}(s)$ (4), denotes the amount of traffic arriving in time interval (*s, t*]. Hence, the next Sleep-time duration for *Node (i)* can be evaluated as:

$$L_i(n+1) = \frac{\delta(C(T) \mid A_{t_f}(s,T))}{C_{total}} \cdot T_{prev}, \forall \delta(C(T)) > 0 \quad (5)$$

For the case that the $\delta(C(T)) < 0$ it stands that:

$$\delta(C(T)) = C_{total} - C_1, \forall C_{total} < C_1, T \in \{\tau - 1, \tau\} \quad ,$$

and $\frac{\delta(C(T))}{C_{total}} \cdot T_{prev} < 0, \forall T_{prev} > T_{prev}(\tau - 1)$, the $C_{N_i} < 0$ and the total active time increases gradually according to the following estimation:

$$T_{sleep} = T(\tau - t_1) - (-C_{N_i}) = T(\tau - t_1) + T_{C_{N_i}} \quad (6)$$

the $T_{C_{N_i}}$ is the estimated duration for the capacity difference for $C_{N_i} < 0$, whereas the Sleep-time duration decreases accordingly with Equations (5) and (6), iff the $C_{N_i} < 0$. Considering the above estimations the traffic flow can be expressed as in [23] as

$$A_{t_f}(T) = m_{t_f}(T) + \hat{Z}_{t_f}(T) \quad (7)$$

where $m_{t_f}(T)$ is the mean arrival rate and $\hat{Z}_{t_f}(T) = \sqrt{a_{t_f} m_{t_f}(T)} \cdot \hat{Z}_{t_f}(T) \cdot a_{t_f}$. The coefficient a_{t_f} is the variance coefficient of $A_{t_f}(T)$. $\hat{Z}_{t_f}(T) \cdot$ is the smoothed mean as in [17], and with $\mathrm{E}(\hat{Z}_{t_f}(T)) = 0$ satisfying the following variance and covariance functions:

$$\begin{aligned} v_{t_f} &= a_{t_f} m_{t_f} \cdot T^{2H_{t_f}} \\ \sigma_{t_f}(s,T) &= \frac{1}{2} a_{t_f} m_{t_f} \cdot (T^{2H_{t_f}} + s^{2H_{t_f}} - (T-s)^{2H_{t_f}}) \end{aligned} \quad (8)$$

$H_{t_f} \in \left[\frac{1}{2}, 1\right]$ is defined as the *Hurst* parameter, indicating the degree of self-similarity. Estimations in (8) can only be valid if the capacity of the *Node* (*i*-1) can host the aggregated traffic destined for Node (*i*) satisfying the $\sup_{s \le T} \left\{ \sum_{t_f=1}^{N} A_{t_f}(s,T) - C_{t_f}(T) \right\}$, for traffic flow t_f at time T and $C_{t_f}(T)$ represents the service capacity of the Node (*i*-1) for this time duration.

2) Two-level Backward Difference Traffic Moments and Sleep-time duration estimation

According to the one-level Backward Difference of the Traffic, the difference in the capacity measure can be estimated as the difference of the traffic while the Node (*i*) is set in the sleep-state. This corresponds to the admitted nodal traffic for a period, as the estimations of the one-moment traffic difference set above. Therefore, in order to estimate the second level Backward Traffic Difference, we need to associate the T_1, T_2, T_3 traffic moments with the volume of admitted traffic $\nabla C_{N_i(t)}$ and define the difference as follows:

$$\begin{aligned} \nabla C_{N_i(0)} &= T_3(\tau) - T_1(\tau - 2) \\ \nabla C_{N_i(1)} &= T_2(\tau) - T_1(\tau - 1) \\ \nabla C_{N_i(2)} &= T_3(\tau - 1) - T_2(\tau - 2) \end{aligned} \quad (9)$$

Figure 4 shows the two level traffic moments and the association between the T_1, T_2, T_3 traffic moments through the Backward slots that are associated with the traffic.

According to Equation (9) the second level Backward Traffic Difference defines the moments that a specified volume of traffic traverses the *Node(i)*. Therefore, the second level Backward Traffic Difference can impact the evaluated sleep duration if the traffic has increased or the sleep-time duration of the *Node(i)* has increased, by avoiding the node to become saturated [8], [9], [24] and [25]. This estimation after consecutive statistical mean estimations has been found to be an estimation of the time as:

$$L_i(n+1) = \frac{\nabla C_{N_i(0)} + \nabla C_{N_i(1)} + \nabla C_{N_i(2)}}{3 \cdot d_p} \quad (10)$$

or as a general form for the j-slot of a *Node(i)* as:

$$L_i^{\,j}(n+1) = \frac{\nabla C_{N_i(j-2)} + \nabla C_{N_i(j-1)} + \nabla C_{N_i(j)}}{3 \cdot d_p} \quad (11)$$

In (11), d_p is the maximum delay in the end-to-end path from a source to a destination where the reference (i.e., A) node lays in, T is the round/cycle for which t_{idle} is evaluated, and *n* is the number of hops. d_p is calculated as:

$$d_p = \sum_{i=0}^{i-1} \delta_i + T_i \quad (12)$$

δ_i is the duration where the requested data was hosted onto *i*-node, and T is the transmission delay.

In order to avoid node's capacity diversities and saturations, each node re-evaluates the sleep-time duration by applying idle listening slots. These slots occur when a sensor wireless node listens to an idle channel to receive possible traffic. Hence, in order to evaluate the idle time t_{idle} for each node that will get into the idle state, the following estimation takes place:

$$t_{idle} = \frac{(T - \max(d_p))}{n} \quad (13)$$

The basic steps of the proposed scheme can be summarized in the pseudocode of the Table 1.

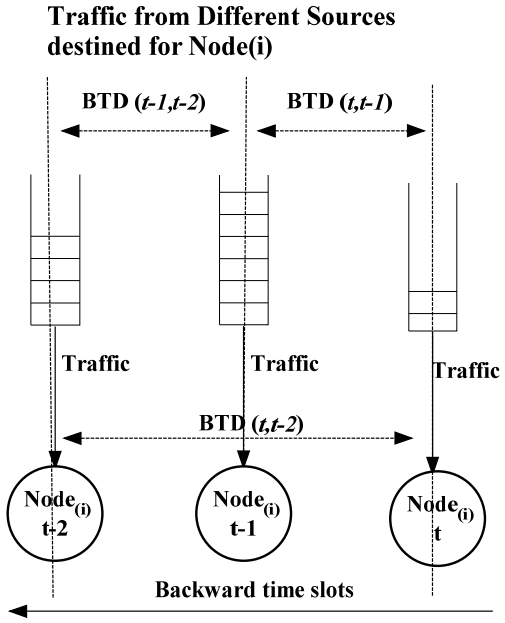


Figure 4. Two-level traffic moments for *node(i)* and the association between the T_1, T_2, T_3 nodal traffic moments.

TABLE I. BASIC STEPS OF THE PROPOSED TWO-LEVEL BTD SCHEME

```
1: for Node(i) that there is C(t)>0 {
2: while (C_{N_i(t)}>0) { //cached traffic
   measurement
3: Evaluate (∇C_{N_i(1)});
4: Calc(δ(C(T)) = C_total − C_1, ∀C_total > C_1, T ∈ {τ−1, τ})
5: if (Activity_Period=(δ(C(T))/C_total)·T_prev >0)
   //Measure Sleep-time duration
6: Evaluate L_i(n+1) = (δ(C(T)|A_{i_j}(s,T))/C_total)·T_prev, ∀δ(C(T)) > 0
7: else if (C_{N_i(t)} > C_{N_i(t−1)}) {
8: while(t_idle = (T − max(d_p))/n) {//Provided that t
   idle is satisfied
```

```
   //second level Backward Traffic
9: Evaluate  L_i^j(n+1) (∇C_{N_i(j-2)} + ∇C_{N_i(j-1)} + ∇C_{N_i(j)}) / (3·d_p)
   //Difference can impact the evaluated sleep
   duration,
   //according to the two-level BTD scheme
   proposed.

   }//if
   }//while
10: else if (δ(C(T)) < 0)
11:      T_sleep = T(τ − t_1) − (−C_{N_i}) = T(τ − t_1) + T_{C_{N_i}} ;
12: Sleep (T_sleep);
13: } //for
14: }//while
```

Taking into consideration the above stochastic estimations, the Energy Efficiency EE_{t_f} can be defined as a measure of the capacity of the *Node(i)* over the *Total Power consumed* by the *Node*, as:

$$EE_{t_f}(T) = \frac{C_{t_f}(T)}{TotalPower} \quad (14)$$

In addition, the Energy Efficiency should satisfy the minimum energy regions for wireless devices defined in [9] where the following is applied:

$$\arg\max(EE_{t_f}(T)) = \min\left[P_{thresshold}\right] \forall i, j \quad (15)$$

$P_{thresshold}$ is the consumed power in the resource interexchange region and should not exceed a certain threshold (as in [9]), and *i* and *j* are the streaming source and destination nodes respectively. Equations (14) and (15) above can be defined as the primary metric for the lifespan extensibility of the wireless node in the system.

IV. Real Time Performance Evaluation Analysis, Experimental Results and Discussion

In this section, the effectiveness of the proposed BTD approach is demonstrated and the accuracy of the developed scheme is validated in real-time by comparing the analytical results of the scheme to those obtained from extensive simulation experiments in work done in [8],[9] and [10]. Towards evaluating the proposed scenario in real time, the MICA2 sensors nodes have been used [27] configured to be manipulated as Peer devices hosting the proposed BTD scheme. These sensors were equipped with the MTS310 sensor boards.

The MICA2 features a low power processor and a radio module operating at 868/916 MHz enabling data transmission at 38.4Kbits/s with an outdoor range of maximum set to 50 meters-taking both no-fading and fading obstacles in-between for better evaluation of the signal strength. The TinyOS operating system is hosted onto MICA2 using the Nested C (NesC) language. As the sensors are application-specific, they can only host a single application. TinyOS does not support memory management or internal process management and, therefore, it discourages applications from allocating or using dynamic memory. This feature enables to evaluate the trade-off between the periodic sleep-wake slot assignments and the proposed scheme which uses a variable and dynamic Sleep-slot assignment. A dynamic topology with the mobility expressed in Section IV.A is implemented, where the BTD scheme assigns the traffic-oriented Sleep and Wake durations. Furthermore, MICA2 supports an expansion connector for attaching various sensor boards on it. The MTS310 board was utilized, which supports a sensor board with a variety of sensing modalities including sounder and an overclocking alarm. The sounder was used to extract sound when needed and denoting the overload of the node or other determined functionalities. Towards evaluating the proposed scheme the signal strength measures were taken into account as developed in [8] and [9], as well as the minimized ping delays between the nodes in the end-to-end path according to the $d_p = Min\sum_{i=1}^{n} D_i$, where *D* is the delay from a node *i* to node *j*, and d_p is the minimized evaluated delay in the end-to-end available path. Moreover, considering the need of bandwidth and the limited battery power for wireless devices, it is necessary to apply efficient routing algorithms to create, maintain and repair paths, with least possible overhead production. The underlying radio technology supports the Cluster-based Routing Protocol (CRP) [28]. A common look-up application is being developed to enable users to share resources on-the-move that are available by peers for sharing. This application hosts files of different sizes that are requested by peers in an opportunistic manner.

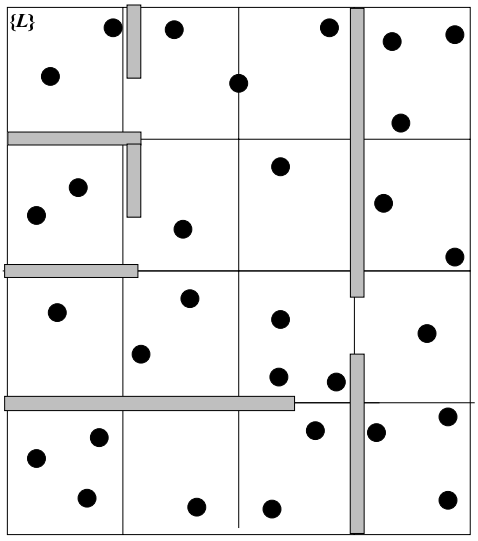


Figure 5. Topology and the location of the sensor nodes in the experimental room.

A. Mobility Model used for mobile peers

Unlike the predetermined Landscape in [31], in this work, the mobility scenario used is the Fractional Random Walk. The random walk mobility model was derived from the Brownian motion, which is a stochastic process that models random continuous motion [30], [31]. In this model, a mobile node moves from its current location with a randomly selected speed in a randomly selected direction as real time mobile users act. However, the real time mobility that the users express, can be defined by spotting out some environmental elements (obstacles, point-of-interest et.c) where users' decisions may be affected. In the proposed scenario, the new speed and direction are both chosen from predefined ranges, $[v_{min}, v_{max}]$ and $[0, 2\pi)$, respectively [1], [30]. The new speed and direction are maintained for an arbitrary length of time randomly chosen from$(0, t_{max}]$. At the end of the chosen time, the node makes a memoryless decision of a new random speed and direction. The movements are expressed as a Fractional Random Walk (FRW) on a Weighted Graph, utilized in the same way as in [32]. The topology *L* and the sample location of the sensor nodes in the experimental room are shown in Figure 5. In addition, this work uses the Random walk as a Markov chain (designed as in the theoretical foundations in [30]) for the motion of each one of the nodes. This allows nodes to continue their path/journey according to their initial selection-decision from a point, i.e., A to a point B in *L*. By using the Random walk as a Markov chain model it denotes that the last step made by the random walk influences the next one based on the stationarity and the correlations between the movements. Under the condition that a node has moved to the right the probability that it continues to move in this direction is then higher than to stop movement. This leads to a walk adjusted to a walk mobility that leaves the starting point much faster than the original random walk model. The probabilities are defined as follows: assuming that a device is currently at location l_i, the next location of the node l_j is chosen from among the neighbors of *i* with probability:

$$p_{ij}^{L} = \frac{w_{ij}}{\sum_{k} w_{ik}} \quad (16)$$

Equation (16) presents the p_{ij} which should be proportional to the weight of the edge (l_i, l_j) and defines *k* as the destination location. In turn the sum of the weights of all edges in the landscape *L* is:

$$w^{L}_{ij} = \sum_{i,j:j>1} w_{ij} \quad (17)$$

All nodes have asymmetry in the signal strength and obstacles within their communication with other nodes whereas, they are moving with random walks. The mobility of each nodes is generated via mechanical robots using the Lynxmotion Track Robot Kit [35], which are programmed to follow the pattern of probabilistic Fractional Random Walk model. The control of the Robot is performed through a 2.4GHz Spectrum radio controller for the movements in all directions.

B. Real-time performance testing and evaluation using the MICA2 sensors equipped with the MTS310 sensor boards

In this section, the results extracted after conducting the real time evaluation runs of the proposed scenario, are presented. In the utilized scenario, 30 nodes were used with each link (frequency channel) having max speed reaching data transmission at 38.4Kbits/s. The wireless network is organized in 6 overlapping clusters, which may vary in time in the active number of the nodes. Each source node transmits one 512-bytes (~4Kbits-light traffic) packet asynchronously and randomly each node selects a destination. The speed of each device can be measured with the resultant direction unit vector [9] and the speed. Each device has an asymmetrical storage capacity compared with the storages of the peer devices. The ranges of the capacities for which devices are supported are set in the interval 1MB to 20MB[1].

Figure 6 shows the average Throughput in contrast to the number of nodes in the streaming zone that were evaluated in real-time using comparatively the periodic sleep-time durations, the scheme in [33] and the proposed scheme. It is undoubtedly true that the proposed scheme enables higher Average Throughput response in the system whereas, comparing with the results extracted from Figure 7, the proposed scheme enables greater network lifetime by using the proposed activity traffic-based scheme. Moreover, Figure 8 shows the Successful packet Delivery Ratio (SDR) in regards to the simultaneous requests in the intra-cluster communicating path for the proposed two-level BTD scheme. The results extracted in Figure 8 are characterizing both statically located nodes (where no movement exists) and mobile nodes where Fractional Random Walk is applied. It is important to note that when the number of mobile nodes increases the SDR drops dramatically. After conducting controlled real-rime evaluations it was noticed that, the significant decrease in the SDR appears, if the total number of moving nodes exceeds the 60% of the nodes in the cluster. This is due to the promiscuous caching policy that it affects the active nodes to prolong their active-time duration causing thereafter to prolong their sleep-time duration which in turn, results in a significant drop in the SDR. Figure 9 shows the Average Throughput with the Total Transfer Delay in (μsec) is shown, for different mobility models. Figure 9 presents the different Throughput responses that the proposed scheme exhibits in contrast to the mobility characteristics, for full node mobility, moderate and low (30%) mobility. It is important to mark that in the cases of full node and moderate mobility, the Throughput decreases when the delays experienced are increased. This is expected, as when the delays on nodes increase the overall

[1] The capacity for each device can be tuned according to the volume of the Traffic in the configuration process.

cluster throughput drops. This is due to the transfers' end-to-end delay metric that characterizes the sensitivity of the transfer in delay.

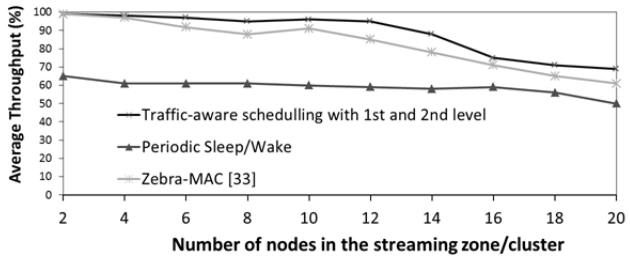


Figure 6. The average Throughput with the number of nodes in the cluster zone for delay bounded transmissions. Evaluation takes place for different comparable schemes.

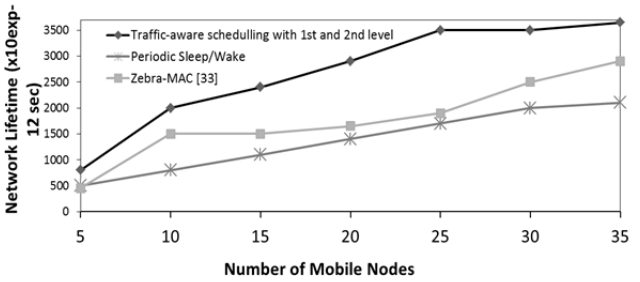


Figure 7. The fraction of the remaining Energy through time using real-time evaluation for different schemes.

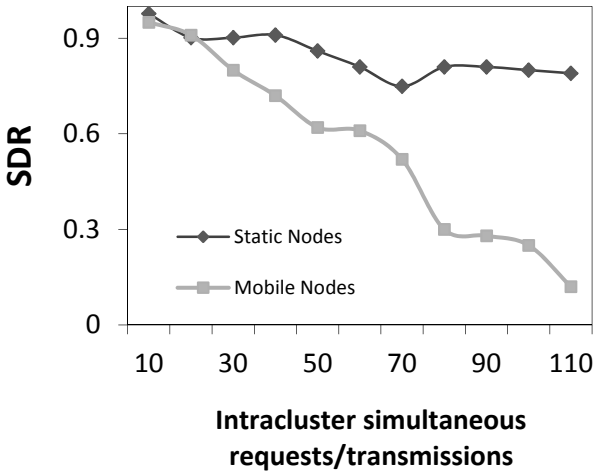


Figure 8. The Successful packet Delivery Ratio (SDR) with the simultaneous requests for the proposed two-level BTD scheme.

Figure 10 shows the lifespan of each node with the number of hops for different schemes provided by Real-Time evaluated comparisons. Measures for the Delay requests with the corresponding Energy efficiency are presented in Figure 11. Results obtained in Figure 11 show that the network lifetime can be significantly prolonged when the 2nd level BTD is applied. By comparing the results obtained through real-time experiments for the scheme developed in [33] as well as with the periodic Sleep/Wake scheduling, the proposed scheme offers greater Energy-Efficiency, while it minimizes the delay per request.

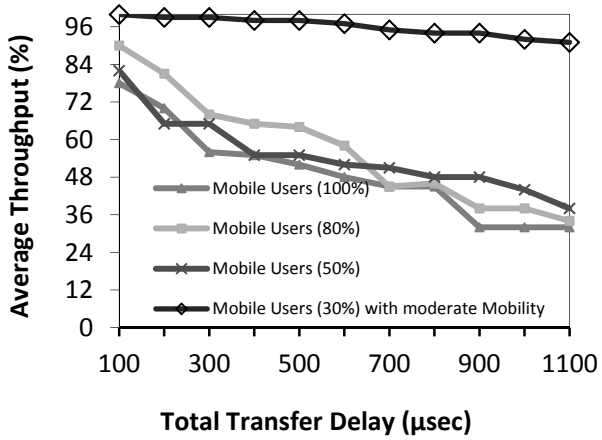


Figure 9. The Average Throughput with the Total Transfer Delay (µsec).

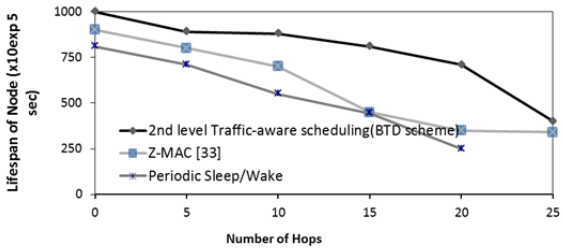


Figure 10. Lifespan of each node with the number of hops using different schemes under real-time evaluations.

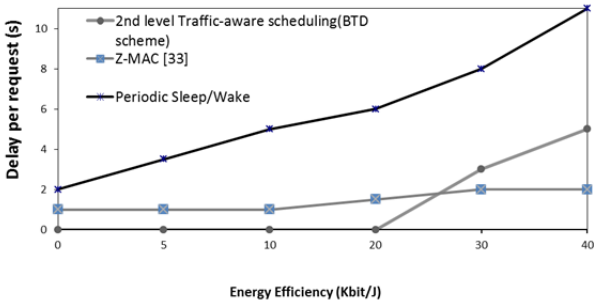


Figure 11. Evaluations for the delay requests with the corresponding Energy efficiency.

Figure 12 shows the fraction of the remaining Energy through time, in contrast to the comparison for different schemes and the associated evaluations during the real-time experimentation. As all schemes aim to reduce the Energy consumption, the proposed scheme behaves satisfactorily in contrast to the scheme developed in [33]. The End-to-End Latency with the number of requests for the users during real-time experimental evaluation is

shown in Figure 13, indicating the number of users that are utilized in the system in the presence of high mobility. Likewise, Figure 14 shows the respective Complementary Cumulative Distribution Function (CCDF or simply the tail distribution) with the Mean download Time for requests over a certain capacity. The later results were extracted in the presence of fading and no-fading communicating obstacles. Figure 15 shows the network lifetime with the number of Mobile Nodes for two schemes. It is important to notice that the network lifetime is significantly extended by using the proposed one and two level BTD for enabling Energy Conservation.

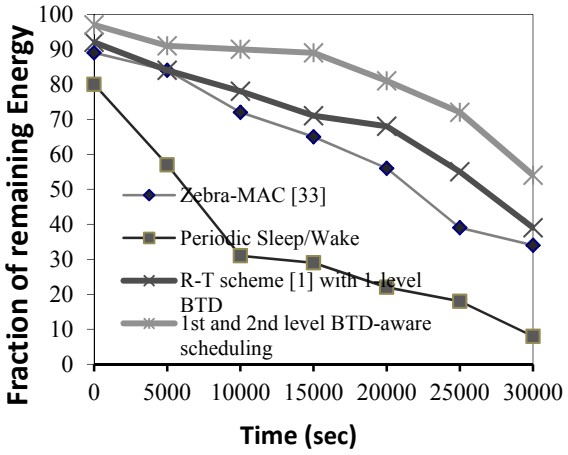


Figure 12. The fraction of the remaining Energy through time using real-time evaluation for different schemes.

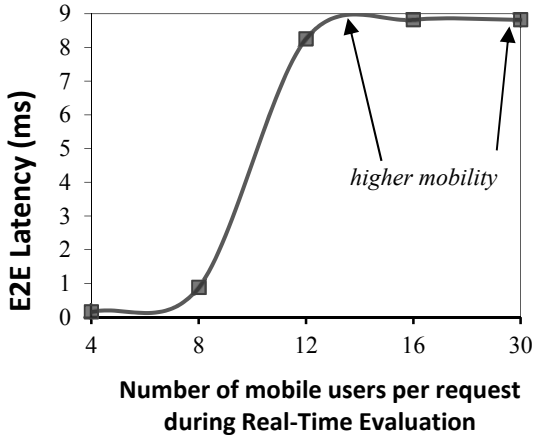


Figure 13. The End-to-End Latency with the number of requests for the users during real-time evaluation.

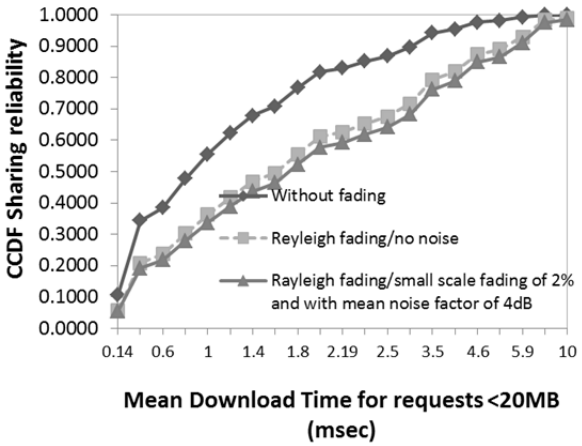


Figure 14. The CCDF for the Sharing Reliability with the Mean download Time for requests over a certain capacity.

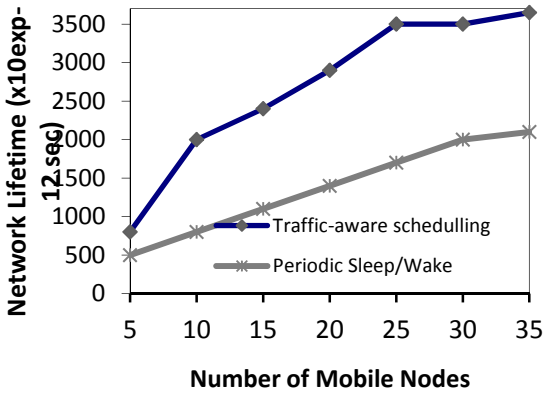


Figure 15. Network Lifetime with the Number of Mobile Nodes.

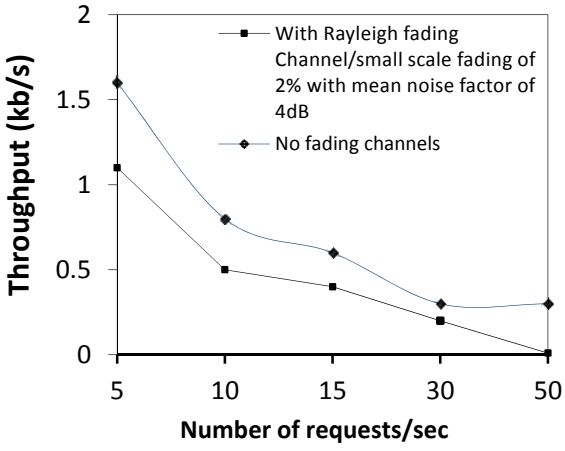


Figure 16. Throughput response of the system hosting the proposed scheme with the Number of requests for certain fading measures' characteristics.

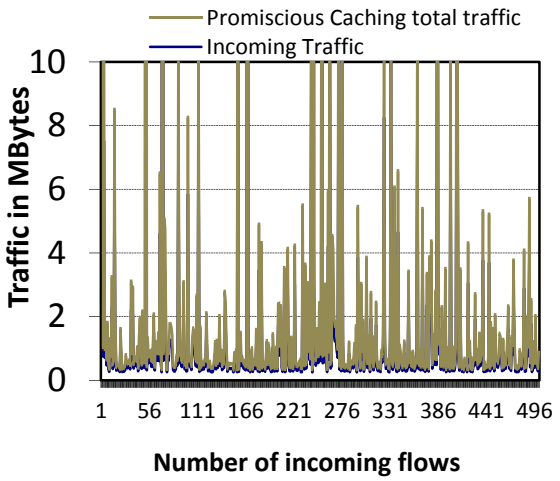


Figure 17. Number of incoming flows on each node in real-time with the incoming traffic in MB, for both the cached traffic and the traffic that a node is expecting to receive.

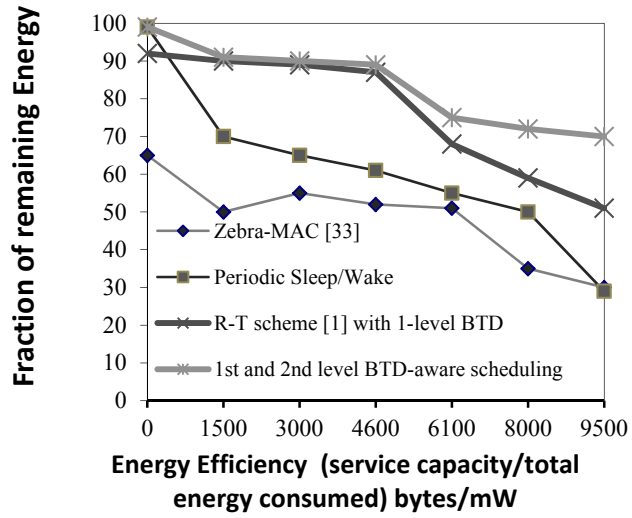


Figure 18. Energy Efficiency (service capacity/total energy consumed) bytes/mW with fraction of the remaining energy onto each node.

The Throughput response of the system under the evaluated two-level BTD in contrast to the number of requests for certain fading measures' characteristics is shown in Figure 16. The proposed scheme shows that in the case of Rayleigh fading characteristics the evaluated Throughput for the two-level BTD in contrast to the number of requests remains at relatively high levels when the number of requests is decreased. Contrarily when the number requests increases the Throughput drops as the Rayleigh fading takes place. This is somehow expected as in the presence of Rayleigh fading the packet transmission and service rate of the wireless channels drops dramatically. In Figure 17, the number of incoming flows on each node in real-time with the incoming traffic, for both the cached traffic and the traffic that a node is expecting to receive is presented. It is important to notice that the cached traffic is not negligible compared with the total volume of traffic that traverses the node. The promiscuous caching enables recoverability of the cached traffic that is forwarded to the destination node. It is obvious that the promiscuous caching enables high SDR rates as data can be recovered, however, it aggravates the energy conservation mechanism by prolonging the next active time duration of the node causing energy consumption.

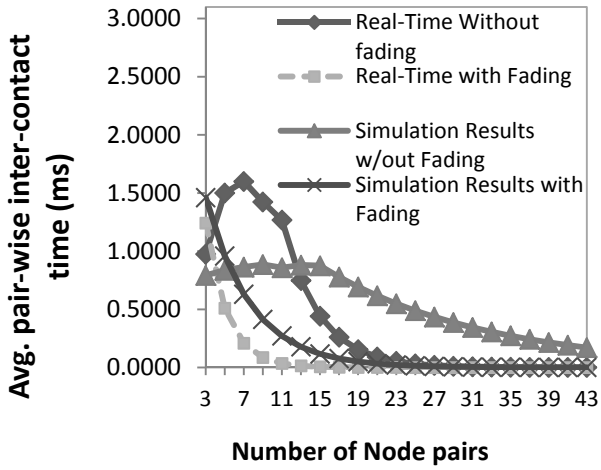


Figure 19. Avg. pair-wise inter-contact time (ms) with the number of pairs that nodes are communicating.

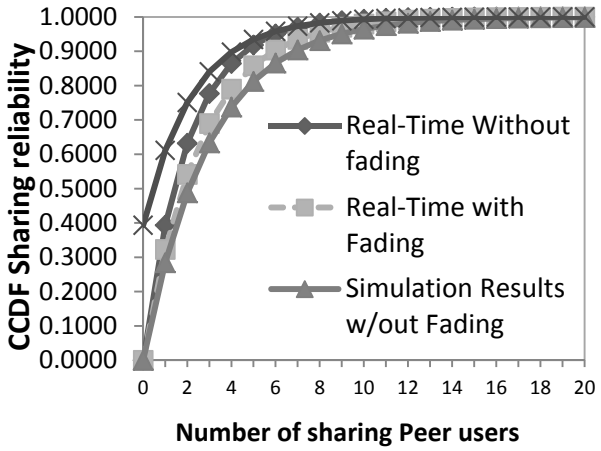


Figure 20. CCDF Sharing Reliability with the Number of sharing Peer-users.

The Energy Efficiency (bytes/mW) which is defined as the service capacity/total energy consumed as in Equation (14), with fraction of the remaining energy onto each node is shown in Figure 18, for 4 different schemes. The proposed framework is shown to have the higher remaining energy for each node in the system whereas, compared to the scheme in [1] it is shown to have an

optimized Energy-Efficiency behavior as it allows greater Energy-Efficiency in contrast to the remaining Energy of each node. Scheme adopted by [33] is shown to have the lowest Energy-Efficiency behavior compared also with the periodic sleep-wake schedules.

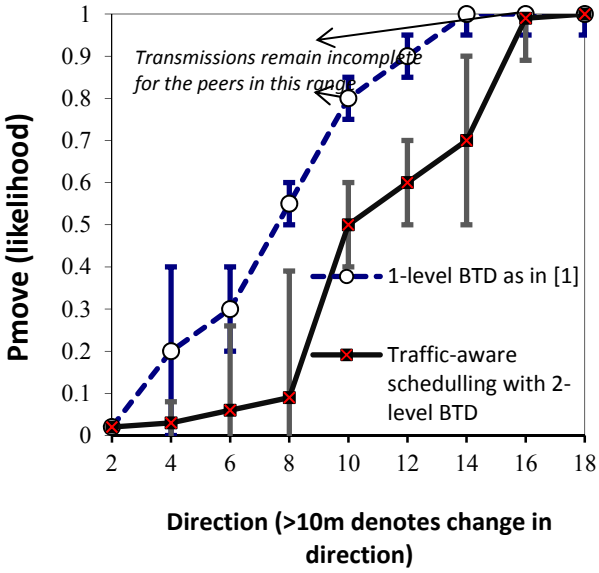


Figure 21. Using the likelihood of Fractional Random Walk (FRW) on a Weighted Graph compared for BTD and two-level BTD schemes.

The main reason that the work in [33] behaves in this way is probably the communication structure of Zerba-MAC which it still relies on time slots similar to TDMA-based solutions. Hence, each slot is tentatively assigned to a node whereas, it can be stolen by other nodes if it is not used by its owner. In Figure 19, the average pair-wise inter-contact time (msec) with the number of pairs of participating nodes (1-hop nodes that are supporting the promiscuous caching process) is presented.

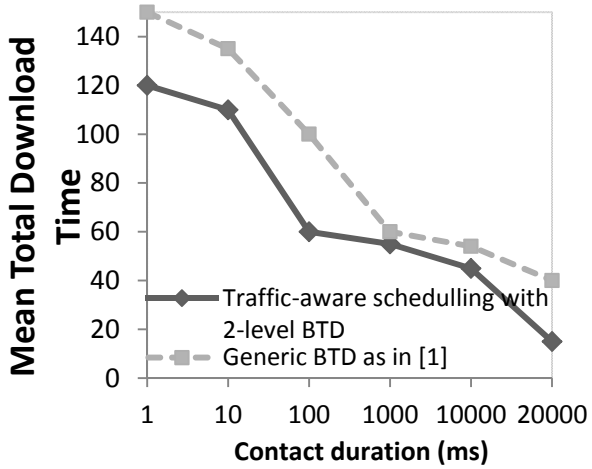


Figure 22. Mean total download time with the peer-contacts and their respective durations.

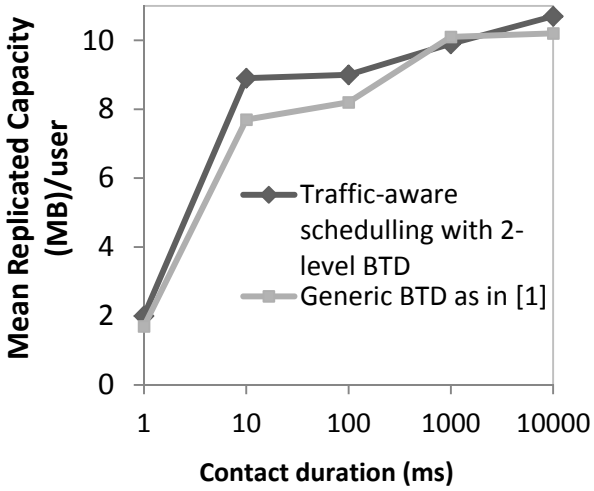


Figure 23. Mean replicated capacity in MB per user with the peer-contacts and their respective durations.

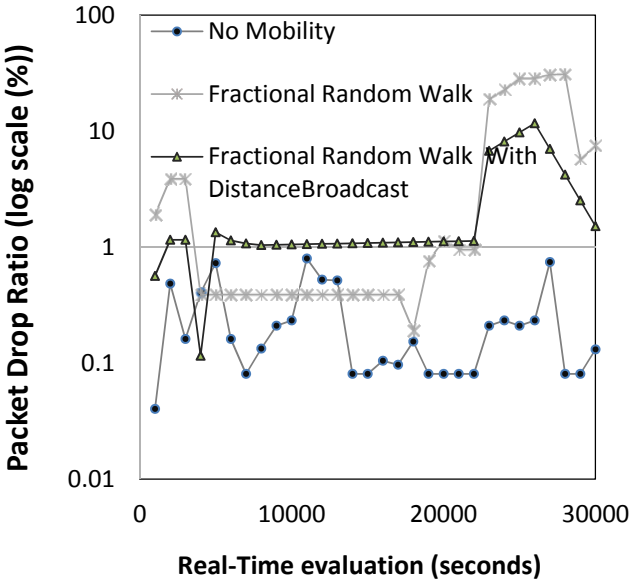


Figure 24. Packet drop ratio during real-time evaluation experimentation for two mobility models and for statically located nodes.

Figure 19 shows that a small number of nodes are directly communicating in the formed cluster within the evaluated area. This inter-contact time is considered very important metric, since participating nodes enable the multi-hop communication for a certain time duration and hence the establishment of End-to-End connectivity. Moreover, the latter enables us to evaluate the performance and robustness of the proposed scheme regarding the support of delay sensitive transmission handling and recoverability effectiveness. Results obtained and presented in Figure 20 show the CCDF Sharing Reliability with the Number of sharing Peer-users; and the

Average Throughput with the number of Nodes in the streaming zone. The results extracted for CCDF Sharing Reliability with the Number of sharing Peer-users were for both Simulation experiments and real-time estimations similar with minor and expected variations (within the confidence interval of 5-7% for the conducted simulation experiments as in [12] and [19]). It can be depicted that the simulated results and the results extracted through real-time traffic and experiments are experiencing 8-12% real values variations.

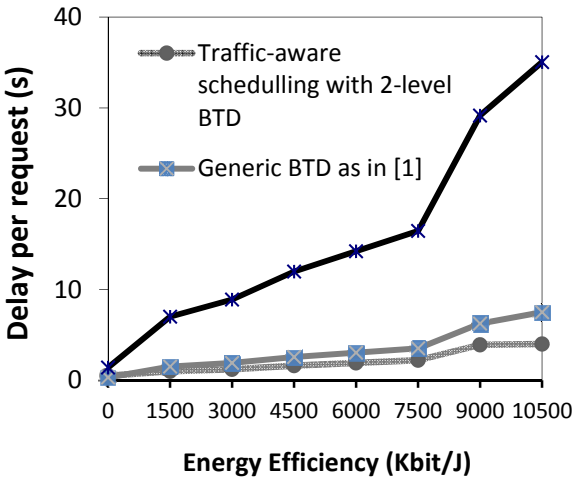


Figure 25. Delay requests in seconds with the Energy Efficiency (Eq. (14)-(15)) compared for three different schemes.

Figure 21 shows the comparison of the two schemes in reference with respect to the likelihood of Fractional Random Walk (FRW) on a Weighted Graph with the average distance for each one of the nodes. The transmissions are interrupted when the transmission distance of the node is increasing whereas, node in reference may find alternative paths to complete the transfers. Figure 22 shows the mean total download time with the contact time for the peers. It is important to mark-out that the proposed optimized scheme in this work outperforms the scheme proposed in [1] allowing less mean total download time and minimizing their respective durations.

The mean replicated capacity in MB per user with the peer-contacts and their respective durations is shown in Figure 23. During the replicated evaluation measures parameters such as the *promiscuous caching* threshold parameter σ_i introduced in [9] were used in order to avoid saturation of nodes [9] and capacity failure due to repeated caching to intermediate nodes. Packet drop ratio during real-time evaluation experimentation for two mobility models is shown in Figure 24. Figure 24 shows that statically located nodes are shown to have the least possible packet drops whereas the proposed scheme behaves satisfactorily when the Fractional Random Walk with Distance broadcast takes place.

The Energy Efficiency is obviously the most important measure for the performance and the energy effectiveness of the proposed scheme. In Figure 25, the delay requests in seconds with the Energy Efficiency (as presented in Equations (14) and (15) compared for three different schemes are shown. It is important to emphasize in the optimization of the Energy Efficiency measures exhibited by the proposed scheme whereas, at the same time to underline the energy differences by the scheme proposed in [1] as well as with the periodic schedules. Figure 25 shows the Energy Efficiency levels in contrast to the delay for each particular request (transfer). It can be depicted that the results extracted for the Energy Efficiency show that the proposed scheme hosting the two-level BTD outperforms the one-level and the periodic sleep, offering in almost all Energy Efficiency levels, minimized delay for the users' requests/transfers.

V. CONCLUSIONS AND FURTHER RESEARCH

This work proposes an adaptive traffic-based mechanism taking into consideration the active moments, and measures the incoming traffic by using the active-time comparisons. Moreover, the proposed two-level backward traffic-difference is presented. This work considers in real-time the one-level BTD scheme and compares the performance with the proposed two-level BTD evaluation hosted on wireless nodes. The proposed methodology takes place during the resource exchange process taking into account the promiscuous caching characteristics and the associated transmission aspects (delay and capacity characteristics) for establishing and maintaining reliable communication. The research framework proposes and provides comparative evaluation of a two-level backward estimation model for allocating the sleep-time duration to a certain node based on the volume of traffic that the node is expecting and admitting (traversed traffic). The proposed scheme aims to further enable Energy Conservation during the resource sharing process of the wireless nodes. The scheme uses the promiscuous caching mechanism for guaranteeing the requested resources and utilizes the one-level and based on delay criteria, the two-level BTD model for the Sleep time estimation. Based on the extracted real-time results and compared to the simulated and results extracted in [1], the designed model guarantees the end-to-end availability of requested resources while it reduces significantly the Energy Consumption. Moreover, the proposed two-level BTD maintains the requested scheduled transfers while, at the same time it increases the throughput response of the system. The evaluated results obtained in real-time show that this method uses optimally the network's and system's resources in terms of capacity and EC and offers high SDRs particularly in contrast with other similar existing Energy-efficient schemes as well as the one-level BTD scheme in [1].

Next steps and on-going work within the current research context will be the expansion of this model into a variable level-based BTD which will encompass a multi-level Markov Fractality Model (MFM). This fractal model

will potentially associate the different moments of the traffic activity and it will be able to extract the Sleep-time estimations for the nodes, in order to enable them to conserve Energy, while it will maintain a reliable resource sharing process on-the-move.

ACKNOWLEDGMENTS

(i) We would like to thank the postgraduate/research assistants Marios Charalambous at the Mobile Systems Lab@Unic (MOSys@University of Nicosia) for assisting the authors with the real-time experimentation.

(ii) We would like to thank the European FP7 ICT COST Action IC1105, 3D-ConTourNet-3D Content Creation, Coding and Transmission over Future Media Networks (WG-3), for the active support and cooperation.

REFERENCES

[1] C. X. Mavromoustakis, C. D. Dimitriou, and G. Mastorakis, "Using Real-Time Backward Traffic Difference Estimation for Energy Conservation in Wireless Devices", Proceedings of the IARIA Fourth International Conference on Advances in P2P Systems (AP2PS 2012), September 23-28, 2012 - Barcelona, Spain, pp. 18-23.

[2] A. Awad, O. Nasr, M. Khairy, "Energy-aware routing for delay-sensitive applications over wireless multihop mesh networks," 7th International Wireless Communications and Mobile Computing Conference (IWCMC), 2011, 4-8 July 2011, pp.1075,1080, doi: 10.1109/IWCMC.2011.5982690.

[3] M. Haahr, R. Cunningham, and Cahill, (1999) Supporting CORBA Applications in a Mobile Environment (ALICE). 5th International Conference on Mobile Computing and Networking (MobiCom), August 1999; ACM Press, Seattle, WA.

[4] M. Papadopouli, and H. Schulzrinne, "Design and Implementation of a Peer-to-Peer Data Dissemination and Prefetching Tool for Mobile Users", First New York Metro Area Networking Workshop, IBM T. J. Watson Research Center, Hawthorne, New York, 12 March 2002.

[5] K Stefanidis, G Koutrika, and E Pitoura, A survey on representation, composition and application of preferences in database systems, ACM Trans. Database Syst 36 (4) 2012.

[6] A Montresor, E Pitoura, A Datta, and S Voulgaris Topic 7: Peer to Peer Computing, Euro-Par 2012 Parallel Processing, 2012.

[7] F. Guo and J. Xu,"Research on Diffusion Strategy About Resource Index of MP2P", IJWMT, vol.2, no.2, pp.1-6, 2012.

[8] C. X. Mavromoustakis, "On the impact of caching and a model for storage-capacity measurements for energy conservation in asymmetrical wireless devices", IEEE Communication Society (COMSOC), 16th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2008), September 25 & 26 2008, "Dubrovnik", September 27, Split and Dubrovnik, pp. 243-247.

[9] C. X. Mavromoustakis and K. G. Zerfiridis, "On the diversity properties of wireless mobility with the user-centered temporal capacity awareness for EC in wireless devices". Proceedings of the Sixth IEEE International Conference on Wireless and Mobile Communications, ICWMC 2010, September 20-25, 2010-Valencia, Spain, pp.367-372.

[10] C. X. Mavromoustakis and H. D. Karatza, "Real time performance evaluation of asynchronous time division traffic-aware and delay-tolerant scheme in ad-hoc sensor networks", International Journal of Communication Systems (IJCS), Wiley, Volume 23 Issue 2 (February 2010), pp. 167-186.

[11] J. Yu and A. P. Petropulu, "Study of the effect of the wireless gateway on incoming self-similar traffic," IEEE Trans. Signal Proceesing, vol. 54, no. 10, pp. 3741-3758, 2006.

[12] C. X. Mavromoustakis and H. D. Karatza, "A tiered-based asynchronous scheduling scheme for delay constrained energy efficient connectivity in asymmetrical wireless devices", The Journal of Supercomputing, Springer USA, Volume 59, Issue 1, (2012), Page 61-82.

[13] Q. Cao, T. Abdelzaher, T. He, and J. Stankonic, "Towards optimal sleep scheduling in sensor networks for rare-event detection", the 4th international symposium on Information processing in sensor networks, 2005.

[14] X.-Y. Li, P.-J. Wan, and O. Frieder, "Coverage in wireless ad hoc sensor networks", IEEE Transactions on Computers, 2003.

[15] W. Ye, J. Heidemann and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks", 21st Conference of the IEEE Computer and Communications Societies (INFOCOM), 2002.

[16] T. Dam and K. Langendoen, "An Adaptive Energy-efficient MAC Protocol for Wireless Sensor Networks,", the 1st international conference on Embedded networked sensor systems, 2003.

[17] H. Cunqing and P. Y. Tak-Shink," Asynchronous Random Sleeping for Sensor Networks", ACM Transactions on Sensor Networks (TOSN), 2007.

[18] I. Jawhar, J. Wu, and P. Agrawal, Resource Scheduling in Wireless Networks Using Directional Antennas, appears in: IEEE Transactions on Parallel and Distributed Systems, Sept. 2010, Volume: 21 Issue:9, pp. 1240 - 1253.

[19] C. X. Mavromoustakis, "Synchronized Cooperative Schedules for collaborative resource availability using population-based algorithm", Simulation Practice and Theory Journal, Elsevier, Volume 19, Issue 2, February 2011, pp. 762-776.

[20] Q. Cao, T. Abdelzaher, T. He, and J. Stankonic, "Towards optimal sleep scheduling in sensor networks for rare-event detection", the 4th international symposium on Information processing in sensor networks, 2005.

[21] Y. Liu and Z. Wang, Maximizing energy utilization routing scheme in wireless sensor networks based on minimum hops algorithm, Computers & Electrical Engineering, Elsevier, Volume 38, Issue 3, May 2012, Pages 703–721.

[22] M. Rashid, M. Mahbub, and C. Hong, "Energy Conserving Passive Clustering for Efficient Routing in Wireless Sensor Network," Advanced Communication Technology, The 9th International Conference on , vol.2, no., pp.982,986, 12-14 Feb. 2007, doi: 10.1109/ICACT.2007.358523.

[23] A. Feldmann, A. Gilbert, P. Huang, and W. Willinger, Dynamics of IP traffic: A study of the role of variability and the impact of control, 1999, ACM SIGCOMM pp. 301-313.

[24] K. Vijayaraghavan and R. Rajamani, "Active control based energy harvesting for battery-less wireless traffic sensors: theory and experiments," in Proceedings of the American Control Conference (ACC '08), pp. 4579–4584, June 2008.

[25] Y. K. Tan, K. Y. Hoe, and S. K. Panda, "Energy harvesting using piezoelectric igniter for self-powered radio frequency (RF) wireless sensors," in Proceedings of the IEEE International Conference on Industrial Technology (ICIT '06), pp. 1711–1716, Mumbai, India, December 2006.

[26] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor," in Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03), pp. 1–13, Los Angeles, Calif, USA, November 2003.

[27] http://www.xbow.com/pdf/ containing the specifications for the associated Crossbow Technology, Industry's First End-to-End, Low-Power, Wireless Sensor Network Solution for Security, Monitoring, and Tracking Applications, last accessed on 20/2/2013.

[28] A. Boukerche, R. Nelem-Pazzi, and A. Martirosyan, Energy aware and cluster-based routing protocols for large-scale ambient sensor networks, Proceedings Ambi-Sys '08 Proceedings of the 1st international conference on Ambient media and systems 2008, pp. 306-311.

[29] C. Mavromoustakis, "Optimizing the End-to-End Opportunistic Resource Sharing using Social Mobility", To appear in the First International Conference on Intelligent Systems and Applications, INTELLI 2012, April 29-May 4, 2012 - Chamonix / Mont Blanc, France.

[30] O. Ibe, Markov Processes for Stochastic Modeling, ISBN-10: 0123744512, Academic Press (September 16, 2008), pp.512.

[31] C. X. Mavromoustakis and H. D. Karatza, "Performance evaluation of opportunistic resource sharing scheme using socially-oriented outsourcing in wireless devices", The Computer Journal, Volume 56, Number 2, February 2013, pp. 184-197.

[32] A.L. Barabási and R. Albert. Emergence of scaling in random networks. Science, 286(5439):509–512, 1999.

[33] A. Rhee, M. Warrier, J. Aia, and M. Sichitiu, Z-MAC: a hybrid MAC for wireless sensor networks in IEEE/ACM Transactions on Networking vol. 16, no. 3, pp. 511-524, 2008.

[34] X. Zhuang and S. Pande, "A scalable priority queue architecture for high speed network processing," in Proc. 25th IEEE International Conf. Computer Commun. (INFOCOM'06), 2006, pp. 1-12.

[35] http://www.lynxmotion.com/c-155-rc-combo-kit.aspx, last accessed on March 13th 2013.

.

A Quasi-Random Multirate Loss Model Supporting Elastic and Adaptive Traffic under the Bandwidth Reservation Policy

Ioannis D. Moscholios*, John S. Vardakas†, Michael D. Logothetis‡, and Michael N. Koukias‡
*Dept. of Informatics & Telecommunications, University of Peloponnese, 221 00 Tripolis, Greece.
Email: idm@uop.gr
†Iquadrat, Barcelona, Spain.
Email : jvardakas@iquadrat.com
‡Dept. of Electrical and Computer Engineering, University of Patras, 265 04 Patras, Greece.
Email: {mlogo, mkoukias}@upatras.gr

***Abstract*—In this paper, we propose a multirate teletraffic loss model of a single link that accommodates elastic and adaptive services whose calls come from a finite traffic-source population. This call arrival process is known as a quasi-random process and is used in traffic modelling when the number of users who generate traffic is relatively small compared to the system capacity. In-service elastic and adaptive calls can tolerate bandwidth compression by extending their remaining service-time (elastic calls) or not (adaptive calls). In this loss system, we study the effect of the bandwidth reservation policy on time congestion probabilities, call congestion probabilities and link utilization. The bandwidth reservation policy is considered when a certain quality of service for each service-class is required and is essential to be guaranteed. The proposed model does not have a product form solution, and therefore we propose approximate but recursive formulas for the efficient calculation of the above mentioned performance measures. The accuracy and consistency of the proposed model are verified by simulation and is found to be quite satisfactory. Finally, we generalize the proposed model to include calls from both finite and infinite number of traffic sources.**

Keywords-Markov chain; quasi-random process; elastic-adaptive traffic; recursive formula; time-call congestion; bandwidth reservation.

I. INTRODUCTION

IN contemporary communication networks, the traffic environment is composed mostly of multirate services of elastic and adaptive traffic. The co-existence of this type of services makes the call-level performance analysis and evaluation of modern telecom networks much more complicated and challenging. It is therefore essential to have proper multirate teletraffic loss models, for the call-level QoS assessment of such networks [1]. Based on analytical tools, telecom engineers can include various services in the network according to their offered traffic-load, decide on proper network dimensioning and avoid link over-dimensioning [2].

In [1], we present a multirate loss model for elastic and adaptive services with finite traffic-source population. By the term elastic traffic (e.g., file transfer), we refer to in-service calls that have the ability to compress their bandwidth, while simultaneously expanding their service time. On the other hand, adaptive traffic refers to in-service calls that tolerate bandwidth compression, but their service time does not alter (e.g., adaptive video) [3]. In both cases, we assume that bandwidth compression is permitted down to a minimum value. As far as the consideration of traffic sources with finite population is concerned, it results in a quasi-random call arrival process, which is, in many cases, a more realistic consideration than a random (Poisson) process (where infinite population of traffic sources is assumed). As the Markov chain analysis shows, the existence of the bandwidth compression/expansion mechanism destroys the Markov chain reversibility, and because of this, the model has no product form solution. Therefore, we propose approximate but recursive formulas for the efficient calculation of the call-level performance metrics, such as time and call congestion probabilities and link utilization. The consistency and the accuracy of the model are verified through simulation and found to be quite satisfactory.

In this paper, we extend [1], by studying the effect of the Bandwidth Reservation (BR) policy on time and call congestion probabilities, as well as on link utilization. The BR policy can achieve the equalization of call blocking probabilities among calls of different service-classes, or guarantee a certain QoS for each service-class. For instance, to equalize blocking probabilities between different service-classes, the BR policy ensures greater link bandwidth to high speed service-classes. Applications of the BR policy in wired (e.g., [4]-[6]), wireless (e.g., [7]-[9]) and optical networks (e.g., [10], [11]) show the policy's significant role in teletraffic engineering. As an example, in wireless networks the BR policy can ensure certain QoS for handoff traffic, while, it is worth mentioning that in optical networks, the term "bandwidth reservation" refers to "wavelength reservation" [10].

The Markov chain analysis is extensively used in the call-level analysis of communication networks. Contrary to the analysis of the complete sharing policy where the stationary probabilities have a product form solution, the BR policy cannot be analysed by the use of a product form solution. This is because one-way transitions appear in the state space of the Markov chain, which destroy the reversibility of the Markov chain [12]. Therefore, the reason why the proposed model in this paper does not have a product form solution is twofold:

not only the existence of the bandwidth compression/expansion mechanism but also the existence of the BR policy. Because of the absence of a product form solution we resort to an approximate solution, and propose a recursive formula for the efficient calculation of the link occupancy distribution. This formula simplifies the determination of all performance measures. We have evaluated the accuracy and consistency of the proposed model through simulation, and found them to be quite satisfactory. Herein, we name the model of [1], Extended Finite - Erlang Multirate Loss Model (EF-EMLM), because it is based on the classical EMLM (also known as Kaufman-Roberts model – more details are referred to the next Section) [13], [14]. Hence, the proposed new model is named EF-EMLM/BR. Furthermore, we generalize the EF-EMLM/BR to include calls from both finite and infinite number of traffic sources.

Potential applications of the proposed new model are mostly in the area of cellular networks, where calls come from finite sources and their bandwidth can be compressed (e.g., [15]–[20]). More precisely, a Wideband Code Division Multiple Access (W-CDMA) network, like Universal Mobile Telecommunications System (UMTS), supports not only streaming traffic of voice service, but also data traffic of an elastic or adaptive type (transferring messages or images) that can tolerate bandwidth compression. A single base station of this network can be modelled as a multirate loss system. The number of users in a cell is rather realistic to be considered finite, due to the limited coverage of a cell; this is even more realistic in the case of microcells. The BR policy can be applied to the system in order to reserve a part of the cell resources especially for handoff incoming traffic, which has to be serviced with a higher priority than the traffic originated inside the cell.

This paper is organised as follows: Section II contains related work. In Section III, we: a) present the basic assumptions and the call admission control, b) show the recursive formula for the link occupancy distribution and c) provide formulas for the various performance measures of the EF-EMLM. In Section IV, we: a) consider the application of the BR policy in the EF-EMLM, b) show the recursive formula for the link occupancy distribution, c) show how the EF-EMLM is related to other multirate loss models and d) provide formulas for the calculation of various performance measures of the EF-EMLM/BR. In Section V, we provide numerical results whereby the new model is compared to existing models and evaluated through simulation results. In Section VI, we generalize the proposed model to include calls from both finite and infinite number of sources. We conclude in Section VII. In Appendix A, we prove the recursive formula for the link occupancy distribution. Finally, we tabulate (as Appendix B) all the symbols used in this paper.

II. Related Work

Multirate teletraffic loss models of a single link that accommodates elastic and adaptive calls have been proposed in [3], [21]-[24]. In [3], the call arrival process is Poisson. A new call is accepted in the link with its peak bandwidth requirement when the occupied link bandwidth together with the peak bandwidth of that call does not exceed the capacity of the link. Otherwise, the new call is accepted in the link by compressing its peak-bandwidth, as well as the bandwidth of all in-service calls (of all service-classes). Call blocking occurs when the minimum bandwidth requirement of a call (achieved after the maximum bandwidth compression) is higher than the link's available bandwidth. The minimum bandwidth requirement of a call is a proportion of its peak-bandwidth; this proportion is common to all service-classes. When an in-service call departs from the system, then the remaining in-service calls, whose bandwidth has been compressed, expand their bandwidth in proportion to their peak-bandwidth requirement. The Markov chain analysis of this model shows that the bandwidth compression/expansion mechanism destroys the Markov chain reversibility, and therefore the model has no product form solution. However, according to [3], a reversible Markov chain that describes the model in an approximate way does exist, and leads to a recursive formula for the determination of link occupancy distribution and, consequently, call blocking probabilities and link utilization. This formula resembles the classical Kaufman-Roberts formula used in the EMLM, where Poisson arriving calls of different service-classes have fixed bandwidth requirements (stream traffic), and compete for the available link bandwidth under the complete sharing policy [13], [14]; thus, we name the model of [3], Extended EMLM (E-EMLM). In [21], the E-EMLM is extended to include retrials, i.e., blocked calls may retry one or more times to be serviced with reduced bandwidth. In [22], new calls, upon their arrival, may reduce their bandwidth according to the occupied link bandwidth. In [23], [24], the E-EMLM is further extended to include the Batched Poisson call arrival process, which is used to approximate arrival processes that are more "peaked"and "bursty" than the Poisson process.

In [1], we consider an extension of the E-EMLM, the EF-EMLM, whereby calls arrive in the link according to a quasi-random process [25]. The latter is smoother than the Poisson process and is used in traffic modelling when the number of users (sources) who generate traffic is finite. As an application example of the quasi-random process one may think of a microcell in a cellular network, where the number of users roaming in the cell's vicinity can be considered finite [19], [26]. Recently, a multirate loss model that includes Poisson calls of stream, elastic and adaptive traffic has been proposed in [20]; however, the presence of stream traffic prohibits the recursive calculation of link occupancy distribution or the other call-level performance measures.

III. The Extended Finite EMLM (EF-EMLM)

In the following subsections, we present basic assumptions for call admission control, the recursive calculation of the link occupancy distribution and the consequent calculation of the call-level performance measures of the EF-EMLM, under the complete sharing policy (without QoS guarantee).

A. Notation, basic assumptions and call admission

We study a link of capacity C bandwidth units that accommodates elastic and adaptive calls of K different service-

classes. Let K_e and K_a be the set of elastic and adaptive service-classes ($K_e + K_a = K$), respectively. Calls of service-class k ($k = 1, \ldots, K$) come from a finite source population N_k and request b_k bandwidth units (peak-bandwidth requirement). The mean arrival rate of service-class k idle sources is $\lambda_k = (N_k - n_k)v_k$ where n_k is the number of in-service calls and v_k is the arrival rate per idle source. This call arrival process is a quasi-random process [25]. If $N_k \to \infty$ for $k = 1, \ldots, K$ and the total offered traffic-load remains constant, then a Poisson process arises. To introduce bandwidth compression, the occupied link bandwidth j may virtually exceed C up to T bandwidth units.

The description of call admission is based on a new service-class k call that arrives in the system when the occupied link bandwidth is j bandwidth units. Then:

i) If $j + b_k \leq C$, the call is accepted in the system with b_k bandwidth units for an exponentially distributed service time with mean μ_k^{-1}.

ii) If $j + b_k > T$, the call is blocked and lost.

iii) If $T \geq j + b_k > C$, the call is accepted in the system by compressing its peak-bandwidth requirement, as well as the assigned bandwidth of all in-service calls (of all service-classes). After compression has taken place, all calls share the C bandwidth units in proportion to their peak-bandwidth requirement, while the link operates at its full capacity C. This is the processor sharing discipline [27], [28].

When $T \geq j + b_k > C$, the compressed bandwidth b_k^{comp} of the newly accepted service-class k call, is calculated by:

$$b_k^{\text{comp}} = r b_k = \frac{C}{j'} b_k \tag{1}$$

where $r = \frac{C}{j'}$ denotes the compression factor and $j' = j + b_k$.

Since $j = \sum_{k=1}^{K} n_k b_k = \mathbf{nb}$, $\mathbf{n} = (n_1, n_2, ..., n_K)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_K)$, the values of r are expressed by $r \equiv r(\mathbf{n}) = \frac{C}{\mathbf{nb} + b_k}$. The bandwidth of all in-service calls is also compressed by the same factor $r(\mathbf{n})$ and becomes equal to $b_i^{\text{comp}} = \frac{C}{j'} \cdot b_i$ for $i = 1, \ldots, K$. After bandwidth compression, the occupied link bandwidth is $j = C$. All adaptive calls do not alter their service time. On the other hand, all elastic calls increase their service time so that the product *service time* by *bandwidth* remains constant. The minimum bandwidth that a call of service-class k ($k = 1, \ldots, K$) tolerates, is:

$$b_{k,\min}^{\text{comp}} = r_{\min} b_k = \frac{C}{T} b_k \tag{2}$$

where $r_{\min} = \frac{C}{T}$ is the minimum proportion of the required peak-bandwidth and is common to all calls.

When an in-service call of service-class k, with bandwidth b_k^{comp}, departs from the system, then the remaining in-service calls of each service-class i ($i = 1, \ldots, K$), expand their bandwidth to b_i^{expan}, in proportion to their peak-bandwidth b_i:

$$b_i^{\text{expan}} = \min \left(b_i,\ b_i^{\text{comp}} + \frac{b_i}{\sum_{k=1}^{K} n_k b_k} b_k^{\text{comp}} \right) \tag{3}$$

To illustrate the previous bandwidth compression mechanism consider the following simple example. Let $C = 3$ bandwidth units, $T = 5$ bandwidth units, $K = 2$ service-classes, $b_1 = 1$ bandwidth unit, $b_2 = 2$ bandwidth units, $N_1 = N_2 = 10$ sources, $v_1 = v_2 = 0.1$ and $\mu_1^{-1} = \mu_2^{-1} = 1$ time unit. The first service-class is elastic while the second service-class is adaptive. The permissible states $\mathbf{n} = (n_1, n_2)$ of the system are 12; they are presented in Table I together with the occupied link bandwidth, $j = n_1 b_1 + n_2 b_2$, before and after compression has been applied. Note that compression is applied if $T \geq j > C$ (bold values of the 3^{rd} column of Table I). After compression has been applied, we have that $j = C$ (bold values of the 4^{th} column of Table I). For example, assume that a new 2^{nd} service-class call arrives while the system is in state $(n_1, n_2) = (1, 1)$ and $j = C = 3$ bandwidth units. The new call is accepted in the system, since $j' = j + b_2 = T = 5$ bandwidth units, after bandwidth compression has been applied to all calls (new and in-service calls). The new state of the system is now $(n_1, n_2) = (1, 2)$. In this state, and based on (2), calls of the 1^{st} and 2^{nd} service-class compress their bandwidth to the following values:

$$b_{1,\min}^{\text{comp}} = r_{\min} b_1 = \tfrac{3}{5} b_1 = 0.6 \ , \ b_{2,\min}^{\text{comp}} = r_{\min} b_2 = \tfrac{3}{5} b_2 = 1.2$$

so that $j = n_1 b_{1,\min}^{\text{comp}} + n_2 b_{2,\min}^{\text{comp}} = 0.6 + 2.4 = 3 = C$. The value of μ_1^{-1} becomes $\frac{\mu_1^{-1}}{r_{\min}}$ so that $b_1 \mu_1^{-1}$ remains constant, while the value of μ_2^{-1} does not alter.

Consider now that the system is in state $(n_1, n_2) = (1, 2)$ and a 2^{nd} service-class call departs from the system. Then, its assigned bandwidth $b_{2,\min}^{\text{comp}} = 1.2$ is shared to the remaining calls in proportion to their peak-bandwidth requirement. Thus, in the new state $(n_1, n_2) = (1, 1)$ the 1^{st} service-class call expands its bandwidth to $b_1 = 1$ bandwidth unit and the 2^{nd} service-class call to $b_2 = 2$ bandwidth units. Thus, $j = n_1 b_1 + n_2 b_2 = C = 3$ bandwidth units. Furthermore, the service time of elastic calls only is decreased to $\mu_1^{-1} = 1$ time unit.

In Fig. 1, we present the state transition diagram of this example. If we consider the four adjacent states (n_1, n_2) : $(2, 0), (2, 1), (3, 1)$ and $(3, 0)$ and apply the Kolmogorov' s criterion (*flow clockwise* = *flow counter-clockwise*) [27], it is obvious that this criterion does not hold. This means that the Markov chain is not reversible. Generally speaking, the bandwidth compression mechanism destroys reversibility in the proposed model and therefore no PFS exists. To circumvent this problem, we use, in the following subsection, state dependent multipliers per service-class k, $\phi_k(\mathbf{n})$, which have a similar role with $r(\mathbf{n})$ and lead to a reversible Markov chain.

B. Determination of link occupancy distribution

Let $\mathbf{\Omega}$ be the system's state space $\mathbf{\Omega} = \{\mathbf{n} : 0 \leq \mathbf{nb} \leq T\}$. The fact that the system cannot be described by a reversible Markov chain means that local balance does not exist between adjacent states of $\mathbf{\Omega}$. Therefore, the steady-state distribution $P(\mathbf{n})$ does not have a product form solution. To derive an approximate but recursive formula for

TABLE I. STATE SPACE AND OCCUPIED LINK BANDWIDTH

n_1	n_2	j (before compression) $0 \le j \le T$	j (after compression) $0 \le j \le C$
0	0	0	0
0	1	2	2
0	2	**4**	**3**
1	0	1	1
1	1	3	3
1	2	**5**	**3**
2	0	2	2
2	1	**4**	**3**
3	0	3	3
3	1	**5**	**3**
4	0	**4**	**3**
5	0	**5**	**3**

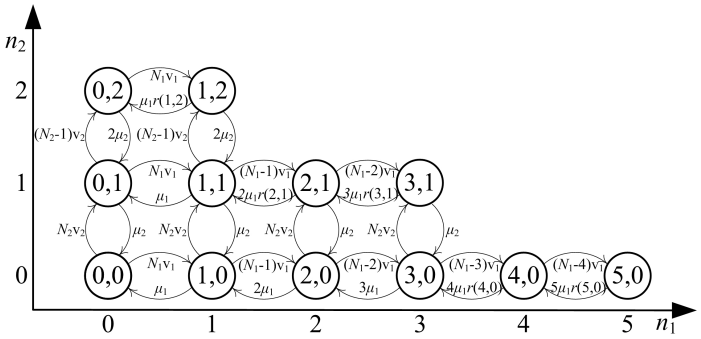


Figure 1. State transition diagram of the tutorial example.

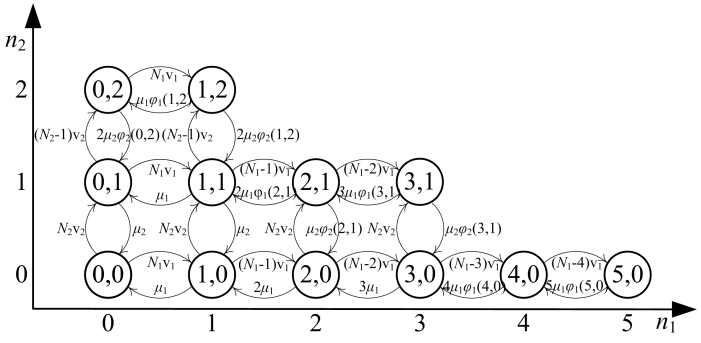


Figure 2. Modified state transition diagram of the tutorial example.

the efficient calculation of the link occupancy distribution, $G(j),\ j = 0, 1, \ldots, T$, we construct a reversible Markov chain that approximates the system by using state multipliers for all states $\mathbf{n} \in \mathbf{\Omega}$. The local balance equations between the adjacent states $\mathbf{n}_k^{-1} = (n_1, n_2, \ldots, n_k - 1, \ldots, n_K)$ and $\mathbf{n} = (n_1, n_2, \ldots, n_k, \ldots, n_K)$ have the form:

$$P(\mathbf{n}_k^{-1})(N_k - n_k + 1)v_k = P(\mathbf{n})\phi_k(\mathbf{n})\mu_k n_k\,,\ k \in K_e \quad (4)$$

$$P(\mathbf{n}_k^{-1})(N_k - n_k + 1)v_k = P(\mathbf{n})\phi_k(\mathbf{n})\mu_k n_k\,,\ k \in K_a \quad (5)$$

where $\phi_k(\mathbf{n})$ is a state-dependent multiplier and is defined as:

$$\phi_k(\mathbf{n}) = \begin{cases} 1 & ,\ \text{when } \mathbf{nb} \le \mathrm{C} \text{ and } \mathbf{n} \in \mathbf{\Omega} \\ \frac{x(\mathbf{n}_k^{-1})}{x(\mathbf{n})} & ,\ \text{when } C < \mathbf{nb} \le T \text{ and } \mathbf{n} \in \mathbf{\Omega} \\ 0 & ,\ \text{otherwise} \end{cases} \quad (6)$$

and

$$x(\mathbf{n}) = \begin{cases} 1 & ,\ \text{when } \mathbf{nb} \le C,\ \mathbf{n} \in \mathbf{\Omega} \\ \frac{1}{C}\left(\sum\limits_{k \in K_e} n_k b_k x(\mathbf{n}_k^{-1}) + r(\mathbf{n}) \sum\limits_{k \in K_a} n_k b_k x(\mathbf{n}_k^{-1})\right) & \\ & ,\ \text{when } C < \mathbf{nb} \le T,\ \mathbf{n} \in \mathbf{\Omega} \\ 0 & ,\ \text{otherwise} \end{cases} \quad (7)$$

where $r(\mathbf{n}) = \frac{C}{\mathbf{nb}}$.

When $C < \mathbf{nb} \le T$ and $\mathbf{n} \in \mathbf{\Omega}$ the bandwidth of all in-service calls is compressed by a factor $\phi_k(\mathbf{n})$ so that:

$$\sum_{k \in K_e} n_k b_k^{\text{comp}} + \sum_{k \in K_a} n_k b_k^{\text{comp}} = C \quad (8)$$

To derive (7), we keep the product *service time* by *bandwidth* of service-class k calls (elastic or adaptive) in state $\mathbf{n}$ of the irreversible Markov chain equal to the corresponding product in the same state $\mathbf{n}$ of the reversible Markov chain. This means that:

$$\frac{b_k r(\mathbf{n})}{\mu_k r(\mathbf{n})} = \frac{b_k^{\text{comp}}}{\mu_k \phi_k(\mathbf{n})} \Rightarrow b_k^{\text{comp}} = b_k \phi_k(\mathbf{n}),\ k \in K_e \quad (9)$$

and

$$\frac{b_k r(\mathbf{n})}{\mu_k} = \frac{b_k^{\text{comp}}}{\mu_k \phi_k(\mathbf{n})} \Rightarrow b_k^{\text{comp}} = b_k \phi_k(\mathbf{n}) r(\mathbf{n}),\ k \in K_a \quad (10)$$

Equation (7) results by substituting (9), (10) and (6), into (8).

Figure 2 shows the modified state transition diagram of our tutorial example, due to the introduction of the state-dependent multipliers $\phi_k(\mathbf{n})$'s. Now, if we consider again the four adjacent states $(n_1, n_2) : (2,0), (2,1), (3,1)$ and $(3,0)$, the Kolmogorov' s criterion (*flow clockwise = flow counter-clockwise*) holds, since:

$$N_2 v_2 (N_1 - 2) v_1 \mu_2 \phi_2(3,1) 3\mu_1 =$$
$$= (N_1 - 2) v_1 N_2 v_2 3\mu_1 \phi_1(3,1) \mu_2 \phi_2(2,1) \Rightarrow$$
$$\phi_2(3,1) = \phi_1(3,1)\phi_2(2,1) \Rightarrow$$
$$x(3,0) = x(2,0) = 1$$

In order to prove a recursive formula for the calculation of $G(j)$'s, we consider two cases: i) states where $0 \leq j \leq C$ (bandwidth compression does not occur) and ii) states where $C < j \leq T$ (bandwidth compression occurs).

When $0 \leq j \leq C$, then $\phi_k(\mathbf{n}) = 1$ and based on (4) and (5), it is proved that [29]:

$$G(j) = \begin{cases} 1 & \text{for } j = 0 \\ \frac{1}{j} \sum\limits_{k \in K} (N_k - n_k + 1)\alpha_k b_k G(j - b_k) & \\ & \text{for } j = 1, \ldots, C \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where: $\alpha_k = \frac{v_k}{\mu_k}$ is the offered traffic-load (in erl) per idle source of service-class k.

Note that (11) comes in fact from the Engset multirate loss model [29], in which calls come from finite sources and compete for the available link bandwidth under the complete sharing policy. The name "Engset" is justified by the fact that for $K = 1$ service-class, (11) can be used to calculate time congestion probabilities (by taking the sum of $G(j)$'s over all blocking states) whose values are the same with the classical Engset formula [30].

When $C < j \leq T$, it can be proved (see Appendix A) that:

$$\begin{aligned} &\sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k G(j - b_k) + \\ &\frac{C}{j} \sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k G(j - b_k) = C G(j) \end{aligned} \quad (12)$$

The combination of (11) and (12) gives the approximate recursive formula of $G(j)$'s, when $1 \leq j \leq T$:

$$G(j) = \begin{cases} 1 & \text{for } j = 0 \\ \frac{1}{\min(j,C)} \sum\limits_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k G(j - b_k) & \\ + \frac{1}{j} \sum\limits_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k G(j - b_k) & \\ & \text{for } j = 1, \ldots, T \\ 0 & \text{for } j < 0 \end{cases} \quad (13)$$

C. Relation of the EF-EMLM to other multirate loss models

(a) When $N_k \to \infty$ for $k = 1, \ldots, K$ and the total offered traffic-load remains constant, then the call arrival process is Poisson and the E-EMLM results. In that case, the formula of $G(j)$'s is given by [3]:

$$G(j) = \begin{cases} 1 & \text{for } j = 0 \\ \frac{1}{\min(j,C)} \sum\limits_{k \in K_e} \alpha_{k,\inf} b_k G(j - b_k) + & \\ \frac{1}{j} \sum\limits_{k \in K_a} \alpha_{k,\inf} b_k G(j - b_k) & \\ & \text{for } j = 1, \ldots, T \\ 0 & \text{for } j < 0 \end{cases} \quad (14)$$

where $\alpha_{k,\inf} = \frac{\lambda_{k,\inf}}{\mu_k}$ (in erl) and $\lambda_{k,\inf}$ is the arrival rate of calls of service-class k.

(b) When $N_k \to \infty$ for $k = 1, \ldots, K$ and the total offered traffic-load remains constant, and $T = C$, then no bandwidth compression is allowed and the classical EMLM arises. In that case, the formula of $G(j)$'s is given by the Kaufman-Roberts recursion [13], [14]:

$$G(j) = \begin{cases} 1 & \text{for } j = 0 \\ \frac{1}{j} \sum\limits_{k=1}^{K} \alpha_{k,\inf} b_k G(j - b_k) & \text{for } j = 1, \ldots, C \\ 0 & \text{for } j < 0 \end{cases} \quad (15)$$

D. Determination of various performance measures

The calculation of $G(j)$'s in (13) requires the value of n_k which is unknown. In other finite multirate loss models (e.g., [29]–[31]) there exist calculation methods for the determination of n_k in each state j through the use of an equivalent stochastic system, with the same traffic description parameters and exactly the same set of states. However, the state space determination of the equivalent system is complex, especially for large capacity systems that serve many service-classes. Thus, we avoid such methods and approximate n_k in state j, $n_k(j)$, as the mean number of service-class k calls in state j, $y_k(j)$, when Poisson arrivals are considered, i.e., $n_k(j) \approx y_k(j)$. Such approximations are common in the literature and induce little error (e.g., [32]–[34]). The values of $y_k(j)$ are given by (16) and (17) in the case of elastic and adaptive service-classes, respectively [3]:

$$\begin{aligned} &y_k(j)G(j) = \\ &= \tfrac{1}{\min(C,j)} \alpha_{k,\inf} b_k G(j - b_k) \left(y_k(j - b_k) + 1\right) + \\ &+ \tfrac{1}{\min(C,j)} \sum_{i=1 \wedge i \neq k}^{K_e} \alpha_{i,\inf} b_i G(j - b_i) y_k(j - b_i) + \\ &+ \tfrac{1}{j} \sum_{i=1}^{K_\alpha} \alpha_{i,\inf} b_i G(j - b_i) y_k(j - b_i) \end{aligned} \quad (16)$$

$$\begin{aligned} &y_k(j)G(j) = \\ &= \tfrac{1}{j} \alpha_{k,\inf} b_k G(j - b_k)(y_k(j - b_k) + 1) + \\ &+ \tfrac{1}{j} \sum_{i=1 \wedge i \neq k}^{K_a} \alpha_{i,\inf} b_i G(j - b_i) y_k(j - b_i) + \\ &+ \tfrac{1}{\min(C,j)} \sum_{i=1}^{K_e} \alpha_{i,\inf} b_i G(j - b_i) y_k(j - b_i) \end{aligned} \quad (17)$$

where the values of $G(j)$'s are calculated by (14).

Having determined $G(j)$'s according to (13), we calculate the following performance measures:

1) The time congestion probabilities of service-class k, denoted as P_{b_k}, which is the probability that at least $T - b_k + 1$ bandwidth units are occupied:

$$P_{b_k} = \sum_{j=T-b_k+1}^{T} G^{-1} G(j) \quad (18)$$

where: $G = \sum\limits_{j=0}^{T} G(j)$ is a normalization constant.

Time congestion probabilities are determined by the proportion of time the system is congested.

2) The call congestion probabilities of service-class k, denoted as C_{b_k}, which is the probability that a new service-class k call is blocked and lost:

$$C_{b_k} = \sum_{j=T-b_k+1}^{T} G^{-1}G(j) \tag{19}$$

where $G(j)$'s are determined for a system with $N_k - 1$ traffic sources.
Call congestion probabilities are determined by the proportion of arriving calls that find the system congested.

3) The link utilization, denoted as U:

$$U = \sum_{j=1}^{C} jG^{-1}G(j) + \sum_{j=C+1}^{T} CG^{-1}G(j) \tag{20}$$

IV. The EF-EMLM Under The BR Policy (EF-EMLM/BR)

In the following subsections, we describe the EF-EMLM/BR, we provide the recursive determination of the link occupancy distribution, we show the relation of the EF-EMLM/BR to other multirate loss models and determine the various call-level performance measures of this model, which assures QoS guarantee (regarding congestion probabilities).

A. Description of the proposed EF-EMLM/BR

The BR policy is used to guarantee a certain QoS for calls of each service-class or attain equalization of call blocking probabilities among different service-classes that share a link by a proper selection of the BR parameters. If, for example, equalization of call blocking probabilities is required between calls of two service-classes with $b_1 = 1$ and $b_2 = 10$ bandwidth units, respectively, then $t(1) = 9$ bandwidth units and $t(2) = 0$ bandwidth units so that $b_1+t(1) = b_2+t(2)$. Note that $t(1) = 9$ bandwidth units means that 9 bandwidth units are reserved to benefit calls of the 2^{nd} service-class. Similarly, if a link accommodates calls of three service-classes with $b_1 = 1, b_2 = 5$ and $b_3 = 10$ bandwidth units, respectively, and equalization of call blocking probabilities is required between calls of the first two service-classes, then $t(1) = 4$ and $t(2) = 0$ bandwidth units so that $b_1 + t(1) = b_2 + t(2)$.

The application of the BR policy in a single link multirate loss model is based on the assumption that the number of calls of certain service-classes is negligible in those states j that form the so-called reservation space. More precisely, in the proposed EF-EMLM/BR the number of service-class k calls is negligible in states $j > T - t(k)$ and is incorporated in the calculation of $G(j)$'s (see (21) below) by the variable $D_k(j - b_k)$ given in (22). Generally speaking, the population of calls of service-class k in the reservation space may not be negligible. In [35] and [36], a complex procedure is implemented in order to take into account this population and increase the accuracy of call blocking probability results in the EMLM and Engset multirate state-dependent loss models, respectively. However, according to [36] this procedure may not always increase the accuracy of the call blocking probability results compared to simulation.

B. Determination of the link occupancy distribution

To apply the BR policy to the EF-EMLM we consider the method of [37]. In that case, the formula for the approximate calculation of $G(j)$ takes the form:

$$G(j)=\begin{cases} 1 & \text{for } j=0 \\ \frac{1}{\min(j,C)} \sum\limits_{k\in K_e} (N_k-n_k+1)\alpha_k D_k(j-b_k)G(j-b_k) & \\ +\frac{1}{j} \sum\limits_{k\in K_a} (N_k-n_k+1)\alpha_k D_k(j-b_k)G(j-b_k) & \\ & \text{for } j=1,\ldots,T \\ 0 & \text{for } j<0 \end{cases} \tag{21}$$

where:

$$D_k(j-b_k) = \left\langle \begin{array}{ll} b_k & \text{for } j \le T-t(k) \\ 0 & \text{for } j > T-t(k) \end{array} \right\rangle \tag{22}$$

and $t(k)$ is the reserved bandwidth (BR parameter) for service-class k calls (elastic or adaptive).

C. Relation of the EF-EMLM/BR to other multirate loss models

(a) When $N_k \to \infty$ for $k = 1, \ldots, K$, and the total offered traffic-load remains constant, then we have the Poisson arrival process and the formula of $G(j)$'s is given by [38]:

$$G(j)=\begin{cases} 1 & \text{for } j=0 \\ \frac{1}{\min(j,C)} \sum\limits_{k\in K_e} \alpha_k D_k(j-b_k)G(j-b_k)+ & \\ +\frac{1}{j} \sum\limits_{k\in K_a} \alpha_k D_k(j-b_k)G(j-b_k) & \\ & \text{for } j=1,\ldots,T \\ 0 & \text{for } j<0 \end{cases} \tag{23}$$

where $\alpha_k = \frac{\lambda_k}{\mu_k}$ (in erl), λ_k is the arrival rate of calls of service-class k and the values of $D_k(j)$ are given by (22).

(b) When all service-classes are elastic and $N_k \to \infty$ for $k = 1, \ldots, K$ and the total offered traffic-load remains constant, then the formula of $G(j)$'s is determined by [39]:

$$G(j) = \begin{cases} 1 & \text{for } j=0 \\ \frac{1}{\min(j,C)} \sum\limits_{k\in K} \alpha_k D_k(j-b_k)G(j-b_k) & \\ & \text{for } j=1,\ldots,T \\ 0 & \text{for } j<0 \end{cases} \tag{24}$$

(c) When $N_k \to \infty$ for $k = 1, \ldots, K$, and the total offered traffic-load remains constant, and $T = C$, then no bandwidth compression is allowed and the EMLM under the BR policy results. In that case, the formula of $G(j)$'s is given by the Roberts'recursion [37]:

$$G(j) = \begin{cases} 1 & \text{for } j=0 \\ \frac{1}{j} \sum\limits_{k=1}^{K} \alpha_{k,\inf} D_k(j-b_k)G(j-b_k) & \\ & \text{for } j=1,\ldots,C \\ 0 & \text{for } j<0 \end{cases} \tag{25}$$

D. Determination of various performance measures

The calculation of $G(j)$'s in (21) requires the value of n_k which is unknown. We approximate n_k in state j, $n_k(j)$, as the mean number of service-class k calls in state j, $y_k(j)$, when Poisson arrivals are considered, i.e., $n_k(j) \approx y_k(j)$. The values of $y_k(j)$ are given by (26) and (27) in the case of elastic and adaptive service-classes, respectively:

$$\begin{aligned} y_k(j)G(j) &= \tfrac{1}{\min(C,j)}\alpha_{k,\inf}D_k(j-b_k)G(j-b_k)\,(y_k(j-b_k)+1) \\ &+\tfrac{1}{\min(C,j)}\sum_{i=1\wedge i\neq k}^{K_e}\alpha_{i,\inf}D_i(j-b_i)G(j-b_i)y_k(j-b_i) \\ &+\tfrac{1}{j}\sum_{i=1}^{K_\alpha}\alpha_{i,\inf}D_i(j-b_i)G(j-b_i)y_k(j-b_i) \end{aligned} \tag{26}$$

$$\begin{aligned} y_k(j)G(j) &= \tfrac{1}{j}\alpha_{k,\inf}D_k(j-b_k)G(j-b_k)\,(y_k(j-b_k)+1) \\ &+\tfrac{1}{j}\sum_{i=1\wedge i\neq k}^{K_a}\alpha_{i,\inf}D_i(j-b_i)G(j-b_i)y_k(j-b_i) \\ &+\tfrac{1}{\min(C,j)}\sum_{i=1}^{K_e}\alpha_{i,\inf}D_i(j-b_i)G(j-b_i)y_k(j-b_i) \end{aligned} \tag{27}$$

where the values of $D_k(j)$ are given by (22) and the values of $G(j)$'s by (23).

Note that in (26) and (27), the mean number of service-class k calls in state j, $y_k(j) = 0$, if $j > T - t(k)$ as it is implied by (22).

Having determined the values of $G(j)$'s according to (21), we can calculate the link utilization based on (20) and the time and call congestion probabilities as follows:

$$P_{b_k} = \sum_{j=T-b_k-t_k+1}^{T} G^{-1}G(j) \tag{28}$$

$$C_{b_k} = \sum_{j=T-b_k-t_k+1}^{T} G^{-1}G(j) \tag{29}$$

where: $G = \sum_{j=0}^{T} G(j)$ is a normalization constant.

Note that, in order to calculate the call congestion probabilities of service-class k, $G(j)$'s, the system should be determined with $N_k - 1$ traffic sources – hence, the similarity between (28) and (29).

V. Evaluation

In this section, we present an application example and compare the analytical results of the Time Congestion (TC) probabilities and link utilization obtained from the E-EMLM, the EF-EMLM and the EF-EMLM/BR. The corresponding simulation results, presented for the EF-EMLM and the EF-EMLM/BR, are mean values of 6 runs. The resultant reliability ranges of the simulation measurements (confidence intervals of 95%) are very small (less than two orders of magnitude) and, therefore, they are not presented. Simulation is based on Simscript III simulation language [40].

We consider a single link of capacity $C = 90$ bandwidth units (b.u.) that accommodates calls of three service-classes. The first two service-classes are elastic, while the third service-class is adaptive. The traffic characteristics of each service-class of the EF-EMLM are:

- 1^{st} service-class: $N_1 = 200,\ v_1 = 0.10,\ b_1 = 1$ b.u.
- 2^{nd} service-class: $N_2 = 200,\ v_2 = 0.04,\ b_2 = 4$ b.u.
- 3^{rd} service-class: $N_3 = 200,\ v_3 = 0.01,\ b_3 = 6$ b.u.

In the case of the E-EMLM, the corresponding Poisson traffic loads are: $\alpha_{1,\inf} = 20$ erl, $\alpha_{2,\inf} = 8$ erl and $\alpha_{3,\inf} = 2$ erl. We also consider two values of T:

a) $T = 90$ b.u., where no bandwidth compression takes place, and
b) $T = 100$ b.u., where bandwidth compression takes place and $r_{\min} = \frac{C}{T} = 0.9$.

In the case of the EF-EMLM/BR model, we choose $t(1)=5$, $t(2)=2$ and $t(3)=0$ in order to achieve blocking equalization between calls of all service-classes since: $b_1+t(1)=b_2+t(2)=b_3+t(3)$. In the x-axis of all figures, v_1 and v_2 increase in steps of 0.01 and 0.005 erl, respectively, while v_3 remains constant. So in Point 1 we have $(v_1, v_2, v_3) = (0.10, 0.04, 0.01)$, while in Point 6 $(v_1, v_2, v_3) = (0.15, 0.065, 0.01)$. In the case of the E-EMLM, the corresponding Poisson traffic loads in Point 1 and Point 8 are $(\alpha_{1,\inf}, \alpha_{2,\inf}, \alpha_{3,\inf}) = (20, 8, 2)$ and $(\alpha_{1,\inf}, \alpha_{2,\inf}, \alpha_{3,\inf}) = (30, 13, 2)$, respectively.

In Figs. 3-4, we present the analytical and the simulation TC probabilities of the 1^{st} service-class for $T = 90$ b.u. and $T = 100$ b.u., respectively. To better compare the corresponding TC probabilities results (while having numerical values), we present in Table II and Table III, only for Points 1 and 6, an excerpt of the results of Fig. 3 and Fig. 4, respectively. Similarly, in Figs. 5-6 and 7-8, we present the corresponding results of the 2^{nd} and 3^{rd} service-class. In the legend of all figures, the term $N = \inf.$ refers to the E-EMLM where the number of traffic sources is infinite for each service-class. Likewise, the term BR in all figures refers to the EF-EMLM/BR. Note that the call congestion probabilities of service-class k $(k = 1, 2, 3)$ are quite close to the corresponding TC probabilities of service-class k (since they are obtained for a system with $N_k - 1 = 199$ traffic sources) and, therefore, are not presented herein. The interested reader may resort to [1], where call congestion probabilities are presented for the EF-EMLM.

All figures show that:

i) analytical and simulation results of TC probabilities are very close to each other,
ii) the application of the compression/expansion mechanism reduces congestion probabilities compared to those obtained when $C = T = 90$ b.u. (compare, e.g., Figs. 3-4, Figs. 5-6 and Figs. 7-8),
iii) the co-existence of the BR policy and the compression/expansion mechanism reduces congestion probabilities compared to those obtained when $C = T = 90$ b.u.,

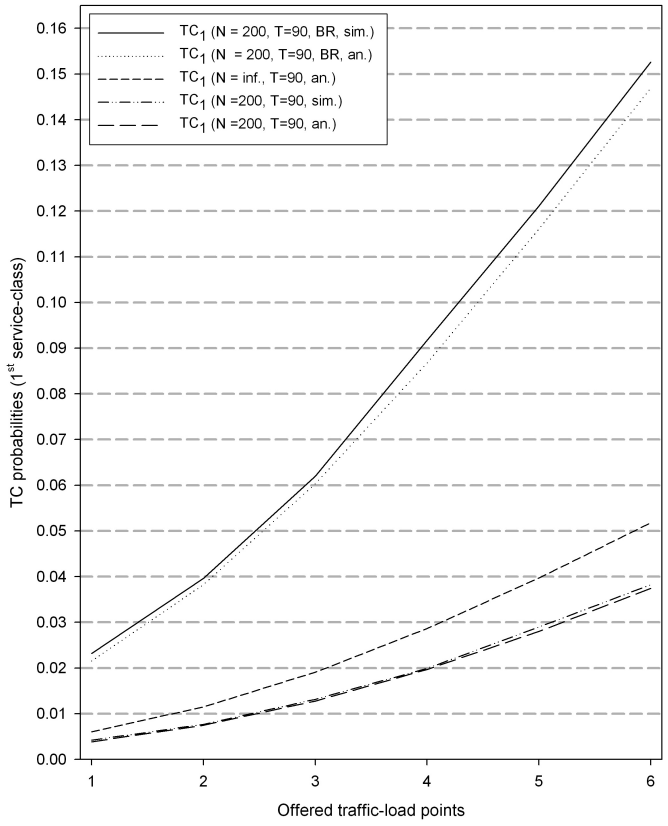


Figure 3. Time congestion probabilities of the 1^{st} service-class (T=90 b.u.).

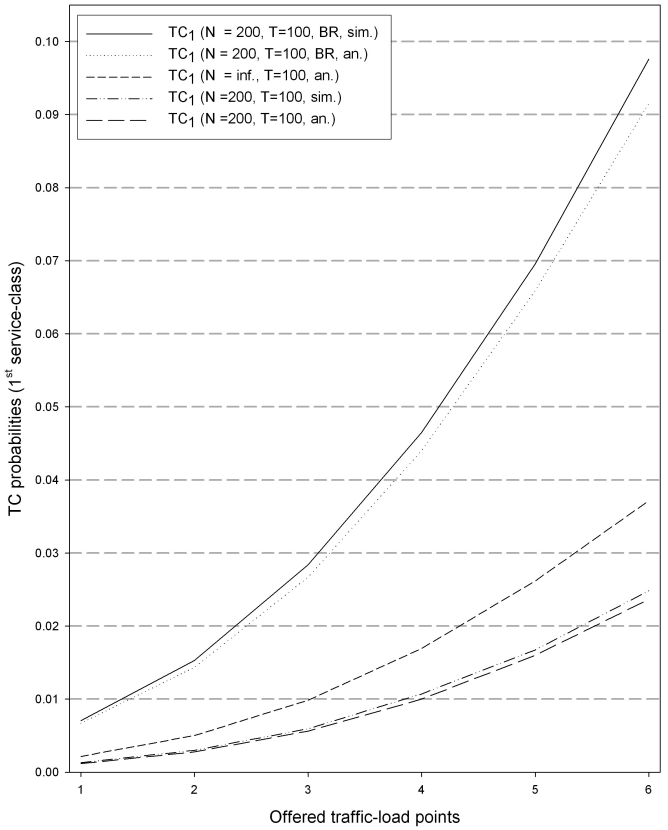


Figure 4. Time congestion probabilities of the 1^{st} service-class (T=100 b.u.).

iv) the congestion probabilities obtained by the EF-EMLM/BR and the EF-EMLM show that the BR policy favors calls of the 3^{rd} service-class, as expected, and
v) the results obtained by the E-EMLM fail to approximate the corresponding results obtained by the EF-EMLM.

TABLE II. EXCERPT OF THE RESULTS OF FIG. 3

$T = 90$	TC_1 $N = 200$ BR, sim.	TC_1 $N = 200$ BR, anal.	TC_1 $N = \inf.$ anal.	TC_1 $N = 200$ sim.	TC_1 $N = 200$ anal.
Point 1	0.02320	0.02153	0.00604	0.00420	0.00385
Point 6	0.15260	0.14692	0.05178	0.03822	0.03743

TABLE III. EXCERPT OF THE RESULTS OF FIG. 4

$T = 100$	TC_1 $N = 200$ BR, sim.	TC_1 $N = 200$ BR, anal.	TC_1 $N = \inf.$ anal.	TC_1 $N = 200$ sim.	TC_1 $N = 200$ anal.
Point 1	0.00704	0.00668	0.00215	0.00125	0.00117
Point 6	0.09760	0.09151	0.03716	0.02480	0.02367

Finally, in Figs. 9-10, we present the analytical and simulation results of the link utilization (in b.u.) for $T = 90$ and $T = 100$, respectively. It is clear, that the application of the bandwidth compression/expansion mechanism increases link utilization, since it decreases call congestion probabilities.

VI. GENERALIZATION OF THE EF-EMLM

Consider a link of capacity *C* b.u. that accommodates calls of finite and infinite number of sources. Let $K_{e,\inf}$ and $K_{a,\inf}$ be the set of elastic and adaptive service-classes ($K_{e,\inf} + K_{a,\inf} = K_{\inf}$) whose calls arrive in the link according to a Poisson process. Similarly, let $K_{e,\text{fin}}$ and $K_{a,\text{fin}}$ be the set of elastic and adaptive service-classes ($K_{e,\text{fin}} + K_{a,\text{fin}} = K_{\text{fin}}$) whose calls arrive in the link according to a quasi-random process. Then, the calculation of the link occupancy distribution, *G*(*j*), can be done by the following formula:

$$G(j)=\begin{cases} 1 & \text{for } j=0 \\ \frac{1}{\min(j,C)}\sum\limits_{k\in K_{e,\text{fin}}}(N_k-n_k+1)\alpha_k b_k G(j-b_k) & \\ +\frac{1}{j}\sum\limits_{k\in K_{a,\text{fin}}}(N_k-n_k+1)\alpha_k b_k G(j-b_k) & \\ +\frac{1}{\min(j,C)}\sum\limits_{k\in K_{e,\inf}}\alpha_{k,\inf} b_k G(j-b_k) & \\ +\frac{1}{j}\sum\limits_{k\in K_{a,\inf}}\alpha_{k,\inf} b_k G(j-b_k) & \\ & \text{for } j=1,...,\text{T} \\ 0 & \text{for } j<0 \end{cases} \quad (30)$$

The proof of (30) is based on the combination of the proofs proposed for the corresponding finite model in [1] and the infinite model of [3], and therefore is omitted. Such a mixture of service-classes does not destroy the accuracy of the model,

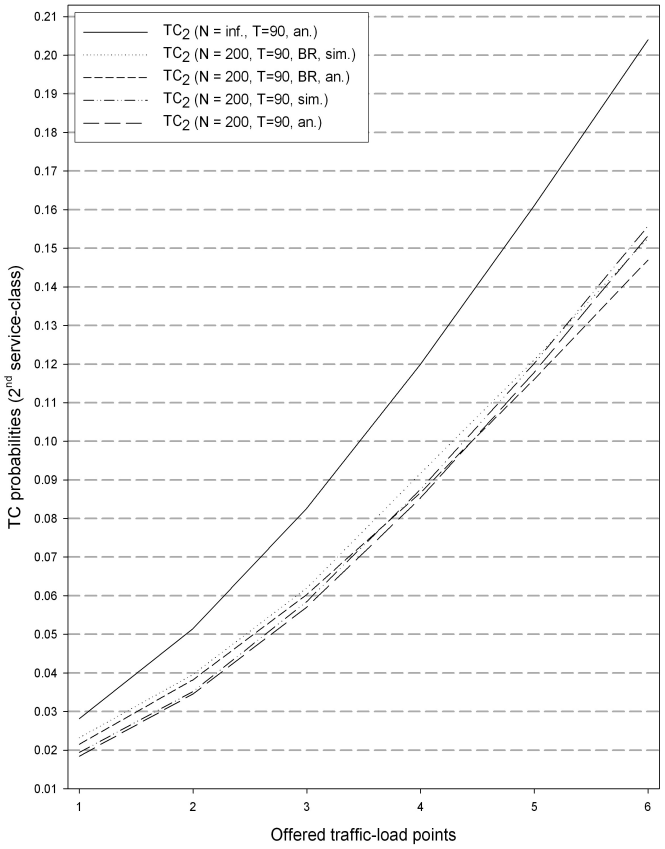


Figure 5. Time congestion probabilities of the 2^{nd} service-class (T=90 b.u.).

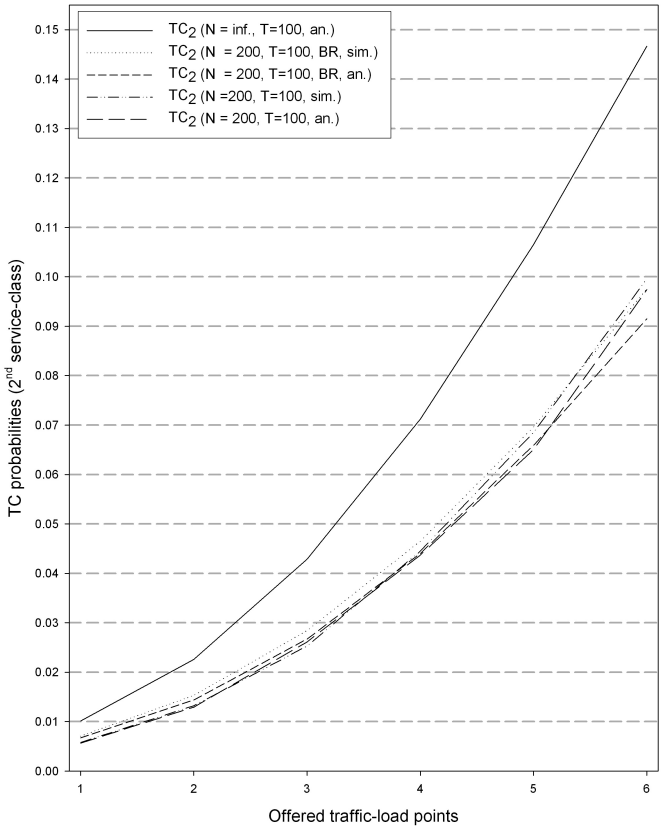


Figure 6. Time congestion probabilities of the 2^{nd} service-class (T=100 b.u.).

since both the E-EMLM and EF-EMLM give quite satisfactory results compared to simulation. The TC probabilities, CC probabilities and the link utilization in the generalized model can be determined by (18), (19) and (20), respectively.

When the BR policy is applied in the generalized model, the calculation of the link occupancy distribution is based on the following recursive formula:

$$G(j)=\begin{cases} 1 & \text{for } j=0 \\ \frac{1}{\min(j,C)} \sum\limits_{k\in K_{e,\text{fin}}} (N_k-n_k+1)\alpha_k D_k(j-b_k)G(j-b_k) \\ +\frac{1}{j} \sum\limits_{k\in K_{a,\text{fin}}} (N_k-n_k+1)\alpha_k D_k(j-b_k)G(j-b_k) \\ +\frac{1}{\min(j,C)} \sum\limits_{k\in K_{e,\text{inf}}} \alpha_{k,\text{inf}} D_k(j-b_k)G(j-b_k) \\ +\frac{1}{j} \sum\limits_{k\in K_{a,\text{inf}}} \alpha_{k,\text{inf}} D_k(j-b_k)G(j-b_k) \\ & \text{for } j=1,\ldots,T \\ 0 & \text{for } j<0 \end{cases} \tag{31}$$

where the values of $D_k(j)$ are given by (22).

As far as the TC probabilities, CC probabilities and the link utilization are concerned, they can be determined by (28), (29) and (20), respectively.

VII. Conclusion

We propose an analytical model for the call-level performance assessment of telecom networks, when elastic and/or adaptive calls of different service-classes come from finite traffic-sources and compete for the available bandwidth of a single link with fixed capacity. Because of the existence of the bandwidth compression/expansion mechanism for handling the elastic/adaptive traffic, the proposed model does not have a product form solution. Therefore, we propose approximate but recursive formulas for the calculation of the most important performance measures, namely time congestion and call congestion probabilities, and link utilization. In addition, we incorporate in our model the bandwidth reservation policy (whereby a part of the link's available bandwidth is reserved to benefit calls of higher bandwidth requirements), and study its effect on the performance measures. Simulation results verify the analytical results and prove the accuracy and the consistency of the proposed model. Furthermore, we show the relation of the proposed model to other multirate loss models and generalize it to include a mixture of service-classes of finite and infinite number of traffic sources. Potential applications of the proposed model are in the environment of wireless networks that support elastic and adaptive traffic. As a future work, we would like to incorporate into the proposed model the peculiarities of wireless networks.

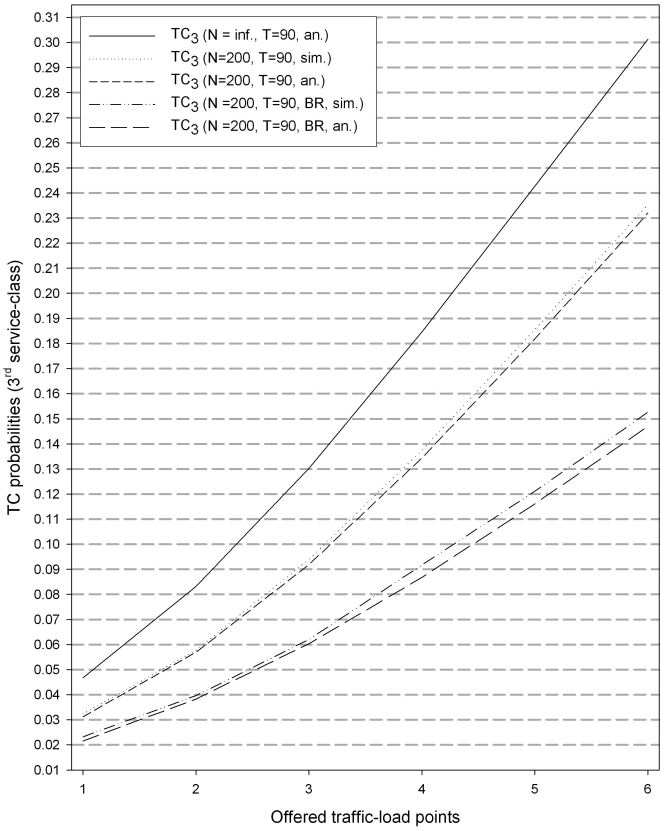


Figure 7. Time congestion probabilities of the 3^{rd} service-class (T=90 b.u.).

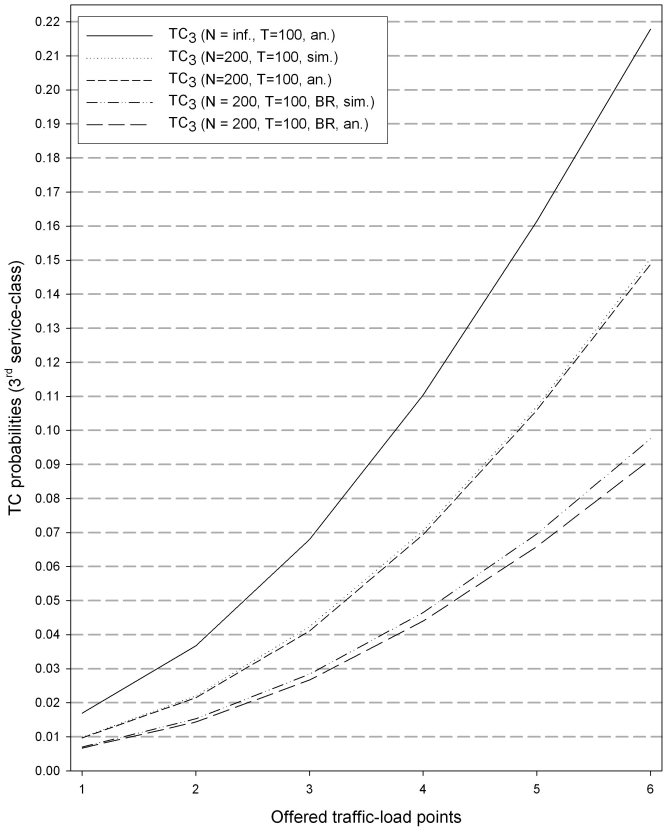


Figure 8. Time congestion probabilities of the 3^{rd} service-class (T=100 b.u.).

APPENDIX A
PROOF OF (12) FOR THE DETERMINATION OF $G(j)$' S WHEN $C < j \leq T$

When $C < j \leq T$, we multiply both sides of (4) by b_k^{comp} and sum over $k = 1, \ldots, K_e$ to have:

$$\sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k^{\text{comp}} P(\mathbf{n}_k^{-1}) = P(\mathbf{n}) \sum_{k \in K_e} n_k b_k^{\text{comp}} \phi_k(\mathbf{n}) \quad (32)$$

Based on (6) and (9), (32) is written as:

$$x(\mathbf{n}) \sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) = P(\mathbf{n}) \sum_{k \in K_e} x(\mathbf{n}_k^{-1}) n_k b_k \quad (33)$$

We continue by multiplying both sides of (5) by b_k^{comp} and sum over $k = 1, \ldots, K_a$ to obtain:

$$\sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k^{\text{comp}} P(\mathbf{n}_k^{-1}) = P(\mathbf{n}) \sum_{k \in K_a} n_k b_k^{\text{comp}} \phi_k(\mathbf{n}) \quad (34)$$

Based on (6) and (10) and since $r(\mathbf{n}) = \frac{C}{j}$, (34) is written as:

$$x(\mathbf{n}) \frac{C}{j} \sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) = P(\mathbf{n}) \frac{C}{j} \sum_{k \in K_a} x(\mathbf{n}_k^{-1}) n_k b_k \quad (35)$$

Adding (33) and (35), we have:

$$\begin{aligned} & x(\mathbf{n}) \left(\sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) \right) + \\ & + x(\mathbf{n}) \left(\tfrac{C}{j} \sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) \right) \\ & = P(\mathbf{n}) \left(\sum_{k \in K_e} x(\mathbf{n}_k^{-1}) n_k b_k + \tfrac{C}{j} \sum_{k \in K_a} x(\mathbf{n}_k^{-1}) n_k b_k \right) \end{aligned} \quad (36)$$

Due to (7), (36) can be written as:

$$\begin{aligned} & \sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) + \\ & + \tfrac{C}{j} \sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) = CP(\mathbf{n}) \end{aligned} \quad (37)$$

To introduce the link occupancy distribution $G(j)$ in (37), let $\mathbf{\Omega}_j = \{\mathbf{n} \in \mathbf{\Omega} : \mathbf{nb} = j\}$ be the state space where exactly j bandwidth units are occupied. Then, since $\sum_{\mathbf{n} \in \mathbf{\Omega}_j} P(\mathbf{n}) = G(j)$, summing both sides of (37) over $\mathbf{\Omega}_j$, we obtain:

$$\begin{aligned} & \sum_{\mathbf{n} \in \mathbf{\Omega}_j} \sum_{k \in K_e} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) + \\ & \tfrac{C}{j} \sum_{\mathbf{n} \in \mathbf{\Omega}_j} \sum_{k \in K_a} (N_k - n_k + 1)\alpha_k b_k P(\mathbf{n}_k^{-1}) = CG(j) \end{aligned} \quad (38)$$

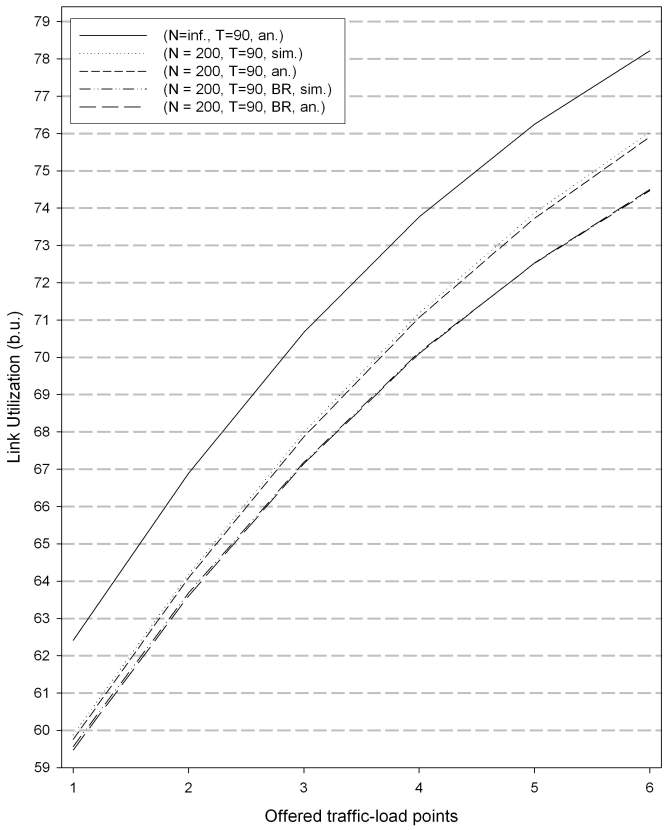


Figure 9. Link utilization (in b.u.), when T=90 b.u.

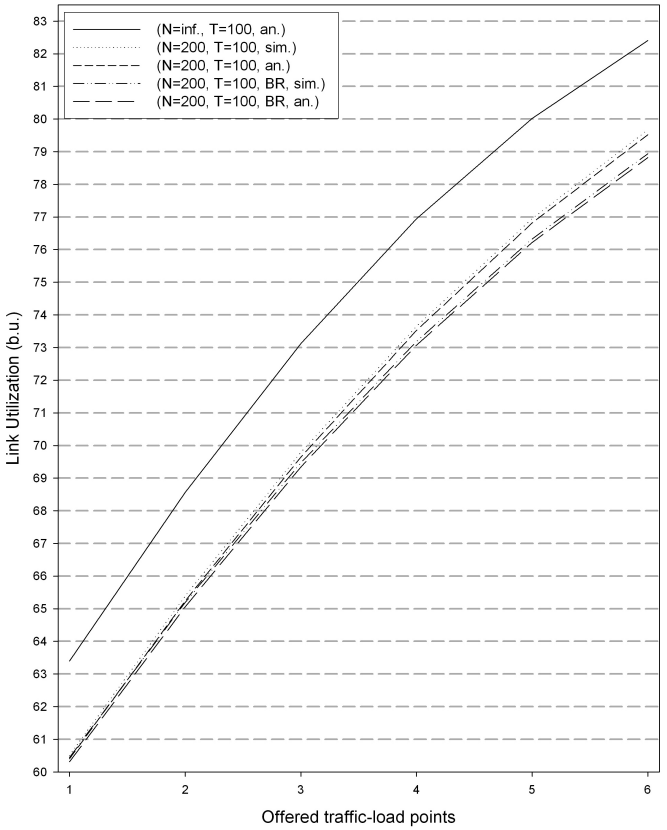


Figure 10. Link utilization (in b.u.), when T=100 b.u.

Interchanging the order of summations in (38) and assuming that each state has a unique occupancy j, we have:

$$\sum_{k\in K_e}(N_k-n_k+1)a_kb_k\sum_{\mathrm{n}\in\Omega_j}P(\mathbf{n}_k^{-1})+ \frac{C}{j}\sum_{k\in K_a}(N_k-n_k+1)a_kb_k\sum_{n\in\Omega_j}P(\mathbf{n}_k^{-1})=CG(j) \quad (39)$$

which can become exactly the same with (12):

$$\sum_{k\in K_e}(N_k-n_k+1)\alpha_kb_kG(j-b_k)+ \frac{C}{j}\sum_{k\in K_a}(N_k-n_k+1)\alpha_kb_kG(j-b_k)=CG(j) \quad (40)$$

ACKNOWLEDGMENT

The authors would like to thank Dr. V. Vassilakis (University of Surrey, U.K.) for his support. Also, the authors thank anonymous reviewers for their valuable remarks and suggestions.

REFERENCES

[1] I. Moscholios, J. Vardakas, M. Logothetis and M. Koukias, 'A Quasi-random Multirate loss model supporting elastic and adaptive traffic', Proc. of 4th Int. Conference on Emerging Network Intelligence, EMERGING 2012, Barcelona, Spain, pp. 56-61, 23-28 Sep. 2012.

[2] K. Ross, Multiservice loss models for broadband telecommunication networks, Springer, 1995, ISBN 3-540-19918-7.

[3] S. Rácz, B. Gerö, and G. Fodor, 'Flow level performance analysis of a multi-service system supporting elastic and adaptive services', Performance Evaluation, vol. 49, pp. 451-469, Sep. 2002.

[4] W. Bziuk, 'Approximate state probabilities in large shared multirate loss systems with an application to trunk reservation', European Transactions on Telecommunications, vol. 16, no. 3, pp. 205-216, May/June 2005.

[5] M. Glabowski, A. Kaliszan and M. Stasiak, 'Asymmetric convolution algorithm for blocking probability calculation in full-availability group with bandwidth reservation', IET Circuits, Devices & Systems, vol. 2, no. 1, pp. 87-94, Feb. 2008.

[6] I. Moscholios, V. Vassilakis, M. Logothetis and M. Koukias, 'QoS Equalization in a Multirate Loss Model of Elastic and Adaptive Traffic with Retrials', Proc. IARIA 5th Int. Conference on Emerging Network Intelligence, EMERGING 2013, Porto, Portugal, 29 Sep. 4 Oct. 2013.

[7] C. Oliveira, J. B. Kim, and T. Suda, 'An adaptive bandwidth reservation scheme for high-speed multimedia wireless networks', IEEE J. Select. Areas Commun., vol. 16, pp. 858-874, Aug. 1998.

[8] G. Raskutti, A. Zalesky, E.W.M Wong and M. Zukerman, 'Enhanced Blocking Probability Evaluation Method for Circuit-Switched Trunk Reservation Networks', IEEE Communications Letters, vol.11, no. 6, pp.543-545, June 2007.

[9] M. Stasiak, P. Zwierzykowski, D. Parniewicz, 'Modelling of the WCDMA Interface in the UMTS Network with Soft Handoff Mechanism', IEEE Globecom 2009, pp. 1-6, Nov. 2009.

[10] K. Kuppuswamy and D. C. Lee, 'An Analytic Approach to Efficiently Computing Call Blocking Probabilities for Multiclass WDM Networks', IEEE/ACM Trans. on Networking, vol. 17, no. 2, pp. 658-670, Apr. 2009.

[11] E. W. M. Wong, J. Baliga, M. Zukerman, A. Zalesky, and G. Raskutti, 'A New Method for Blocking Probability Evaluation in OBS/OPS

Appendix B – List of Symbols

Symbol	Meaning
C	Capacity of the link (in bandwidth units)
T	Virtual capacity of the link (in bandwidth units)
j	Occupied link bandwidth (in bandwidth units), $j = 0, \ldots, T$
$G(j)$	Link occupancy distribution
G	Normalization constant
K_e	Set of elastic service-classes
K_a	Set of adaptive service-classes
K	Set of service-classes, $K = K_e + K_a$
k	Service-class k $(k = 1, \ldots, K)$
N_k	Finite number of sources of service-class k
b_k	Peak-bandwidth requirement of service-class k calls
$\mathbf{b}$	Vector of the required peak-bandwidth per call of all service-classes, $\mathbf{b} = (b_1, b_2, \ldots, b_K)$
λ_k	Mean arrival rate of service-class k idle sources
$\lambda_{k,\inf}$	Mean arrival rate of Poisson service-class k calls
v_k	Arrival rate per idle source of service-class k
μ_k^{-1}	Mean of the exponentially distributed service time of service-class k calls
α_k	Offered traffic-load (in erl) per idle source of service-class k, $\alpha_k = v_k/\mu_k$
$\alpha_{k,\inf}$	Offered traffic-load (in erl) of Poisson service-class k calls, $\alpha_{k,\inf} = \lambda_{k,\inf}/\mu_k$
n_k	Number of in-service calls of service-class k
$\mathbf{n}$	Vector of all in service calls of all service-classes, $\mathbf{n} = (n_1, n_2, \ldots, n_K)$
$P(\mathbf{n})$	Steady state distribution
b_k^{comp}	Compressed bandwidth of service-class k calls
r	Compression factor
b_k^{expan}	Expanded bandwidth of service-class k calls
$t(k)$	Bandwidth reservation parameter of service-class k
$\phi_k(\mathbf{n})$	State-dependent multiplier of service-class k
$x(\mathbf{n})$	State-dependent variable
$y_k(j)$	Mean number of Poisson service-class k calls in state j
P_{b_k}	Time Congestion probabilities of service-class k
C_{b_k}	Call Congestion probabilities of service-class k
U	Link utilization

Networks With Deflection Routing', J. Lightwave Technol., vol. 27, issue 23, pp. 5335-5347, Dec. 2009.

[12] F. Kelly, Reversibility and Stochastic Networks, Wiley Series in Probability and Mathematical Statistics. Wiley: New York, 1979.

[13] J. Kaufman, 'Blocking in a shared resource environment', IEEE Trans. Commun. vol. 29, Oct. 1981, pp. 1474-1481.

[14] J. Roberts, 'A service system with heterogeneous user requirements', in: G. Pujolle (Ed.), Performance of Data Communications systems and their applications, North Holland, Amsterdam, 1981, pp. 423-431.

[15] G. Kallos, V. Vassilakis, I. Moscholios, and M. Logothetis, 'Performance Modelling of W-CDMA Networks Supporting Elastic and Adaptive Traffic', Proc. 4th IFIP HET-NETs, Ilkley, U.K., pp. 09/1-09/10, 11-13 Sep. 2006.

[16] G. Fodor and M. Telek, 'Bounding the Blocking Probabilities in Multirate CDMA Networks Supporting Elastic Services', IEEE/ACM Trans. on Networking, vol. 15, pp. 944-956, Aug. 2007.

[17] M. Logothetis, V. Vassilakis and I. Moscholios, 'Call-level Performance Modeling and QoS Assessment of W-CDMA Networks', Wireless Networks: Research, Technology and Applications, Ed. Jia Feng, Nova Science Publishers, New York, USA, 2009, pp. 57-90.

[18] M. Glabowski, M. Stasiak, A. Wisniewski, and P. Zwierzykowski, 'Blocking Probability Calculation for Cellular Systems with WCDMA Radio Interface Servicing PCT1 and PCT2 Multirate Traffic', IEICE Trans. Commun., vol. E92-B, pp. 1156-1165, Apr. 2009.

[19] G. Kallos, V. Vassilakis and M. Logothetis, 'Call-level performance analysis of a W-CDMA cell with finite population and interference cancellation', European Trans. Telecom., vol. 22, no. 1, pp. 25-30, 2011.

[20] B. P. Gerö, P. L. Pályi and S. Rácz, 'Flow-level performance analysis of a multi-rate system supporting stream and elastic services', Int. J. Commun. Syst., Wiley, doi: 10.1002/dac.1383, 2012.

[21] I. Moscholios, V. Vassilakis, J. Vardakas and M. Logothetis, 'Call Blocking Probabilities of Elastic and Adaptive Traffic with Retrials', Proc. of 8th Advanced Int. Conf. on Telecommunications, AICT 2012, Stuttgart, Germany, 27 May-1 June 2012, pp. 92-97.

[22] V. Vassilakis, I. Moscholios and M. Logothetis, 'The extended connection-dependent threshold model for call-level performance analysis of multi-rate loss systems under the bandwidth reservation policy', Int. J. Commun. Syst., Wiley, vol. 25, no. 7, pp. 849-873, July 2012.

[23] I. Moscholios, J. Vardakas, M. Logothetis and A. Boucouvalas, 'QoS Guarantee in a Batched Poisson Multirate Loss Model Supporting Elastic and Adaptive Traffic', Proc. of IEEE ICC 2012, Ottawa, Canada, 10-15 June 2012, pp. 1296-1301.

[24] I. Moscholios, J. Vardakas, M. Logothetis and A. Boucouvalas, 'Congestion Probabilities in a Batched Poisson Multirate Loss Model Supporting Elastic and Adaptive Traffic', Annals of Telecommunications, vol. 68, issue 5, pp. 327-344, 2013.

[25] H. Akimaru and K. Kawashima, Teletraffic Theory and Applications, 2nd edition, Springer-Verlag, Berlin, 1999.

[26] I. Moscholios, M. Logothetis and M. Koukias, 'A State-Dependent Multi-Rate Loss Model of Finite Sources with QoS Guarantee for Wireless Networks', Mediterranean Journal of Computers and Networks, vol. 2, no. 1, pp. 10-20, Jan. 2006.

[27] M. Stasiak, M. Glabowski, A.Wisniewski, and P. Zwierzykowski, Modeling and Dimensioning of Mobile Networks, Wiley & Sons, 2011.

[28] S. Yashkov and A. Yashkova, 'Processor sharing: a survey of the mathematical theory', Automation and Remote Control, vol. 68, no. 9, pp. 1662-1731, Sep. 2007.

[29] G. Stamatelos and J. Hayes, 'Admission control techniques with application to broadband networks', Computer Communications, vol. 17, no. 9, pp. 663-673, 1994.

[30] I. Moscholios, M. Logothetis and P. Nikolaropoulos, 'Engset Multi-Rate State-Dependent Loss Models', Performance Evaluation, vol. 59, issues 2-3, pp. 247-277, Feb. 2005.

[31] I. Moscholios, M. Logothetis and M. Koukias, 'An ON-OFF Multirate Loss Model of Finite Sources', IEICE Trans. on Commun., vol. E90-B, no.7, pp. 1608-1619, July 2007.

[32] M. Glabowski and M. Stasiak, 'An approximate model of the full-availability group with multi-rate traffic and a finite source population', Proc. of 12th MMB&PGTS, Dresden, Germany, pp. 195-204, Sep. 2004.

[33] V. Vassilakis, G. Kallos, I. Moscholios and M. Logothetis, 'Call-Level Analysis of W-CDMA Networks Supporting Elastic Services of Finite Population', Proc. of IEEE ICC 2008, Beijing, China, 19-23 May 2008.

[34] I. Moscholios, M. Logothetis, V. Stylianakis and J. Vardakas, 'The Priority Wavelength Release Protocol for Dynamic Wavelength Allocation in WDM-TDMA PONs Supporting Random and Quasi-Random Bursty Traffic', Proc. of the 18th NOC 2013, Graz, Austria, 10-12 July 2013.

[35] M. Stasiak and M. Glabowski, 'A simple approximation of the link model with reservation by a one-dimensional Markov chain', Performance Evaluation, vol. 41, pp. 195-208, July 2000.

[36] I. Moscholios and M. Logothetis, 'Engset multi-rate state-dependent loss models with QoS guarantee', Int. J. Commun. Syst., vol. 19, pp. 67-93, Feb. 2006.

[37] J. Roberts, 'Teletraffic models for the Telecom 1 Integrated Services Network', Proc. of ITC-10, Mondreal, Canada, 1983.

[38] I. Moscholios, V. Vassilakis, M. Logothetis and J. Vardakas, 'Bandwidth Reservation in the Erlang Multirate Loss Model for Elastic and Adaptive Traffic', Proc. IARIA 9th Advanced Int. Conference on Telecommunications, AICT 2013, Rome, Italy, 23-28 June 2013.

[39] I. Moscholios, V. Vassilakis, M. Logothetis and A. Boucouvalas, 'Blocking Equalization in the Erlang Multirate Loss Model for Elastic Traffic', Proc. IARIA 2nd Int. Conference on Emerging Network Intelligence, EMERGING 2010, Florence, Italy, 25-30 Oct. 2010.

[40] Simscript III, http://www.simscript.com *(retrieved: December 2013).*

Integrated Fuzzy Solution for Network Selection using MIH in Heterogeneous Environment

Ahmad Rahil, Nader Mbarek, Olivier Togni
Laboratoire d'Electronique, Informatique et Image
UMR 6306, University of Burgundy
Dijon, France
ahmad.rahil | nader.mbarek | olivier.togni@u-bourgogne.fr

Mirna Atieh
Département Informatique, Faculté des Sciences
Économiques et de Gestion, Lebanese University
Beirut, Lebanon
matieh@ul.edu.lb

***Abstract*—Seamless handover between networks in heterogeneous environment is essential to guarantee end-to-end QoS for mobile users. A key requirement is the ability to select seamlessly the next best network. Currently, the implementation of the selection algorithm of the IEEE 802.21 standard by National Institute of Standards and Technology considers only the signal strength as a parameter to select the best destination network. In this paper, we improve the implementation of the existing selection algorithm by proposing an integrated solution to select the best destination network. Our proposed solution consists of proposing a Multi Criteria Selection Algorithm that modifies the current implemented algorithm by including additional parameters such as available bandwidth, mobile node speed and type of network. This first solution is complemented with a fuzzy logic model, which includes a new controller entity where parameters such as signal strength, signal quality and available bandwidth of the destination networks are considered as inputs. The inference rules for the controller entity are derived from a detailed analysis made on a large number of data retrieved from the servers of Alfa mobile telecommunications company. The results, initially obtained using Network Simulator, show that there is a need for a new framework taking into account additional parameters to guide network selection process during handover in order to provide mobile users with better QoS. They were then complemented with a model that qualified each candidate network in the vicinity of the mobile based on a scale of 0 (the least advisable network) and 1 (the most recommended network). This will lead to the mobile node choosing the network that has the maximum likelihood estimation between a set of recommended networks.**

Keywords-seamless vertical handover; QoS parameters; IEEE 802.21 MIH; fuzzy logic modeling

I. INTRODUCTION

Communicating from anywhere at any time is becoming a requirement of great importance for mobile users. However, the rapid expansion of wireless network technologies creates a heterogeneous environment. Nowadays, mobile users would like to acquire, directly from the device, different kinds of services like internet, audio and video conferencing, which sometimes require switching between different operators or network types. Moreover, user preferences are different. Some are interested in service costs only; others will be satisfied with broadband networks that cover large geographic areas, etc. Consequently, to satisfy the above requirements, user mobility should be covered by a set of different overlapping networks forming a heterogeneous environment. A mobile device should be able to choose, from all available networks in its environment, the one that meets its needs and ensures accordingly the transition from one cell to another in the same technology (horizontal handover) or between different types of technologies (vertical handover). During this handover period, the challenge is to conserve the QoS parameters guarantee. QoS is the capability of operators to provide satisfactory services for a given user in terms of data rates, call blocking, delay and throughput.

The remainder of this paper is organized as follows: Section II describes the background. Section III describes the main components of IEEE 802.21 standard and its implementation using NS2 simulator. Section IV provides an overview of wireless protocols used in our simulation environment. Section V describes the simulation scenarios and results. The fuzzy logic system is described in Section VI. Section VII discusses the fuzzy logic handover decision algorithm and we conclude in Section VIII.

II. BACKGROUND

In this section, we will present the MIH and Fuzzy Logic related works to select the destination network during handover. Our contributions to improve the selection of the destination network during handover are summarized at the end of this section.

A. MIH Related Work

The IEEE 802.21 [1][2] is an emerging standard, also known as Media Independent Handover (MIH) that supports management of seamless handover between different networks in a heterogeneous environment. The current implementation of the IEEE 802.21 standard for the network simulator (NS-2) by National Institute of Standards and Technology (NIST) based on draft 3 [3][4] considers only the Radio Signal Strength Indicator (RSSI) as a unique parameter to determine the best network [5]. We argue in this paper that this parameter alone is not sufficient to

satisfy users' needs. Indeed, signal strength, available bandwidth (ABW), traffic on the serving network and packet loss ratio are among the other parameters that have an impact on the mobile user in terms of QoS. For example, a bad QoS, when using a real time application in a handover process, may be due to a lack of ABW because of high load in the host network while the signal strength is good.

Several attempts have been made to improve the handover within the MIH framework. Chandavarkar et al. [6] proposed an algorithm for network selection based on the strength of the battery, the speed of the mobile, and the coverage radius of the network in order to avoid power loss during handover and to improve the efficiency of seamless handover. Siddiqui et al. [7] proposed a new algorithm named TAILOR that uses different QoS parameters based on user preferences to select the destination network. Also, this algorithm optimizes the power consumption.

Jiadi et al. and Ying et al. [8][9], modified the MIH where handover is performed in three steps: initiation, selection and execution. The proposed process aims to improve the handover delay by adding new events to the initiation step that can be generated from the application layer instead of lower layer upon the user's satisfaction. Moreover, they added a new algorithm at the selection step based on price, delay, jitter, signal noise ratio (SNR) and available data rate within the MADM (Multi Attribute Decision Making) function to improve the QoS during the selection process.

The research work initiated in [10][11] proposed a selection algorithm based on the willingness of users to pay for a given service, while Cicconetti et al. [12] provided an algorithm based on three parameters: connectivity graph, connectivity table between nodes and the current geographical position of the serving network. The proposed algorithm reduces the handover time and the energy consumption of mobile node (MN) due to scanning.

The Media Independent Information Server (MIIS) component (see Section III) of MIH is not fully implemented by NIST. Arraez et al. [13] implement this service and install it on each access point (AP) allowing users to save the energy of the battery by just activating a single interface. According to the IEEE 802.21 standard, an MIH user communicates, through the link layer, with its MIHF, which sends a query to MIIS to retrieve the list of all networks in the vicinity. Alternatively, the authors of [14][15] developed a new method to communicate with the MIIS through the upper layers using Web Services.

Moreover, 802.11 protocols [16] define 11 channels for communication and force the MN during the handover to scan all channels looking for the active one. Khan et al. [17] proposed a new algorithm based on MIIS, to provide user with a list of only active channels to be scanned in order to save time during handover.

An et al. [18] added two new parameters to MIH that allow FMIPV6 to save the steps of proxy router solicitation and advertisement (RtSolPr/PrRtAdv). This resulted in a decrease of handover latency and improvement of packet loss ratio.

B. Fuzzy Logic Related Work

A fuzzy logic model has been also used for handover solution in heterogeneous environments. Unnecessary handovers, when oscillating between two networks, are commonly known in the literature as the Ping-Pong effect. Kwong et al. [19] show that the handover decision based only on RSSI may exhibit a drawback such as the Ping-Pong effect. P. Dhand [20] and Pragati et al. [21] proposed respectively a Fuzzy Controller for Handoff Optimization (FCHO) and a fuzzy algorithm based on multiple parameters to minimize the unnecessary handover and eliminate the Ping-Pong effect.

Several works have been done to select the best destination network by combining different types of parameters as input to a fuzzy model. Ling et al. [22] make use the RSSI and the distance between the MN and the base station. Yan et al. [23] use the velocity of MN and the ABW. Alternatively, Vasu et al. [24] use QoS parameter within a multi-criteria algorithm through a fuzzy logic controller (FLCs) rules. Sadiq et al. [25] propose a fuzzy logic based handover decision based on the RSSI of the AP and relative direction of the MN toward the APs. It also showed that using this schema, the handover latency at L2 level is improved. Authors of [26] use multiple parameters like bandwidth, SNR, traffic load and battery power to propose a fuzzy based vertical handover algorithm NG-VDA between LTE and WLAN. The research initiated in [27] proposed a modular fuzzy-based design for Handover Decision System (HDS) to deal with the large number of fuzzy inference rule base. Authors of [28] use the user preference and network parameters as input to the fuzzy system. Also it introduces fuzzy logic rules at different phases of the handover process. While authors of [29] proposed a fuzzy Q-Learning algorithm to find the optimal set of fuzzy rules in a Fuzzy Logic Controller (FLC) for traffic balancing in GSM-EDGE Radio Access Network (GERAN). To optimize the fuzzy logic algorithm without requiring an expert knowledge, Foong et al. [30] proposed a newer approach using Adaptive Network Fuzzy Inference System (ANFIS) where the training element is incorporated into the existing fuzzy handover algorithm. This training element helps in optimizing and modeling the membership function and the inference rules base.

C. Contributions

Different research work aimed to improve the standard itself or the NIST implementation of the standard. In this paper, we present two algorithms to improve the NIST implementation of the MIH standard.

The first contribution demonstrates, through simulation, that there is a need for additional input parameters with the RSSI to better select a destination network. This research work is complemented by a new fuzzy logic algorithm to better select the most appropriate destination network.

Within our first contribution we investigate, by experiment, the effect of the inclusion of three parameters with the RSSI into the selection mechanism during handover. These parameters are: ABW, type of network and mobile speed. As far as we know, these parameters have not

been investigated at the same time before. As it will be detailed in Section V, our first experiment will show that by including the ABW, the packet loss ratio will be improved. The second experiment will show that based on the type (WIFI, WIMAX) of the current and destination network, we can save on packet loss. The third experiment will show that it is worthily significant to consider the velocity of the MN while selecting a new destination network. As a result, there is a need for a new model based on more than one input parameter to select the best destination network.

Concerning our second contribution, we propose a new model based on a fuzzy logic system that takes three parameters as an input to select the best destination network. This solution has been explored earlier in the literature. But the twist here lies in the fact that an accurate choice of the best destination network depends also on the accuracy of the membership function and the inference rules base used in the fuzzy logic model. Our added value for this contribution is that we construct our inference rules base after a detailed observation on more than 9500 records retrieved from the real server of an existing Lebanese mobile telecommunication company (Alfa). Among these 9500 records, we have 100 cases of vertical handover between GPRS network (2.5G) and UMTS (3G) network. Unfortunately, the LTE installation is still in progress for the all Lebanese Companies. After analysis of the QoS parameters value during handover, we conclude that the 100 cases of handover follow a set of 18 rules. These 18 rules constitute our inference rule. Also the membership function used is based on the exact values and threshold determined by Alfa. The findings are anticipated to be mirrored and extended to WIFI-WIMAX handovers, something that was not possible due to the lack of available data in such networks.

III. IEEE 802.21 STANDARD

The IEEE 802.21 standard, also known as Media Independent Handover (MIH), provides mobility management at layer 2.5 by being inserted between layer 2 and layer 3. As depicted in Figure 1, the Media Independent Handover Function (MIHF) is the main entity of the standard that allows communication in both directions between lower and upper layers through three services: event (MIES), command (MICS) and information (MIES) [4][31].

A. Media Independent Event Services, MIES

This service detects changes in the lower layers (physical and link) to determine if it needs to perform handover. Two types of events can occur: "MIH Event" sent by the MIHF to the upper layers (3 +), and "Link Event" that spreads from the lower layers to the MIHF.

B. Media Independent Command Services, MICS

This service uses two types of events. The "MIH Commands" transmitted by the user towards the MIHF and "Link Commands" sent by MIHF to lower layers.

C. Media Independent Information Services, MIIS

The MIIS let the mobile user discover and collect information about features and services offered by neighboring networks such as network type, operator ID, network ID, cost, network QoS, and much more. This information helps in making a more efficient handover decision across heterogeneous networks.

IV. WIFI, WIMAX, GPRS AND UMTS STANDARDS

In this section, we describe an overview of the emerging wireless technologies that we have used within the handover scenarios to validate our contributions.

A. IEEE 802.11, WIFI

IEEE 802.11, Wireless Fidelity (WIFI) [32], is a wireless local network technology designed for a private LAN with a small coverage area (hundreds of meters typically). Different versions of 802.11 communicate on different frequency bands with different bit rates. In all simulations that we performed during our research work, we use IEEE 802.11b version. Mobility support in conventional IEEE 802.11 standard is not a prior consideration and horizontal handover procedure does not meet the needs of real time traffic [33]. WIFI's QoS is limited in supporting multimedia or Voice over Internet Protocol (VoIP) traffic and several research activities have been carried out in an attempt to overcome this limitation [34].

B. IEEE 802.16, WIMAX

IEEE 802.16, WIMAX (Worldwide Interoperability for Microwave Access), technology is for metropolitan area network (MAN) covering a wide area at very high speed. QoS in WIFI is relative to packet flow and similar to fixed Ethernet while WIMAX define a packet classification and a scheduling mechanism with four classes to guarantee QoS for each flow: Unsolicited Grant Service (UGS), Real-Time Polling Service (RTPS), non-real-Time Polling Services (nrtPS) and Best Effort (BE). WIMAX mobile (802.16e) adds a fifth one called extended real-time Polling System (ertPS) [35]. WIMAX supports three handover methods: Hard Handover (HHO), Fast Base Station Switching (FBSS) and Macro-Diversity Handover (MDHO). The HO process [36] is composed of several phases: network topology advertisement, MS scanning, cell reselection, HO decision and initiation and network re-entry [37][38].

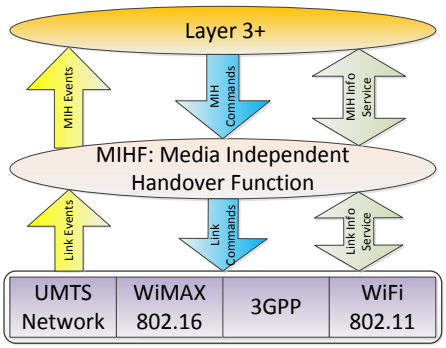


Figure 1. MIH Architecture

C. General Packet Radio Service, GPRS 2.5G

The General Packet Radio Service (GPRS) is an improved version of Global System for Mobile Telecommunications (GSM). GPRS was originally standardized by European Telecommunications Standards Institute (ETSI) and now maintained by the 3rd Generation Partnership Project (3GPP) [39][40]. GPRS Introduces two new elements [41] to the existing GSM architecture: the serving GPRS Support Node (SGSN) to control the communications and mobility management between the mobile stations (MS) and the GPRS network; and the Gateway GPRS Support Node (GGSN) that acts as an interface between the GPRS network and external packet switching networks such as Internet, or GPRS networks of different operators [42].

GPRS has several enhancements vs. GSM. (1) It introduces services based on packet-switching technique instead of the circuit-switching network. (2) It eliminates the monopolization of the GSM channel, reducing by that the communication cost and improving the transmission speed [43].GPRS assign a static IP address for the user reducing by that the time of session establishment and access to the service comparatively to GSM. GPRS enables billing by volume [44] (the number of exchanged packets) or based on the content (e.g., by image sent). Finally, GPRS introduces more sophisticated security mechanism than GSM [45].

D. Universal Mobile Telecommunications System, UMTS

UMTS is the third generation evolution of the GSM/GPRS systems; standardization work is done at a worldwide level within the 3GPP. UMTS, by integrating packet and circuit data transmission [46], allows the interoperability with GSM and its evolution. With the use of the Wideband Code Division Multiple Access protocol (W-CDMA), UMTS provide high transmission rate that can reach 2 Mbit/s allowing a better use of e-commerce, multimedia and Visio conference application from anywhere at any time. UTRAN is the great innovation of UMTS and is in charge of control and radio resource management. It enables the exchange of information between the mobile terminal and the core network. UTRAN consists of two main entities: (1) the RNS that contains one or more base station (Node B) and the Radio Network Controller (RNC), (2) the Serving GPRS Support Node SRNC controlling the mobility management. UMTS manage seamlessly two types of handover: soft and hard handover [47].

V. MIH PERFORMANCE EVALUATION

In this section, we will present three scenarios to assess the impact of the ABW, type of network, and user velocity on selecting a destination network during handover. For the three scenarios, the decision for handover is totally taken by MIH.

A. Simulation Environment

To show the limits of using one parameter to select an access network and to motivate the need of advanced selection methods that combine several constraints, we present several simulation scenarios using NS2, v2.29, which support the MIH module implemented by NIST.

The covered scenarios focus on criteria other than RSSI when evaluating network in the vicinity for handover. The first scenario investigates and assesses the impact of the selected network available bandwidth. The second one deals with the type of destination network. While the third scenario addresses the speed effect of the MN on QoS during handover.

Various simulation parameters are summarized in Table I. The traffic used has a constant bit rate (CBR), which allows for calculating the number of packet loss. It also could be used to simulate voice traffic. Packet size is always constant to 1500 bytes and the throughput is determined by varying the interval of sending packet during simulation.

B. Scenario I: NIST Selection Weakness

1) Topology Description: The topology of this scenario, shown in Figure 2, consists of two WIFI Access Points AP1 and AP2 (802.11b) located inside an 802.16 base station (BS) coverage area and one MN equipped with multiple interfaces. It is important to note that other streams of traffic source are connected to AP2 consuming its bandwidth. By doing that, I would simulate the connection of more than one mobile. Initially, the MN connected to AP1, starts moving to the center of the BS coverage area and on its way detect AP2. According to the NIST handover algorithm, that selects a new network based on the RSSI only, AP2 is considered as better network than WIMAX and the MN will make a handover from AP1 to AP2. Once the MN reaches the limit coverage area of AP2, the handover to WIMAX base station occurs.

2) Scenario I Results: By increasing the throughput of traffic generated by the CBR application on the MN, we observe an overall greater number of packet losses. Figure 3 shows the packet loss during HO. When a MN loses the signal on AP1 it needs to make a HO to another network, it has 2 choices: handover to AP2 or to WIMAX. According to NIST algorithm, which selects a new network based on the signal strength only, AP2 is selected and Figure 3 shows the number of Packet Loss (PL) during handover AP1-AP2. When the MN reaches the limit coverage area of AP2 and makes the handover to WIMAX. At this point, we observe another value of PL during HO AP2-WIMAX.

3) Critics of the NIST algorithm: The main issue with this algorithm lies in the fact that it selects a destination network based on the signal strength received by the MN, which is unsatisfactory. Indeed, a MN, near to an overloaded base station, receives a strong signal. According to NIST algorithm, the MN handover to this base station will occur and will result in a high packet loss ratio due to a lack of ABW.

C. Multi Criteria Selection Algorithm

In this section, a new selection algorithm named Multi Criteria Selection Algorithm (MCSA) will be proposed. It is

a modified version of the algorithm proposed by NIST to select a destination network based on two criteria: RSSI and ABW of the destination network. We assume that the user preference is mainly composed of selecting a network with the largest ABW whatever the cost is within a predefined maximum limit. Then, we compare the number of packet loss during HO between MCSA and NIST algorithm.

1) Strategy of our MCSA algorithm: A MN that is connected to a serving network receives beacons and Router Advertisement (RA) from WIFI and WIMAX networks in the vicinity. According to our proposed algorithm, MN will select the network that has the biggest ABW. In order to get the value of ABW to the MN, we added to the structure of the beacons and RA in NS2 a new field that holds the value of ABW.

2) MCSA results: in order to compare MCSA and NIST results, we use the same topology of simulation cited in Figure 2. By using our proposed MCSA algorithm, which aims to find among the visible list of networks, the one that have the largest ABW, WIMAX is selected instead of AP2 and the total number of handovers decreases resulting in improving the total number of PL and the Quality of Service is thus preserved during the mobility of the MN.

TABLE I. SIMULATION PARAMETERS

WIFI Access Point AP1 and AP2 Parameters	
Transmission Power (Pt_)	0.027 W
Receiving Threshold (RXThresh)	1.17557e-10 W
Carrier Sending Threshold (CXTresh)	1.058.13 e-10 W
Coverage Radius	150 meters
Radio Propagation Model	Two-RayGround
Frequency (Freq)	2.4 GHz
Sensitivity to link degradation (lgd_factor_)	1.2
Physical Data Rate	11 Mbps
WIMAX Parameters	
Transmission Power (Pt_)	30 W
Receiving Threshold (RXThresh)	3e-11 W
Carrier Sending Threshold (CXTresh)	2.4 e-11 W
Coverage Radius	1500 meters
Radio Propagation Model	Two-RayGround
Frequency (Freq)	3.5 GHz
Sensitivity to link degradation (lgd_factor_)	1.2
Antenna Type	Omni Antenna
Modulation	OFDM
Physical Data Rate	30 Mbps

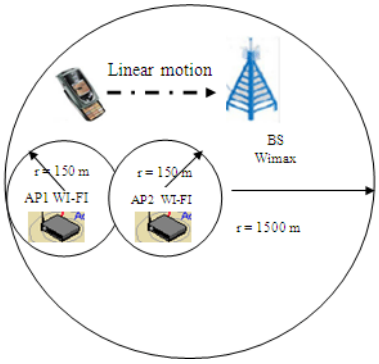


Figure 2. Scenario I topology

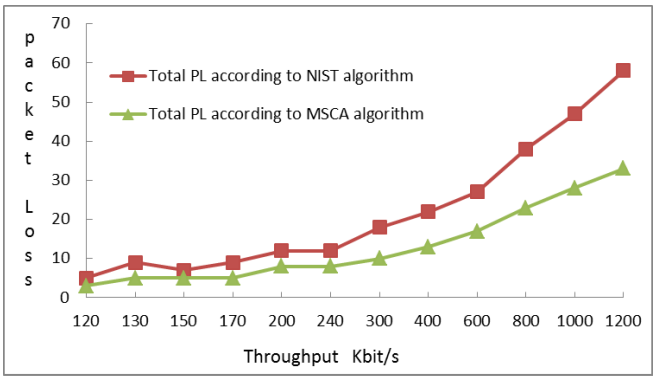


Figure 3. Packet loss according to NIST and MCSA algorithm.

For a user who gives more importance to the number of Packet Loss rather than type of network (WIFI or WIMAX), it is better to follow the strategy of our proposed MCSA algorithm that improves the packet loss ratio by 33% with respect to the NIST. Table II shows the improvement concerning the number of HO and the PL with MSCA for a given throughput.

We can conclude that selecting a destination network using only RSS as indicator does not meet the needs of all users. A more accurate choice of the destination network during handover would consider the ABW of the considered network. A new framework is needed to consider the values of different criteria while taking a decision in order to make a better choice concerning the destination network during handover.

In order to better understand the sequence of events that an MN and a network perform during a successful HO, we provide a short description of messages sequence chart in Figure 4. The dashed and non-dashed blocs represent the flow of handover messages according to NIST and MCSA algorithm. By using our MCSA algorithm, we can save all messages in the dashed bloc, which enables less signaling over the network and improves packet loss.

A detailed description of the events sequence according to the implementation of the IEEE 802.21 standard by NIST, corresponding to our simulation scenario and taking into account our MCSA algorithm or not, is summarized as follows:

1) MIH user on the MN sends MIH Capability Discovery Request to discover the link capability supported (events and commands) for each MAC on each node.
2) MIH user on the MN sends MIH Register Request to register to the local and remote MIHF.
3) MIH User on the MN sends MIH Get Status requesting the available network interface; it discovers the presence of two interfaces (WIFI and WIMAX) supporting events and commands services of MIHF.
4) MIH user on the MN sends MIH Event Subscribe request to subscribe to the events on the given links for local and remote MIHF. This latter sends MIH Event Subscribe response to the MIH User of the MN.

TABLE II. COMPARISON OF HO NUMBER AND PL WITH EACH ALGORITHM

According to NIST algorithm		According to MCSA algorithm	
Number of HO	*Total PL*	*Number of HO*	*Total PL loss*
2 (AP1 to AP2 and AP2 to WIMAX)	20 AP1 to AP2:9 and AP2 to WIMAX:11	1 (AP1 to WIMAX)	10 AP1 to WIMAX:10

5) Once the BS decides the reservation of bandwidth, it informs the MN of the frame structure in the uplink and downlink. It sends the DL-MAP/UL-MAP to the WIMAX interface of the MN. The WIMAX base station is detected and generates a Link Up event toward the MIHF of MN.
MIHF of the MN order the WIMAX interface of MN to connect to the BS.
6) In this case, a router solicitation is sent from the MIPV6 module of MN to the neighbor discovery module of the BS.
7) Neighbor discovery module of the BS replies by sending RA to the MIPV6 module of MN with the network prefix of WIMAX base station = 3.0.0; router-life time= 1800s.
8) MN's WIFI interface receives a beacon message with a power above the threshold value and triggers a Link Detect event; the ABW of AP1 is largely available (not consumed by any other traffic), according to both algorithm MCSA and NIST, AP1 is considered as a better network.
9) MIHF of MN sends a Link Connect message to the WIFI interface of MN; exchanges of association Request/Response between MN and AP1.
10) The WIFI interface of the MN, in its sends a Link Up message to the MIHF and MIH user of MN.
11) Exchanging of RA and router solicitation between the MIPV6 of MN and the neighbor discovery module of AP1 (first WIFI access point).
12) Starting of traffic flow between the WIFI interface of the MN and the correspondent node through the AP1 access point.
13) Once MN reaches the limit coverage of the AP1, it starts receiving the beacon message coming from AP2. Detect the presence of a beacon power above the defined threshold.
14) WIFI interface of the MN sends a Link Going Down and Link Down to the MIH user of MN.
15) MIH user of MN sends a Link Scan request to the MIHF of MN.
16) The WIFI interface of MN sends a probe request and starts scanning the 11 channels of WIFI interface looking for an active one.
17) This message received by AP2, which reply by sending a probe response to the MIH user of MN through its MIHF. MIH user of MN detects the presence of AP2.

According to NIST algorithm, that considers this AP as a better network, decides to handover to it (and continues with step 19). But according to MCSA algorithm, which evaluates the ABW of AP2 before handover to it, find its ABW, consumed by other traffic, very small comparatively to WIMAX, ignore this network and handover to WIMAX directly (jump to step number 20).

18) MIH user sends to MIHF an MIH Link ConFig. This generates a Link Connect to the WIFI interface of MN (connection to AP2).
19) MIH user sends to the MIHF a MIH Link Disconnect, which disconnects the connection between the WIFI interface of MN and AP1. According to NIST algorithm, we continue with step 21 and according to MCSA we jump to step 28 saving by that all steps between 21 and 27. Thus, the signaling overhead decreases.
20) The WIFI interface of MN sends a Link Handover Imminent message to the MIHF of MN.
21) MIH user of MN sends Link Handover Complete to the MHIF of MN.
22) WIFI interface of MN sends Link Up indication event to the MIH user of MN through his MIHF announcing the detection of AP2 (second WIFI access point).
23) MIPV6 module of MN sends router solicitation to the WIFI interface of AP2, which answer by a RA with the new prefix (2.0.1).
24) Starting of traffic between the WIFI interface of MN and correspondent node (CN) through AP2.
25) MIH user sends the MIH Capability Discovery Request and response to the MAC layer of AP2 testing if the Events and Commands events list is supported.
26) The MN reaches the limit coverage of AP2, starts a Link Going Down event, the WIFI interface of MN sends a Link Scan event looking for others network (delaying the connection to WIMAX) do not find anyone else WIMAX.
27) MN connects to WIMAX and a Link Disconnect event with WIFI is triggered and the traffic continues to the end of the simulation through WIMAX.

D. Scenario II: Type of Network Impact

In this scenario, we use the NIST algorithm without extension to show that it needs to be improved by considering other parameters with the signal strength.

1) Topology Description: Figure 5 illustrates the topology of scenario II. During this simulation, we compare the delay taken by MN when it makes a HO from WIFI to WIMAX (Figure 5a) versus handover from WIMAX to WIFI (Figure 5b). Measurements are done according to the handover algorithm of NIST only.

During the simulation, the MN moves from WIFI (AP1) toward the center of BS. Once it reaches the limit coverage of AP1, a "Link Going Down" trigger is fired announcing the need for handover. Since the only available network is 802.16 (WIMAX), the handover is made to this network. We also study the same simulation when the mobile moves from WIMAX to WIFI.

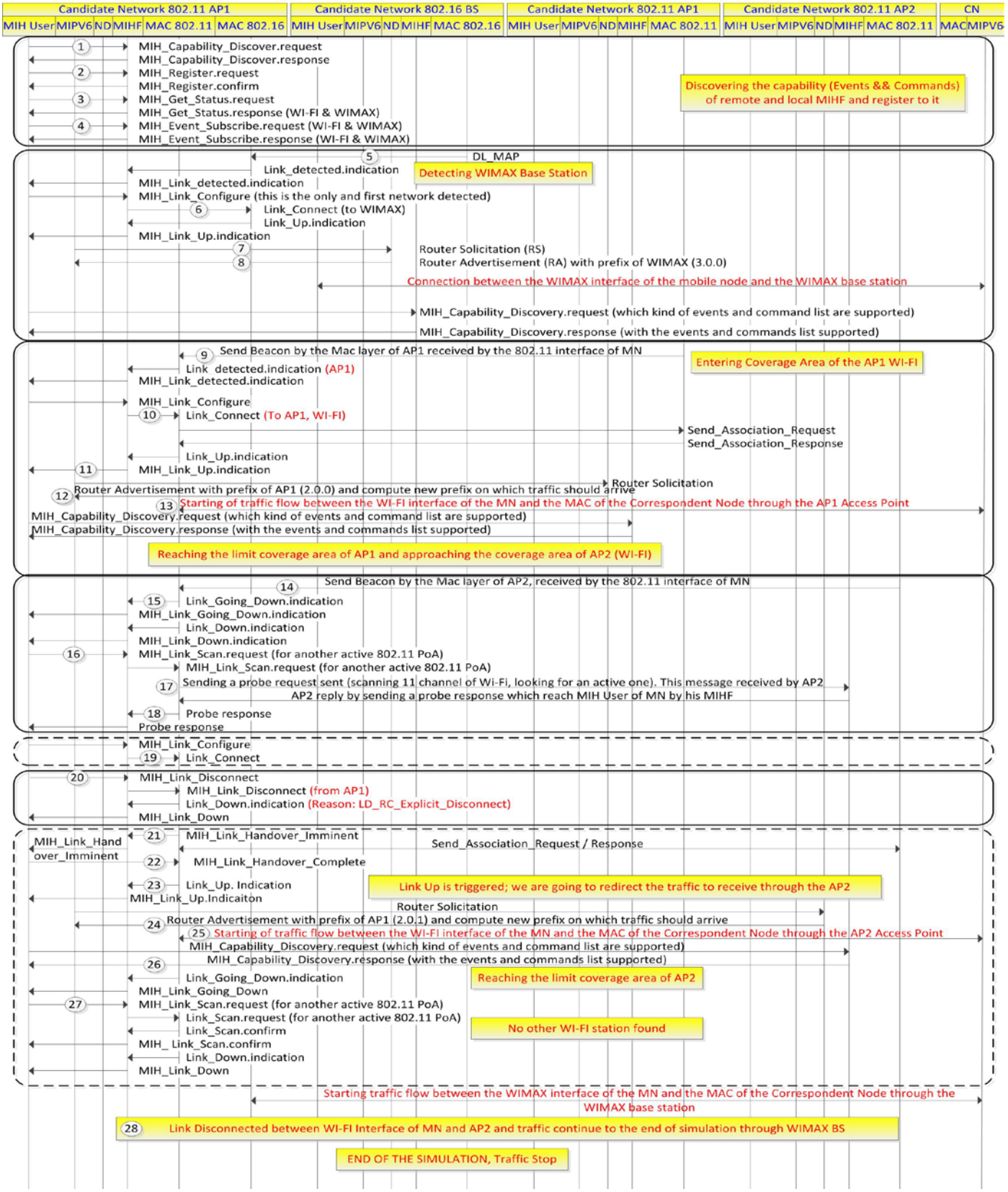


Figure 4. Handover Flow Chart Messages according to NIST and our MCSA algorithm

2) Scenario II Results: Figure 6 shows a decreasing curve of the handover delay as a function of the traffic throughput generated by the MN application. Handover delay is the time difference between the first packet received on the destination network and the last packet received on the current served network. When we increase the throughput, the time between two consecutive packets is smaller and packets reach the destination network earlier, which explains the appearance of the downward curves of handover delay in Figure 6. It shows also that for the same application throughput, the handover delay depends on the type of destination Network (WIFI or WIMAX).

Handover delay from WIMAX to WIFI is smaller than the handover delay from WIFI to WIMAX. When the MN connected to AP1 moves to the center of BS (Figure 5a), it reaches the limit coverage area of AP1 and generates a "Link Going Down" trigger. In this case, a scan process starts looking for a new network delaying the connection to BS (Figure 6). While for handover from WIMAX to WIFI network (Figure 5b), the MN does not trigger this event because it is still in the coverage area of WIMAX (no loss of WIMAX signal) and that's why we have less handover time (Figure 6).

As a conclusion of this experiment, we can say that based on the type of destination network, we can have different values of handover delay and consequently different value of PL.

As shown in Figure 6, we can note that by varying the throughput values between 120Kbit/s and 170Kbit/s, the handover time (WIFI / WIMAX) varies between 275ms and 200ms hence exceeding the maximum acceptable value of the QoS end-to-end delay parameter (150ms) [48] for real time application. This criterion is worthy of consideration when selecting a new network during HO.

E. Scenario III: Speed Impact

1) Topology Description: In this scenario, shown in Figure 7, we study the effect of MN speed on the packet loss during HO. At the beginning, the MN connected to WIMAX, moves to the center of the BS, resulting on a handover to AP1 and AP2 according to NIST algorithm. Once the MN reaches the limit coverage of AP2, it returns to WIMAX network.

2) Scenario III Results: For the three different experimented speeds, the packet loss on WIMAX is null because 802.16e WIMAX is designed to support high speed mobile user [48][49]. Once an MN starts moving toward the center of the BS, it detects the presence of AP1. According to the NIST algorithm, it makes a HO to AP1. Some PL occurs during this HO and the value of this PL increases with mobile node speed (Figure 8) because WIFI, unlike WIMAX, is limited in high-speed transport communications environment [50]; and does not support high speed mobility. Indeed, for a speed of 20m/s we can see a great impact of Doppler Effect on the system performance [51], which is a source of quality of service deterioration.

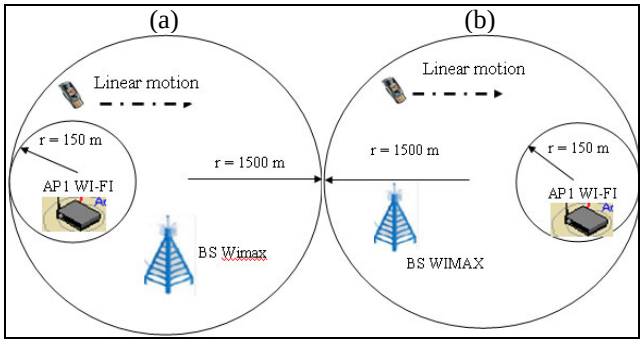


Figure 5. (a) Handover WIFI-WIMAX, and (b) Handover WIMAX-WIFI

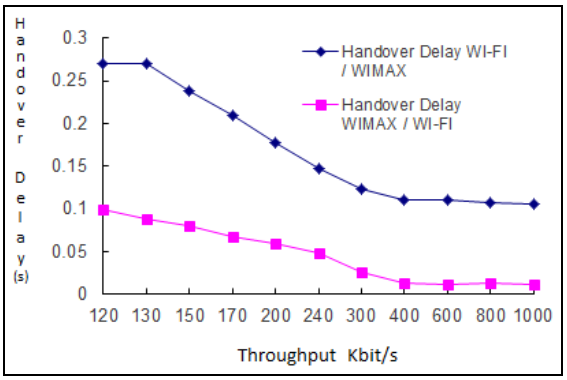


Figure 6. Handover Delay Curves

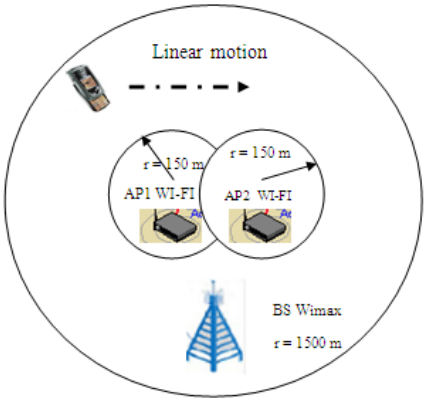


Figure 7. Scenario III topology

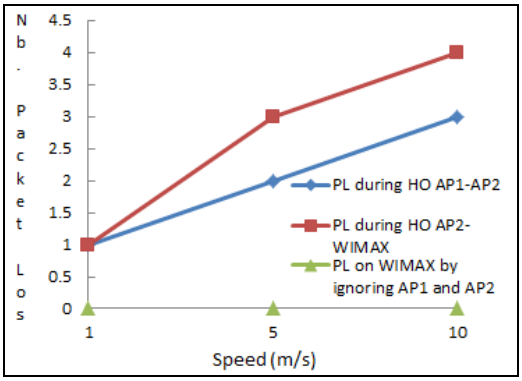


Figure 8. Packet loss as a function of mobile speed

The same process happens during handover from AP1 to AP2 as we experienced other number of packet loss that increases with mobile speed. Moreover, when the MN handover from AP2 to WIMAX some packet loss occur whose number increase with mobile speed. Accordingly, we conclude that users who give importance to the number of packet loss and MN speed would prefer to stay on WIMAX and never stream through AP1 or AP2. Hence, we conclude that NIST fails to meet the requirement of mobile user moving at a speed higher than the pedestrian speed (1m/s). Thus, we argue that there is a need for a new framework that takes into account the user speed.

As a conclusion of the above three experiments, we can say that selection algorithm provided by NIST must be improved by introducing more QoS parameters during selection. As such, a fuzzy logic system complementing the proposed algorithm will be introduced in the subsequent sections.

VI. FUZZY LOGIC SYSTEM

Fuzzy logic is the theory to deal with the multivalued sets and the uncertainty principle [52]. As Figure 12 shows, the fuzzy logic system is composed of three processing units: fuzzification, fuzzy inference rules base engine and a defuzzification unit [53]. A membership function gives the image of the value for a fuzzy set in the range 0 to 1. This value is called Membership Degree. During the fuzzification process, the crisp value of each input parameter is mapped into the appropriate fuzzy set using the corresponding membership function. The input parameters for our proposed model are: Received Signal Code Power measured in UMTS (RSCP), the signal strength on GPRS (RXLEVEL), the ratio of the received energy per chip measured on UMTS (Ec/Io), the signal quality received on GPRS (RXQUAL) and the ABW. A membership function can take different forms: triangular, trapezoidal, Gaussian and sigmoidal [54]. The membership functions of the input parameters in our system are shown in Figure 9, Figure 10 and Figure 11, respectively. For (Ec/Io) and RSCP the input values are transformed into one of the four fuzzy sets (Bad, Acceptable, Good and Very good) while the ABW is mapped into one of the three fuzzy set (Low, Medium and High). Due to their simplicity and computational efficiency, the triangular form combined with the trapezoid one is used for the membership functions. Moreover, this form of membership function has been widely used in real time applications [55]. The universe of discourse of each input parameter is depicted in Table IV.

Two schemas exist for the fuzzy inference rules base namely Sugeno [56] and Mamdani [57]. The former schema is ideal for linear technique and gives a crisp value as a result while the latter is a good pattern for an expert knowledge system in the form of IF-THEN [58] but gives a symbolic value as a result. The fuzzy rule base, in our case, is a collection of IF-THEN rules that help to choose the best network in the context of QoS guarantee for a given user. Our fuzzy rules base is extracted from the observation done on more than 9500 voice data records. This data was retrieved from the server of an operating mobile telecommunications company (Alfa) after a long drive test. Among the 632 cases of handovers we received, 100 cases were for vertical handover. As the scope of our research is for vertical handover, we consider only these 100 cases between GPRS and UMTS networks.

After analysis, we find that these 100 cases of handover follow a set of rules that will constitute our fuzzy rules base (see Table III). Moreover, these rules were completed thanks to experts from the Alfa telecom company in order to cover all remaining handover scenarios. The fuzzy inference engine will be applied on the fuzzy rules base to help choosing the best network during handover.

The role of the deffuzifier is to compile the output of the fuzzy inference engine and convert it from natural language to a crisp value using the centroid method. This method computes the gravity center of the membership function for a given fuzzy value. The final crisp output corresponds to a scoring value for each candidate network between 0 (the worst network) and 1 (the best network).

For the sake of simplicity, the following abbreviations are used for the fuzzy sets: H for High, M for Medium, L for Low, B for Bad, A for Acceptable, G for Good and VG for Very Good.

VII. FUZZY LOGIC HANDOVER DECISION ALGORITHM

We consider the scenario given by Figure 13. The MN connected to Network 1, reaches its limit coverage area and needs to select the best network among the available ones in its vicinity. Our proposed algorithm will use the three parameters Ec/Io or RXQUAL, RSCP or RXLEVEL and ABW of the destination networks as input, see Figure 12. Values of the input parameters for each candidate network are given by Table V. The fuzzification process maps, for each candidate network, the three input parameters values to their name(s) of membership function(s) and memberships degree(s) in the function(s). For example, Figure 11 shows the memberships degree of the Ec/Io input parameter for Net. #2 (-11 dB) with the membership functions Good (G, 0.25) and Acceptable (A, 0.75). Table VI shows the (Membership-FN, Membership-Degree) for each candidate network. Each triplet (EC/Io, RSCP and ABW) of input parameters can fire one or more rules in our base with different strength α_i. Before calculating the crisp output value by defuzzification, we must calculate the firing strength of each rule as the minimum of the triplet input values (see Table VII). For each value of a handover output (Ho output), a numerical value between 0 and 1 is assigned. (Highly Recommended (HR) = 1, Recommended (R) = 0.5, Lowly Recommended (LR) = 0.25 and Not Recommended (NR) = 0). Networks with bad (B) or low (L) value for any input parameters are considered as not recommended. Our inference base looks only for recommended networks. Finally, the crisp value that represents the score for each candidate network (between 0 and 1) is given by the following formula [59]:

$$z_0 = \sum_{i=1}^{n} \alpha_i z_i \Big/ \sum_{i=1}^{n} \alpha_i \quad (1)$$

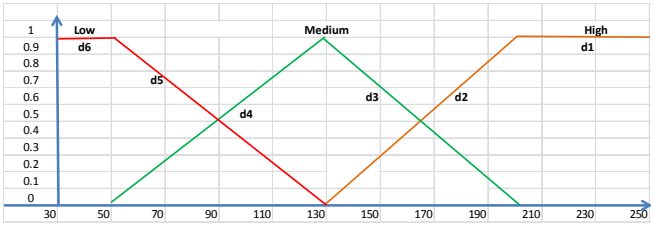


Figure 9. Membership Function for ABW

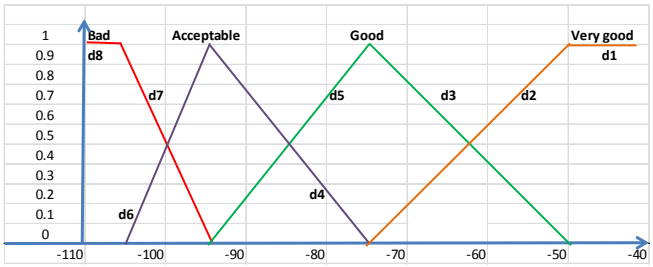


Figure 10. Membership Function for RSCP

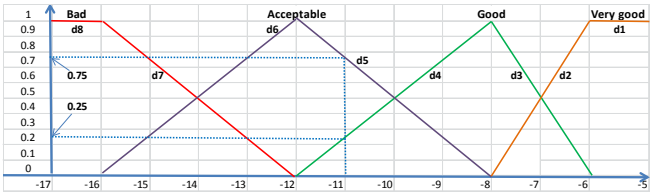


Figure 11. Membership Function for EC/Io

TABLE III. INFERENCE RULES BASE

Rule No.	ABW	Ec/Io	RSCP	Ho output
1	L	A	M	LR
2	L	A	H	R
3	L	G	M	R
4	L	G	H	R
5	L	VG	M	R
6	L	VG	H	R
7	M	A	M	R
8	M	A	H	R
9	M	G	M	R
10	M	G	H	HR
11	M	VG	M	HR
12	M	VG	H	HR
13	H	A	M	R
14	H	A	H	R
15	H	G	M	HR
16	H	G	H	HR
17	H	VG	M	HR
18	H	VG	H	HR

TABLE IV. UNIVERSE OF DISCOURSE

RSCP (dBm)	Bad < -105	Acceptable -95 to -105	Good -95 to -75	Very Good >-50
EC/Io (dB)	Bad < -16	Acceptable -16 to -12	Good -12 to -8	Very Good -8 to -6
ABW (Mbps)	Low < 50	Medium 50 to 130	High 130 to 250	

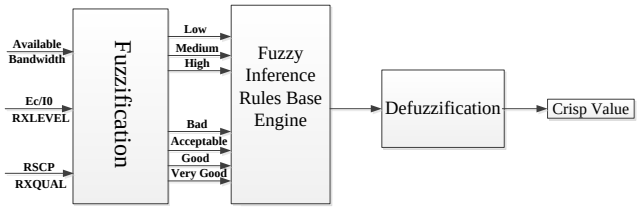


Figure 12. Fuzzy Logic Processing Units

Where α_i is the firing strength for a given rule and z_i is the numerical value assigned to the handover output value of each rule. The crisp scoring value for each network should be calculated. The network that has the nearest value to 1 is the most recommended one. For example, calculation of the crisp scoring value for Net. 2 are as follows: the input parameters of the second network fire rules number 1, 2, 3 and 4 of our base with different strength. The firing strength of each rule is calculated as the minimum of all input parameter's value for a given rule. Table VII shows only the fired rules with strength greater than zero. It would be pointless to show rules whose firing strength is null. Table VII shows the scoring value for each recommended network in the vicinity of the MN. These scores are calculated according to formula given in (1). For example, score of the Network #2 is calculated as follow:

$$\frac{\left[(0.4 \text{ x } 0.25)+(0.5\times 0.5)+(0.25\times 0.5)+(0.25\times 0.5)\right]}{(0.4+0.5+0.25+0.25)} = 0.42$$

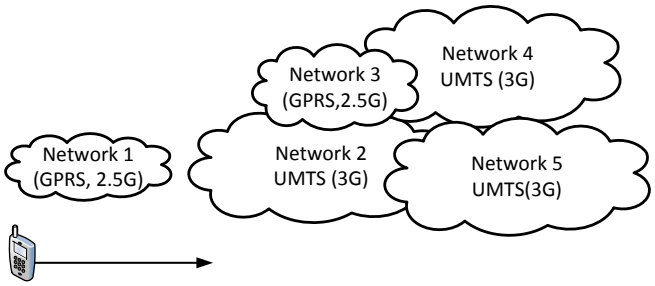


Figure 13. Studied scenario

TABLE V. PARAMETERS VALUES FOR CANDIDATE NETWORKS

	Net. 2	**Net. 3**	**Net. 4**	**Net. 5**
RSCP (dBm)	-100	-80	-90	-70
EC/Io (dB)	-11	-14	-7.5	-7.5
ABW (Mbps)	170	110	220	90

TABLE VI. (MEMBERSHIP-FN, MEMBERSHIP-DEGREE) FOR EACH CANDIDATE NETWORK

	Net. 2	**Net. 3**	**Net. 4**	**Net. 5**
RSCP (dBm)	(A, 0.5) (B, 0.5)	(A, 0.25) (G, 0.75)	(A, 0.75) (G, 0.25)	(VG, 0.2) (G, 0.8)
EC/Io (dB)	(G, 0.25) (A, 0.75)	(A, 0.5) (B, 0.5)	(VG, 0.25) (G, 0.75)	(VG, 0.25) (G, 0.25)
ABW (Mbps)	(H, 0.6) (M, 0.4)	(L, 0.25) (M, 0.75)	(H, 1)	(L, 0.5) (M, 0.5)

TABLE VII. FIRING STRENGTH OF EACH RULE AND NETWORK SCORING FOR DIFFERENT CANDIDATE NETWORKS

Net. #	Fired Rules #	Strength of the fired rules	Network scoring
2	1	Min (0.5, 0.75, 0.4) = 0.4	0.42
	2	Min (0.5, 0.75, 0.6) = 0.5	
	3	Min (0.5, 0.25,0.4) = 0.25	
	4	Min (0.5, 0.25, 0.6) = 0.25	
3	1	Min (0.25, 0.5, 0.75) = 0.25	0.41
	7	Min (0.75, 0.5,0.75) = 0.5	
4	4	Min (0.75, 0.75, 1) = 0.75	0.66
	6	Min (0.75, 0.25, 1) = 0.25	
	10	Min (0.25, 0.75, 1) = 0.25	
	12	Min (0.25, 0.25, 1) = 0.25	
5	9	Min (0.8, 0.75, 0.5) = 0.5	0.73
	11	Min (0.8, 0.25, 0.5) = 0.25	
	15	Min (0.2, 0.75, 0.5) = 0.2	

By a simple comparison of the network scoring column of Table VII, we found that Net. #5 is the most recommended one among all available networks in the vicinity of the MN.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we have evaluated the effect of some parameters like Radio Signal Strength, available bandwidth, type of network (802.11 or 802.16) and mobile speed for choosing the best network in the vicinity. We conclude that choosing a network based on the Radio Signal Strength only is not always a good strategy. The experiments that we conducted using the NS2 showed that the inclusion of additional parameters significantly improves the packet loss ratio and so the QoS guarantee for mobile users. Even with the significant improvements that were introduced with MCSA algorithm, we investigated a model that is based on fuzzy logic to address the short-falls of our modified and enhanced algorithm. The new integrated system provides a better comprehensive solution. The limitation of the fuzzy logic work lies in the fact that the records were collected and experimented with cover 2G & 3G networks HO, so we have to extend these records to those of WIFI and WIMAX. However, our theoretical calculations prove that irrespective of networks type, the results should be similar to a great extent. It is worth mentioning that several attempts were made to obtain needed data from the USA, but without much success due to confidentiality and intellectual property concerns.

In future work, we will propose a framework with a generic model that takes into consideration different levels of constraints such as network parameters with users and operators preferences to improve the selection of the best candidate network and optimize QoS parameters in terms of packet loss ratio, delay and jitter for real time applications. In addition, attempts will be made to secure data regarding WIFI and WIMAX handover in order to validate the fuzzy logic model findings and prove the prescribed assumptions put forward in this research work. Furthermore, the proposed fuzzy logic algorithm will be implemented in NS2. At that point, a concrete comparison will be conducted among MCSA, the fuzzy logic algorithm and a customized model based on multiple linear regression strategy that is under investigation to better select a destination network.

ACKNOWLEDGMENT

The authors would like to thank the Lebanese ministry of communications, and Mrs. Wafaa Bazzi, Head of RAN Research & Studies Unit at ALFA-ORASCOM TELECOM for providing the voice data records. We would like also to thank Dr. Khalil Fakih from Ericsson, Dr. Bassam Hussein and Dr. Adnan Harb, from the Lebanese International University for all their consultation and helps.

REFERENCES

[1] A. Rahil, N. Mbarek, and O. Togni, “Smart Network Selection and Packet Loss Improvement during Handover in Heterogeneous Environment”, the Ninth International Conference on Networking and Services - ICNS, Mar. 2013 pp. 185-192.

[2] IEEE 802.21, Media Independent Handover Standard http://standards.ieee.org/ getieee802/ download /802.21-2008.pdf. 14.12.2013.

[3] IEEE P802.21/D03.00, “The Network Simulator NS-2 NIST add-on—IEEE 802.21 model (based on IEEE P802.21/D03.00)”, National Institute of Standards and Technology (NIST), Jan. 2007.

[4] M. M. Rehan, “Investigation of IEEE 802.21 Media Independent Handover”, PhD thesis, Mohammad Ali Jinnah University, 2009.

[5] K. Taniuchi, Y. Ohba, V. Fajardo, S. Das, M. Tauil, Y. Cheng, A. Dutta, D. Baker, M. Yajnik, and D. Famolari, “IEEE 802.21: Media Independent Handover: Features, Applicability, and Realization”, IEEE Communications Magazine, vol. 47, pp. 112–120, Jan. 2009, doi:10.1109/MCOM.2009.4752687.

[6] B. R. Chandavarkar and D. G. Reddy, “Improvement in Packet Drop during Handover between WiFi and WiMax”, International Conference on Network and Electronics Engineering IPCSIT, vol. 11, Sep. 2011, pp. 71-75, doi:10.7763/IPCSIT.

[7] F. Siddiqui, S. Zeadally, H. El-Sayed, and N. Chilamkurti, “A dynamic network discovery and selection method for heterogeneous wireless networks” Int. J Internet Protocol Technology, vol. 4, pp. 99-114, Jul. 2009, doi:10.1504/IJIPT.2009.027335.

[8] F. Jiadi, J. Hong, and L. Xi, “User-Adaptive Vertical Handover Scheme Based on MIH for Heterogeneous Wireless Networks”, Wireless Communications, Networking and Mobile Computing, WiCom 09. 5th International Conference, Sep. 2009, pp. 1-4, doi:10.1109/WICOM.2009.5302424.

[9] W. Ying, Z. Yun, Y. Jun, and Z. Ping, “An Enhanced Media Independent Handover Framework for Heterogeneous Networks”, IEEE Vehicular Technology Conference, VTC, Spring 2008, pp. 2306–2310, doi:10.1109/VETECS.2008.512.

[10] O. Ormond, G. Muntean, and J. Murphy, “Network Selection Strategy in Heterogeneous Wireless Networks”, Proc. of IT&T 2005: Information Technology and Telecommunications, Oct. 2005.

[11] E. Bircher and T. Braun, “An Agent-Based Architecture for Service Discovery and Negotiations in Wireless Networks”, 2nd Intl. Conf. on Wired/Wireless Internet Communications, vol. 2957, Feb. 2004, pp. 295-306, doi:10.1007/978-3-540-24643-5_26.

[12] C. Cicconetti, F. Galeassi, and R. Mambrini, “Network-Assisted Handover for Heterogeneous Wireless Networks”,

GLOBECOM Workshops (GC Wkshps), IEEE, Dec. 2010, pp. 1-5, doi:10.1109/GLOCOMW.2010.5700294.

[13] J. M. Arraez, M. Esseghir, and L. M. Boulahia, "An Implementation of Media Independent Information Services for the Network Simulator NS-2", the 8th Annual IEEE Consumer Communications and Networking Conference - Wireless Consumer Communication and Networking, Jan. 2011, pp. 492–496, doi:10.1109/CCNC.2011.5766519.

[14] V. Andrei, E. C. Popovici, and O. Fratu, "Solution for Implementing IEEE 802.21 Media Independent Information Service", 8th International Communications Conference on, IEEE, Jun. 2010, pp. 519–522, doi:10.1109/ICCOMM.2010.5509008.

[15] V. Andrei, E. C. Popovici, O. Fratu, and S. V. Halunga, "Development of an IEEE 802.21 Media Independent Information Service", Automation Quality and Testing Robotics (AQTR), IEEE International Conference on, vol. 2, 2010, pp. 1-6, doi:10.1109/AQTR.2010.5520819.

[16] A. Ghittino, N. Di Maio, and D. Di Tommaso, "WiFi network residual bandwidth estimation: A prototype implementation", Wireless On-demand Network Systems and Services (WONS), 2012, pp. 43-46, doi:10.1109/WONS.2012.6152234.

[17] M. Q. Khan and S. H. Andresen, "An Intelligent Scan Mechanism for 802.11 Networks by Using Media Independent Information Server (MIIS)", Advanced Information Networking and Applications (WAINA), IEEE Workshops of International Conference on, Mar. 2011, pp. 221–225, doi:10.1109/WAINA.2011.26.

[18] Y. Y. An, B. H. Yae, K. W. Lee, Y. Z. Cho, and W. Y. Jung, "Reduction of Handover Latency Using MIH Services in MIPv6", Proc. of the 20th IEEE International Conference on Advanced Information Networking and Applications (AINA 06), vol. 2, Apr. 2006, pp. 229-234, doi:10.1109/AINA.2006.283.

[19] C. F. Kwong, T. C. Chuah, and S. W. Lee, "Adaptive Network Fuzzy Inference System (ANFIS) Handoff Algorithm", International Journal of Network and Mobile Technologies ISSN 1832-6758 Electronic Version, vol. 1, no. 2, pp. 195–198, Nov. 2010.

[20] P. Dhand and P. Dhillon, "Handoff Optimization for Wireless and Mobile Networks using Fuzzy Logic", International Journal of Computer Application, vol. 63, no.14, pp. 0975–8887, Feb. 2013.

[21] P. T. Kene and M. S. Madankar, "FLC Based Handoff Mechanism for Heterogeneous Wireless Network: A Design Approach", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, ISO 9001:2008, Certified Journal, vol. 3, Issue 2, Feb. 2013.

[22] T. C. Ling, J. F. Lee, and K. P. Hoh, "Reducing Handoff Delay In Wlan Using Selective Proactive Context Caching", Malaysian Journal of Computer Science, vol. 23, no. 1, pp. 49-59, 2010.

[23] Z. Yan, H. Luo, Y. Qin et al., "An adaptive multi-criteria vertical handover framework for heterogeneous networks," in Proceedings of the International Conference on Mobile Technology, Applications, and Systems, Sep. 2008, pp. 141–147.

[24] K. Vasu, S.Maheshwari, S. Mahapatra, and C. S. Kumar, "QoS aware fuzzy rule based vertical handoff decision algorithm for wireless heterogeneous networks," in Proceedings of the National Conference on Communications (NCC '11), Jan. 2011, pp. 1–5.

[25] A. S. Sadiq, K. Abu Bakar, and K. Z. Ghafoor, "A Fuzzy Logic Approach for Reducing Handover Latency in Wireless Networks", journal of Network Protocols and Algorithms, ISSN 1943-3581, vol. 2, no. 4, 2010.

[26] A. Aziz, S. Rizvi, and N .M. Saad, "Fuzzy Logic based Vertical Handover Algorithm between LTE and WLAN", Intelligent and Advanced Systems (ICIAS) International Conference on, 2010, pp. 1-4, doi:10.1109/ICIAS.2010.5716261.

[27] T. Thumthawatworn, A. Pervez, and P. Santiprabhob, "Adaptive Modular Fuzzy-based Handover Decision System for Heterogeneous Wireless Networks", International Journal of Networks and Communications, 2013, pp. 25-38, doi:10.5923/j.ijnc.20130301.04.

[28] V. S. Krishna and L.Rajesh, "Implementation of Fuzzy Logic for Network Selection in Next Generation Networks", Recent Trends in Information Technology (ICRTIT), International Conference on, 2010, pp. 595–600, doi:10.1109/ICRTIT.2011.5972475.

[29] P. Munoz, R. Barco, I. de la Bandera, and S. M. Luna-Ramírez,"Optimization of a Fuzzy Logic Controller for Handover-based Load Balancing", Vehicular Technology Conference (VTC Spring), IEEE 73rd, 2011, pp. 1–5, doi:10.1109/VETECS.2011.5956148.

[30] K. C. Foong, C. T. Chee, and L. S. Wei, "Adaptive Network Fuzzy Inference System (ANFIS) Handoff Algorithm", Future Computer and Communication, ICFCC 2009, International Conference on, 2009, pp. 195–198, doi:10.1109/ICFCC.2009.95.

[31] H. Silva, L. Figueiredo, C. Rabadão, and A. Pereira, "Wireless Networks Interoperability - WIFI Wimax Handover", Proc. Systems and Networks Communications, Fourth International Conference on, (ICSNC 09), Sep. 2009, pp. 100–104, doi:10.1109/ICSNC.2009.99.

[32] WIFI Alliance, http://www.WIFI.org 14.12.2013.

[33] H. Velayos and G. Karlsson, "Techniques to reduce the IEEE 802.11b handoff time", Proc. Communications, IEEE International Conference on, Jul. 2004, pp. 3844–3848, doi:10.1109/ICC.2004.1313272.

[34] N. T. Dao, R. A. Malaney, E. Exposito, and X. Wei, "Differential VoIP Service in WIFI Networks and Priority QoS Maps", IEEE Globecom, vol. 5, Dec. 2005, pp. 2653-2657, doi:10.1109/GLOCOM.2005.1578241.

[35] B. XieI, W.Zhou, and J.Zeng, "A Novel Cross-Layer Design with QoS Guarantee for WiMAX System", Pervasive Computing and Applications, ICPCA, Third International Conference on, vol. 2, Oct. 2008, pp. 835-840, doi:10.1109/ICPCA.2008.4783726.

[36] IEEE P802.16, "IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems", Feb. 2009.

[37] IEEE P802.16m/D4, "Air Interface for Fixed and Mobile Broadband Wireless Access Systems: Standard IEEE P802.16e", 2010.

[38] M. A. Awal and L. Boukhatem, "WiMAX and End-to-End QoS Support", White Paper, Univ. of Paris-Sud 11, CNRS. May 2009.

[39] 3GPP, 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface (Release 6) Technical Specification 29.060 V6.19.0, (2008-09).

[40] 3GPP, 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Numbering, addressing and identification (Release 9) Technical Specification 23.003 V9.0.0 (2009-09).

[41] N. Purohit and S. Tokekar, "Survivability index for a GPRS network", Networks 2008 ICON 2008, 16th IEEE International Conference on, 2008, pp. 1–4, doi:10.1109/ICON.2008.4772660.

[42] W. Cheng, L. Chen, Y. Dou, and Z. Lei, "Mobile User Time-Of-Day Regularity Analysis in GPRS Network", Communication Technology and Application (ICCTA 2011), IET International Conference on, 2011, pp. 713–717, doi:10.1049/cp.2011.076.

[43] http://www.3gpp.org/3GPP. 14.12.2013.

[44] W. Liu, Y. Liu, R. Li, and P. Wang, "Research and development of communication between PC and mobile base on embedded system and GPRS", Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2nd International Conference on, 2011, pp. 4180–4183, doi:10.1109/AIMSEC.2011.6010034.

[45] ETSI ETR 332, Security Techniques Advisory Group; Security requirements capture, http://www.etsi.org. 14.12.2013.

[46] 3GPP TS 29.060. General Packet Padio Service (GPRS) Tunnelling Protocol (GTP) across the Gn and Gp Interface.

[47] M. G. S. Bhuyan, M. S. H. Mollah, Z. Ahmad, and S.M. Rahman, "A New Approach of Efficient Soft Handover Management for Proposed UMTS Network Architecture", Environmental and Computer Science, ICECS '09. Second International Conference on, 2009, pp. 353–355, doi:10.1109/ICECS.2009.89.

[48] International Telecommunication Unit (ITU) G.114 specification, 1996.

[49] S. Murawwat and T. Javaid, "Speed & Service based handover Mechanism for cellular WIMAX", Computer Engineering and Technology (ICCET), 2nd International Conference on, vol. 1, April 2010, pp. V1-418-V1-422, doi:10.1109/ICCET.2010.5486067.

[50] Z. Zhao, "WIFI in High-Speed Transport Communications", Intelligent Transport Systems Telecommunications, (ITST), 9th International Conference on, Oct. 2009, pp. 430–434, doi:10.1109/ITST.2009.5399314.

[51] M. Thaalbi and N. Tabbane, "Vertical Handover between WiFi Network and WiMAX Network According to IEEE 802.21 Standard", Technological Developments in Networking, Education and Automation, pp. 533-537, 2010, doi:10.1007/978-90-481-9151-2_93.

[52] J. Wang, J. B. Yang, and P. Sen, "Safety analysis and synthesis using fuzzy sets and evidential reasoning", Reliability Engineering & System Safety, vol. 47, Issue 2, pp. 103-118, 1995.

[53] J. S. Martínez, R. I. John, D. Hissel, and M. Péra, "A survey-based type-2 fuzzy logic system for energy management in hybrid electrical vehicles Original Research Article", Information Sciences, vol. 190, pp. 192-207, 2012.

[54] J. Zhao and B. K. Bose, "Evaluation of membership functions for fuzzy logic controlled induction motor drive", IECON 02, Industrial Electronics Society, IEEE 2002 28th Annual Conference of the IEEE Industrial Electronics Society, vol.1, 2002, pp. 229-234, doi:10.1109/IECON.2002.1187512.

[55] L. L. Bello, G. A. Kaczynski, and O. Mirabella, "Improving the real-time behavior of ethernet networks using traffic smoothing", Industrial Informatics, IEEE Transactions on, vol. 1, Issue: 3, pp.151-161, 2005, doi:10.1109/TII.2005.852071.

[56] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control", IEEE Transactions on Systems Man And Cybernetics, vol. 15, no. 1, pp. 116–132, 1985.

[57] E. Mamdani, and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller", International Journal of Man-Machine Studies, vol. 7, no. 1, pp. 1–13, 1975.

[58] A. Kaur and A. Kaur,"Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system", International Journal of Soft Computing and Engineering, vol. 2, no. 2, 2012.

[59] E. Kozlowska, "Basic Principles of Fuzzy Logic", ISSN 1214-9675, http://access.feld.cvut.cz/view.php?cisloclanku=2012080002. 14.12.2013.

Information Visibility in Public Transportation Smart Card Ticket Systems

Maja van der Velden
Department of Informatics
University of Oslo
Oslo, Norway
e-mail: majava@ifi.uio.no

Alma Leora Culén
Department of Informatics
University of Oslo
Oslo, Norway
e-mail: almira@ifi.uio.no

***Abstract*—This paper discusses the role of information visibility in public transportation ticket systems. The case is the replacement of paper tickets with smart card tickets in the Oslo region in Norway. By contrasting the visibility of ticket information to users of paper tickets and smart card tickets, this paper describes the move from local information on paper tickets to distributed information on smart cards. We used a qualitative content analysis of reader comments to an online opinion article on the new smart card system to analyse the effect of the loss of ticket information. Using the concept of 'networked visibility', this paper argues that the move from paper to smart card ticket resulted in less informed travelers and more informed providers. We discuss issues around the visibility of ticket information and present diverse design solutions, including an augmented reality application, which may address the ticket information needs of public transportation users.**

Keywords—Information Visibility; Smart Card; Ticket Information; Public Transportation; Mobile Apps

I. INTRODUCTION

In 2009, Ruter, the public transportation company in the Oslo region in Norway, began to replace the paper tickets with a contactless smart card [1]. By February 2013, Ruter stopped the sale of the last of the paper tickets, the coupon card. This prompted a former user of the coupon card to send a letter to the editor to one of Norway's national newspapers. In *We don't want this!*, the user expresses her frustration with the new smart card system [2]. She argues that the new ticket system leaves her permanently insecure about the validity of her ticket.

This user was not alone in her critique. Ruter has experienced a high level of user dissatisfaction with the new system, as well as critique for the way the company dealt with it. We identify information visibility as an important factor contributing to the wide spread dissatisfaction. In design of information systems and usability studies, the visibility of information is related to the visibility of a system's status. Being informed about a system's status is one of the ways in which users receive feedback on a system's use or performance. Studies of the visibility of information on smart cards have been implemented in several sectors, such as supply chain management, the automotive industry, and the healthcare sector [3]–[6]. In public transportation studies, information visibility is discussed as part of radio-frequency identification (RFID) applications for identifying and tracking vehicles, e.g., [7]. We are not aware of information visibility studies of ticketing systems in the public transportation sector, although some other interesting and related issues have been reported in [8]–[10].

Rust and Kannan [6] consider ubiquitous computation to offer a great opportunity towards enhancement of user experience. How could a smart card enhance user experience? A smart card is a credit card size plastic card containing a microchip with antenna for contactless communication with a card reader; see Fig. 1 for the card and Fig. 2 for card readers. The embedded sensor technology provides an opportunity to use the card for several purposes, integrating diverse systems into one, e.g., transportation and event services, or the annual subscription to a museum. It also enables service providers to store information about user's behavior on the card, in order to offer better one-to-one services.

When moving from paper-based practices [11] to practices where computing and communications technologies are embedded into everyday objects, there are many ways *"to capitalize on our familiarity, skill and experience in dealing with the everyday world around us"* writes Dourish [12]. New values, new possibilities, but also new concerns may emerge when interacting with these familiar objects with and without embedded technology. In the case of the RFID-enhanced transportation ticket, we could not capitalize on familiarity, because, one major characteristic of the ticket was lost: the visibility of information.

In this paper, we investigate the role of information visibility in the use of the smart card ticket. Our study was guided by two questions:

1. What is the role of information visibility in a public transportation ticketing system?
2. What design solutions are possible to increase information visibility in the smart card ticket?

Our case is the 'Reisekort', the smart card ticket used in the public transportation system in the Oslo region. Our focus is on the basic information that is needed to use the smart card as a valid transportation ticket. We identified three basic ticket information needs of public transportation users: the type of the ticket, the value of the ticket, and the duration of the ticket (see Table I). There are other ticket information needs, such as the price of one trip, an overview or log of implemented trips, and an overview of past travel expenses, but they do not add significantly to our argument.

In *We don't want this!*, the author proposes one solution to the lack of information visibility of the smart card: "receive a receipt every time you use the card". The rational behind this solution is that one will always have a visual confirmation of basic ticket information: what type of card is used and how much money and time are left on the card.

Many readers responded to *We don't want this!*. In this paper, we analyze the reader comments and discuss new proposals for design. We start with a discussion of ticket information needs, comparing paper tickets and smart card tickets. In Section III, we discuss our method and present our analysis of reader comments. In Section IV, we discuss our findings and in Section V, we use the analysis to propose or discuss some design interventions.

Although our case is a local phenomenon, our findings may contribute to the study of networked near field communication (NFC) services in general. Information visibility, we argue, is an important aspect in user satisfaction and for guiding the (re-) design of smart card tickets.

II. FROM PAPER TICKET TO SMART CARD

In *We don't want this!*, the author compares the paper coupon card with the pay-as-you-go smart card. Since the author is a pensioner, she used to have a discount coupon card, recognizable by a different color than the regular coupon card. The card was unregistered and could be used by several people traveling together. The monetary value of the ticket was printed on the card and the empty spots on the card visualized how many trips could still be made with the card. Once the coupon card was stamped in a ticket machine on board the bus or tram, or on the platform of the train or metro, the validity of the ticket could be read on the timestamp.

Figure 1. A smart card ticket used in public transportation in Norway.

In terms of information visibility, the user had the following information available at all times and without the use of extra technology (see Table I):

- The type of ticket: shape, color, name of ticket
- The value of the ticket: printed on the ticket
- The duration of the ticket: date and timestamp on the ticket

This information has become invisible in the smart card ticket. All smart cards, be it a pay-as-you-go card, a weekly, monthly or day-pass, a regular or discount card, have the same shape, color, and name (see Fig. 1). The type of ticket, value of the ticket, and duration of the ticket can only become visible when an external reader is used (see Fig. 2).

TABLE I. BASIC TICKET INFORMATION NEEDS OF PUBLIC TRANSPORTATION USERS

Type of ticket   The type of the ticket refers to the different kinds of tickets available. We can differentiate between types of tickets based on the number of trips and types of tickets based on the particular period they cover independent of the number of trips (day, week, month, and year). Other types are registered or unregistered (anonymous) tickets, and regular and discount tickets (for children and people over 65 years old).   • A popular paper ticket in Oslo was the unregistered 8 trip-ticket, the so-called *Flexicard*, which was available as a regular ticket and a discount ticket. The Flexicard could be used by more than one traveler at the same time. There is no smart card variation of this ticket.   • A popular smart card ticket is the pay-as-you-go card, which can be topped up when needed. This ticket can only be used by one traveler at the time. There is no paper variation of this ticket.
Value of ticket   The type of ticket decides the monetary value of the ticket. In the case of the pay-as-you-go card, the value of the ticket depends on how much money the user has put on the card. The monetary value of all tickets diminishes with use. A ticket has zero value when the duration of the ticket has expired or when the monetary value is below the price of a ticket. In the case of pay-as-you-go cards, any amount less than the value of a single ticket may be left on the card. Paper tickets did not have this characteristic.
Duration of ticket   The duration of the ticket is decided by the date and time stamp of a ticket and varies for the different ticket types. Registered monthly paper tickets were sent automatically by mail to the user before the monthly ticket expired. Registered smart cards can be automatically topped up (in case of a pay-as-you-go card) or extended (in case of the 30 days card).

Figure 2. Smart card ticket readers at metro stations and on the bus.

How does this affect the use of the smart ticket? In this section, we look at information visibility in three activities public transportation usually users engage in: purchasing a ticket, using a valid ticket and having an expired ticket.

A. *Purchasing a ticket*

When a ticket is purchased, the three pieces of basic information (type of the ticket, value of the ticket and duration of the ticket) are given by the user to a sales person or are selected by the user on a vending machine or on the public transportation website. In addition, public transportation users also have the choice between a registered and an unregistered ticket. 'Registered' means that the name and date of birth of the user is registered with the public transportation provider. 'Unregistered' means that the user is anonymous and that the age of the user is unknown.

The information provided when the ticket is purchased is at all times visible on the paper ticket in the form of printed text (type, value, duration), the size of the ticket (type), the color of the ticket (type), and the shape of the ticket (type). For example, the Flexicard, the 8-trip ticket (see Fig. 3), was the only folded paper ticket. It had a pre-printed text to indicate the value of the ticket (kr.180) and the word 'voksne' (adults) to indicate that it was a regular ticket. Also the color of the ticket informed the traveler that it was a regular ticket. Discount tickets have a different color. The printed text on a strip is a timestamp, indicating the time when the one-hour validity of the ticket ends. This timestamp is added to the ticket when a traveler enters a metro platform or a bus or tram and inserts the card in a ticket stamp machine (see Fig. 3).

Figure 3. A paper ticket (left) and a ticket stamp machine (right).

An important difference between the paper ticket and the smart card ticket is that ticket information is never visible on the smart card ticket itself; information becomes visible when the smart card ticket is networked with other devices. This can happen in four different ways: via stationary ticket readers positioned at the entrances of stations and platforms of the metro and train and inside buses and trams, scanners handheld by human ticket controllers, smart card terminals at the point of purchase, and the Internet (only for registered smart card holders, see Fig. 4).

Travelers can always buy more than one ticket. For example, a user of the 30-day ticket may buy a new 30-day ticket before the old one has expired. The smart card ticket can also contain several tickets of the same type, e.g., two 30-day tickets or several different types of tickets

Figure 4. Accessing on the internet the information from the smart card.

B. *Using a Ticket*

When one is traveling, the value and duration of the ticket change. On the paper ticket this information is at all times visible, while travelers with a smart card need to use ticket readers to access this information on their card. The smart card readers are also used to validate a ticket and give information about the type of card, expiration date or remaining value of the card, and expiration time. This information is visible for two seconds at the time of validation. This is often too short. If the user tries to scan the card again, an error message is displayed. The user has to wait 2 minutes after validation in order to display the information again. The fact that the type of ticket is not visible without scanning it, presents the risk of traveling with a wrong card, e.g., a parent can use a child ticket without knowing it. This becomes possible because travelers with a week, 30-days, or year smart card ticket are not obliged to validate their ticket every time they use the public transportation system.

When a traveler validates a ticket, the ticket reader can provide the wrong information. For example, an 11-year-old girl, who travels alone on a tram to her dance school, uses her pay-as-you-go smart card twice a week. Incidentally, her mother accompanies her one-day and notices that the child pays the adult fee instead of the discounted fee for children. The child's birthday was recorded at the time of the purchase of the smart card ticket and the card has been working well over a long period of time. The mother and the daughter walk into the public transportation service centre. The customer representative scans the card. All the trips, and the fees paid for them, appear on the screen. It becomes apparent that somehow the discount child's smart card was read as a regular card. The customer representative counts the number of wrongly charged trips, fills a paper based refund form, and issues the overcharged amount in cash.

Because a smart card user can have several tickets stored on the card, the ticket reader will also give information about the type of ticket being in use. However, there has been a lot of confusion over the validation of period tickets, such as the 7 or 30-days ticket. No user would stamp such a paper ticket

before the old ticket was expired, but many smart card ticket users assumed they had validated their new period smart card ticket before the old one was expired. They thus assumed that they were traveling with a valid ticket after their old ticket had expired. Many of these travelers were fined for traveling without a valid ticket.

C. The Expired Ticket

A paper ticket is expired when the timestamp on a ticket has expired. The user of a smart card will not be able to see if the ticket is still valid. The ticket has to be read (see above). If the ticket is a registered smart card (only 10% of all smart card tickets are registered [22]), the validity of the ticket can also be checked by logging onto the public transportation system's website.

At the moment, travelers have no way of checking the validity of their smart card ticket when they leave their home or office or when they are inside a metro train or regular train, unless they have a registered card and Internet access. Smart card users taking the bus or tram find out if their ticket has expired or not by using a card reading located inside the bus or tram. Our observations with smart card readers located with the bus driver (regional buses) made clear that many travelers are surprised to find out that they have not enough funds on their card and that they were attempting to travel with an expired card (see Fig. 5). In that case the user needs to buy an expensive one-time paper ticket from the bus driver or has to leave the bus.

Figure 5. Validating a smart card ticket on the bus.

III. WHAT SMART CARD TICKET USERS SAY

A. Methodology

The Norwegian news media (newspapers and online editions) have covered the transition from paper tickets to smart card tickets extensively. From the introduction of the smart card ticket (Reisekort) in 2009 until September 2013, 392 articles were published [13]. These articles usually triggered many comments from readers, e.g., [14], [15]. Studies in reader comments take largely place in journalism and media studies. Research has shown that news that covers public affairs, that have high social impact, and that is negative, receives the most reader comments [16]. Readers are more willing to comment when they are involved in the issue under discussion [17].

One of the news stories and commentaries we read, the *We don't want this!* article, attracted our attention. It generated many reader comments in short period of time. Secondly, because the author proposed a solution for the lack of basic ticket information, some readers contributed with their own ideas for the design of information visibility in the public transportation ticket. The article plus reader comments provided us with a rich body of use experiences, plus design ideas based on these experiences.

We used qualitative content analysis [18], [19] to study the content of *We don't want this!* and its reader comments. Generally, it is not clear if reader opinions, such as this one, are presenting the general opinion. American research has shown that such opinions are often written by people who are older, better educated, and more conservative than the general public, but, on the other hand, such letters on controversial topics may reflect public opinion [20]. A second limitation to our approach is that is often difficult to generalize the findings of qualitative methods such as content analysis to the general public at large. Thirdly, we have no way of knowing if our online sample, the article author and the commentators on the article, is representational of the general public.

We dealt with the limitations of the qualitative content analysis by using other methods to supplement our findings. We have used direct observations of people purchasing or validating their tickets and we conducted semi-structured interviews with 20 randomly chosen public transport users. Both observations and interviews, summarized in [1], lead to the same conclusion: the lack of ticket information visibility is problematic for travelers.

The article was published on June 1, 2013 on the Opinion section of the newspaper's website. The article received 232 comments written by 105 unique registered commentators. One hundred and one of the comments contained a negative opinion about the new smart card ticket, 38 comments contained a positive opinion, while 90 comments discussed related aspects of the new ticket system or expressed opinions that did not fall under the category 'positive' or 'negative' or they were deleted (3 comments). The entire discussion (now closed) was coded independently by two researchers and the results were discussed and synthesized. The analysis of the reader comments shows that many public transportation users experience the loss of ticket information as problematic.

B. Analysis

In this section, we present our analysis of the online material. We first describe and present the main themes in the reader comments that emerged in the content analysis related to the design of the system, the lack of user's perspective, validation of the ticket, nostalgia over the paper based solution, cognitive complexity, comparison with other systems and solution proposals. A discussion of these findings in the context of information visibility and the implications for design will be presented in the next section.

1) Design

Several readers comment on the design of the new ticket systems, both the smart card tickets as well as the readers, the placement of the readers:

"We also have small children who need a card. Which card shall we give them? Do we need to go down to one of the stations to find out which is a regular card, which one a children [discount] card, and how much money is left on the card? And sometimes we have guests and we experience the same problems." (#14: 24 likes)

"Several times a week the system [card readers] doesn't work when you get on [the bus] and many of the buses have older equipment so you can't read [the screen] if you don't bend over to a 1.40 m. height for the one second that the reader gives you information." (#56: 75 likes/1 dislike)

"*What kind of problem is solved with the [smart card ticket]? I used to have a [paper] monthly pass and I could always see its stamp and I was sure that the same date would still be on the ticket when there was ticket control."* (#83: 12 likes)

"I struggle using these validation machines [card readers]. Only seldom I am able to read what is written on them before it disappears. I am often unsure if I have paid or not. Shameful." (#207: 57 likes)

2) Users' perspective

Several readers mention that the system is developed without having the interests of the users in mind:

"Don't understand why they who decide in our little Oslo can't come down from their prestige horse and actually deliver a system that takes the users' interests in consideration." (#73: 71 likes)

"[H]ave made thousands of trips with tram, metro, bus and train, also in other countries, but never experienced problems with paying that come close to being as foolish and user-unfriendly as in Oslo." (#36: 67 likes)

"The reason for making a new system must be to make it simpler, not for making it more difficult. Ruter [the public transportation company] *has made it difficult for many customers. Tried to complain, but they don't pay attention to the individual customer."* (#15: 22 likes)

3) Validation

Many of the reader comments focused on validation of the ticket. Users experienced that card readers did not work so they couldn't validate their card. The comments also show that there is confusion about the act of validating a ticket. Although validating a smart card ticket has the same function as stamping a ticket, many readers don't seem to understand the concept of validating a smart card. Some equate paying for a ticket with validating a ticket, while this is only true for the pay-as-you-go card:

"This is BTW called 'buy ticket' too; that's what is written in the display when you validate a pay-as-you-go card." (#61)

"After a few days my mother was using the card [a 2-zone card] from zone 2 to zone 1, but then it became clear that I had topped up the card in zone 1 […] the card couldn't be validated in zone 2." (#148: 48 likes)

4) Paper is best

Many readers argue that the paper ticket system was a better system:

"*Paper is king. Paper doesn't need electricity and it is easier to fill the ink cartridge in empty stamp machines than to cruise around town trying to find out where the mistake is when all ticket machines stop working."* (#80: 76 likes/1 dislike)

"I miss the 8-trip ticket. Always overview over remaining trips, the stamp machines at the stations worked. Simple and effective." (#109: 73 likes)

Not everyone was satisfied with the paper ticket and its stamps:

"And there where different kind of stamps and explanations of what was the line number, the date and time, and the number of zones etc. … And now we have tickets whose 'content' is invisible!:-)." (#172: 4 likes)

5) Cognitive challenges

The new system of smart card, card readers, and buying tickets is perceived as too complex:

"I don't know how the Ruter system works (the same as for any tourist coming to Oslo), I have no idea where and how to buy a ticket." (#5: 12 likes)

"I have no idea what I have to do to buy a ticket for the bus/metro/tram. […] I miss the paper ticket." (#34: 41 likes)

6) Other places – different smart card ticket systems

Many of the readers mention smart card systems in other countries. They have experienced systems that are simpler because they have only one type of ticket or the traveler pays per distance and checks in and out each time the card is used:

"Seoul, Bussan (both South-Korea), Hong Kong (+ many Chinese cities) Tokyo (+ many Japanese cities), Singapore, most of Europe (this are many cities), Bangkok, Moscow, St. Petersburg (Russia), Kiev (Ukraine), Medellin (Colombia), Rio de Janeiro, Sao Paulo (both Brazil), New York (USA), Sydney (Australia) have all brilliant metro systems, where it is easy to buy a ticket and understandable systems that always work. I have tried all of them and never had any problems. In contrast, I have lots of problems with the system in Oslo." (#221: 103 likes)

7) Solutions

Several readers propose solutions to deal with the problems they experience with the smart card ticket system. Several propose to stop investing money in an expensive smart card system and ongoing ticket control in order to

provide free public transportation or at least free transportation for senior citizens:

> *"Wouldn't it be simpler and cheaper for all parties if public transportation was completely free??"* (#74: 24 likes)

Based on experiences in other countries, several readers propose a simpler system:

> "*Let people pay for each time they travel, dependent on the distance. Gladly with plastic card or app. Thus we need only one option in the machines (Put money on)."* (#102: 3 likes)

Others have started to use the new mobile phone app, which allows one to order a ticket on the spot:

> *"On my mobile I can see the whole time how much time I have left again [on my ticket]."* (#57: 3 likes/1 dislike)

On the other hand, people with older mobile phones, including older models of smart phones, are unable to use the app and with the app more responsibility is transferred to the user. The user now needs to make sure that the valid ticket can be shown on demand by a ticket controller:

> *"I prefer the smart card. It is not as vulnerable for mistakes as the phone app, for example something as simple as an empty battery. I don't say that the app is badly made, but it puts more responsibility and risk on the user".* (#178: 9 likes)

Another commentator proposed a smart card with a small 'ink screen' that could keep the ticket information visible:

> *"So can one always see how much one has again …"* (#170: 6 likes)

8) Responsibility to learn a new system

Many of the readers who are positive about the smart card system state that users who have problems with the new system are lazy or do not take responsibility for learning something new:

> *"This is new, not complicated […]. It doesn't take a long time to learn oneself the new system, not even the elderly. This is just laziness."* (#9: 11 likes/3 dislikes)

> *"Don't people no longer have responsibility? This case and the other cases that were in the media lately show that people don't want to familiarize themselves with the new system, they don't want to learn and they don't take responsibility*." (#18: 10 likes/8 dislikes)

One reader describes explicitly what it means to take responsibility:

> *"The only thing I need to remember is to bring my card, to scan it when I go on board, and to make a note when I buy the ticket (I put the alarm on in my mobile phone an hour ahead in case I am not able to go out and back again in one hour)."* (#87: 15 likes / 1 dislike)

IV. Discussion: The Visibility of Ticket Information

The majority of reader comments critical of the new smart card system address the issue of loss of basic ticket information. Not only the timestamp is now removed from the ticket, also other visual pointers, such as the shape and color that indicated the type of ticket, have disappeared. The answer to simple questions, such as "is this my child's ticket" or "how many more trips can I make on this ticket*"* now involve a variety of devices and places. Ticket information, once located on a piece of paper in the hand of the user, is now distributed and networked. Stalder [21] calls this *networked visibility*, which is "created by the capacity to record, store, transmit, access communication, action, and states generated through digital networks". Stalder presents two types of visibility: *horizontal visibility*, pertaining to information becoming visible to users; and *vertical visibility*, pertaining to what information the service providers can see. While users can often manage their horizontal visibility, e.g., what information about themselves becomes visible to others, they have no control over the vertical visibility of their information. Service providers have access to the information of all users, but this visibility is one way, it is invisible to the users.

Based on Stalder, we can differentiate between the horizontal and vertical visibility of ticket information. We understand *horizontal visibility* as the visibility of ticket information to the user of the public transportation system. The paper ticket user has immediate horizontal visibility, at all times and places. The ticket information is directly visible on the paper ticket – when the ticket is in use, not in use, or expired. The smart card user's horizontal visibility is limited to particular places: when the ticket is purchased, when it is read or scanned, or when it is checked on the web (only for registered cards). As we saw in the previous section, many travelers are insecure about the validity of their smart card. Travelers taking the bus or metro can only found out if their ticket is still valid after they have boarded, unless they have a registered ticket (only 10% of the smart card ticket holders) or are close to a ticket machine at metro stations and main bus stations.

We understand *vertical visibility* as the visibility of the ticket information to the provider of the public transportation system. The provider has other ticket information needs than the user. The provider is interested in *use information*, such as the users' frequency, time, and destination of travel, and what type of ticket they use. This information is the basis for organizing public transportation schedules, the frequency of departures, and the number of routes. The provider had only limited vertical visibility when the paper tickets were in use and therefore had to implement user surveys to get this information. With the introduction of the smart card, the provider has full access to ticket information.

A. Networked Visibility

In our case, the networked visibility created by embedding computing and networking capabilities in a

public transportation ticket has decreased the horizontal visibility of the users and significantly increased the vertical visibility of the provider. The loss of horizontal visibility negatively affects a large number of travelers who are uncomfortable using their smart card. They cannot transfer their familiarity, skill, and experience in using the paper ticket to the smart card. This becomes especially clear in the reader comments on validation. Travelers do not see the similarity between stamping a paper ticket and validating a ticket via a card reader. They experience new problems, such as card readers that do not work or cannot validate a smart card ticket. If they do validate or read their card, the basic ticket information is only visible for 2 seconds. Therefore, many smart card users express their insecurity about the validity of their tickets when they travel and several relate stories of unpleasant experiences during ticket control.

B. Mobile Phone App

None of the travelers, who were positive regarding the new smart card system, addressed the issue of ticket information visibility of the smart card ticket. They had found 'work arounds' for the loss of basic ticket information, e.g., setting an alarm on the mobile phone or validating the ticket every time they used public transportation, even if their ticket type did not require this. Many of the positive travelers are now using Ruter's mobile app, which was launched in December 2012. The app provides all basic ticket information, including a count down function for travelers who bought a single ticket with one-hour validity (see Fig. 6).

Figure 6. An adult single trip ticket: the green zone states: valid and the timer shows the exact amount of validity time left

C. New Concerns

The increase of the vertical visibility of the provider can create new concerns in terms of privacy as the whereabouts of smart card users can be recorded, stored, and transmitted. These data can be accessed or aggregated for uses not directly related to the public transportation system, such as marketing. This seems less of a concern in Norway. At the introduction of the smart ticket, Ruter wanted all cards to be registered (linked to a person) and required cards to be validated before each trip. These demands soon disappeared after intervention by the Norwegian Data Protection Authority in 2009. At the moment 90% of more than 600.000 smart cards used in public transportation in the Oslo region are not registered and there are strict rules about storage and use of data produced by the 10% registered cards.

V. Design Alternatives

The design of the smart card ticket builds forth on some of the characteristics of the paper ticket: tickets need to be validated before use and the points of validation can be found at the same locations as the paper tickets. The main difference is the visibility of ticket information. On paper tickets, information was visible at all times and places because it was a *local information*, it was locally stored on the paper ticket. As we have seen in the previous section, there are serious issues with visibility of information on smart card tickets. Can we make this characteristic of the paper ticket available on the smart card ticket and thus capitalize on the familiarity with the paper ticket?

A. ARTick: Augmenting the Smart Ticket

As described in [1], in order to improve the user experience with smart tickets and offer local visibility of the ticketing information, we made a simple prototype: ARTick (Augmented Reality Ticket) (Fig. 7). ARTick turns any smart phone into a mobile smart card reader using NFC standards, see also [10]. NFC is a short-range wireless technology, enabling one-way and two-way communication between smart phones or between smart phones and other wireless devices, in our case the contactless smart card [23].

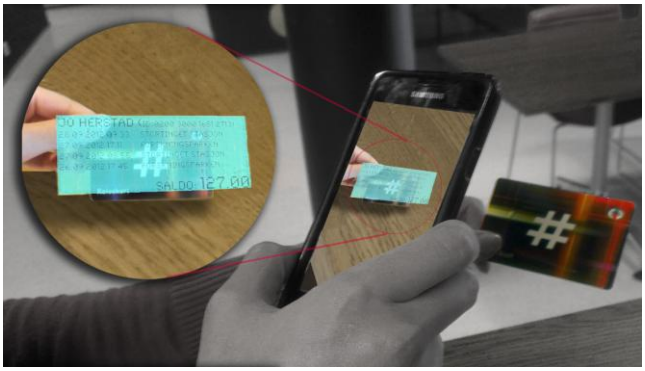


Figure 7. A 3D rendering of smart card ticket information using a NFC-enabled smart phone with ARTick [1].

On NFC-enabled mobile phones, the ARTick application uses NFC to read the information off the card. The application enables the user to check the type of ticket, the value and duration of the ticket, as well as the latest transactions. ARTick enables ticket information to be read

in 2D and 3D, augmented using the camera as shown in Fig. 7, as well as audio for the visual impaired.

Non-NFC-enabled smart phones use the camera to take an image of the card number on the back of the smart card ticket, see lower left corner on Fig. 1, and use Optical Character Recognition (OCR). This card number corresponds with ticket information stored on the website of the public transportation provider. Once the number is recognized, a user can access the same information as the provider.

The addition of audio is particularly interesting for users who have issues with their vision, whether it is related to sight challenges or various forms of dyslexia. ARTick follows universal design principles [24]. ARTick thus enables all mobile phones with camera, smart or not, to check the validity of the smart card ticket. After the launch of the mobile app [25], this solution's appeal diminished. The smart phone Ruter app solves the problem of visibility. ARTick may, however, still be of interest for groups with special needs or travelers without smart phones.

B. Solution Suggestions by Online Commentators

In the Introduction, we stated that the author of *We don't want this!*, proposes one solution: "receive a receipt every time you use the card". She writes:

> *"The system is not based on legal principles. It does not take into account two important principles:*
> 1. *You must have a receipt each time you pay a commodity.*
> 2. *This should be so not only when you buy a ticket for some period, or even buy a single ticket or just put money on your card, but also whenever you use your card. I see from other posts that you are referring to the Canary Islands, where this is a practice on their buses. I have spent so much, I have so much left. This principle applies in most other cases in business. When I put my money on the card, I borrow this money to Router until I use it to travel."*

As mentioned in Section IV, a smart phone based ticketing service is now available, see Fig. 10. The app won 'The Best App award' at Mobile Trends conference [25]. However, not everyone owns a smart phone. In the analysis of the customers' comments we found the following conversation, pointing to several issues regarding this solution, e.g., expense related to acquisition of a smart phone, usability issues regarding the purchase of the ticket:

> *"Mobile app is definitely a future oriented and good solution. Ruter / NSB should enter into an agreement with Telenor* [telecommunications provider] *as affordable smart phone and subscription to those who lack this. Let those born before 1940 go free, and sell day passes to tourists. And by all means, the app must be working in the whole country as soon as possible." (#54: 8 likes)*

> *"This phone, that the app is attached to, do they send it to you in mail, free of charge?" (#55: 40 likes/1 dislike)*

> *"Of course, but first you must use an order app and an email app on your existing mobile phone, or use password app to receive a password to a website where you can obtain the user name to an app, which provides access to a password that you can use to log onto a website ..." (#56)*

Among user commentaries, we find several that propose alternative solutions. Most notably, a solution based on SMS:

> *"Why can not the ticket be paid through a regular SMS, for those who do not have smart phones? I'm not saying that mobile is the only way to pay, but since there is app-ticket requiring smart phone, it should have been SMS-ticket too! You send an SMS with the typical B* [Barn - Norwegian for children] *for children, H for seniors* [Honor citizens] *V for adult* [Voksen - adult] *and get a code (which ticket controllers can check) with an expiration time and ticket type specification. If you need to have multiple zones, simply send B2, V2 etc - very easy." (#78: 21 likes)*

Another design suggestion among commentators is:

> *"What if you have a small ink display cards that get power from the computer right after you check, just to refresh the ink on the screen and become passive again? So you can always see how much is left." (#86: 6 likes)*

> *"Why not have a solution where I do not have to check or validate. E.g., what about selling 100 trips (more or less) and an optional notification about the number of remaining trips? The more trips you buy the higher discount and flexibility for those who travel infrequently." (#44: 11 likes).*

> *"Introduce the city tax, get all ticket controllers fired, as well as everyone who worked on this s... ticket system, and make the collective traffic free."* (#58: 3 likes)

All of the above suggestions, explicitly or implicitly, address the visibility of the information. On a mobile app solution, it is already there. The solution by the author of *We don't want this!*, the SMS solution proposed by the commentator #78, as well as the ink display on the ticket suggestion by commentator #86, all have explicit visibility at all times, and build on a familiarity with the old system. In the old system, the stamp in itself was a receipt for transaction. The last two suggestions simply remove the validation, or tickets, and thus address the visibility indirectly.

C. Wearables: an Emerging Trend in Smart Ticketing

Massachusetts based Sesame Ring project, supported by Massachusetts Bay Transportation Authority (MBTA), started in 2013, see [26]. The designers involved in the project managed to reduce the size of the RFID tag and

place it inside a ring, see Fig. 8. Rings are custom sized and users may create their own designs for the front of the ring.

Figure 8. Wearable technology seems to be the direction for transportation solutions. Photo: Kickstarter, [27].

A similar solution has been developed for London's transit smartcard Oyster, see [28]. Fig. 9 shows the card worn as a wrist-band attachment.

The MBTA and London transport services are possibly different than services offered in Oslo, but the wearable tickets offer one possibility to code type of the card: color. One could then instantly and easily identify those carrying monthly, weekly, daily or any other type of ticket offered by Ruter. However, the problems related to visibility of expiration time would still remain unsolved, assuming current rules and regulations around the use of the card.

Figure 9. A wearable solution replacing the plastic smart card is proposed for London transport. Photo: via Yanko Design

Many large smart phone producers, e.g., Samsung, Apple and Sony, see [29], are in a race to push their smart watches on the market. The watches could easily provide visibility of both ticket type, and validity period. As with a smart phone, though, this solution will not fit all travelers. It would involve investment into the device and skills to operate it. For many travelers though, also in Oslo, this could provide a solution scoring high on user satisfaction.

VI. CONCLUSION AND FUTURE WORK

The move from paper-based to smart card-based public transportation tickets did not result in enhanced user experience. Travelers could not use their familiarity with the paper ticket when the smart card ticket was introduced. Our study showed that the lack of information visibility, the immediate and continuous access to ticket information, is a prerequisite for users to capitalize on their familiarity with the paper ticket.

The loss of this visibility resulted in negative user experiences in the three main ticketing activities: purchasing a ticket, using a ticket, and knowing when a ticket is expired. This became especially clear in the discussion of the notion of the validation of the ticket. The loss of information visibility of the smart card user led to a situation in which the familiarity with stamping a paper ticket could not be transferred to the notion of validating a smart card ticket.

Also new concerns emerged after the move from paper tickets to smart card tickets. Initially, Ruter wanted every traveler to register (personalized) the smart card ticket and to validate the ticket every time the public transportation system was used. In this way, the provider would gain access to user and use information (vertical visibility), creating new concerns about privacy. Ruter could not maximize its vertical information visibility, after intervention by the Norwegian Data Protection Authority, but still has access to more use data than was available when the paper tickets were still in use.

The concepts of networked visibility and horizontal and vertical visibility helped us to understand the emergence of new user experience issues and concerns when computing and networking technology is embedded in a transportation ticket. These concepts were also used to explore solutions, such as how to restore the horizontal visibility of ticket information, the immediate and continuous ticket information to the users. We discussed several solutions that increase the information visibility of the smart card ticket, such as ARTick, an app that turns a smart phone into a smart card reader, a smart card with display, and the addition of a paper receipt every time the smart card is validated. We also discussed wearable smart tickets, which we believe will enhance user experience but not necessarily information visibility. Ruterbillett, Ruter's mobile phone app, fully restores information visibility to the user, but can only be used on latest models mobile phones.

In further research, we will continue studying the role of information visibility in smart technologies.

REFERENCES

[1] M. van der Velden, A. L. Culén, J. Herstad, and A. Atif, "Networked visibility: The case of smart card ticket information," Proc. Sixth International Conference on Advances in Computer-Human Interactions 2013 (ACHI 2013), March 2013, pp. 228–233.

[2] "Ruter endrer smartbilletten - Osloby." [Online]. Available: http://www.osloby.no/nyheter/Ruter-endrer-smartbilletten-6744324.html#.UjIGkD-568A. [Accessed: 05-Dec-2013].

[3] B. A. Aubert and G. Hamel, "Adoption of smart cards in the medical sector: the Canadian experience," Social Science & Medicine, vol. 53, no. 7, pp. 879–894, Oct. 2001.

[4] C. R. Plouffe, J. S. Hulland, and M. Vandenbosch, "Research report: Richness versus parsimony in modeling technology adoption decisions--Understanding merchant adoption of a

smart card-based payment system," Information Systems Research, vol. 12, no. 2, pp. 208–222, June 2001.

[5] F. Resatsch, U. Sandner, J. M. Leimeister, and H. Krcmar, "Do point of sale RFID-based information services make a difference? Analyzing consumer perceptions for designing smart product information services in retail business," Electron. Market., vol. 18, no. 3, pp. 216–231, August 2008.

[6] R. T. Rust and P. K. Kannan, "E-service: a new paradigm for business in the electronic environment," Commun. ACM, vol. 46, no. 6, pp. 36–42, June 2003.

[7] S.-Y. Chou and Y. Ekawati, "Cost reduction of public transportation systems with information visibility enabled by RFID technology," in Global Perspective for Competitive Enterprise, Economy and Ecology, S.-Y. Chou, A. Trappey, J. Pokojski, and S. Smith, Eds. London: Springer, 2009, pp. 553–561.

[8] H. Iseki, A. Demisch, B. D. Taylor, and A. C. Yoh, "Evaluating the costs and benefits of transit smart cards," California PATH Research Report, Berkely: University of California, 2008.

[9] N. Mallat, M. Rossi, V. K. Tuunainen, and A. Öörni, "An empirical investigation of mobile ticketing service adoption in public transportation," Personal Ubiquitous Comput., vol. 12, no. 1, pp. 57–65, Jan. 2008.

[10] F. Morgner, D. Oepen, W. Müller, and J.-P. Redlich, "Mobile smart card reader using NFC-enabled smartphones," in Security and Privacy in Mobile Information and Communication Systems, A. U. Schmidt, G. Russello, I. Krontiris, and S. Lian, Eds. Springer Berlin Heidelberg, 2012, pp. 24–37.

[11] A. J. Sellen and R. H. R. Harper, The Myth of the Paperless Office. The MIT Press, 2001.

[12] P. Dourish, "What we talk about when we talk about context," Personal Ubiquitous Computing, vol. 8, no. 1, pp. 19–30, 2004.

[13] Retriever, "Retriever research: ruter AND reisekort". [Online]. Available: http://web.retriever-info.com/services/archive.html. [Accessed: 05-Dec-2013].

[14] "Du risikerer å måtte betale for gammelt Ruter-rot," Dagbladet.no. [Online]. Available: http://www.dagbladet.no/2011/10/26/nyheter/innenriks/kollektivtrafikk/18775825/. [Accessed: 05-Dec-2013].

[15] "Feil i Ruters systemer ga Marianne bot på t-banen," Dagbladet.no. [Online]. Available: http://www.dagbladet.no/2012/08/16/nyheter/kollektivtrafikk/ruter/bot/22983916/. [Accessed: 05-Dec-2013].

[16] P. Weber, "Discussions in the comments section: Factors influencing participation and interactivity in online newspapers' reader comments," New Media Society, doi 1461444813495165, Jul. 2013.

[17] N. Diakopoulos and M. Naaman, "Towards quality discourse in online news comments," in Proceedings of the ACM 2011 conference on Computer supported cooperative work, New York, NY, USA, 2011, pp. 133–142.

[18] S. Elo and H. Kyngäs, "The qualitative content analysis process," Journal of Advanced Nursing, vol. 62, no. 1, pp. 107–115, 2008.

[19] V. Braun and V. Clarke, "Using thematic analysis in psychology," Qualitative Research in Psychology, vol. 3, no. 2, pp. 77–101, 2006.

[20] M. McCluskey and J. Hmielowski, "Opinion expression during social conflict: Comparing online reader comments and letters to the editor," Journalism, vol. 13, no. 3, pp. 303–319, April 2012.

[21] F. Stalder, "Politics of networked visibility," Acoustic Space, no. 10, pp. 13–19, 2011.

[22] J. B. Bakken, "Aner ikke hvem 90 prosent av kundene er," Dagens IT. [Online]. Available at http://www.dagensit.no/k/mappami/article2618470.ece. [Accessed: 05-Dec-2013].

[23] D. K. Finkenzeller, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication, Third Edition. John Wiley & Sons, 2010.

[24] S. Goldsmith, Universal Design. Routledge, 2012.

[25] "RuterBillett er kåret til årets app | Ruter." [Online]. Available: https://ruter.no/verdt-avite/presse/pressemeldinger/ruterbillett-er-arets-app/. [Accessed: 05-Dec-2013].

[26] "MIT students create MBTA Charlie Card ring," Boston Magazine. [Online]. Available: http://www.bostonmagazine.com/news/blog/2013/08/22/mbta-charlie-card-ring-kickstarter/. [Accessed: 05-Dec-2013].

[27] L. LaBianca, "Sesame ring - Where will it take you?," Kickstarter. [Online]. Available: http://www.kickstarter.com/projects/1066401427/sesame-ring-where-will-it-take-you. [Accessed: 05-Dec-2013].

[28] "Accessorize with London's fashionable new Oyster Card," TheCityFix. [Online]. Available: http://thecityfix.com/blog/accessorize-with-londons-fashionable-new-oyster-card/. [Accessed: 05-Dec-2013].

[29] "Samsung Galaxy Gear wrist watch | Technologist." [Online]. Available: http://www.technogist.com/2013/08/samsung-galaxy-gear-wrist-watch.html. [Accessed: 05-Dec-2013].

Figure 10. The new Ruter mobile app. One can clearly see the type of the ticket and the expiration date/time: 30-day ticket expires in 24 hours. Photo: Ruter.

Quality of Service Aware Configuration of Network Equipment in Industrial Environments

György Kálmán
ABB Corporate Research
Norway
gyorgy.kalman@no.abb.com

***Abstract*—Industrial Ethernet offers greater flexibility and potentially lower deployment costs than traditional fieldbuses. Ethernet is already the preferred communication technology from the controller and is expected to penetrate the instrument area also. Engineering and operation of these networks is introducing new challenges in the industrial automation field, including the lack of appropriate Quality of Service (QoS) metrics for these applications. This paper presents an overview of the industrial Ethernet landscape, shows the challenges around QoS parameters and shows an example engineering support function. Through this example, it presents different approaches and decisions taken for the proof of concept implementation. An overview about the issues related to representation and generation of configuration data, support of multiple vendors in the engineering phase and also during operation. The paper concludes with showing the differences between QoS metrics in industrial and office networks. The implemented proof-of-concept tool shows that bulk configuration of devices opens a QoS aware deployment process.**

***Keywords*—industrial Ethernet, QoS, metrics, engineering, infrastructure, switch, configuration, life cycle, multi vendor**

I. Introduction

This paper is an extended version of [1], Mass Configuration of Network Devices in Industrial Environments presented at ICN2013. In addition to the original, the scope of the paper is extended with industrial QoS aspects to show, why automated or aided configuration of network devices is crucial in industrial environments.

A modern industrial communication system contains a considerable amount of nodes interconnected with Ethernet and current trends point towards moving the Ethernet connectivity down to instrument level. Having an all-Ethernet infrastructure offers several advantages over traditional fieldbus-based or Ethernet-fieldbus mixed networks. These include simpler deployment by using the same connectors and wires over the whole network, ample bandwidth, wide range of communication hardware and easy connectivity towards office networks or the internet.

One of the main drawbacks is a result of the inherently different network topology compared to office environments. In industry, the bus-like structure has proven to reduce costs with cutting cabling need. In these scenarios, the backbone is usually composed as a ring and the devices, sub networks or other devices are connected to this with small switches (up to approx. 10 ports).

In such structures, switched Ethernet is still operational but not at it's optimal parameters, as, e.g., collisions are eliminated but the traffic aggregation on the ring interfaces can lead to queueing. Important QoS parameters, e.g., convergence time and jitter are both negatively affected by the typical industrial topologies.

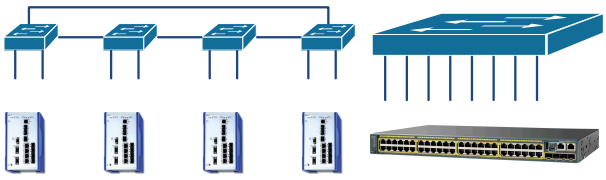


Fig. 1. Comparison of typical switch sizes in industry and other fields

The use of small switches lead also to deep and sparse spanning trees which limit the performance of the Rapid Spanning Tree Protocol (RSTP) and have a negative impact on the reconfiguration time in case of link failure. Time synchronization of devices also suffers from the long distance between different part of the networks and has been mitigated by introducing the more precise IEEE 1588 Precision Time Protocol (PTP).

To mitigate the issues associated with the deep spanning trees and the growing RSTP convergence time, several industrial redundancy protocols were introduced from special versions of RSTP through proprietary ring protocols, doubled networks to overlay networks.

Engineering of networks composed from small switches results in typically a magnitude more devices than a comparable office network (e.g., a bigger refinery can have several hundreds of switches with a typical branching factor of 4-7) as shown on Figure 1. During engineering and Factory Acceptance Test (FAT), the effort of configuring these devices is high and severely influences the competitiveness. In the majority of cases, the actual configuration of the devices can be described with setting port-VLAN allocations, RSTP priorities, Simple Network Management Protocol (SNMP) parameters and performance monitoring. These steps currently require manual work, which is increasing cost during engineering and also leads to increased resource usage during FAT as configuration errors may happen.

QoS parameters are often evaluated at the end of the engineering process as part of the FAT which might result in an iterative process with changing structures. The main showstopper as the author can see, is that the QoS metrics used in office or telecommunication networks cannot be used directly in industrial networks.

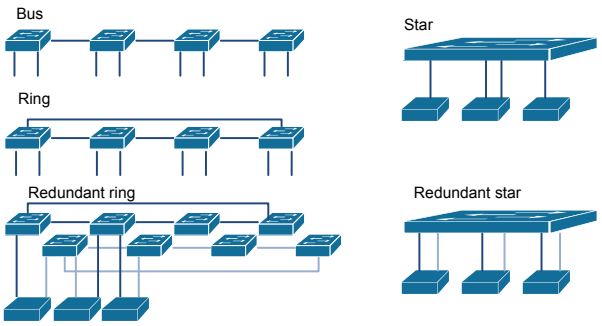


Fig. 2. Typical industrial Ethernet topologies

The primary aspects, which cause differences in the operation and hence the QoS metrics to be used are described in the following sections.

A. Topology

A key area, where industrial networks do differ considerably from their office counterparts is the topology used. In an office environment, the network is structured to resemble an equalized tree as much as possible. Also, high port density switches are used to lower the hierarchy levels in the network.

The industrial environment, as stated earlier, resembles more a bus-like topology. Ring-based redundancy solutions [2], traditional planning and cabling cost both force network engineering towards the use of rings as backbone and small switches to connect the few nodes which are located close.

Ring structures are beneficial for redundancy, but are problematic for traffic engineering. These rings aggregate traffic and force longer paths in the network than in a comparable office counterpart.

Ring mitigates the main risk of a bus topology by allowing the failure of one link and still keeping the network in operation. The long paths introduced by a bus however are still present. Where the constraints allow, it is beneficial to use a redundant star network [3].

B. Network segmentation

The traffic aggregation of rings do cause other issues too, especially if multi- or broadcast traffic is involved. In a typical installation, several industrial protocols are in use. In order to reach a more stable network and avoid that nodes are receiving unnecessary traffic, these networks are often segmented into several Virtual LANs (VLANs). The convergence of different networks on the same physical media also makes network management different compared to legacy systems.

C. Configuration and Maintenance

Current industry practice builds on a detailed network drawing and unit-to-unit configuration of the network devices as part of the deployment. Here, in most cases, the built-in web configuration solutions of the different vendors are used, although some provide their tools for own product lines which enable configuration of multiple units.

From the engineering viewpoint, setting up these devices one-by-one is a great risk, as the chance of human error is high. This risk is mitigated with additional resources, meaning more work hours to check the actual setup [4].

From maintenance viewpoint, this situation is even worse. Most installations have a long life expectancy and, therefore future maintenance engineers will either face 10-15 year old web interfaces if they have to modify something or the the problems associated with replacing the old device and migrating the configuration to a new one.

The purpose of this paper is to give an overview of the issues with the use of Ethernet in the industry-typical engineering practices and scenarios. These include network layout, QoS requirements, e.g., redundancy, time synchronization or maintenance and bulk configuration.

II. Quality of Service in Industrial Ethernet

Ethernet is expected to overtake the role as first choice for communication in industry installations for the majority of the cases. Although it is superior in bandwidth compared to any fieldbus used before, the past history of lacking determinism has raised concerns in the industry. The problems associated with traffic scheduling, prioritization and loss have been explored since both in industrial and other networks, mainly related to audio and video (AV) applications.

QoS is an objective measure of the network performance based on a defined set of metrics. For the AV applications the defined metrics include bandwidth, jitter, delay and loss. These are all relevant to the industrial applications, however, the weights cannot be mapped directly and in some cases the requirements are more strict in an industrial environment. An example would be the tolerance for jitter in industrial applications, in motion control (typically less than 1 ms) or factory automation (typically up to 10 ms). Since data in these examples are used in machine to machine communication, failing under delivery might lead to production stop resulting in direct economic loss instead of reduced user experience.

QoS solutions can be categorized into the two classes defined by IntServ and DiffServ.

A. IntServ

IntServ aims to implement QoS features that can enable to implement circuit switched like services on a packet switched network. It offers a fine grained system, where all nodes in the core network run IntServ and by using the Resource Reservation Protocol (RSVP) to create communication channels with end-to-end QoS. Resources along the whole path are reserved for the specific stream fulfilling absolute timing and bandwidth requirements.

B. DiffServ

DiffServ provides relative traffic prioritization and thus no absolute QoS parameters. Guarantees are not given and the traffic classification is only valid for the specific device. No end-to-end guarantees are given, also not for the insertion into a specific priority queue. The solution is more scalable, as the intermediate nodes do not need to keep track of all streams,

there is no end-to-end resource reservation and only a few priority queues are used.

The two classes show the fundamental difference between absolute and relative QoS guarantees.

Currently, DiffServ is the preferred solution for internet traffic as this solution is scalable and offers good enough service quality assuming appropriate over provisioning of resources. The relative traffic priorities given by DiffServ however are not a perfect match for industrial applications. What industry expects is much closer to the granularity of what Asynchronous Transfer Mode (ATM) provided in the QoS field, or if some of the most critical applications are covered by technologies with intrinsic QoS (e.g., EtherCAT), by IntServ. Although not scalable, IntServ-like QoS can still be a valid solution for industry as critical traffic is either used only inside LANs or is transmitted over Synchronous Digital Hierarchy (SDH) or Multiprotocol Label Switching (MPLS) links with strict Service Level Agreements (SLAs). The implementation of an IntServ-like absolute traffic prioritization is however at the moment not a prime objective in industrial environments. The over provisioning of resources and the possibility to take demanding processes into a segment with intrinsic QoS allows the use of standard Ethernet with the existing, relative traffic prioritization and store-and-forward switching. If ample bandwidth is provided, a probabilistic upper bound with good confidence can be given on transmission parameters.

In addition to the AV-typical metrics, there are some network properties, which have to be taken into account in an industrial environment and can be addressed in the engineering phase. These are mostly caused by the specific structures and processes used. An example is the low branching factor. During engineering it is possible to aim for shorter paths in the network between critical nodes or to try to equalize at least parts of the network (e.g., using an AVL-tree [5]).

C. Industrial Ethernet overview

Industrial Ethernet is already the communication technology expected to be present from the controller to the operator workstations and beyond. In the field level, there are still some applications preferring fieldbuses but the trend here also shows the growing market share of Ethernet. In the following, industrial Ethernet is used to reference the use of Ethernet technology in industrial environments (on OSI Layer 2) despite that several of the industrial Ethernet protocols are in fact on upper layers.

Using switched Ethernet offers several benefits in the industrial field, including:

- high bandwidth,
- off the shelf, low cost technology,
- seamless integration with office and telecommunication networks,
- network convergence incorporating automation, security and safety.

In [3], the traditional split of three different networks used in the industrial field is shown. The structure shows a heritage

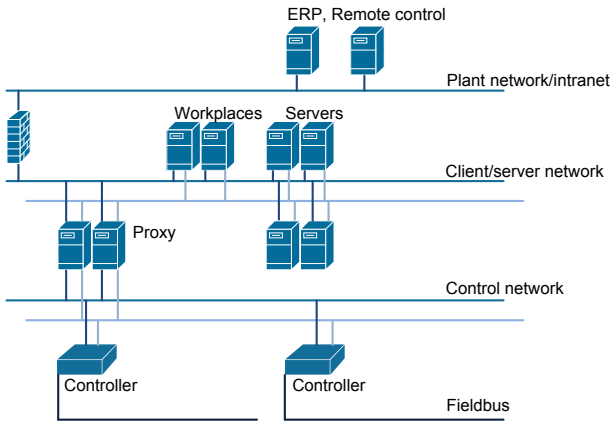


Fig. 3. Automation network hierarchy

of the field buses what explains the frequently used bus and ring topologies.

Considerable efforts in industrial network research are spent on the fields of QoS, time synchronization, redundancy and network convergence.

One of the main assumed drawbacks of Ethernet in automation was the CSMA/CD algorithm used for multiple access. Although the problem of collisions is not present any more (as all networks are implemented as switched, full-duplex Ethernet and thus CSMA/CD is not being run), the probabilistic nature of Ethernet has raised severe concerns on Ethernet's capability to replace strictly scheduled fieldbuses despite the much higher available bandwidth [7]. Industrial applications are now shifting towards the use of 1G links also for the field and control network level, thus traffic, except high frequency control and sampling, can be carried with standard Ethernet equipment.

Data refresh rates differ depending on the field and usage of the industrial network.

Class	Grace time	Desciption
Uncritical	<10 s	ERP, Manufacturing
Automation	<1 s	human interface
Benign	<100 ms	process, manufacturing
Critical	<10 ms	synchronized drives

For high refresh rates and hard-real time applications several technologies have been developed, where EtherCAT is one of the most popular solutions.

EtherCAT implements a ring with a master device connected to both ends and slaves chained on the ring. The master sends out the frames on the network and the slaves, while the signals are passing through the network interface, are exchanging information (Figure 4). The frame is not stored in any way on the slaves and the latency in crossing the EtherCAT ring is fixed. EtherCAT is offering intrinsic QoS, as the jitter is practically 0, the cycle times can be calculated prior to deployment and time synchronization is provided through a distributed clock synchronization algorithm.

The master can be connected to the rest of the industrial network with an additional network interface.

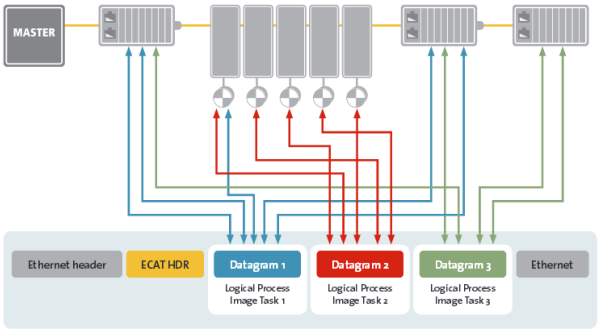


Fig. 4. Structure of the EtherCAT frame [8]

D. Time synchronization

The need for synchronized operation throughout the network is required by most applications, although the precision need and the impact of keeping the limit differs. In a typical non-industrial example system, like a stock exchange system or Enterprise Resource Planning (ERP) the precision limit is more relaxed than, e.g., controlling a surveyor belt, where the high-speed belt is driven by hundreds of electric motors which need to operate synchronous to change the overall speed. The required degree of accuracy also depends on the specific application.

In non-industrial applications, typically the precision reached by Network Time Protocol (NTP) is adequate, while industrial efforts and lately also audio-video standardization is expecting IEEE 1588 Precision Time Protocol (PTP) to serve as synchronization protocol. The combined effort of industry and AV fields are also shown in, that the original IEEE Audio-Video Bridging task group has changed name to Time-Sensitive Networking incorporating also automation.

The synchronization of the local and the system clock is achieved by periodic exchange of messages. Both SNTP and PTP offer by default a system-wide relative clock synchronization which is enabling the operation of the process. In case absolute time synchronization is required (e.g., logging of events also between different networks), a suitable external time source is required. As an example, the GPS service or a land-based clock signal can serve as a high precision time source. Synchronization to a global time reference is also required if the system is composed from different installations where there is no possibility of a direct network-wide time synchronization solution.

1) Network Time Protocol: NTP can provide a satisfactory synchronization service [9] if the required precision is in the 10s of milliseconds range or millisecond range if SNTP is used [10]. The protocol is implemented as a software-only component [11] and is widely supported. Despite the relative precision of the protocol, in most cases the precision requirement of industrial Ethernet networks does require higher precision, which requires the use of PTP.

2) Precision Time Protocol: PTP was defined to allow much more precise time synchronization [12] and as such allow the use of Ethernet for applications where the time precision throughout the network needs to read nanosecond range. To reach the required precision, PTP is using hardware support, a timestamping mechanism integrated into the network adapter of the nodes. The close connection to the NIC is also a limitation, as currently it can only be used on Ethernet networks [13]. The upcoming v3 splits the protocol into a hardware dependent and independent part and opens for different physical bearers.

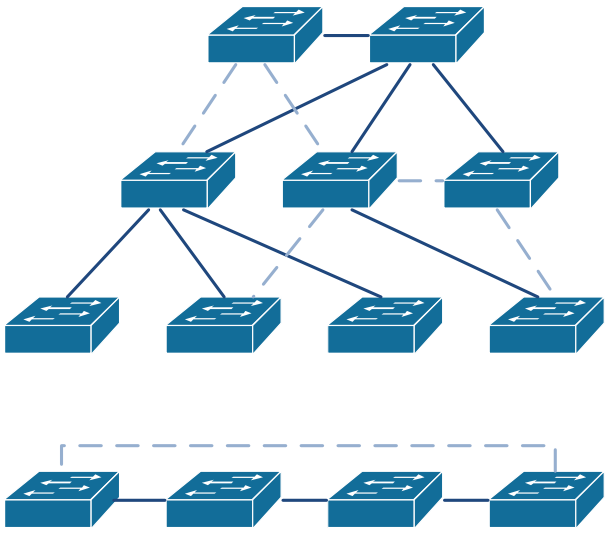


Fig. 5. RSTP redundancy is provided by stand-by links

3) IEEE 802.1AS: The protocol implements a strictly-defined subset of PTP while extending the usage area towards wireless LANs and other physical media than wired Ethernet. The objective was to provide a precision timing solution for AV purposes. As the split of the protocol between hardware dependent and independent layers was successful, in PTP v3, a similar approach is suggested.

III. Redundancy

Network redundancy is an important availability requirement for industrial applications. In upper network levels, the controller and the client/server network, dual-homed devices are common. The actual network redundancy protocol is however dependent on the application area and the supplier.

The simplest solutions offer the use of RSTP and implement stand-by redundancy by offering backup links, which can be enabled in case the primary fails. Typical physical topologies include bus, ring and tree structures.

The bus structure is not preferred as one failure might render large parts of the network inaccessible, but it might be used as one segment of a redundant network.

The reconfiguration time is a decisive QoS parameter for selecting the correct redundancy protocol.

A. Rapid Spanning Tree

RSTP is calculating a minimal-cost spanning tree of an Ethernet network. It is an IEEE protocol incorporated into the IEEE 802.1D standard. While RSTP was designed primarily for loop-avoidance, it is the primary choice for network redundancy in cases where a moderate but not bumpless

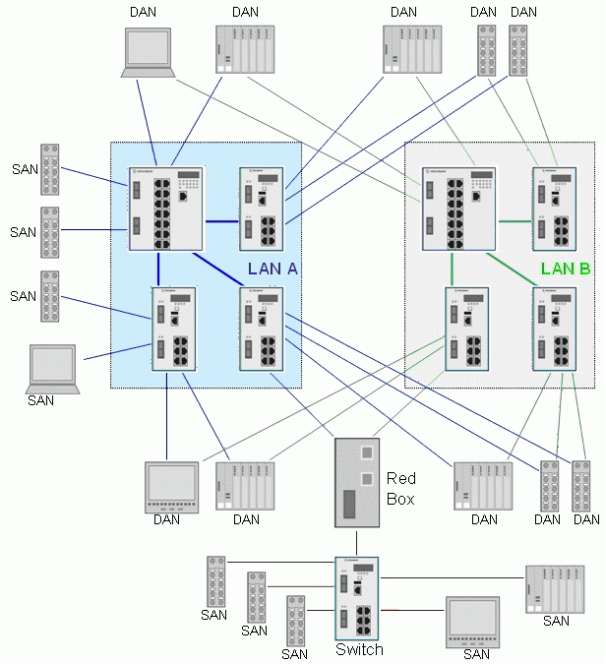


Fig. 6. PRP redundancy [14]

reconfiguration of the network path is acceptable (e.g., process automation).

Extensive research has been executed for evaluation RSTP performance and many automation network equipment suppliers have developed their own flavor of RSTP. The biggest advantage using RSTP is, that it does not require special support in the core network or at the end nodes. The default performance is acceptable where grace periods of several seconds are allowed, which is extended to a part of the factory automation field by the vendor-specific enhancements.

B. IEC 62439

As RSTP was unable to meet some redundancy requirements, a wide range of proprietary redundancy protocols were introduced for industrial Ethernet. IEC has initiated a standardization effort for high availability automation network redundancy, which resulted in the IEC 62439 family of standards. The standard describes several protocols and also references RSTP. From the availability viewpoint, in addition to the standby redundancy provided by, e.g., RSTP, IEC 62439-3 defines two seamless redundancy protocols, Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR). These two protocols provide zero switchover time, as the data is sent always on two networks in the same time.

1) PRP: is one of the two bumpless protocols defined in IEC 62439 and as such offer the highest QoS for redundancy with the deployment of a full reserve network. The two networks are in parallel operation and data is transmitted continuously over the two interfaces. A merge layer is included between the link and network layer to suppress duplicate frames at the receiver. The topology of the network is a tree with double-homed nodes. Single-attached nodes (without redundancy) are also supported.

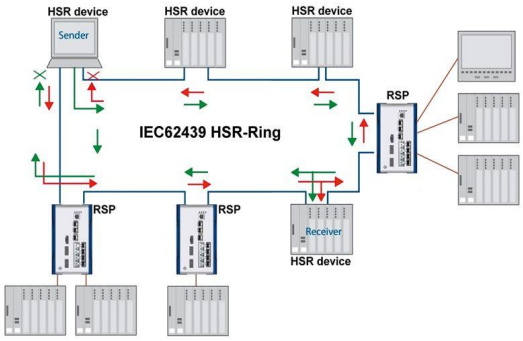


Fig. 7. HSR redundancy [14]

2) HSR: implements a two-directional ring and is sending traffic in both directions simultaneously. This solution does not require a double network infrastructure, but is using more bandwidth, as the core has to be able to carry all traffic aggregated and doubled. Special hardware for all nodes is required, single-attached nodes are supported through the use of a redundant coupler module (RedBox).

C. Convergence in Industrial Ethernet

Ethernet offers a key feature for further optimization of networks, the possibility to use it as a single communication solution which carries both automation, safety, security, communication and other network traffic. The available bandwidth and prioritization solutions allow effective use the network without compromising service quality [3]. An example for safety integrated systems is Safety over EtherCAT [8].

The use of shared infrastructure is still seen as problematic from the QoS viewpoint. It is accepted that in a typical case Ethernet can provide the necessary QoS levels but the lack of composite traffic models and scepticism for shared links is limiting the spread of using the same physical links for different classes of traffic. Most of the issues are raised in connection with Safety Integrated Systems (SIS), where the safety function is using the same communication medium as other parts of the automation network. Stakeholders are concerned about that the probabilistic transmission or network failure will stop the safety function. In contrast, the operation principle used in SIS is that if the safety message is not arriving in time, the system is going into safe state, thus the safety function is intact, but the uptime of the process suffers. From the QoS viewpoint this behavior results in both delay and loss requirements, but probabilistic QoS might be acceptable as only the process uptime is threatened but not the safety in case of the network is failing to meet the QoS parameters.

Using QoS aware planning and appropriate traffic models can however change the scepticism and result in better overall performance. To prove that engineering can be supported with tooling to achieve better QoS a proof-of-concept tool was implemented. The tool shows that engineering aspects can be used to ensure the use of available prioritization solutions, time synchronization and redundancy solutions.

IV. PROOF OF CONCEPT

The motivation behind this work was to reduce engineering costs and to explore possible solutions for provider-

independent configuration representation and setup of multiple devices in the same time [15], [16]. The potential cost reduction in the engineering phase is expected to reach 20-25% of the total cost, not counting the life cycle support.

The review of a project portfolio revealed that in most installations 2-3 vendors are involved in supplying network infrastructure based on various preferences. Although the planning of the network is done independently from the actual manufacturers, the configuration and acceptance checks do depend on per vendor knowledge and tools.

The expected result of the research task was in addition to explore possible solutions, to create a proof-of-concept tool, which can compose, deploy and modify configuration of one or multiple Ethernet switches in the same work session.

In long-term, the vision of a common configuration and management tool was defined, where planning, configuration, as-planned checking, monitoring and life cycle management was provided. Such a tool could offer a common interface to plan a network with defining the segmentation and port distribution (this covers the current network drawing step), generate configuration for the devices (which is done typically by engineers), deploy and then through discovery, check that the network has the same structure as planned (for example the VLANs are set up correctly). During operation, the tool could read out the current configuration from a device and upload it to a replacement unit, even if these are from different manufacturers.

V. HARDWARE

To explore configuration possibilities, remote configuration features of selected product lines were reviewed:

- *RuggedCom RS9xx* [17]: This switch line supports configuration update using a built-in Trivial FTP (TFTP) client or server, depending on requirements. In addition, Secure Copy (SCP), terminal with Command Line Interface (CLI) and SNMP is supported for file and configuration manipulation. As all of the reviewed managed switches, this unit offers a web interface. A vendor-specific tool for monitoring is available.
- *Hirschmann RSRxx* [18]: This switch line offers a TFTP Client, CLI access through telnet or the web interface, a java-based web interface, SCP file transfers and a proprietary Automatic Configuration Adapter. This adapter, if physically connected to the device, uploads or downloads configuration enabling easy replacement from the same vendor.
- *Moxa EDS-508* [19]: Has TFTP server and client, CLI, SCP transfers and offers a web interface. A proprietary Auto-Backup Configurator is offered for backup and restore, allowing easy replacement from the same vendor.

The research also showed that SNMP is supported on all units, although the features were focused on monitoring and not on configuration.

The review showed considerable differences between web interface structures and the available options. The differences were big enough to limit reuse of configuration knowledge and proved to support the initial assumption about cost reduction potential.

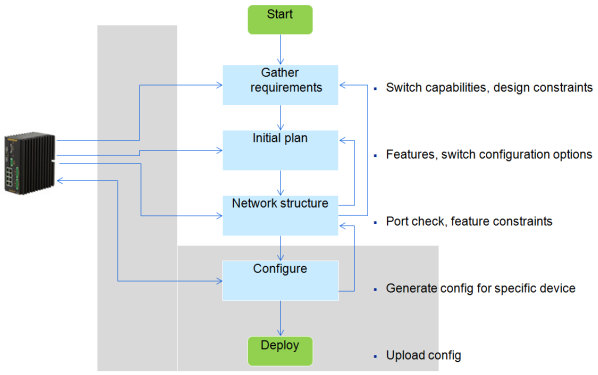


Fig. 8. Proof of concept coverage

Configuration data was accessible on all devices as structured text files, which were human readable and could be a base for the configuration tool design. In Figure 8, the expected coverage of a configuration tool is shown. The objective was to allow up- and downloading, manipulation and storage of configuration information.

A. Multiple unit configuration

One of the most important features was to check the feasibility of configuring multiple units in the same time and to explore the possible issues.

As part of the planning, a feature set was identified, which were set the same on all devices or could be calculated automatically. An example is the selection of the RSTP root bridge.

Other questions were risen in connection with the long paths and rings used in these networks. It was assumed, that depending on the behavior of the devices, the configuration might need to be topology aware.

The user interface was also a crucial point, as the objective was to reduce engineering cost, which pointed towards a simpler interface than most of the switches offered. This request was supported by, that only a handful of features needed to be set and most of the parameters were left at factory defaults.

VI. ENGINEERING TOOL EXAMPLE

The implementation was focused on a subset of the possible features. Based on feedback from engineering, configuration of multiple devices and support of multiple vendors were selected as key features, which should be supported by a simple user interface. In Figures 9 and 10, the test user interface is shown for single- and multi-unit mode.

The planned system was designed to cover tasks associated with configuration and deployment stages of the engineering process. To ensure, that options, which are not being used by the system are preserved, the tool only replaces relevant parts of the original configuration files with new data 11.

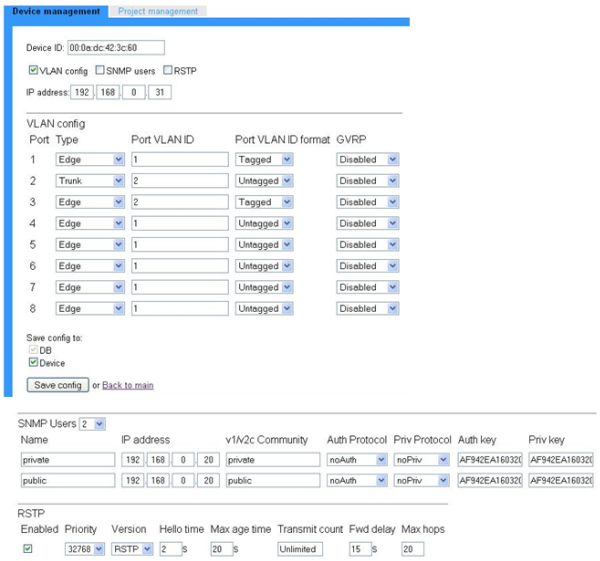


Fig. 9. Single unit configuration

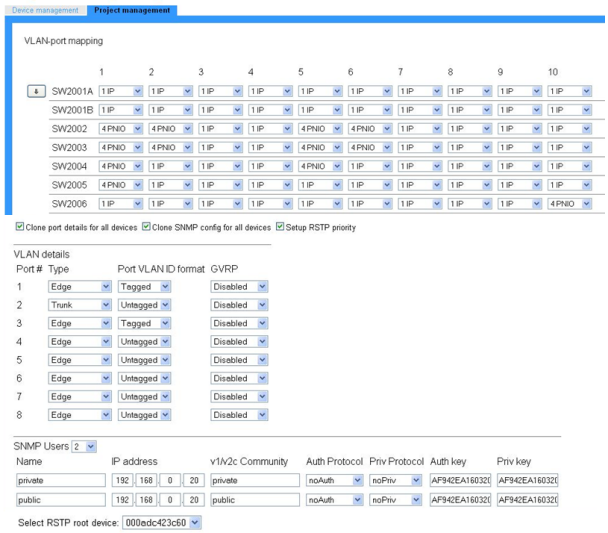


Fig. 10. Multiple unit configuration

A. Requirements

- vendor independent, simple user interface
- remote configuration of one or multiple devices
- life cycle support with configuration versioning and cloning
- configure selected features

B. Features

A subset of available features on the switches was selected based on experiences from engineering. This set was planned to cover most of the engineering needs without resulting in a complex interface.

The feature set was defined for both single and multi mode as:

- *Host IP*: to be able to set the device's IP
- *Port-based VLAN*: allow the setup of per-port VLANs
- *SNMP setup*: configure SNMP access rights and community memberships
- *Spanning Tree*: select STP protocol and allow changes in bridge priority

To support documentation, an automatic network documentation generator function was also included.

A single unit configuration section was included for practical purposes and served also as a testbed for checking the configuration generation capabilities.

The system was designed so, that it would preserve changes made outside the configuration tool (thus allowing device specific configuration for features not covered by the tool), so the composition of the configuration data was implemented in a way, that it is only changing the relevant part and keeps the rest of the data untouched (Figure 11.

C. Multiple vendor support

Enabling support for multiple vendors has risen several issues, which were not foreseen. Even if all the switches covered were complying the same IEEE standards, the actual implementation and availability of features depends on the vendor.

As a result, a vendor independent representation of the configuration data was needed and the configuration generation process had to be split into storage, representation and actual configuration data.

D. Multiple device support

Configuring multiple devices in one session was considered as the most important feature, as this would result in the highest cost cut. Covering multiple devices also meant that the difference between the per unit web interfaces and the configuration tool might be the most emphasized.

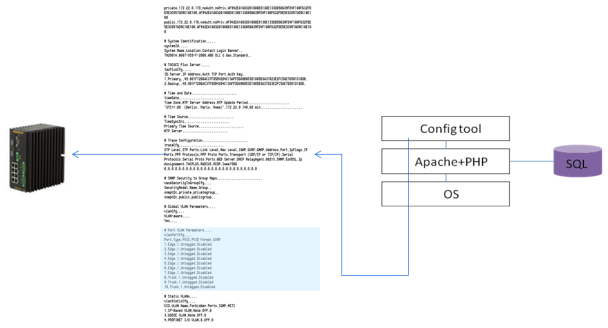


Fig. 11. Composition of the configuration

For the IP configuration and VLAN settings, a matrix of switches and VLANs was generated. This offers a single-screen overview of a typical network in the evaluated projects. The drawback of this representation is, that if a large number of ports and switches are used, the size of the matrix is getting large. This limitation was found acceptable in this case, as in

a typical industrial environment low port count switches are used, so adding more switches will result in a longer matrix, but the width will stay limited.

The tool offers cloning of the port and SNMP settings to all devices and setting the root RSTP bridge.

E. Connectivity

The review of connectivity methods has shown that it is problematic to choose one specific solution. Even in case of just the three product lines reviewed, different protocols turned out to be easier to use.

For the proof of concept, for one vendor (RuggedCom) TFTP was chosen for up- and downloading configuration data. For the other vendor (Hirschmann), CLI-based configuration and telnet. While being aware, that none of these protocols provide secure transfers, this requirement was relaxed for the current version. This decision was supported by that the tool is intended to be used during engineering, where these networks operate as isolated islands.

VII. Lessons learned

There are several important issues that were identified during the evaluation and development of the configuration tool.

A. Vendor independent configuration data

In order to support multiple vendors, the configuration data needs to be stored in an independent format. Generation of the appropriate configuration file or script depends on the vendor's implementation and there might be considerable differences.

Changes between vendors in most of the cases results in information loss about the configuration of the device. An example is the support of vendor specific spanning tree protocols. The use of these proprietary protocols is beneficial if the network is homogeneous, but might cause problems if multiple vendors are present. If the original configuration was set up, e.g., to use RuggedCom's eRSTP and the device is replaced with another manufacturer's switch, the configuration tool has to fall back on, e.g., RSTP, as that is the nearest standard protocol which is supported by the new device.

If later the device is changed back (e.g., a device needed to be taken out from the network and was temporarily replaced by another), if the configuration storage depends on the vendor, then only RSTP will be used even if eRSTP is available, as the migration process will only create a representation of the current configuration in the new device.

B. Topology-awareness

An interesting issue with configuration was raised while the tests of the multiple unit configuration were executed. In single unit mode, there were no problems, the configuration was updated, the device was reset and after some seconds, network operation was restored. The same happened if multiple units were configured in an office-like topology (equalized tree), where only a few levels of switches were involved and the longest path was 3-4 hops. In case of industry-typical rings, anomalies and connectivity errors happened.

The investigation showed that while the update operation itself is done in a fraction of a second and it takes approximately 2-3 seconds for a device to reset, this was too short to update all members of the ring. In the tree topology, the devices were updated before the first unit decided to reset. In the ring, however, these resets happened before all members were updated. The result was that the network was falling into fractions and in some cases one had to approach each *lost* device separately.

As a result, it was identified that it would be beneficial if in case of multiple unit configuration, the update would be done with respect to the topology, starting from the leaves and progressing upwards in the tree. The same approach can be used in rings, as these will be represented as a long unbalanced tree (in normal operation RSTP is disabling the redundant link to avoid a loop).

C. Identical configurations

Although the switches used in this work were not the most complex units available, it turned out to be a complicated task to reach exactly the same configuration on devices from different vendors.

A typical example is the configuration of a trunk port. In one case, this option was available directly on the web interface and in the configuration file, but on a different switch at least 6 commands in the CLI were required.

Another example is the above mentioned case of RSTP. In practice, all major vendors have their own enhancements to RSTP to achieve better convergence times. This also means, that these proprietary solutions can only used on homogeneous fractions of the network. If a device is replaced by a device from a different vendor can result in weaker performance, as all the units have to fall back on the first standard solution, which in most cases will be RSTP.

VIII. Applying QoS functions in engineering

Using the proof-of-concept as starting point, possible extensions towards QoS aware design were explored. The objective was to support more of the typical engineering tasks while optimizing the use of available QoS functions. These include traffic prioritization, optimization, topology optimization and time synchronization [20], [21].

The first field was topology awareness. One of the important aspects were to support the automatic inclusion of default spares, which can be later used either for redundancy or for extensions. Other properties were the automatic inclusion of redundant links to reach a specific level of redundancy in a spanning tree structure.

A. Topology generation

The size of the network has a negative impact on the achievable QoS. The engineering tool could, within defined limits, aim for reducing the network's critical parameters and provide an updated approximation of QoS parameters. In topology engineering, the primary goal with regard to delay and reconfiguration time is to reduce the longest critical path.

First the critical path can be identified using Critical Path Method (CPM) and then equalized with a tree-equalization algorithm. The equalization can be supported by the configuration tool with allowing the exchange of network infrastructure with low effort (e.g., a larger port count switch instead of a smaller one).

When the topology is finalized, protocols running on the network might be optimized further, e.g., by manual assignment of the RSTP root bridge, which has an optimal place in the *middle* of the network: the point most centrally located with regard to network paths.

If the QoS requirements cannot be fulfilled with the actual design (e.g., redundancy offered by RSTP), the engineer can be informed and the network topology can be transformed in order to meet the requirements. This can, e.g., result in changing a ring network into a redundant star (shorter paths with the expense of higher cabling cost) or adding a secondary network (if bumpless redundancy is required).

In case the QoS requirements of the critical path cannot be fulfilled with topology manipulation, a technology with intrinsic QoS can be recommended. The proof-of-concept although supports configuration of several devices, the connection between them is implicit and the tool has no information about the planned topology.

B. Traffic prioritization

Traffic prioritization in Ethernet is offering a feature which can be easily configured and achieves a level of DiffServ-like operation with relative-priority traffic classes. Network devices support a number of priority queues and important traffic is preempting less important frames. Depending on the prioritization scheme, a probabilistic value for network delay and loss can be given for the specific traffic classes. These classes can be shared between several different traffic sources, which might result in internal queueing. The correct selection of the traffic to priority mapping is important for the desired operation. Strict priorities might lead to excessive delays on low priority traffic if the higher priority traffic is utilizing most of the available bandwidth. To avoid exhaustion, the loose policy is supporting not only priority and First In First Out (FIFO) scheduling of the priority queues but also ageing between the queues.

In an engineering tool, the priority-traffic mappings could be done automatically in addition to the selection of scheduling policy.

C. Traffic optimization

Multicast and broadcast is often used in automation protocols. Aggregation of traffic on LANs is less important in a non-industrial environment, primarily because of the less strict QoS requirements and the flatter architecture. In the industry-typical ring topologies, where traffic is aggregated between a high number of nodes, it can be beneficial to use traffic aggregation to avoid queueing. During engineering, multicast grouping protocols like Internet Group Management Protocol (IGMP) can be used.

D. Time synchronization

Time synchronization is a critical feature on industrial networks and some aspects have to be supported in the engineering phase. The selection of the synchronization protocol used poses different requirements towards the infrastructure elements. SNTP is generally supported in all network equipment, and since it is software only, it can be also deployed in nodes which does not support it by default. PTP on the other hand requires hardware support for the preferred precision and thus limits the possible range of network equipment used.

IX. Engineering support

There is a considerable potential to cut costs in network engineering if appropriate tools are available. Although network management software are available and widely used in office environments, their resource need and cost render them unrealistic for industrial deployment.

The proof of concept implementation of a configuration tool, which can partially automate the setup of Ethernet switches aims to reduce the engineering complexity and to support QoS aware planning and deployment of networks. The main difference compared to proprietary solutions is, that this tool supports multiple vendors and with a vendor independent representation of configuration data, also allows future extensions.

Testing of the tool revealed several issues associated with device configuration, especially related to problems caused by the topology and the complexity of generating identical configurations for switches from different vendors.

X. Conclusion

QoS is an important property of a communication network, independently whether it is used in an industrial or another environment. The issues associated with providing specific service levels on an inherently non-deterministic packet switched network are the same, although the importance of the metrics differ.

The uncertainty related to the probabilistic transfer time bounds provided by Ethernet initially cause scepticism towards the extended use of the technology. Most of these negative opinions have roots in the past and refer to problems which are mostly non-existent on current, typically 1G, full-duplex, switched networks. To ensure, that Ethernet can successfully overtake as primary communication technology on all levels of industrial networks, a set of QoS metrics need to be defined with appropriate weights to allow calculation of expected network performance.

Current metrics and weights are focused on different applications, mostly AV, whereas industrial uses require a different weight composition. The paper has pointed out several of these metrics, e.g., redundancy, time synchronization, prioritization, packet loss and delay which are more important in the industrial field than in AV in contrast to, e.g., bandwidth, which in the majority of cases is not a primary concern in industrial environments.

Industrial topologies are also introducing problems which were considered as non-existent in the office domain because

of different network structures. Rings and deep trees are possibly avoided in other networks but are preferred in the industry sector. The depth of an industrial network can be approximated with, e.g., $(O)(\log_6 n)$ versus the office $(O)(\log_{16} n)$ where 6 and 16 is the branching factor, *n* is the number of nodes. This results in a weaker operation of several protocols (like RSTP) and result in a worse QoS.

A proof-of-concept implementation of a bulk configuration tool is shown, which can serve as a basis for a more complex network engineering and support system. Bulk configuration of devices is the first step towards using already existing QoS functions in industrial applications without requiring expert knowledge during planning and commissioning.

The possible extension of the tool with topology manipulation will enhance the QoS of the system by transformations using well-known algorithms and open for selecting critical areas, where the required parameters can only be reached by using intrinsic QoS technologies.

REFERENCES

[1] Gy. Kálmán, Mass Configuration of Network Devices in Industrial Environments, in Proceedings of International Conference on Networks (ICN) 2013, pp. 107-111

[2] Kleineberg et.al., Automatic device configuration for Ethernet ring redundancy protocols, Emerging Technology and Factory Automation (ETFA) 2009.

[3] Industrial Ethernet: A Control Engineer's Guide, Cisco White Paper, 2010.

[4] C. Rojas, P. Morell, Guidelines for Industrial Ethernet infrastructure implementation: A control engineer's guide, IEEE IAS/PCA 2010.

[5] G. Adelson-Velskii, E. M. Landis, An algorithm for the organization of information, Proceedings of the USSR Academy of Sciences 146, pp. 263-266. (Russian) English translation by Myron J. Ricci in Soviet Math. Doklady, 3:1259-1263, 1962.

[6] H. Kirrmann, Highly Available Automation Networks, Standard Redundancy Methods, Rationales behind the IEC 62439 standard suite, ABB, 2012.

[7] Communication Networks and Systems in Substations - Part 5 Communication Requirements for Functions and Device Models, IEC 61850-5, 2003.

[8] EtherCAT Technology Brochure, EtherCAT Technology Group, November, 2012.

[9] D.L. Mills, A Brief History of NTP Time: Confessions of an Internet Timekeeper, ACM SIGCOMM, Computer Communication Review, vol. 33, no. 2, pp. 9-22, April 2003.

[10] M. Ussoli, G. Prytz, SNTP Time Synchronization Accuracy Measurements, Emerging Technology and Factory Automation (ETFA) 2013.

[11] P. Ferrari, A. Flammini, S. Rinaldi, A. Bondavalli, F. Brancati, Experimental Characterization of Uncertainty Sources in a Software-only Synchronization System, IEEE Transactions on Instrumentation and Measurement, vol. 61, no 5, pp. 1512-1521, May 2012.

[12] T. Neagoe, V. Cristea, L. Banica, NTP Versus PTP in Computer Networks Time Synchronization, in Proceedings of IEEE International Symposium on Industrial Electronics, 2006, vol. 1, pp. 317-362.

[13] IEEE Standard for a Precision Clock Synchronization Protocol of Networked Measurement and Control Systems, IEEE Std 1588-2008, 2008.

[14] Media Redundancy Concepts, High Availability in Industrial Ethernet, Hirschmann White Paper, 2011.

[15] Imtiaz et.al., A novel method for auto configuration of Realtime Ethernet Networks, Emerging Technology and Factory Automation (ETFA) 2008.

[16] Reinhart et.al., Automatic Configuration (Plug & Produce) of Industrial Ethernet Networks, INDUSCON 2010.

[17] Rugged Operating System v3.10 User Guide, Ruggedcom, January 19th, 2012.

[18] Reference Manual, Command Line Interface Industrial Ethernet Gigabit Switch Release 7.0, Hirschmann, 2011.

[19] Datasheet, EDS-508, Moxa, 2010. May 5th

[20] P. Ferrari, A. Flammini, S. Rinaldi, E. Sisinni, On the Seamless Interconnection of IEEE 1588-based Devices Using a PROFINET IO Infrastructure, IEEE Transactions on Industrial Informatics, vol. 6, no. 3, pp. 381-392, August 2010.

[21] G. Gaderer, P. Loschmidt, T. Sauter, Quality Monitoring in Clock Synchronized Distributed Systems, in Proceedings of IEEE International Workshop on Factory Communication Systems, pp. 13-21.

A Novel Approach to Interior Gateway Routing

Yoshihiro Nozaki, Parth Bakshi, and Nirmala Shenoy
College of Computing and Information Science
Rochester Institute of Technology
Rochester, NY, USA
{yxn4279, pab8754, nxsvks}@rit.edu

Abstract— **Most ISPs and Autonomous Systems (AS) on the Internet today use Open Shortest Path First (OSPF) or Intermediate-System-to-Intermediate-System (IS-IS) as the Interior Gateway Protocol (IGP). Both protocols use the Link-State routing approach and require distribution of link state information to all routers in a network or in an area. Any topological changes require redistributing updates and refreshing routing tables. This results in high convergence times. During convergence, packet routing becomes unreliable. During the years as network sizes have grown, the routing table sizes have also exhibited a linear growth. This is indicative of scalability issues in the current routing approaches and could be a limiting factor for future growth. Future Internet initiatives, which were started worldwide almost a decade ago, have enabled novel approaches to address the routing problem. In this article, such a novel interior gateway routing approach is presented. The approach leverages the tiered structure existing among ISP networks, AS, and in general most networks. The routing protocol was thus named the Tiered Routing Protocol (TRP). Though TRP can be used for both inter- and intra-AS routing, in this article, it is presented as a candidate protocol for intra-AS routing. TRP operations are supported by a tiered addressing scheme. Use of TRP replaces both the Internet Protocol (IP) and the routing protocol. The rationale for TRP and its details followed by its evaluation over the US national testbed namely Emulab are presented in this article. TRP's performance is compared with OSPF to highlight its major contributions to address routing scalability.**

Keywords-Intra-domain Routing; Network Convergence; Internetworking Architectures; Tiered Architectures; Routing Table sizes; Interior Gateway Routing

I. INTRODUCTION

This article is an extension of the conference paper [1] aiming at providing a detailed analysis of ISP network and transition platform between Internet Protocol (IP) routing and the proposed routing protocol.

IP provides best effort reachability for communication across networks and nodes connected to the Internet. In IP networks, routers use routing protocols to discover and maintain routes and also to recover from route failures. Routing tables maintained by current routing protocols increase almost linearly with increase in network size and is an unhealthy trend indicating scalability issues, which can manifest as performance degradation. Also, the time taken for the network to adapt to topological changes increases with increase in network size resulting in higher convergence times during which routing is unpredictable and unstable. With more and more users connecting to networks today, this poses a serious problem. Patch and evolutionary solutions have been and are being proposed and implemented to address the problem both at the inter domain and intra domain level [2][3].

Interior Gateway Protocols (IGP) such as Routing Information Protocol (RIP) and OSPF were designed to work with IP. RIP is a distance vector (DV) protocol and can be used in networks with a maximum diameter of 15 hops. Large ISP networks thus use Link-State (LS) IGPs such as IS-IS or OSPF, which uses the area concept to segment networks into manageable size. LS routing protocols require periodic updates and redistribution of updates to all routers in the network or in an area on link state changes. Each router running the LS routing protocol executes the Dijkstra's algorithm on the collected link state information to populate routing tables. Dissemination of network-wide (or area-wide) link state information also adversely impacts scalability and convergence times in the networks using OSPF. In some cases, the physical location of areas requires use of virtual links to the backbone area further limiting the versatility of OSPF.

A primary contribution in this work is to decouple the dependency of routing table sizes from the network size. However, this had to be found on a solution that would also be acceptable to the Internet service provider community. Thus, the proposed routing protocol adopts an internetworking model that derives from the structures used by ISPs to define their business relationships namely the *tiers*. The routing protocol so proposed is called the *tiered routing protocol* (TRP). A new tiered addressing scheme to enable efficient operation of the TRP was also introduced. The tiered address inherits attributes of the tiered structures and expresses them explicitly in the address to be used for TRP operation and packet forwarding. To decouple dependencies between connected network entities, and enable their easy movement and connections to other entities a nesting concept is introduced [4].

Traditionally, the Internet Protocol was designed to provide logical addresses, application transparency and forwarding of packets based on routing table entries. A routing protocol was thus needed to populate the routing tables. For this purpose, different types of routing algorithms and protocols based on these algorithms were developed. Examples are distance vector algorithm based RIP, link state algorithm based OSPF and path vector algorithm based Border Gateway Protocol (BGP). TRP has been designed to replace both the IP and routing protocol. This is true for inter- and intra-AS routing. Thus, interworking functions and

complexities due to the interworking of two protocols are reduced.

In this article, the rationale and detailed operation of TRP as an IGP are described. TRP has also been evaluated and its performance compared with OSPF using the US national testbed – Emulab [4]. In this article, TRP is applied to an AS and the process of identifying tiers, tiered address allocation, population of routing tables, packet forwarding and failure handling is described. TRP implemented in an AS also provides the network setting for performance evaluation and comparison with OSPF. To provide a more comprehensive performance evaluation, the following metrics were evaluated: initial convergence times, convergence times after link failures, routing tables sizes, and control overhead during initial convergence and convergence after link failure.

Section II describes related work for the reduction of convergence times in IGPs. Section III describes the two routing protocols namely OSPF and TRP. The description is limited to the performance studies targeted, the convergence times, routing table sizes and control overhead. Some foundational studies, which led to the proposal of the tiered routing approach, are also included in this section. Related work on TRP is covered more extensively as compared to OSPF as details about OSPF are RFC standards [2]. Section IV discusses the use of Multi Protocol Label Switching (MPLS) as a transition facilitator and the mechanisms that can enable a successful transition. Section V provides details of the emulations tests and the techniques adopted to collect results in the Emulab testbed. The two AS topologies evaluated are also described. Section VI provides the averaged results collected from several emulation runs over several test sites. This is followed by a detailed analysis of the results for both TRP and OSPF operation. In Section VII, the conclusions and intended future work are discussed.

II. Related Work

Significant research effort can be noticed towards improving and enhancing IGP performance. Some these efforts were directed to the reduction and optimization in IGP convergence time subsequent to link state changes in the network or area. Work in this regard can be broadly categorized into: (a) reducing failure detection time and (b) reducing routing information update time.

A. Reduction in Failure Detection Time

Layer-2 notification is used to achieve sub-second link/node failure detection. However, this relies on types of network interfaces and does not apply to switched Ethernet [6].

Layer-3 notification is the more adopted method for link failure detection. For this purpose, the *Hello* protocol is used. The hello protocol, besides being used to disseminate neighbor information, is also used to identify link/node failure in many routing protocols and is the layer-3 failure detection mechanism. OSPF sends *hello* packets to adjacent routers at an interval of 10 sec by default. The hello packet contains information on all links that a router is connected to. On missing four *hello* packets consecutively from a neighbor, OSPF routers recognize an adjacency failure with that neighbor router. Reducing *hello* packet interval time to sub-seconds can significantly reduce the failure detection time, but at the expense of increased bandwidth usage due to increase in the number of periodic *hello* packets. Increased number of hello packets in a short interval can also increase possibility of route flaps.

B. Reduction in Link State Propagation Time

Although link/node failure detection time can be reduced to sub-seconds, propagating the link status to all routers in the network takes time and is dependent on the network size.

To reduce such delays, an approach that suggests the use of several pre-computed back up routing paths was proposed. Pan et al. [7] proposed the MPLS based on a backup path to reroute around failures. However, having all possible MPLS back up paths in a network is not efficient. Multiple Routing Configurations (MRC) [8] uses a small set of backup routing paths to allow immediate packet forwarding on failure detection. A router in MRC maintains additional routing information on alternative paths. However, MRC guarantees recovery only from single failures. Liu at el. [9] proposed the use of pre-computed rerouting paths if the same can be resolved locally. Otherwise, multi-hop rerouting path had to be set up by signaling to a minimal number of upstream routers. Another approach limits the propagation area of link state update after failure. Narvaez [14] proposed limited flooding to handle link failures. When a link failure occurs, the descendants of the failed link in the shortest path tree are determined and the new shortest path without the failed link is calculated. Then, the updated information is propagated in only the area of descendant nodes.

The two delays discussed above are significant. However, the SPF recalculation time can also be almost a second in large networks [6]. As packet loss/delay or routing loops occur during convergence, it is important to reduce this time. Novel routing approaches under the future Internet initiatives thus provide the opportunity to view the routing problem from a fresh perspective and thus design solutions that are not constrained by the current architectures or implementations.

III. Routing Protocols and Operations

In this section, we describe the operations of the two protocols studied in this article namely OSPF and TRP. In the case of OSPF, only a few basic operations necessary to explain the performance metrics are presented. Details of OSPF operation are publicly available in the Related RFC documents [2].

TRP operation is explained in detail for intra-domain routing. This includes implementing tiered structures within an AS, tiered address allocation to devices in the tiers, routing table population and maintenance with TRP, and the packet forwarding algorithm and link failure handling. Some properties of the tiered address, which makes TRP robust and a few TRP features that result in low convergence times and small routing table sizes are also discussed.

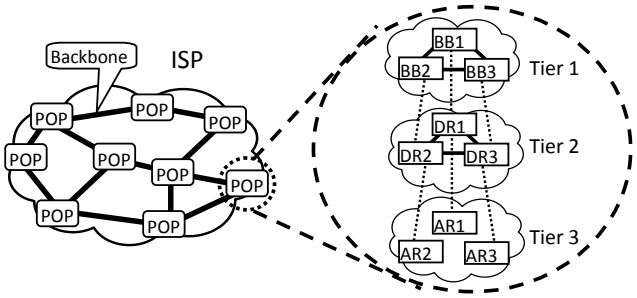


Figure 1. Tiered Topology within an ISP

A. Open Shortest Path First (OSPF)

Link State routing protocols offer faster convergence with theoretically no hop and no network size limitations compared to distance vector (DV) based routing protocol. The small sized update packets consume less bandwidth, as compared to the DV protocols. However, the update packets have to reach all routers in the network or area for successful convergence and stability in routing tables. The routing table update using Dijkstra's algorithm is complex and CPU intensive if the number of entries in the link state database is high [2].

Basic operations of OSPF include: (a) establishing adjacencies with neighbor routers and electing a Designated Router (DR) and a Backup DR (BDR); (b) maintaining Link State Database (LSDB) and; (c) executing the Dijkstra's algorithm on the LSDB to populate the forwarding database or routing tables. These operations are invoked during startup and also when there are link state changes. Convergence in the two cases is impacted differently and thus described separately below.

1) Initial Convergence in OSPF

a) Establishing Adjacencies: OSPF starts by establishing adjacencies with direct neighbor routers using the *Hello* protocol. *Hello* packets are sent on each interface using a multicast address to neighbor routers. Once *Hello* packets are exchanged, each router recognizes if they are connected via a point-to-point network or a multi-point network such as Ethernet, where several routers are in the same subnet. In the case of multi-point networks, OSPF will elect a DR and a BDR using the router *prioity* and the router *ID*. This is necessary to reduce the number of adjacent direct neighbors and the traffic to establish / maintain them.

b) Maintaining Link State Databases: Hello protocol is also used for link state check between established adjacent neighbor routers. On link state establishment as routers come up, distribution of adjacency information to all routers is initiated by flooding Link State Advertisements (LSA). Each router records all the recived link state information that was flooded in a LSDB.

c) Populating Routing Tables: Using the topology information in the LSDB, each router then locally computes the shortest paths from itself to all other routers in the network (or area), using the Shortest Path First (SPF) or Dijkstra algorithm to populate the routing tables or Forwarding Information Bases (FIB).

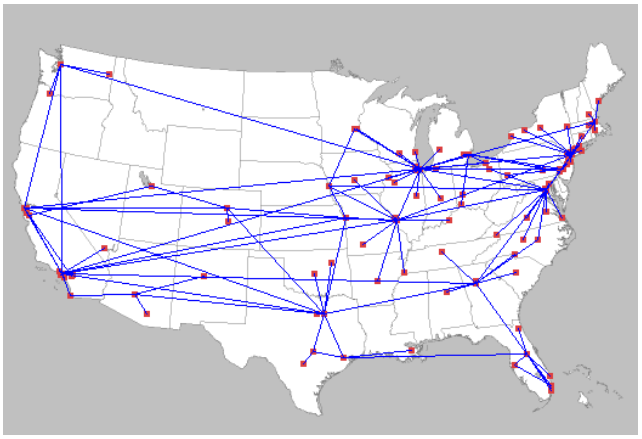


Figure 2. AT&T POP Level Network in the US

2) Convergence After Link / Node Failures

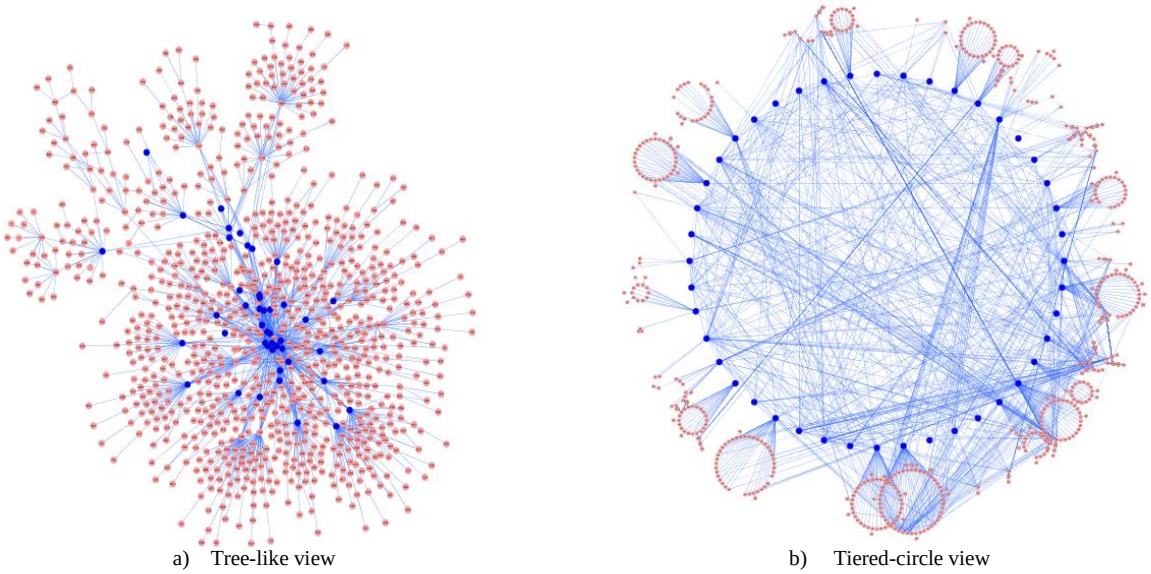
a) Failure Detection: Missing 4 consecutive *Hello* packets from a neighbor indicates link or router failure on that link and hence is one mechanism for failure detection. This is the layer-3 failure detection mechanism and has been adopted predominantly.

b) LSA Propagation: Subsequent to a failure detection, a router generates new LSAs. The LSAs have to be propagated to all routers in the network (area). The time for generating new LSAs on a single failure is between 4 to 12 milliseconds (ms) [9]. OSPF specifies that LSAs cannot be generated within 5 seconds from the last LSA generation time. This provides sufficient time to update the LSDB from the last event and run the Dikjstra alogorithm. LSA propagation time also depends on the number of hops between the routers in the network and the processing delay at each router and transmission delay at each hop.

c) SPF Recalculation Time: When new LSAs update the LSDB they trigger new SPF calculations to update the FIB. Two parameters delay SPF calculations; a *delay timer*, which is 5 seconds and a *hold timer*, which is 10 seconds by default. *Delay timer* is the time between the *new LSA arrival time* and *start of SPF calculation time. Hold timer* limits the interval between two SPF calculations.

B. Tiered Routing Protocol (TRP)

The underlying operational principles of TRP derive from the tiered structure existing in our networks today. However, TRP can run on physical meshed network by creating logical tree-like hierarchical topology through the use of Tiered Routing Addresses (TRA). Hence, in this section, we first describe the process adopted to identify tiers in a given network topology. In large ISP and AS networks, there are backbone routers that connect to one another and extend the connectivity to distribution routers. The distribution routers in turn connect to access routers or sub-networks. In this network scenario, the set of backbone routers can be designated as tier 1 routers, the distribution routers would be the routers at tier 2 and the access routers and sub-networks that they connect would be tier 3. This is the tiered structure adopted for implementing TRP within an AS.



a) Tree-like view b) Tiered-circle view

Figure 3. NY-POP Router-level network in AT&T

To understand the extension of the tiered concept to ISP networks, Fig. 1 is used. Fig. 1 shows a 3-tier structure of that can be identified within an ISP network. Typically, inside of an ISP, there are several Point of Presences (POPs), which form the backbone of that ISP. Each POP in turn comprises several routers, some of which are backbone routers that are primarily meant to connect to other backbone routers in other POPs. Inside of an ISP POP, there is a set of backbone (BB) routers as shown in the projected BB cloud (on the right side of the picture), which can be associated to tier 1 within the POP. The BB routers connect to distribution routers (DR), which can be associated to tier 2. The distribution routers in the DR cloud provide redundancy and load-balancing between backbone and access routers (AR). The ARs then connect to customer or stub networks. The ARs and the stub network can thus be associated to tier 3.

1) Validating the Tiered Approach: In this subsection, we validate the use of the tiered approach using the tiered structure adopted in ISP networks. For this purpose, we conducted some studies using data from the Rocketfuel dataset [15]. This dataset has router-level connectivity information of ISPs. From the Rocketfuel dataset, we imported the AT&T router connectivity information using Cytoscape [16] that also helps to visualize AT&T's router-level topology on the US map (this excludes Hawaii and Alaska). The dataset contains not only the connectivity information, but also the router's location (city) information. Thus, we were able to map each router and city in the visualization shown in Fig. 2.

In total, 11,403 routers and 13,689 links interconnecting the routers were identified under this study. Each city in the topology visualization is a POP that has a large number of routers. A total 110 POPs were identified in the AT&T ISP network in Fig. 2. In each POP, routers connecting with routers in other POPs were identified as BB routers.

2) Associating Routers to Tiers: One of the biggest POP in the AT&T ISP network is the New York POP (NY-POP), which has 946 routers. Among these, 44 of them were identified as BB routers that have link(s) to other POPs. NY-POP router-level topology visualized as a tree structure is shown in Fig. 3 (a). The slightly large dots belong to a node (router) in the tree that has numerous branches. These routers are thus ideal candidates to be the BB routers in tier 1. Using Cytoscape, the visualization was changed to the one shown in Fig. 3 (b), where the BB routers now form the inner circle. Routers that are one hop or a maximum of 5 hops from BB routers were identified as the distribution routers (DR). Some DRs had multiple connection to the BB routers. The edge routers are the access routers that were associated to tier 3 in the POP.

Based on the NY-POP topology observation and the studies conducted, we could identify a total 44 BB routers, 542 DR routers, and 360 AR routers. Once the tiered structure has been identified and the routers associated to tiers explicitly, the tiered address can be allocated as described next.

Once tier 1 nodes are identified, an automated Tiered Routing Addresses (TRA) allocation process can be initiated [10]. This process is explained in the next section. Below we discuss some inherent features of the TRA and the resulting impacts on TRP.

3) TRA Allocation: TRA depends on the tier level in a network and carries the tier value explicilty as the first field. The tier levels can be assigned as described above. Routers closer to a backbone or default gateway have lower tier value and routers near the network edge have higher tier value. TRA can be allocated to a *network cloud* (that comprises of a set of routers used for a specific purpose, such as backbone, distributions and so on) or a router. They are however not allocated to a network interface. Network interfaces are identified by port numbers. However, a router or end node can have multiple TRAs based on its connection to several upper tier routers or networks. This helps to support multi homing.

4) TRA Guarantees Loop-Free Routing: The automated TRA allocation starts from a node at a lower value tier to nodes at higher value tiers. The parent node's address (without the tier value) is part of a child node's address and

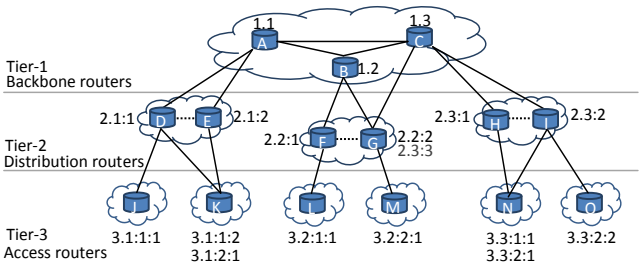


Figure 4. Example Tiered Topology and TRA

precedes a child's unique identifier. As TRAs determine the packet forwarding paths, this feature in a TRA avoids packet looping. However, this dependency can be decoupled at any level through *nesting* without affecting the loop-free packet forwarding.

5) Nested TRA: Let us consider the case where a TRA is assigned to a network cloud. A new tiered structure and TRA can be started for entities within the network cloud, allowing nesting of TRAs. If a network administrator wishes to incorporate clouds in a cloud, nested TRAs can be used where the TRA of an inner cloud does not depend on the TRA of the outer cloud. This decoupling introduces as high level of scalability and flexibility in the internetwork routing operations.

6) Inherent Routing Information: A TRA carries the path information between a lower tier entity and an upper tier entity due to the fact that a child inherits a parent's TRA (without tier value) as part of its address. Thus, a route between two communicating enitites or nodes can be identified by comparing the nodes' TRAs. If a node has multiple TRAs, a sender node may select a communication path based on criteria such as a shorter path or path with better resources.

7) TRP Convergence Time: TRP does not require distribution of routing information due to the inherent route information carried by the TRA. Network convergence in TRP is the time required for direct neighbors to recognize the topology changes in the one-hop neighborhood (in some cases a little more delay may be incurred as infromation may have to propagate down/up a tree branch). However, this time will thus be several magnitudes less than the converegence times experienced by current routing protocols. The extent of information dissemination can also be controlled for optimized operation.

8) TRP Routing Table Size: The packet forwarding decision in TRP is based on next-hop tier level in the direction of packet forwarding, and has only three choices: same tier level, upper tier level, and lower tier level. Thus, the routing table has to be minimally populated with the directly connected neighbor networks / routers. Further optimization is possible by including the two-hop or three-hop neighbors.

C. TRP Operation

Several of the TRP operations such as address allocation, packet forwarding, link / node failure detection / recovery, address re-assignment, and addition / deletion of nodes are explained in this section.

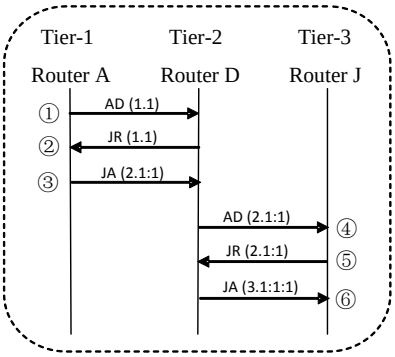


Figure 5. TRA allocation process

TABLE I. ROUTING TABLES OF ROUTER F AND G FROM FIGURE 4

Router F {2.2:1}						Router G {2.2:2, 2.3:3}					
Uplink		*Down*		*Trunk*		*Uplink*		*Down*		*Trunk*	
Port	*Dest*	*Port*	*Dest*	*Port*	*Dest*	*Port*	*Dest*	*Port*	*Dest*	*Port*	*Dest*
1	1.2	3	3.2:1:1	2	2.2:2, 2.3:3	1	1.2	3	3.2:2:1	4	2.2:1
Dest – directly connected neighbor						2	1.3				

1) Address Allocation Process: TRA allows automatic address allocation by a direct upper tier cloud or node. Once tier 1 nodes acquire their TRAs (or have been assigned their TRAs), tier 2 nodes will get their TRA from the serving tier 1 node.

a) TRA Allocation: The process starts from the top tier, i.e., tier 1. A tier 1 node advertises its TRA to all its direct neighbors. A node, which receives an advertisement, sends an address request and is allocated an address. For example in Fig 5, Router A with TRA 1.1 sends Advertisement (AD) packets to Routers B, C, D, and E. Routers D and E send Join Request (JR) to Router A because they do not have a TRA yet. Router B and C do not request address to Router A because they are at the same tier level. Router A allocates a new address (2.1:1) to Router D using a Join Acceptance (JA) packet. Another new address (2.1:2) is allocated to Router E. The last digit of the new address is maintained by the parent router, i.e., Router A. Once Router D registers its TRA, it starts sending AD packets to all its direct neighbors and address assignment continues to the edge routers.

b) Multi-Addressing: If a router has multiple parents, like Router G in Fig. 4, it can get multiple addresses. A router with multiple addresses may decide to use one address as its primary address to allocate addresses to its children routers. This implementation was adopted in the work presented in this article.

2) Routing Tables: TRP maintains three routing tables based on the type of link it shares with its neighbors. In a tiered structure, links between routers are categorized into three different types: up-link that connects to an upper tier router; down-link that connects to a lower tier router; and trunk-link that connects to routers in the same tier level. A router can identify the type of link from which the AD packet arrives by comparing its tier value with the tier value in the received packet.

```
1: if( R.TV == P.TV) then
2:   if( R.TA.the_last_digit == P.TA.the_first_digit) then
3:     if( port_num = find ( P.TA.the_second_digit, down-link table ) ) then
4:       remove( P.TA.the_first_digit);
         P.TV++;
         forward( P, port_num );
         return();
5:     end if
6:   else if( R.TV == 1 ) then  //at Tier1
7:     if( port_num = find ( P.TA.the_first_digit, up-link table ) ) then
8:       forward( P, port_num ),
         return();
9:     end if
10:  end if
11: else if( R.TV– P.TV == 1 && R.TA.the_parent_digit == P.TA.the_first_digit) then
12:   if( port_num = find ( P.TA.the_second_digit, trunk-link table ) ) then
13:     remove( P.TA.the_first_digit);
        P.TV++;
        forward( P, port_num );
        return();
14:   end if
15: else if( R.TV < P.TV) then
16:   discard( P); //wrong packet
      return();
17: end if
18: if( port_num = find( up-link table ) ) then
19:   forward( P, port_num );
      return();
20: end if
21: discard( P);  //no entry in routing tables
    return();
```

Algorithm 1. Packet forwarding at router *R* and incoming packet *P*.

Router F has three different types of links to Routers B, G, and L on port numbers 1, 2, and 3 respectively. Advertisement from Router B is received at port 1 and compared with the tier level of Router B (which is 1) and its own tier level (which is 2). Since tier level of Router B is less than tier level of Router F, the link connected on port number 1 is recognized as up-link and the information is stored in the up-link table. Likewise, information about Router G is stored in the trunk-link table, and information about Router L is stored in the down-link table.

In Table I, the '*port*' column shows the port number of the router and '*dest*' column shows the TRA of direct neighbor obtained from the advertisements. There are multiple entries against a single port in the trunk-link table of Router F because Router G has two TRAs. The routing table for Router G is also shown.

The TRA carries the shortest path information inherently. Hence, initial convergence time in TRP is significantly lower than OSPF because, with one advertisement packet from each direct neighbor, the routing tables converge. This also results in less number of control packets and traffic.

In the network in Fig. 4, three tier levels have been identified, and the TRA for the routers in this network are noted beside them. The TRA is made up of *TV*. *TA*, where *TV* is the tier value to identify the tier level and Tree Addresses (TA) is the address of the router. A '**.**' notation in the tiered address separates a TV and the TA. Thus, the TRA starts with a TV followed by '**:**' separated addresses, which form the TA. Thus, TRA 3.1:1:1 has TV=3 and TA= 1:1:1.

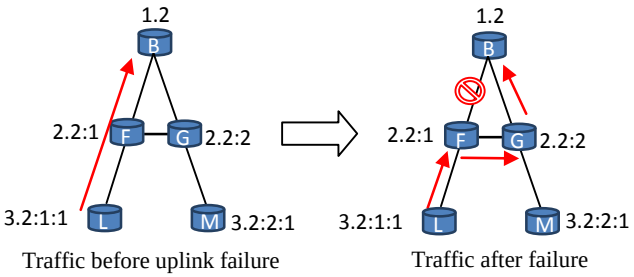


Figure 6. Failure handling with uplink

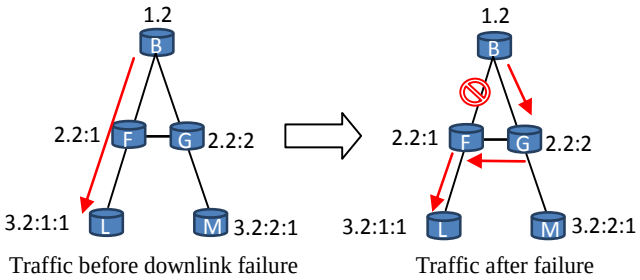


Figure 7. Failure handling with downlink

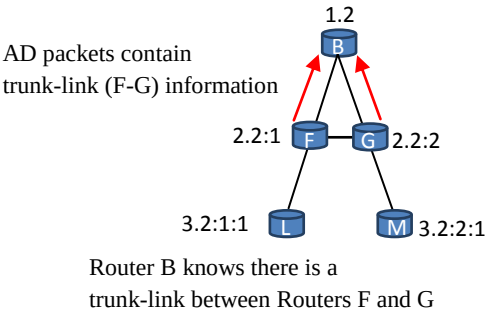


Figure 8. Trunk-link information sharing by the parent router

3) Packet Forwarding in TRP: Packet forwarding in routers running TRP is done as follows. The source router compares the source and destination TRAs to determine the TV of a common parent (grandparent) router between them. Assume source is Router L and destination is Router M in Fig. 4. Source Router L compares TA in its TRA namely ***2***:1:1 with the TA of the destination router's TRA namely ***2***:2:1 from left to right to find the common digit in these addresses. In this case, it happens to be the **1st** digit *2* (shown bold italic character) in the *first place*. This provides the information that a common parent (grandparent) between the two routers resides at *tier 1*. The TV in the forwarding address is thus set to *1*. To this TV is then appended the TA of the destination router to provide the forwarding address 1.2:2:1. Another example, for a forwarding address between source Router J ***1:1***:1 and the destination Router K ***1:1***:2 will be ***2***.1:2 because a common parent is identified at *tier 2*. The pseudo code for the forwarding decisions at a TRP router is provided in Algorithm 1 and it is self-explanatory.

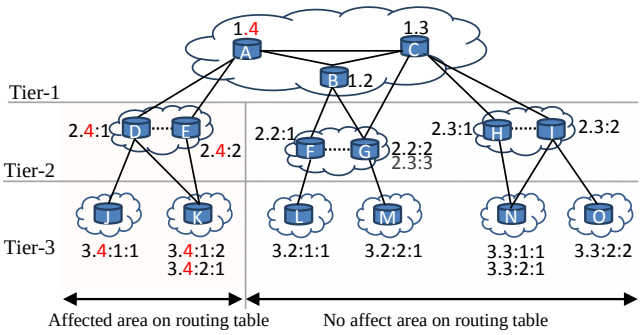


Figure 9. Address changes in TRP

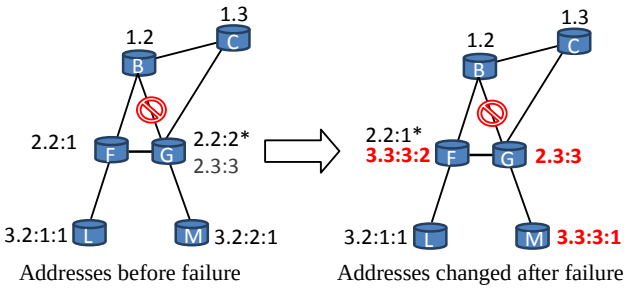


Figure 10. Primary address change

D. Failure Detection and Handling

Failure detection in TRP is *hello* packet based, i.e., typical of layer 3 notification proposed for use with current routing protocols. In TRP, 4 missing AD packets is recognized as link/node failure. A TRP router tracks all neighbors AD packets times and if ADs from a neighbor is missing 4 consecutive times, the TRP router updates its routing table accordingly.

However, in TRP packet forwarding on link/node failure a router does not have to wait for the 4 missing AD packets. An alternative path, if it exists, can be used immediately on missing a single AD packet irrespective of the routing table update. With the current high speed and reliable technologies, it is highly improbable to miss AD packets and redirecting packets on missing one AD packet is justified. However, for a fair comparison with OSPF we adopted the 4 missing hello packets to indicate a link/node failure.

1) Uplink failure: If a node detects an uplink failure and has a trunk link, it can use the trunk link, because trunk link exists between routers that have the same parent route, or it can use an uplink if one exists. In Fig. 6, the sibling router connected to Router F derives its address from the same parent. So, Router F knows that the uplink router on Router G will be its parent Router B.

2) Down link failure: Let a link failure occur between Routers B and F in Fig. 7. To detour around the link failure, down link traffic between Router B and F needs to take a path Router B-G-F. To achieve this, Router B needs to know if there exists a trunk link between Router F and G. A parent router must know all trunk links between its children routers. The trunk link information can be set in AD packets to help a parent router maintain all trunk link information as described in Fig. 8. Due to inheritances, routers can assume responsibilities to forward to their directly connected neighbors as the TRAs carry relationship information.

3) Address Changes: Address changes can happen because of node failure, topology change, or administrative decisions. In TRP, address changes affect limited area and incur very low latency as no updates have to be propagated.

For example, if Router A changed its TRA from 1.1 to 1.4 in Fig. 9, all neighbor Routers B, C, D, and E notice the change from the AD packet sent by Router A. Router D and E will change their TRAs without notifying Router A. Therefore, children of Router A can change their addresses rapidly. The same procedure continues to Routers J and K by the next AD packet from Routers D and E. The pruning operation is triggered on change detection.

4) Primary Address Change: If a node has multiple addresses and a link to a primary address failed, the node changes one of its secondary address to primary address and advertises the same. The child of the node also changes its address in the same manner as described in the case above and keeps the last digit. For example, Router G has two addresses and let 2.2:2 be the primary address in Fig. 10. When a failure occurs between Routers B and G, Router G changes its primary address to 2.3:3 and then advertises it. As a result, Router M changes its address to 3.3:3:1.

IV. TRANSITION WITH MPLS

One major contribution of our work was the study of MPLS as a transition platform to introduce TRP and replace IP and its routing protocols. MPLS achieves similar goals in terms of replacing IP and the routing protocols, but uses the routes from IP routing tables to determine the MPLS paths. Once the paths are established MPLS bypasses the use of IP in the MPLS aware routers. Another feature of MPLS that aided the transition studies was the use of label and label stacking, where in the proposed transition the labels serve to carry the TRP addresses, and label stacking was used to achieve the tiered functionalities, i.e., forwarding across tiers. The packet forwarding decision is the same as Algorithm 1. In this section, the implementation details are presented.

In Fig 11, there are eight MPLS aware routers, Routers A to H. Of these Routers A and F are Label Edge Routers (LER) and the others are Label Switch Routers (LSR). TRAs were assigned to all MPLS aware routers as shown in the figure. Based on the TRAs, it can be noted that Router C is a tier 1 router, Routers B, E, and D are tier 2 routers, while Routers A, G, H and F are tier 3 routers. To conduct the feasibility study, the MPLS tables were manually populated as shown in Figs. 12 and 13. For real implementations using MPLS, the operation of MPLS and its process of populating the tables have to be modified and are not included in this article.

We first explain the use of the tables. The first table in Fig. 12 (a) is for Router A, which is a LER. This router is connected to the IP network 192.168.1.0/24. However, in order to forward a packet to the destination network 10.100.1.0/24, the forwarding table has a dedicated entry. Interpreting this table; when a packet arrives with 10.100.1.0/24 as the destination address, LER A will *push* two labels *1* and *131* where *1* is the outer label (L-1). This

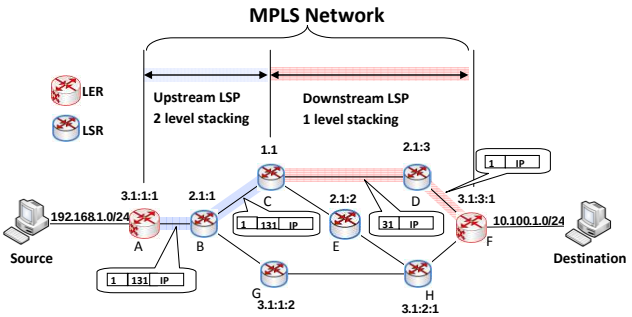


Figure 11. MPLS enabled network with TRP

Destination Network	*Out Label*	*Action*	*Next Hop*
10.100.1.0/24	1(L1) 131(L2)	PUSHx2	Router B

L-1 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
1	IP header	POP	IP address

a) Router A: 3.1:1:1 (LER)

Destination Network	*Out Label*	*Action*	*Next Hop*
192.168.1.0/24	1(L1) 111(L2)	PUSHx2	Router D

L-1 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
1	IP header	POP	IP address

b) Router F: 3.1:3:1 (LER)

Figure 12. LER MPLS tables of Routers A and F

packet will then be sent on to the next hop, which is Router B. If a packet arrives to be delivered to network 10.100.1.0/24 at LER A, Router A will *pop* the L-1 label and then forward the packet to the destination IP address in the packet. Similar table entries can be noted for LER F in Fig. 12 (b), which will also perform operations similar to Router A.

At LSR B, it will check the outer label when a packet arrives from Router A and processes the packet forwarding based on the outer label (L-1, tier 1) table. As per this table, when the packet arrives from Router A, if it has a forwarding address where the tier value is *1* (L-1), then the packet will be sent uplink to Router C with a *swapped* label, which will also have a value *1*. If the outer label (L-1) was *2*, it indicates that the anchor tier level is *2* in the forwarding TRA, and Router B is the anchor router (at which time redirection will take place). Hence, Router B will *pop* the L-1 label and the packet will then be processed as per L-2 label table. In the L-2 label, when a packet is received, Router B will *swap* the incoming labels with new labels to deliver the packet to either Routers A or G. Similar entries can be noticed for Routers C and D and their operations will be similar to that explained for Router B and tables are shown in Fig. 13.

Handling tier based forwarding with MPLS can be summarized as:

- For upstream forwarding, a L-1 label indicates that a MPLS packet is to be forwarded until the upper tier level specified in the label is reached. If L-1 label value is lower than router's tier value, it is forwarded to an upper tier.
- For downstream forwarding, if L-1 label value is the same as router's tier value, the router removes (*pop*) L-1 label and forwards the packet to a lower tier based on L-2 label.

L-1 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
1	1	SWAP	Router C
2	N/A	POP	N/A
11	1	SWAP	Router A
12	2	SWAP	Router G
L-2 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
11	1	SWAP	Router A
12	2	SWAP	Router G

a) Router B: 2.1:1 (LSR)

L-1 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
1	N/A	POP	N/A
L-2 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
111	11	SWAP	Router B
131	31	SWAP	Router D

b) Router C: 1.1 (LSR)
Note: Router C may have more entries

L-1 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
1	1	SWAP	Router C
2	N/A	POP	N/A
31	1	SWAP	Router F
L-2 Label Table			
In Label	*Out Label*	*Action*	*Next Hop*
31	1	SWAP	Router F

c) Router D: 2.1:3 (LSR)

Figure 13. LSR MPLS tables of Routers B, C, and D

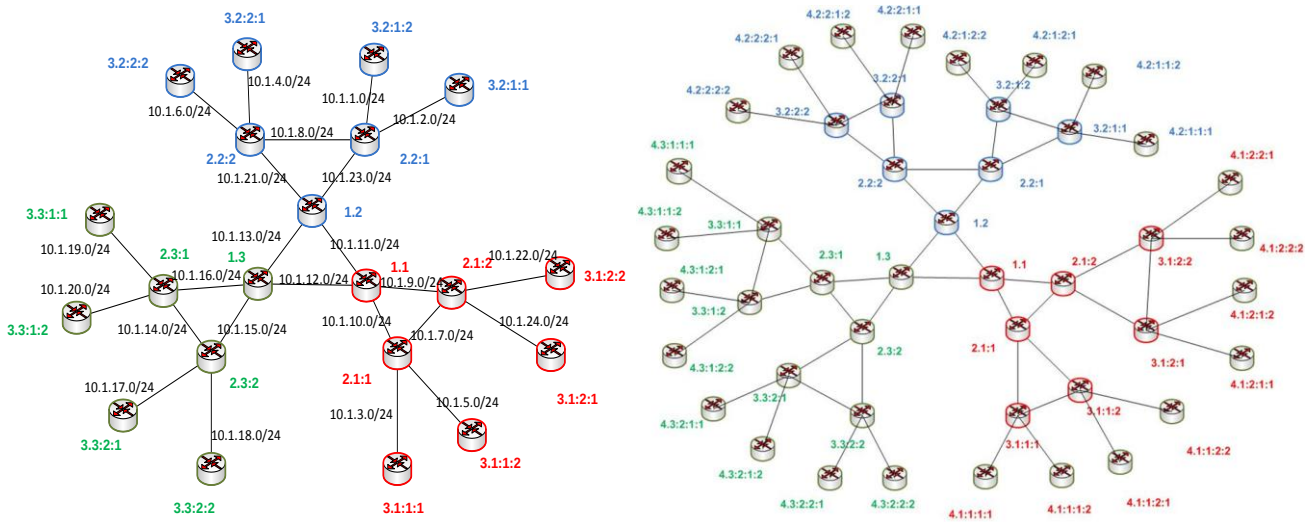
We now work through an example of packet forwarding in the network scenario shown in Fig. 11. Let the source node send a packet to a destination node with destination IP address 10.100.1.x, where x is the host identifier. LER has to be aware of the TRA allocated to network with IP address 10.100.1.0/24. This TRA is 3.1:3:1. Following are the steps.

1) Forwarding TRA calculation: Router A calculates the forwarding TRA to 3.1:3:1 by comparing with own TRA (3.1:1:1) with destination TRA 3.1:3:1. The forwarding TRA will be 1.1:3:1.

2) AddingMPLS header: Router A add two MPLS label to the packet using two *push* operations, where the L-1 label is *1*, L-2 label is *131*. The packet is then forwarded to the next hop Router B.

3) 1st hop: Router B checks the outer label, i.e., L-1 label value of *1*. This is less that Router B's tier value *2*. Thus, the packet will be forwarded to an upper tier based on L-1 label table. In this case, the label will be *swapped* to *1* and then the packet will be forwarded to next hop Router C.

4) 2nd hop: Router C checks L-1 label value of 1. This equals Router C's tier value of 1. Router C will remove the L-1 label through a pop operation and then the packet



a) 21 nodes b) 45 nodes

Figure 14. Testbed Topology with Tiered Addresses

should now be redirected. Router C will hence check the L-2 label, which is *131*, in the packet and compares it with its L-2 label table entry. Then, Router C forwards the packet to the next hop Router D after *swapping* the label from *131* to *31*.

5) 3rd hop: Router D checks L-1 label value *31* and lookups its L-1 label table. It will *swap 31* to *1* and then forward to the next hop Router F.

6) Removing MPLS header: Router F checks the L-1 label value of *1* and lookup its L-1 label table. It will then *pop* (removes the MPLS header from the packet) and checks the IP header to forward to the final destination.

V. EMULATIONS

A. Emulab Test Setup

TRP routers were implemented on Linux machines in Emulab. Emulab is an experimentation facility, which allows setting up networks with different topologies to provide a fully controllable and repeatable experimental environment. Emulab uses different types of equipment for this purpose. Two different types of machines were used during the course of this experiment, as allocated by the Emulab team.

Quagga 0.99.17 [12], a software routing suite for configuring OSPF was used for the comparison studies. IPerf [11] was used to generate data traffic.

A 21-nodes topology is shown in Fig. 14 (a). The configuration details are provided in Table II. In the 45-node topology, the additional 24 nodes were added to the outer circle of the 21 nodes' topology and displayed in Fig. 14 (b). The IP addresses were allocated from address space 10.1.x.x/24 to the segments as shown for OSPF. The TRAs for TRP were allocated using the scheme described in Section III-B.

B. Assumptions

1) More complex or meshed topologies could not be created due to the limitations on the number of interfaces on the Emulab machines. The number of physical network interfaces of Emulab PCs is limited to five, where one interface is used for control. Therefore, only four network interfaces are usable for setting up the test topologies. TRA address allocation mechanism will create logical tree-like topology on a physical meshed topology. Thus, we select tree-like topologies to utilize all links on the emulation because of the limited number of interfaces.

TABLE II. EMULAB TESTBED CONFIGURATIONS

Topology	21 Nodes	45 Nodes
Type of processor	Pentium III	Quad Core Xeon Processor
Number of links	24	54
Connection speed	100 Mbps	100 Mbps

2) TRP code operates on Linux user space and hence the timings and dependent variables such as packet loss during convergence showed a higher value than if the code were run in kernel space. Comparatively the Quagga OSPF code runs in kernel space. However, we present the parameters as collected without any corrections for the higher projected values noted for TRP.

3) To provide a random environment for the tests, they were conducted in two different sets of networks and the experiments repeated five times in each case. For a given 21-node topology or 45-node topology the machines were maintained the same throughout the emulation runs.

4) To emulate link failures, Emulab uses link shaping nodes that can be placed on the segments. We adopted this approach to fail links between Node 1.3 and Node 2.3:2 for both the 21-node and 45-node scenarios.

5) For OSPF evaluations, only one area was defined, as the intention is to demonstrate the performance impacts to increase the number of routers in a network or an area.

C. Tiered Routing Protocol Code

TRP runs above layer 2, *bypassing all layers* between layer 2 and the application layer. It replaces both IP and its routing protocols. To run applications on TRP, a modified

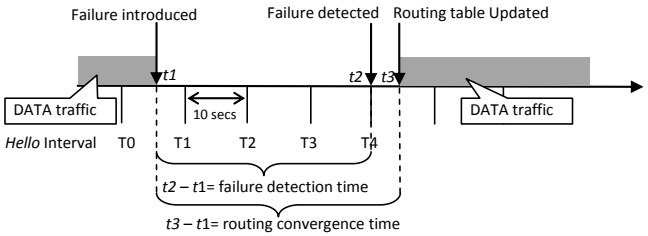


Figure 15. TRP Routing Convergence Time

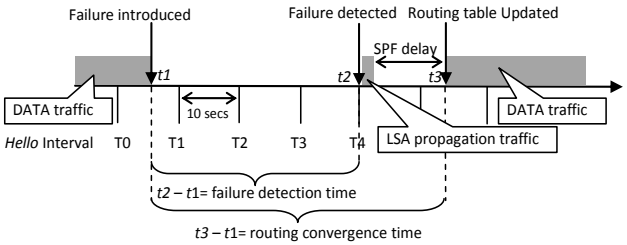


Figure 16. OSPF Routing Convergence Time

clone of IPerf called SIPerf, which allows bandwidth and link quality measurement in terms of packet loss, was used.

D. Initial Convergence Performance Statistics

1) Convergence Times: In OSPF, initial convergence takes place after the FIB update is run on all routers. To improve the veracity of collected data, the timestamps when SPF was run as well as the time when the routing table was updated was logged. For TRP, the timestamp for a new entry in the routing tables is logged and if the routing table at the routers remains unchanged for the next three *hello* intervals then the network was deemed to have converged.

2) Routing Table Size: In OSPF, this value was logged using the built-in commands provided by Quagga. In TRP, this information was logged in a file and sent to the server.

3) Control Overhead: To collect control overhead, Tshark [13], which is similar to Wireshark [13] was utilized to capture packets from which the control packets were accounted for. Tshark is a command-line tool and it was invoked through special scripts during the emulation. Bytes in the packets exchanged during convergence were summed to determine the control overhead at each node and then sent to the server. In TRP, a utility to record the number of control packets exchanged during initial convergence time was built in.

E. Link Failures Performance Statistics

Convergence time after link failure has two components.

1) Link failure detection time: This is the same for OSPF and TRP as they detect a link failure on missing 4 *hello* messages. With a *hello* interval of 10 seconds, this was recorded to be 30 seconds with an additive time - time between the first missing hello packet and the time when the link was actually brought down.

2) Time to update routing tables: This time is different for OSPF and TRP and the differences are explained using Figs. 15 and 16.

3) TRP Response to Link Failures: In Fig. 15, the time t_1 when the link failed is noted along with time t_3 ,which is the time it took to remove the link from the routing table.

Total time for convergence T_c is given by

$$T_c = T_{ru} - T_{fd} \quad (1)$$

where T_{fd} is the failure detection time given by

$$T_{fd} = t_2 - t_1 \quad (2)$$

and T_{ru} is the routing table update time given by

$$T_{ru} = t_3 - t_2 \quad (3)$$

Thus,

$$T_c = t_3 - t_1 \quad (4)$$

T_{fd} will be the same for OSPF, but T_{ru} is negligible in the case of TRP as this is the time for the TRP code to access the routing tables and update its contents. In Figs. 15 and 16, these times are identified based on the operations of TRP and OSPF, respectively.

4) OSPF Response to Link Failure: OSPF uses several timers on link failures, to rerun SPF algorithm and a few other hold times to avoid toggling. They are *Hold_Time*, which is the separation time in milliseconds between consecutive SPF calculations. An *Initial_hold_time* and *Max_hold_time* is also specified. SPF starts with the *Initial_hold_time*. If a new event occurs within the *hold_time* of any previous SPF calculation then the new SPF calculation is increased by *initial_hold_time* up to a maximum of *max_hold_time*.

Let T_{LSA} be the LSA propagation delay, T_{SPF} be the time to run SPF on subsequent LSA messages and T_{TU} be the table update delay, then T_{ru} of OSPF is given by

$$T_{ru} = T_{LSA} + T_{SPF} + T_{TU} \quad (5)$$

T_{SPF}, *initial_hold_time* and *max_hold_time* were set to 200 ms, 400 ms, and 5000 ms respectively for the test. Fig. 16 captures the relationship between the delays for OSPF.

VI. Performance Analysis

The performance of OSPF and TRP, during the initial convergence phase and their response to subsequent link failures are presented in this section. In the histograms, data collected for the two test sites are provided separately, to show the closeness of the two data sets under different environments to reflect the reliability of the experiments.

1) Initial Convergence Times

Fig. 17 records the average initial convergence times in seconds collected from the two test sites and for the two different topologies, one with the 45-router and the other with 21-router. While the convergence times recorded for OSPF range from 55 seconds in the case of the 21-router network to over 60 seconds in the case of the 45-router

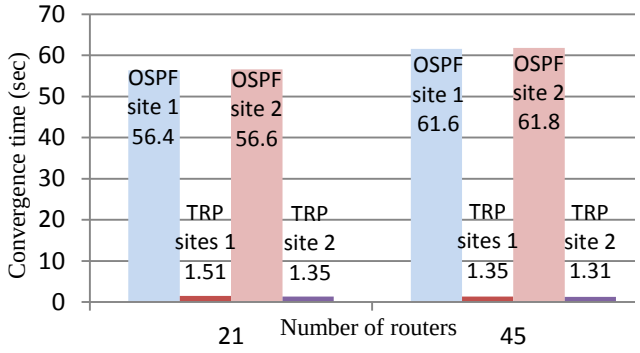


Figure 17. TRP vs. OSPF Initial Convergence Time (sec)

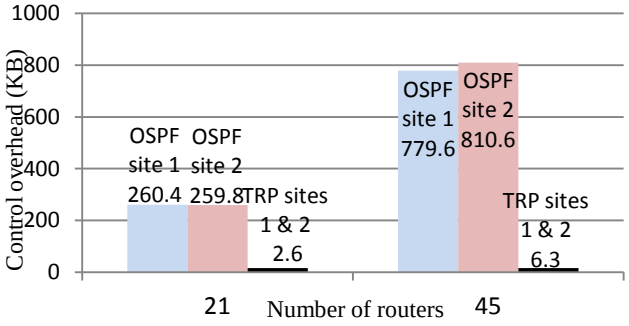


Figure 18. TRP vs. OSPF Routing Control Overhead Size (KB)

network, the convergence times for the network running TRP was around 1 second. While convergence times are stable irrespective of the number of routers running TRP, in the case of OSPF, the convergence times showed an increase by 5 to 6 seconds, indicating dependency of convergence times to the network size. TRP thus has 50-60 times improvement compared to OSPF.

2) Control Overhead During Initial Convergence

Fig. 18 shows the plot of control overhead in Kbytes for OSPF and TRP. Control overhead in the case of OSPF varies from 250 Kbytes for the 21-router network to around 750 to 800 Kbytes for the 45-router network. Increase in overhead almost triples as network size doubles. Control overhead for TRP was 2.6 Kbytes for the 21-router network and around 6 Kbytes for the 45-router network. The improvement achieved with TRP is 100 times in the case of the 21-router network and 120 times in the case of the 45-router network.

3) Routing Table Sizes

In Fig. 19, the routing table sizes collected were the same in the case of OSPF and TRP for the two test sites and hence one graph with the maximum routing table entries is provided. In the case of OSPF, this value is 25 for the 21-router network (as there are 25 segments) and in the case of the 45-router network this value was 55. In the case of TRP, the routing table entries reflects the number of directly connected neighbors, so in both cases, the maximum routing table entry was 4, there is no dependency on the network size.

4) Convergence Time After Link Failure

Fig. 20 displays the routing table update time in seconds subsequent to link failure detection. While OSPF shows an update time of 1.5 to 2 seconds for the 45-router network and

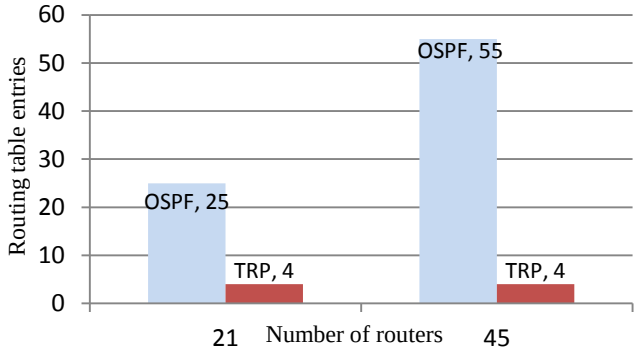


Figure 19. TRP vs. OSPF Routing Table Entry Size

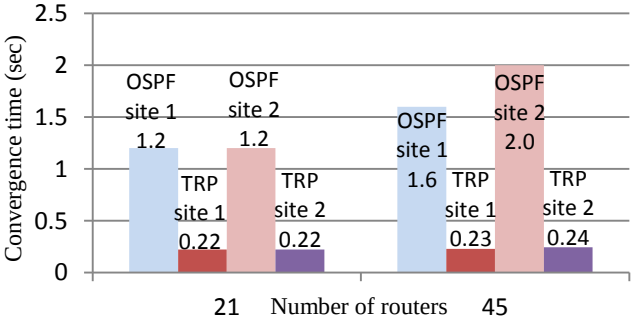


Figure 20. TRP vs. OSPF Convergence Time after Failure (sec)

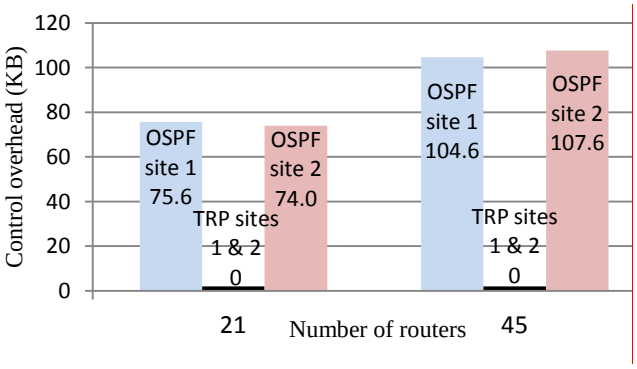


Figure 21. TRP vs. OSPF Control Packet Size after Failure (KB)

around one second for the 21-router network, TRP update times were 200 to 240 milliseconds; a magnitude of 6 improvement for the smaller network and a magnitude of 8 improvement for the larger network. Routing table update time is invariant to the network size in the case of TRP.

5) Control Overhead After Link Failure

Control overhead for TRP and OSPF collected during the convergence times, includes the time to detect a failure and also time to update routing tables. For the given topologies no control overhead was incurred with TRP. In Fig. 21, OSPF required around 100 Kbytes and 70 Kbytes of control packets for the 45-router and 21-router networks respectively. For complex topologies, in TRP change in topology information may have to be propagated to downstream networks. Similarly, upstream router may also have to be informed when a downstream link fails. These features were not tested in the scenarios.

6) Data Packets lost

The packets lost during failure detection will be the same for both protocols as the failure detection time is 4 missing *hello* packets. The time to update routing tables was recorded to be around 0.2 seconds for TRP and 1.2 to 2.0 seconds for OSPF. Thus, the packets lost during routing table update time was a maximum of 1 packet for TRP and a maximum of 10 packets with OSPF at a data rate of 5 packets per second.

From the results presented so far, it would be clear that TRP would be an ideal routing protocol to address scalability concerns as networks grow in number and in size. This is true as the routing table sizes and routing table update time is independent of network size. This in turn will positively impact the routing performance in the network. The convergence times are also very low and changes in network topology do not require network or area-wide dissemination of the changes. This will reduce instability in routing packets and also reduce packet loss.

VII. Conclusions and Future Work

A Tiered Routing protocol was developed under a new tiered Internet architecture. The tiered addresses in this architecture are used by TRP for packet forwarding. In this article, TRP is evaluated as an IGP using Emulab test facility. Initial convergence time and control overhead with networks running TRP is very low as the protocol does not require message flooding or any calculations subsequent to a link status change. Due to the inherent routing information in the tiered addresses, the routing table sizes in TRP are significantly low. Stability in the routing entries and their invariance to network size also indicates the strengths of such new approaches. Comparison with OSPF validates this.

There are several possible directions for future work. OSPF supports area concept for large network, so apply the area concept for larger network to compare with TRP. Validating TRP for inter-domain routing is another direction. Since tier levels in Autonomous System (AS) level topology can also be identified, based on their business relationships such as provider-customer and peer-peer relationship, TRP can be applied for inter-domain routing. Thus, Border Gateway Protocol (BGP) and TRP can be compared to validate TRP as inter-domain routing protocol.

Acknowledgment

This work was sponsored by NSF under grant number 0832008.

References

[1] Y. Nozaki, P. Bakshi, and N. Shenoy, "Tiered interior gateway routing protocol," ICNS 2013, The Ninth International Conference on Networking and Services, pp. 68-75, 2013.

[2] J. Moy, "RFC 1245 - OSPF protocol analysis," RFC Editor, 1991.

[3] M. Yannuzzi, X. Masip-Bruin, and O. Bonaventure, "Open issues in interdomain routing: a survey," Network, IEEE, vol. 19, no. 6, pp. 49- 56, 2005.

[4] Y. Nozaki, H. Tuncer, and N. Shenoy, "A tiered addressing scheme based on floating cloud internetworking model," Distributed Computing and Networking, Lecture Notes in Computer Science, vol. 6522, pp. 382-393, 2011.

[5] "Emulab: network emulation testbed," http:// www.emulab.net. (accessed December 2013)

[6] C. Alaettinoglu, V. Jacobson, and H. Yu, "Towards milli-second IGP convergence," Internet Draft, IETF, 2000.

[7] P. Pan, G. Swallow, and A. Atlas, "RFC 4090 - Fast reroute extensions to RSVP-TE for LSP tunnels," May 2005.

[8] A. Kvalbein, A.F. Hansen, T. Ciˇci´c, S. Gjessing, and O. Lysne, "Multiple routing configurations for fast IP network recovery," IEEE/ACM Transactions on Networking, vol. 17, no. 2, pp. 473-486, 2009.

[9] Y. Liu and A.L.N. Reddy, "A fast rerouting scheme for OSPF/IS-IS networks," In Proceedings of ICCCN, pp. 47- 52, 2004.

[10] N. Shenoy, M. Yuksel, A. Gupta, K. Kar, V. Perotti, and M. Karir, "RAIDER: Responsive architecture for inter-domain economics and routing," GLOBECOM Workshops (GC Wkshps), 2010 IEEE, pp. 321-326, 2010.

[11] "Iperf: the TCP/UDP bandwidth measurment Tool," http://www.iperf.sourceforge.net. (accessed December 2013)

[12] "Quagga software routing suit," http://www.quagga.net. (accessed December 2013)

[13] "Tshark and wireshark," http://www.wireshark.org. (accessed December 2013)

[14] P. Narvaez, "Routing reconfiguration in IP networks," Ph.D. dissertation, MIT, June 2000.

[15] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP topologies with Rocketfuel," IEEE/ACM Transactions on Networking, vol. 12, no. 1, pp. 2-16, 2004.

[16] "Cytoscape," http://www.cytoscape.org. (accessed December 2013)

An Architecture for Wireless Sensor Actor Networks for Industry Control

Yoshihiro Nozaki, Nirmala Shenoy
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY, USA
yxn4279@rit.edu, nxsvks@rit.edu

Qian Li
CAST-Telecommunications Engineering Technology
Rochester Institute of Technology
Rochester, NY, USA
qxl2571@rit.edu

***Abstract*— A robust and reliable architecture for wireless sensor actor networks for industry control (WASANIC) is discussed and described in this paper. The stringent physical constraints in an industry environment are taken into consideration. We proposed an architecture that allows efficient cross-layering between a semi-scheduled medium access control (MAC) protocol called the Neighbor Turn Taking MAC (NTT-MAC) and a routing protocol based on the Multi-Meshed Tree (MMT) routing algorithm that is suited to the WSANIC topology encountered in an industry. The proposed architecture also addresses survivability and security. The cross-layered approach, named NTT-MMT, supports reliable and robust transportation of data. Through simulations, the performance of NTT-MMT was compared with carrier sense multiple access with collision avoidance (CSMA/CA) MAC and dynamic source routing (DSR) protocol.**

Keywords-Sensor Actor Networks; Industry Control; Robust and Reliable Architectures; Cross Layering; Medium Access Control

I. INTRODUCTION

This paper is the extended version of the conference paper [1], and aimed at providing a deep insight into the integration between a Medium Access Control (MAC) protocol, the Neighbor Turn Taking (NTT) [2][3], and routing protocol, the Multi-Meshed Tree (MMT) [4]. Towards this, we describe the physical constraints encountered in a wireless industry environment and propose a suitable topology and an architecture that would address survivability and security. We then highlight MAC functions essential to handle data, task, and event prioritization, which is vital for wireless industry control. Lastly, we identified a secure routing scheme that complements and integrates into the MAC, to provide the requisite connectivity robustness.

Wireless Sensor-Actuator Networks (WSAN) comprise of wireless sensors and actuators (or actors). Typically, sensors are low-processing, low-energy devices that sense data such as temperature, pressure and so on. The sensed data is gathered at a sink to be analyzed and acted upon. In some cases, sensors are low-cost disposable devices. Based on the sensed data, actuators make decisions and take action. Actuators normally have higher processing capacity and are not energy constrained. They may also perform the functions of a sink.

Significant hardware and software technology advances have resulted in major cost reductions in sensors and actuators. This coupled with elegant techniques to overcome challenges in wireless transmissions make WSANs attractive and viable for many applications. Examples are environment / habitat monitoring and control, battlefield surveillance, industry control and automation. In WSAN for environment and habitat monitoring and control, and battlefield surveillance, a large number of sensors are randomly deployed in potentially inaccessible areas, hence they are disposable and should be highly energy conserving. Multi-hop data collection paths, self-configuration and self-healing are predominant features of WSAN in such applications. Importance of security in such WSANs depends on the applications.

Considering a Wireless Sensor-Actuator Network for Industry Control (WSANIC), high survivability and ability to support data, event and task prioritization are predominant requirements. Security is very important because of the critical nature of the application. For example, explosives high power and chemical industries could have serious detrimental effects in terms of cost and / or human loss if tampered with. Due to the fact that sensors and actuators could be placed in least human-frequented areas makes them highly vulnerable to security attacks.

In contrast to the distinctive features mentioned earlier for WSANs, in a WSANIC, sensors and actuators are manually placed, resulting in a more stationary and deterministic topology. Self-configuration and self-healing are required upon device failures or environmental changes. Devices are rarely disposable and batteries can be charged or changed regularly. Thus, some issues that pose serious challenges in WSAN are less problematic in WSANIC [5]. Robustness, interference in communications and data reliability are of major concern in a WSANIC. To improve robustness, one has to look for options other than using powerful antennas as high power transmissions pose danger in inflammable spaces and increase interference effects [6]. In an industry environment, high electromagnetic fields due to heavy electrical devices and power cables are normal to expect, which negates the use of low power transmissions by sensor and actors. Communications interference is also caused due to events such as environment conditions, moving people and objects all of which can impact timely data transmission. Data reliability is critical as corrupted data could result in improper control of machinery and processes, which could be catastrophic. Furthermore, in a WSANIC, some data may have to be transported with least latency, i.e., high priority and without loss, as they may need an immediate action to be taken.

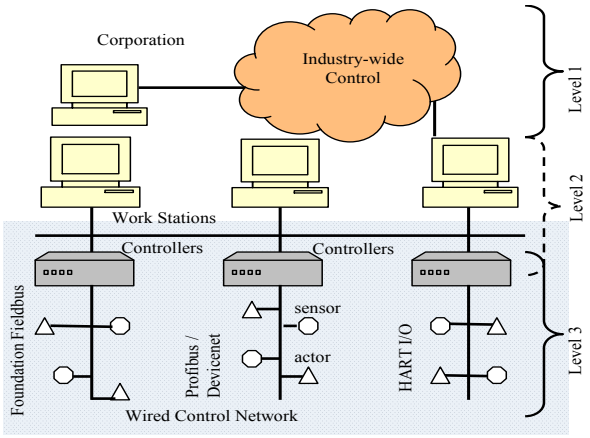


Figure 1. Wired Industry Control Network Architecture

To achieve close to real-time communications between sensors and actuators, a medium access control (MAC) protocol that allows for timely and reliable delivery is necessary. If multi-hop or multi path communications are required to introduce redundancy and robustness in the paths used to deliver data, then a reliable and robust routing protocol is equally important. The two protocols should operate with low complexity and interwork efficiently. We proposed an architecture presented in [1] that allows efficient cross-layering between a semi-scheduled MAC protocol called NTT-MAC and a routing protocol based on MMT algorithm that is suited to the WSANIC topology encountered in an industry. In this article, expanded cross-layering approach, multi-hop and multipath routing maintenance, and security concerns are discussed.

This paper has the following structure: Section II describes current industry control networks. Related works that are addressing WSANIC issues is provided in Section III. Section IV describes about WSANIC. Section V introduces our proposed architecture NTT-MAC and also discusses detailed cross-layering approach and Section VI analyses the result of simulations. Section VII provides the conclusions.

II. CONTROL NETWORKS IN INDUSTRY

Wired Control Networks (CN) have been adequately supporting industry control network requirements till date. However, in industries dealing with explosives, moving, or rotating machinery, some locations are inaccessible or highly inconvenient to monitor using wired sensor and actuator systems. The cabling and conduits for wired sensors and actuators besides being vulnerable to damage can be cost prohibitive - ranging typically to as much as one third to one half of the total system cost [7]. Industrial sensors meanwhile have seen a steady decrease in cost and the eventual driving cost factor in wired industry control networks is the cabling cost rather than the sensor or actuator cost. A low-cost wireless sensor-actuator system with reasonable battery life that provides reliable data collection spanning an entire industry plant, while meeting certain cost objectives would create a paradigm shift in industry control and automation [7]. Such systems would also allow the penetration of computing capabilities in locations that previously would have been cost-prohibitive [8]. In the section below, we discuss some of the most adopted wired industry control network topologies and standards.

A. Wired Control Network

A *Process Control System* in an industry uses sensors to measure the process parameters and actuators to adjust the operation of the process. Control action can be inbuilt into actuators or can be in separate entities called controllers. In industry control, it is convenient to have controllers separate from actuators as the controllers collect data from several sensors, make decision on an appropriate action to take (like proportional, integral, derivative or combinations of these) and actuate several actuators [5].

In Fig. 1, a typical wired industry-wide control network is shown. It has three levels of hierarchical control. The network at level 3 that connects the sensors and actuators to the controllers is of interest to us and we use the term wired CN for this segment. In this article, we propose an architecture and suitable protocols for a wireless CN (earlier termed the WSANIC) that can replace the wired CN and analyze the performance of such a WSANIC.

At level 3 in Fig. 1, Foundation Fieldbus (FF), Profibus and Devicenet are some of the wired CN industry standards being used [6]. The standards assume inherent high predictability and reliability as they operate over wired networks and hence the target of real-time data delivery should be achievable. Real-time and reliable data delivery is very important in industry control, since loss or untimely delivery of scheduled data could result in costly consequences [5]. Other performance affecting factors to consider are data rates, distance and transmission ranges. For example at the physical layer of FF, the official data rate is 31.25 Kbps. A process unit in a plant could span tens to hundreds of meters. Depending on the cable types and whether the controller is mounted close to the sensor / actuator or in a remote room, the distance range of FF is expected to be from 200 to 1900 meters [5]. As a promising alternative to industry control, a WSANIC should have capabilities similar to the wired CN and address the critical targets set by the wired CN standards.

III. RELATED WORK

The frequency spectrum used in current wireless networks can support high data rates. However, long transmission ranges are difficult to achieve as high power transmissions are undesirable in an industry typically those that handle explosives or highly inflammable material. In [8], Enwall T. provides statistics from studies conducted on suitability of major wireless network standards like 802.11g, 802.11s, Zigbee 802.15.4 and WiMax for industry control as per ISA-SP100 standards [9]. From the statistics it is clear that none of the above standards come close to doing what they need to do to fully support industry applications. However, combining Zigbee with a service broker [8] improved Zigbee's rating considerably, though it still fell short in several aspects such as network and messaging security, adequate reporting rates, quality of service in terms of timeliness, delivery ordering and recovery actions

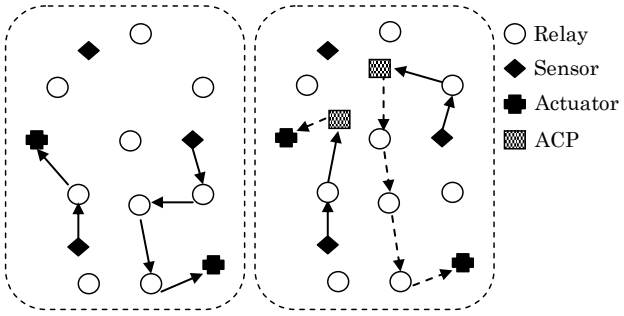


Figure 2. WSANIC architectures

among others.

A survey of related literature reveals that there are few contributions that address WSANIC issues [5] - [8]. The prime focus in these articles are on how best to replace the FF or other similar wired CN [5] with a wireless counterpart.

From an industry and standards perspective, several wireless organizations are investigating solutions and pursuing adoption of wireless standards promoted by them. Among these, Wireless Industrial Network Alliance (WINA), Zigbee, International Society of Automation (ISA) wireless system for automation, and Wireless Highway Addressable Remote Transducer (WirelessHART) protocol are some major ones [6]. However, none of these efforts takes into consideration industry environmental, placement and access restrictions.

In [10], the authors observe that "a WSAN should be robust to node failures and in general exhibit fast dynamic response to topology or connectivity changes". In [11], researchers at Massachusetts Institute of Technology harnessed the robustness inherent in mesh topologies in a WSANIC test bed. These observations indicate that topology and architectural issues are important to consider for WSANIC architecture. High survivability and security are equally important. The varied features are best addressed through suitable architectures and / or topologies.

IV. WIRELESS SENSOR ACTUATOR NETWORKS FOR INDUSTRY CONTROL

We start with three main devices that are essential in a WSANIC, namely sensors, actuators and controllers. We distinguish their functions in an industry control environment to aid in a suitable architecture design. Without loss of generality, it is assumed that sensors and actuators are distinct and separate devices. Sensors are end devices that collect and transmit data while actuators are end devices that receive data and actuate a lever or valve in an industry control process. The controller, which we henceforth call an Access Control Point (ACP) is the data collection device that collects data from several sensors and is the source point of control data to that controls the operations of several actuators. Inter-ACP communication required for industry wide control may be over wireless or wired links is not considered in this architecture. It is reasonable to assume that ACPs will be limited in number and positioned at specific locations. Hence, it may not be possible for all sensors and actuators to have a line of sight communications path to an ACP. For robustness in connectivity, it is further essential that sensors and actuators have routes to multiple ACPs.

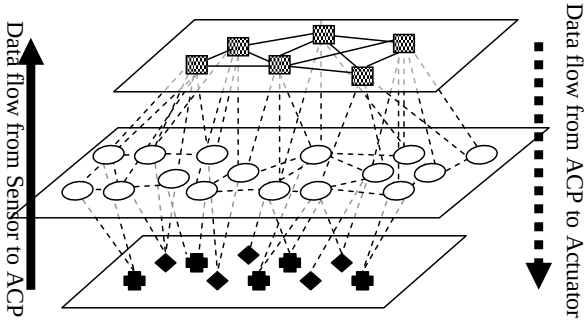


Figure 3. The Semi-automated Architecture

A. The Architecture

To overcome the physical constraints due to communications range, line of sight transmissions and to provision multiple paths between ACPs and the sensors / actuators special devices called 'relays' are introduced. Relays forward data for other devices and will also aid in setting up multiple paths of communications between ACPs and sensors / actors. It has been observed in [12] that multiple types of devices result in complex management due to diversity in techniques, data collection methods and protocols. In the proposed architecture, multiple types of devices are necessary to provide robustness and adaptability. However, complex communications and management are avoided by using a set of medium access and routing protocols that is common to all devices.

The architecture thus designed for WSANIC should include consideration of ACPs, sensors, actuators and the relay mesh. Fig. 2 shows such typical architectures and the topology linking the different devices that can be used for the purpose. As seen in Fig. 2 (a), there is no ACP in the automated architecture because actuators can process collected data and make decision automatically thereby replacing the need for special devices to perform the action. The data flow in this architecture will be a one-way communication from sensors to actuators. While the automated architecture has one-way communication, the semi-automated architecture has two-way communications, i.e., between sensors to ACPs and ACPs to actuators [13], as seen in Fig. 2 (b).

Fig. 3 expands the architecture in Fig 2b, by positioning the devices that also shows the linking and connectivity between the different devices. In this architecture, sensors send data to ACPs, and the collected data from several sensors is processed at the ACPs. Fig. 3 shows the logical view of the architecture. Thus, there are distinct 3 layers, comprising of a top layer, which is a mesh of ACPs, a middle layer, which is a mesh of relay nodes, and a bottom layer, which comprises of sensors and actuators. In an industry, the physical location of the devices may not be separated as seen in Fig. 3.

Between each layer and among the middle layer entities, wireless links are assumed to be used for communications. After the collected data is processed in ACPs, the ACPs will

make decision for proper actions to be initiated in the actuators and forward the commands to the actuators. Since ACPs can be powerful computers and wired to each other, they can process much effective and take collaborative decisions as compared the automated architecture. However, the semi-automated architecture requires route maintenance between sensors and ACPs, and ACPs and actuators. There will be more transmissions than the semi-automated architecture due to the two-way communication. Therefore, the semi-automated architecture needs improved MAC in terms of less collision and latency and robust routing protocols that leverage the multiple paths and multi hops in the architecture.

B. The Protocols

In a typical wired CN standard like the FF, the protocol stack is derived from the OSI 7 layer model, where only the lower two layers namely the physical and the data-link are specified; the network, transport and session layers are removed [4]. The proposed protocol stack for WSANIC also follows the two-layer approach. The lower layer is the physical layer, which is not the focus of this article, and the layer above, i.e., layer 2, has integrated medium access control and routing functions that operate off a single header. This is very attractive in wireless networks as it reduces header overhead, processing requirements and its associated delays, while allowing MAC and routing functions to interwork closely.

C. The Medium Access Control Functions

A MAC protocol for WSANIC should provide timely and near-lossless data delivery that is comparable to wired CN. In wired CN, it is naturally assumed that priority data carrying vital information under alarm conditions will be delivered reliably and in time. However, this assumption is not valid in wireless networks and sensitive, urgent data has to be handled specially to facilitate timely and reliable delivery.

Timely delivery can be achieved through preemptive priority. Preemption requires abortion / delay of other transmissions or receptions on the arrival of high priority data. This capability can be provisioned through the use of a dual channel MAC, one channel to carry high priority data and another channel for normal data. The MAC switches the local processing to handle high priority data on its arrival. However, this requires increased performance capabilities in the wireless nodes.

Reliability can be achieved through the use of acknowledgements and retransmissions on loss of such acknowledgements. However, this should be accomplished within acceptable latency limits. Reliability can be achieved in the routing functions through the use of concurrent multipath transmissions of critical data to increase the probability of its delivery.

A scheduled MAC is more suitable for reliable and timely delivery of data. However, as we advocated a multi-hop mesh topology a scheduled MAC is difficult to implement due to synchronizations issue. Moreover, in an industry environment, an unscheduled MAC will have more flexibility as it can provide combinations of periodic, event-based and query-based data collection and delivery. If an unscheduled MAC is used, then reliability of data delivery has to be achieved via acknowledgements and retransmissions. Given the frequency spectrum used in current wireless networks, the data rates achieved are very high compared to a wired CN data rates (like the FF) and retransmissions on loss of acknowledgements can be processed within acceptable latency limits. The routing scheme to be presented next also support timely and reliable data delivery, as it has the capability to send priority data concurrently on proactively maintained multiple paths.

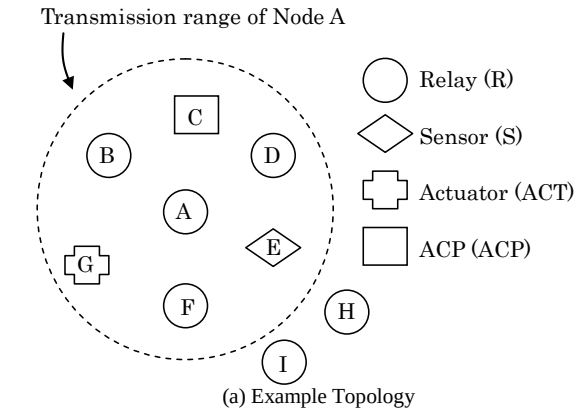
D. Routing Functions

ACPs, sensors and actuators in WSANIC can be stationary or mobile. The set of relays that forward data from sensors to actuators can vary due to mobility of ACPs (which is rare), sensors, and actuators; battery drain at relays or environmental changes which can impact the wireless link between a pair of devices. In this case, a single route is not advisable as data loss due to route failure has a high probability of occurrence. Multiple routes from sensors to ACPs and ACPs to actuators can alleviate this problem. Delays due to new route discovery also cannot be tolerated in such critical situations. Hence, a robust proactive multipath routing scheme with low overheads would be ideally suited. Routing based on the Multi Meshed Tree (MMT) algorithm [4] [14] has these desirable features.

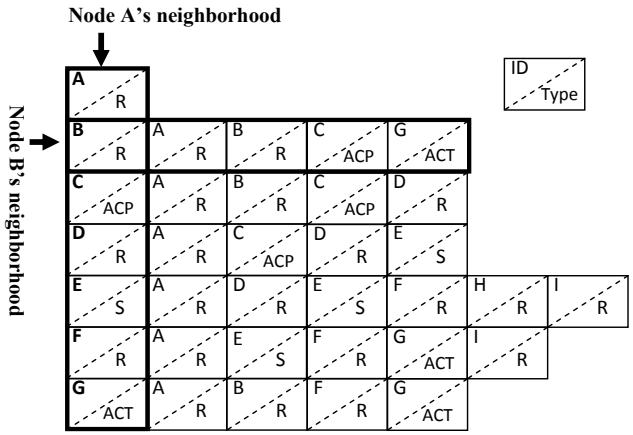
V. IMPLEMENTATION

We stated earlier that the MAC and routing functions would be integrated and operate off a single protocol header. Hence, in this section, we first describe the operational details of the Neighbor Turn Taking (NTT-MAC) and then the operation of the MMT routing protocol. This is followed by the details of integrating the two operations.

The NTT-MAC protocol uses carrier sensing similar to IEEE 802.11 CSMA/CA [15], but adopts a more deterministic medium access approach. In this new approach, nodes take turns to access the media, based on neighbor knowledge and hence is called the Neighbor Turn Taking MAC protocol [2]. This protocol has been previously shown via simulation to perform better than IEEE 802.11 CSMA/CA in terms of end-to-end packet latency and rate of successfully transmitted packets under saturated traffic conditions [3]. The MMT based routing sets up overlapping (meshed) trees originating at the ACPs and ending at the sensors and actuator. The meshed trees provide proactively established multiple robust routes. MMT algorithm also uses neighbor knowledge for its operation. Thus, the cross-layering approach adopted in the proposed architecture integrates the functions.



(a) Example Topology



(b) Neighbor table of node A

Figure 4. Neighbor knowledge example

A. *Neighbor Turn Taking Medium Access Control*

NTT-MAC uses a distributed loosely scheduled approach based on neighbor knowledge and their activities. NTT operation requires two processes, 'neighbor sensing' and 'turn scheduling'. Because there are four different types of nodes - sensors, relays, actuators, and ACPs, the NTT-MAC proposed in [2] has been customized to the new architecture with the four different types of devices. We now explain the different operations in NTT-MAC.

1) Neighbor Sensing: Each node overhears messages sent by its neighbor nodes to calculate its turn to access the medium next. To accomplish this, all nodes in the network advertise themselves and their 1-hop neighbors periodically. Thus, nodes know their neighbor's neighbor information, i.e., 2-hops neighbor information. In addition, node types such as sensor, relay, actuator, and/or ACP is also advertised. This advertisement is also used as a hello protocol [16] to detect any change in the 2-hop neighbors. Fig. 4 (b) shows an example of neighbor knowledge of the topology in Fig. 4(a). Nodes B, C, D, E, F, and G are neighbors of Node A. In Fig. 4 (b), the left most column in the table represents Node A's neighbor list and each row represents each neighbor's neighbor list including itself. For example, Node B's neighbors are nodes A, C, G and their node types are relay (R), ACP, and actuator (ACT). These neighbors' node type information is included in the advertisement.

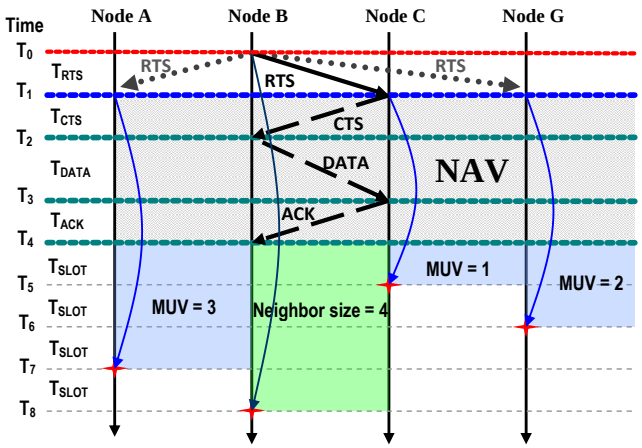


Figure 5. Example of Turn Calculation

2) Turn Scheduling: In NTT-MAC, a turn slot time (T_{SLOT}) is allocated to each node after the computations by the turn scheduling algorithm. Turn scheduling is achieved based on the neighbor table and their activities as described next.

a) Neighbor Activities: Each node calculates its next turn based on the sender node's neighbor list which it overhears from its neighbors transmissions. For example, if Node B in Fig. 4 (a) sends a packet, all neighbors nodes A, C, and G hear the transmission of Node B. They will then calculate their next turn by looking up Node B's neighbor list. The neighbor list indicates the order of each node's turn. Fig. 5 illustrates the turn calculation initiated by Node B's activity namely a data transmission by Node B. A ready-to-send (RTS), clear-to-send (CTS), DATA, and an acknowledgement (ACK) packet are used for data transmission. When Node B has data to send to Node C, Node B sends RTS to Node C. Since Nodes A and G are neighbors of Node B, they also hear the RTS packet at time T_1 in Fig. 5. Then, all neighbors of Node B calculate their next turn. Because there will be a sequence of packet transmissions between Node B and C, the total transmission time is the sum of time to send CTS (T_{CTS}), DATA (T_{DATA}), and ACK (T_{ACK}) transmissions. This is called a network allocation vector (NAV). In addition to the *NAV* time, each node calculates its next turn based on the position in the sender's neighbor list, named medium user value (*MUV*). According to Node B's neighbor list in Fig. 4 (b), the turn taking order is Node A→ B → C → G. When Node B is taking a turn, MUV of Node C, G, and A are 1, 2, and 3 respectively. The time T_{SLOT} is greater than or equal to the time to transmit RTS (T_{RTS}), to provide chance to send an RTS packet. Total wait time (T_{WAIT}) for each node at time T_1 in Fig. 5 can then be calculated as:

$$NAV = T_{CTS} + T_{DATA} + T_{ACK} \quad (1)$$

$$T_{WAIT} = NAV + (MUV \times T_{SLOT}) \quad (2)$$

Based on the type of packet received, the value of *NAV* will be updated. For example, *NAV* at time T_2 will be:

$$NAV = T_{DATA} + T_{ACK} \quad (3)$$

And at time T_3:

$$NAV = T_{ACK} \quad (4)$$

Therefore, the first sender after Node B will be Node C at T_5, and the second sender will be Node G at T_6 if Node C did not send any packets. If Node C sends a packet at time T_5, all neighbors recalculate their next turn based on the types of packet they overhear. In order to synchronize their turns, the order in each neighbor list has to be the same with all neighbors.

In WSANIC, data from a specific sensor and ACP may have higher priority than others. In this case, these nodes can get more chance to send data by adding a duplicate entry for themselves in their neighbor list and advertise it. Thus, they can take turns more frequently.

b) Node's activities: The turn calculation is based on a node's neighbor list size. For example, Node B calculates its next turn to be 4th because its neighbor list size is 3.

c) Updating: Each node has one next turn scheduled at any time. Thus, each node compares previous turn scheduling time and the new turn scheduling time after every turn calculation, and applies the latest scheduled time.

B. Multi Meshed Tree Routing

For routing, the Multi-Meshed Tree (MMT) algorithm is used to create logical meshed trees in the network. These trees are rooted at the ACPs. The ACTs and sensors are the leaf nodes. Since the semi-automated architecture has two-way data flow, sensor nodes need routes to ACPs and ACPs need routes to actuators. In addition, a sensor can communicate with any ACP and any ACP can communicate with any actuator. Hence, both sensors and ACPs are required to maintain routing information. As a result, route maintenance can become complicated and difficult. Most well-known routing protocols (proactive and reactive) in wireless ad hoc networks such as Dynamic Source Routing (DSR) [17] and Optimized Link State Routing (OLSR) [18] are required to maintain routing information at sender nodes. MMT requires only ACPs to maintain route information to ACTs. Sensors have the route information to ACPs, which is inherent in their allocated virtual IDs (VIDs). Inherently in MMT, leaf nodes in the trees such as sensors and actuators can know routes to the root nodes of the trees once they joined the trees as this information is available in the assigned VIDs to the leaf nodes. Likewise, the root nodes such as ACPs know routes for both sensors and actuators. Therefore, sensors do not require maintenance of routing information. Because the logical trees are meshed, MMT routing protocol provides not only overlapping coverage, but also route robustness while avoiding loops in the meshed topology. Loops are avoided due to the path-vector like property of the VIDs. An optimized version of the MMT algorithm presented in [4] is used to reduce control packets of MMT in the proposed architecture.

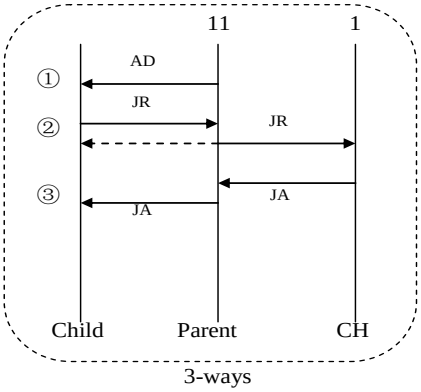
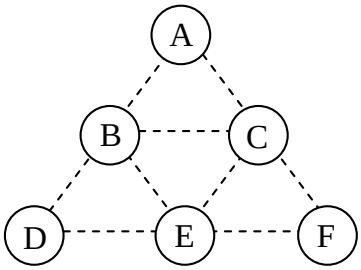


Figure 6. 3-ways handshake in MMT joining process



	A	B	C	D	E	F
VID 1	1	11	12	111	112	123
VID 2		121	113	1211	122	1132
VID 3		1221	1123	1121	1212	1122
VID 4				1222	1131	1223
VID 5					1111	
VID 6					1231	

Figure 7. Example of MMT (Hop limit = 3)

1) Multi-Meshed Trees (MMT)

As mentioned above, in the proposed architecture, trees are grown from root nodes (ACPs) to leaf nodes (i.e., sensors and actuators) through the relay nodes. Each meshed-tree can be viewed as a cluster and the ACP as the cluster head (CH) and all other nodes are the cluster clients because data flows in the semi-automated architecture are from sensors to ACPs and ACPs to ACTs. A 3-ways handshake is adopted by nodes to join the meshed tree.

Fig. 6 shows the 3-ways handshake used by a node during the joining process in MMT. The ACP (CH) node initiates tree creation by broadcasting an advertisement (AD) containing its VID. In general, a node on hearing an AD packet and wants to join the tree will send a join request (JR) to the sender of the AD packet who then becomes the parent node eventually to the joining node. The parent then records the new VID in a JR message and forwards to the CH, which register the new VID to its cluster member list. Because the child node can hear the forwarded JR message, the child can

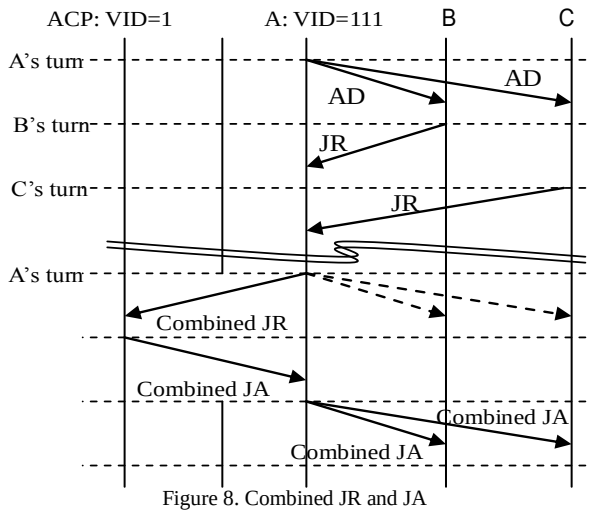


Figure 8. Combined JR and JA

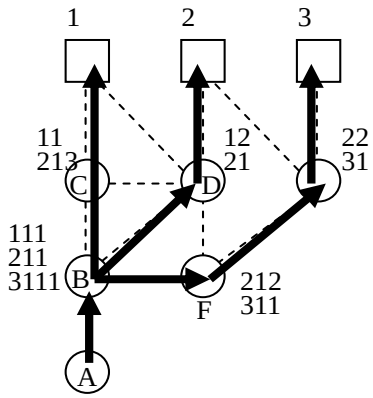


Figure 9. Joining process for multiple CHs

knows the new VID assigned to it at the time. The CH replies with a join acceptance (JA) packet to the parent after registering the new VID. Finally, the parent sends the JA to the child. And then, the child node starts to advertise its new VID to its neighbors. The new VID for a child node is one additional digit appended to the parent's VID.

Fig. 7 shows an example topology and the VIDs allocated using MMT to create the meshed trees. For example, if a CH node A has VID 1, the child VID can be between 11 and 19. So, Node B and C will get VID 11 and 12. Since Node C has 12, its child can be between 121 – 129. In this manner, the VID carries the route information. The total number of digits in a VID indicates the hop distance from CH, and also the route to the CH. The process continues until the tree encounters defined limits such as maximum hop count, cluster size or edge nodes are reached.

To avoid loops in trees, VIDs are not assigned if there is already a child-parent relationship with a particular VID. This VID acceptance rule applies for not only direct parent-child, but also for any grandparents and grandchildren nodes.

As part of the integrated operation of NTT-MAC and MMT routing, the knowledge acquired under the NTT-MAC is used in the MMT joining process by combining the JR and JA during the 3-ways handshake as shown in Fig. 8. Nodes B

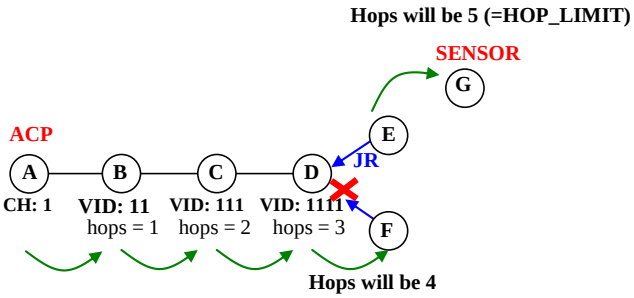


Figure 10. Neighbor Knowledge and Hop Limit

and C are neighbors of Node A, which has VID 111. After Node A broadcasts its VID, Node A calculates its next turn based on its neighbor table. Nodes B and C overhear Node A's AD packet and calculate their next turn based on Node A's neighbor table. As their turn scheduling is based on Node A's neighbor table, next turn scheduling time of Nodes B and C are the time before Node A's next turn. Hence, the JR from all Node A's neighbor can be received before Node A's next turn to transmit. Therefore, Node A can combine all JR messages from its neighbors ideally and assign new VIDs for all the children nodes when Node A gets its next turn and forward the combined JR to the CH. The CH node returns a combined JA to Node A after the new VIDs are registered.

The proposed joining process thus allows the request for multiple VIDs with a single 3-ways hand shake process not only for the same CH, but also for VIDs under different CHs. Fig. 9 shows the scheme for joining different CHs using a single 3-ways handshake. If Node A wants to join all VIDs of Node B namely 111, 211, and 3111, Node A sends a single JR, which contains the request to join all VIDs included. Then, Node B assigns new VIDs under all requested VIDs and broadcasts them in JR message. All neighbors of Node B overhear the JR and look into the requested VIDs. If the VIDs contain their direct child VID, they will forward the JR to their CH. For example, Node C will forward the JR because the JR contains request for VID 111, which is a direct child of Node C's VID 11. Likewise, Node D and Node F will forward the JR packet because VIDs 211 and 3111 are their child VIDs.

C. Interaction between Multi Meshed Tree and Neighbor Turn Taking algorithms

Since MMT uses neighbor knowledge for optimized cluster joining process, MMT interacts with NTT to look up neighbor table. This cross-layering approach is thus named as MMT-NTT. Each node maintains neighbor knowledge, which includes not only the node's VID but also the node type. MMT helps set up routes between sensor to ACP and ACP to actuator. Fig. 10 shows an example scenario. When Node E and F receive an AD packet that contains VID 1111 from Node D, they will make decision whether they should send a JR to request a new VID or not. The VID 1111 is 3 hops away from an ACP (Node A). If Node E and F joined this VID, they will be at 4 hops away from the ACP. If the HOP_LIMIT is set to 5, their neighbor nodes will be at the 5th hop (the last hop allowed under the configuration).

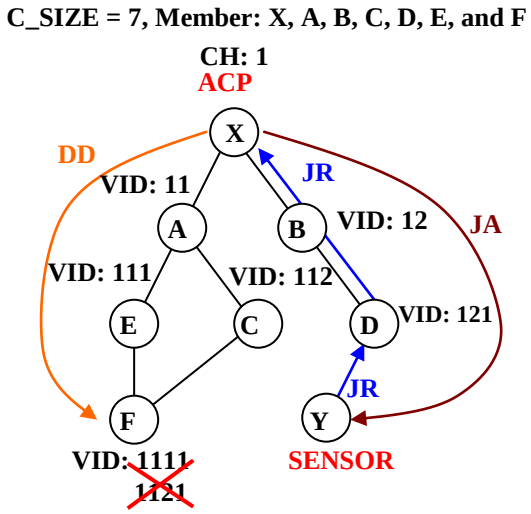


Figure11. Neighbor Knowledge and Cluster Size

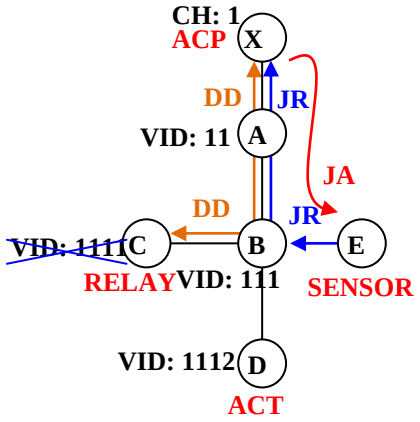
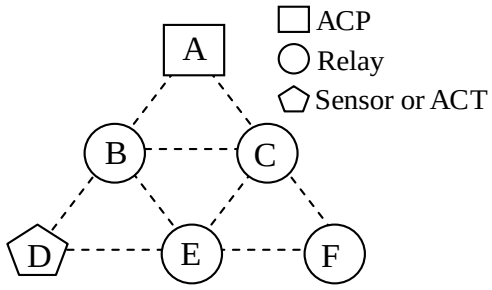


Figure 12. Neighbor Knowledge and Child Size

Therefore, if they do not have a sensor or ACT node in their neighbor list, the new VID will be meaningless and would use up the limited cluster size wastefully. Thus, a node that does not have any sensor or ACT in its neighbors' neighbor list, it will not send a JR if the joining VID is already at HOP_LIMIT−2. In Fig. 10, Node F will not send a JR to Node D because it does not have sensor or ACT in its neighbor list. On the other hand, Node E can join the VID 1111 because a sensor Node G is its neighbor and its newly acquired VID will be 1111x (4hops) and the new VID for Node G will be 1111xx (5hops).

In addition to the hop limit, ACP (CH) can also limit its cluster size (C_SIZE) based on the topology and number of total nodes in the network. Therefore, an ACP will prioritize inclusion of more sensors and ACTs within the C_SIZE limit. MMT-NTT considers priorities to achieve this. If the total number of nodes in the cluster has reached C_SIZE limit, the CH can replace low priority node such as relay nodes to accommodate high priority node such as sensor and ACT. In this case, a CH creates a Direct Delete (DD) packet to tell the low priority node to delete all VIDs related to that



	A	B	C	D	E	F
VID 1	1	11	12	111	112	
VID 2		121	113	1211	122	
VID 3				1121		
VID 4				1221		

Figure 13. MMT for the semi-automated architecture (Hop limit = 3)

TABLE I. MMT CONTROL OVERHEADS COMPARISON

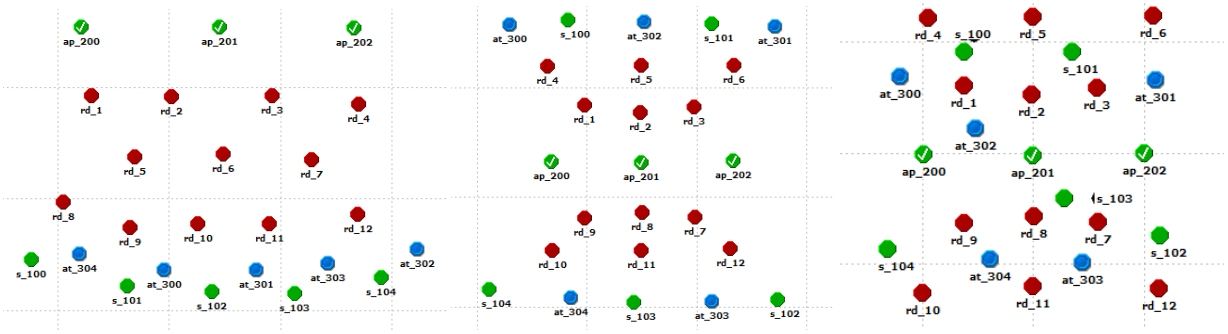
Total number of	VID	AD	JR	JA	Overheads
Original MMT	21	8	30	30	68
MMT-NTT with combined JR JA	11	5	13	10	28

Topology used in Fig. 7 and Fig.13 is used for this comparison

CH. Fig. 11 shows an example scenario for it. When the C_SIZE of the ACP Node X is limited to 7 and it has already reached this value with the member Nodes X, A, B, C, D, E, and F. Sensor Node Y wants to join the cluster, Node Y sends a JR to Node D, and then Node D forwards the JR to the CH, Node X. This JR is accepted by the CH even though it has already reached C_SIZE because Node Y is sensor node and it has higher priority than relay nodes in the cluster. Node X (CH) sends a DD packet to Node F to remove it from the cluster. Thus Node F removes VIDs 1111 and 1121 from its subsequent AD packets. Meanwhile, Node X sends a JA to Node Y.

There is also limit to the maximum number of child nodes under a relay node, based on node density, given by the variable CHILD_SIZE to reduce the number of VIDs. The rationale for this assumption also arises from the fact that if a node has too many children, it could result in a bottleneck. A relay node will give priority in accommodating more sensor and ACT nodes within the CHILD_SIZE limit. Fig. 12 shows an example scenario. When Node B has already accepted the maximum number of children under VID 111 and a sensor Node E wants to join the VID 111, Node B checks Node C's neighbor list and finds that there are no ACT and sensor in this list. Node B can then send a DD message to Node C under the VID 111. Node B will also send a DD packet to Node X to deregister the VID assigned to Node C. Meanwhile, Node B accepts the JR from Node E.

Fig. 13 shows optimized MMT for a sample topology. Sensor and ACT do not support child nodes, so Node D does not have a child VID. Because Node F does not have any sensor and ACT in its neighbor list and Node C and E have already reached 2 hops from the CH, Node F does not join any tree. As a result, total number of control packets is



a) Layered b) Middle c) Random

Figure 14. Relative placement of sensors (s_number), actuators (at_number), relays (rd_number), and ACPs (ap_number)

reduced significantly because total number of VIDs in use is reduced. On the other hand, NTT interacts with MMT to identify sender and destination nodes from the VIDs and to calculate the turn scheduling from neighbor table and its own VIDs.

Table I shows the total number of VIDs assigned by the original MMT and the MMT-NTT that combines JR and JA messages based on the identical topologies in Fig. 7 and Fig. 13. Total number of control packets, i.e., AD, JR and JA is significantly reduced in the MMT-NTT that combines JR and JA messages. Reducing control packets in WSAN and WSANIC reduces number of collision and also achieves low latency in data transfer.

D. Security

As the trees of MMT are rooted at an ACP, the ACP can be used to authenticate the sensors, actuators and relays as they join its tree. In addition, MMT has the following security features.

1) Route spoofing [19] is a common security issue faced by reactive routing protocols. MMT being a proactive routing protocol does not face this issue. Furthermore, during MMT route setup, each node has to register with the ACP, which can employ efficient authentication schemes before admitting nodes to join its tree.

2) Impersonation [20] is easily detected in MMT due to the locality property of the VIDs. If a malicious node A is close to B and learns B's VID by eavesdropping; there are limited ways in which A can use B's VID. If A assumes B's VID in its vicinity, B would recognize this and report (via one of its alternate routes using a different VID) to ACP and the ACP could challenge A. If A takes B's VID and moves away from B to use it, then the VID is invalid because of the locality property of the VIDs. Node A could wait till B moved away and then use the VID, but when B moves away it will acquire and report a new VID to the ACP, and the ACP will know that A is misusing B's VID.

3) Denial of Service (DoS) attacks [21] can be acted upon if the DoS origination point can be located. The affected area can then be quarantined to restrict adverse effects in the rest of the network. In the MMT-based approach, DoS due to flooding or jamming will result in several route failure reports to the ACP. Based on the failure reports in the affected area, the ACP can determine a virtual boundary (of VIDs) of the affected area and isolate that area.

4) Black hole [22] problems are encountered when malicious nodes do not forward incoming packets. An explicit acknowledgement may not resolve this problem as the malicious node can send an acknowledgement for every received data packet without forwarding it. MMT builds routes on links that are bidirectional. At the MAC, forwarding of a data packet can be used as an implicit acknowledgement to the previous sender of the packet and this type of acknowledgement can be used till the packet reaches the destination node, at which point an explicit acknowledgment has to be used. When a node repeatedly fails to forward packets, the parent node reports this to the ACP, which declares that route obsolete by using alternate routes to inform the sensors and actuators that have a route via the defaulting node. A further advantage of using implicit acknowledgements is reduction in the number of transmitted messages.

TABLE II. SIMULATION SETS

	ACT	Sensor	Relay	ACP
NTT-MMT (5)	**5**	**5**	12	3
802-DSR (5)				
NTT-MMT (10)	**10**	**10**	12	3
802-DSR (10)				

TABLE III. SIMULATION SETTINGS

Data size	Data rate	Duration	Transmission	Data Generator
500 bits	0.05 sec	5.0 sec	11 Mbps	ACP, Sensor

VI. ANALYSIS RESULTS

A. The Evaluation Topology and Simulation Scenarios

Fig. 14 shows the topologies used in the OPNET simulations [23] to evaluate the proposed scheme. The

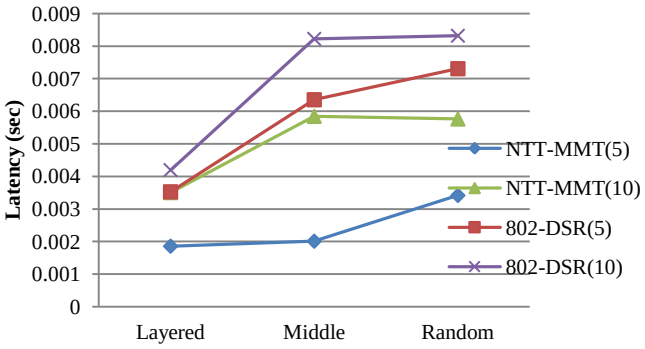


Figure15. End-to-end Latency

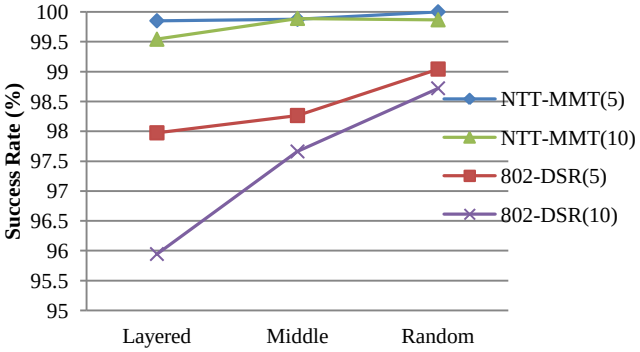


Figure16. Success Rates

topologies show relative placement of the sensors and actuators with respect to the ACPs, which is similar to the semi-automated industry architecture discussed earlier. Node name starting with s_, *at_*, *rd_*, and *ap_* are sensors, ACTs, relays, and ACPs respectively. Nodes in Fig. 14 (a) are placed in a layered manner where the top layer has ACPs, middle layer has relays, and bottom layer has sensors and ACTs. In Fig 14 (b), ACPs are placed in the middle and sensors and ACTs are placed at the edges. In Fig 14(c), all nodes except ACPs are placed randomly.

Several sets of simulations runs were conducted and each set is recorded in Table II. Each set was conducted on the three topologies described in Figure 14(a), (b) and (c). Each simulation was run for 5 seconds, and was repeated for 5 different seeds. Table III records the simulation setting. At the ACPs and the sensors, data was generated at the rate of one packet in 0.05 seconds, with a packet size of 500 bits. The transmission data rate was set to 11 Mbps. Data from sensors were sent to one of the three ACPs and data from ACPs were sent to all of the ACTs. Thus, in the 5 sensors and 5 ACTs scenario, a total of 2000 data packets (5 seconds / 0.05 packets * 5 sensors + (5 seconds / 0.05 packets) * 3 ACPs * 5 ACTs) can be transmitted and a total 4000 data packets can be transmitted in the 10 sensors and 10 ACTs scenario if routes between sensors and ACPs, and ACPs and ACTs are fully maintained by routing protocol.

The proposed architecture, which supports an integrated NTT-MAC and MMT routing protocol, called NTT-MMT is compared with a similar architecture using 802.11 CSMA/CA MAC and DSR routing protocol, called 802-DSR in the plots.

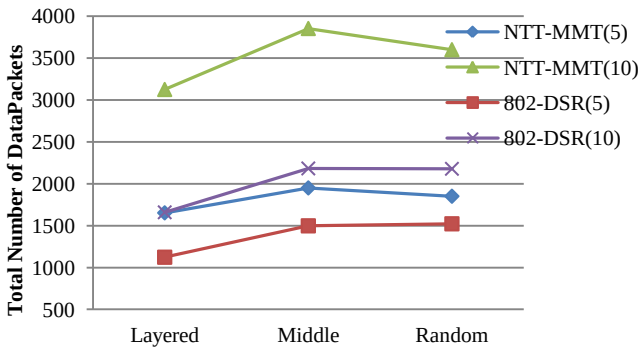


Figure17. Total number of Data packets

B. *Performance Metrics*

1) Average end-to-end latency is the time taken from transmission of a data packet at the sender to its reception at the receiver.

2) Average success rate is calculated as the ratio of total number of packets received correctly at the destination node to the total number of packets sent by the sender node.

3) Average total number of sent data is also recorded to provide data on the traffic loads in the scenario.

C. *Simulation Results*

Fig. 15 shows results of average end-to-end latencies. Latencies of both NTT-MMT and 802-DSR have increased in the 10 sensors and 10 ACTs scenario as compared to the 5 sensor and 5 ACT scenario. This is because the number of nodes and generated data packets are increased for the same simulation field size. Therefore, the 10 nodes scenario is more congested than the 5 nodes scenario.

Fig. 16 shows results of average success rates. Both the 5 nodes and 10 nodes scenario show high success rates in the NTT-MMT scheme. The reason that the random topology has higher success rate compared to others is because the number of hops between sensors and ACPs, ACPs and ACTs are smaller in this topology.

Fig. 17 captures average total number of sent data packets. NTT-MMT sent 1.5 times more data packets than 802-DSR because NTT-MAC has a better utilization of the wireless medium compared to IEEE 802.11 CSMA/CA and MMT routing maintains more routes than DSR routing. Based on the data rate of packet generation, 2000 and 4000 data packets can be generated in 5 seconds simulation for the 5 nodes and the 10 nodes topologies. NTT-MMT could process 91% and 88% of the maximum number of data packets generated for the 5 nodes and the 10 nodes scenario respectively. On the other hand, 802-DSR could process only 69% and 50% of the maximum number of data packets generated.

The NTT-MMT achieves high success rate and low latency at the same time. In addition, NTT-MMT could send more data packets. Robustness in NTT-MMT is high because success rates of NTT-MMT remains high irrespective of the

different topologies and in highly congested network situations.

VII. CONCLUSIONS

The NTT-MAC is contention based but uses a loosely scheduled medium access scheme that does not require strict time synchronization or a central server because it schedules based on neighbor activity. The main performance aspect we targeted when we developed NTT-MAC scheme was to achieve reduced latency and higher success rate. We also introduced a routing protocol based on the MMT algorithm, which is a proactive routing protocol along with the NTT-MAC. MMT is developed to support high route robustness with a quick and easy forwarding approach based on virtual IDs. In industry control, Wireless Sensor-Actuator Ad-hoc Network using NTT-MAC and MMT-routing will provide superior performance. The performance metrics focused were success rate, packet delivery latency, and number of delivered data packets. The simulation results show improved performance of NTT-MMT in terms of success rate and end to end latency compared to DSR operating with IEEE 802.11 CSMA/CA MAC.

REFERENCES

[1] Y. Nozaki, N. Shenoy, and Q. Li, "Wireless sensor actor networks for industry control," ICNS 2013, The Ninth International Conference on Networking and Services, pp. 172-178, 2013.

[2] N. Shenoy, C. Xiaojun, Y. Nozaki, S. Hild and P. Chou, "Neighbor turn taking MAC - a loosely scheduled access protocol for wireless networks," Personal Indoor and Mobile Radio Communications, PIMRC 2007, IEEE 18th International Symposium on, pp. 1-5, 2007.

[3] E. F. Golen, Y. Nozaki and N. Shenoy, "An analytical model for the neighbor turn taking MAC protocol," Military Communications Conference, MILCOM 2008, IEEE, pp. 1-7, 2008.

[4] N. Shenoy, Y. Pan, D. Narayan, D. Ross and C. Lutzer, "Route robustness of a multi-meshed tree routing scheme for Internet MANETs," Global Telecommunications Conference, GLOBECOM 2005, IEEE, vol. 6, pp. 3351-3356, 2005.

[5] D. Chen, M. Nixon, T. Aneweer, R. Shepard, and A. K. Mok, "Middleware for wireless process control systems," Workshop on Architectures for Cooperative Embedded Real-Time Systems, 2004, http://wacerts.di.fc.ul.pt/papers/Session1-ChenMok.pdf. (accessed December 2013)

[6] J. Song, A. K. Mok, D. Chen, and M. Nixon, "Challenges of wireless control in process industry," in Workshop on Research Directions for Security and Networking in Critical Real-Time and Embedded Systems, San Jose, CA, 2006, http://moss.csc.ncsu.edu/~mueller/ftp/pub/mueller/papers/cps06.pdf. (accessed December 2013)

[7] T. Brooks, "Wireless technology for industrial sensor and control networks," Sensors for Industry, 2001. Proceedings of the First ISA/IEEE Conference, pp.73-77, 2001.

[8] T. Enwall, "Deploying wireless sensor networks for industrial automation and control," http://www.eetimes.com/design/industrial-control/4013661/Deploying-Wireless-Sensor-Networks-for-Industrial-Automation-Control. (accessed December 2013)

[9] "ISA-100 wireless system for automation," http://www.isa.org/MSTemplate.cfm?MicrositeID=1134&CommitteeID=6891. (accessed December 2013)

[10] B. P. Gerkey and M. J. Mataric, "A market-based formulation of sensor-actuator network coordination," in Proceedings of the AAAI Spring Symposium on Intelligent Embedded and Distributed Systems, Palo Alto, California, pp. 21-26, 2002.

[11] P. Robert, "Wireless mesh networks," http://www.sensorsmag.com/networking-communications/standards-protocols/wireless-mesh-networks-968. (accessed December 2013)

[12] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," Ad Hoc Networks, vol. 2, pp. 351-367, 2004.

[13] L. Barolli, T. Yang, G. Mino, F. Xhafa and A. Durresi., "Routing efficiency in wireless sensor-actor networks considering semi-automated architecture," Journal of Mobile Multimedia, vol. 6, pp. 97-113, 2010.

[14] N. Shenoy, Y. Pan, and V. G. Reddy, "Quality of service in internet MANETs," Personal Indoor and Mobile Radio Communications, PIMRC 2005, IEEE 16th International Symposium on, vol. 3, pp. 1823-1829, 2005.

[15] IEEE Computer Society LAN MAN Standards Committee. "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," 1997.

[16] V. C. Giruka and M. Singhal, "Hello protocols for ad-hoc networks: overhead and accuracy tradeoffs," World of Wireless Mobile and Multimedia Networks, WoWMoM 2005, 6th IEEE International Symposium on, pp. 354-361, 2005.

[17] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: the dynamic source routing protocol for multi-hop wireless ad hoc networks," Ad Hoc Networks, vol. 5, pp. 139-172, 2001.

[18] T. Clausen, and P. Jacquet, "Optimized link state routing protocol (OLSR)," RFC 3626, IETF Networking Group, 2003.

[19] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "A survey of routing attacks in mobile ad hoc networks," Wireless Communications, IEEE, vol. 14, no. 5, pp. 85-91, 2007.

[20] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "A secure routing protocol for ad hoc networks," Network Protocols, 2002. Proceedings. 10th IEEE International Conference on, pp. 78-87, 2002.

[21] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," Computer, vol. 35, no. 10, pp. 54-62, 2002.

[22] M. Al-Shurman, S. M. Yoo, and S. Park, "Black hole attack in mobile ad hoc networks," In Proceedings of the 42nd annual southeast regional conference, ACM, pp. 96-97, 2004.

[23] "OPNET modeler," http://www.riverbed.com/products-solutions/products/network-performance-management/network-planning-simulation/Network-Simulation.html. (accessed December 2013)

Fault-Tolerant and Energy-Efficient Generic Clustering Protocol for Heterogeneous WSNs

Mandicou Ba, Olivier Flauzac, Rafik Makhloufi, and Florent Nolot
Université de Reims Champagne-Ardenne, France
CReSTIC - SysCom EA 3804
{mandicou.ba, olivier.flauzac, rafik.makhloufi, florent.nolot}@univ-reims.fr

Ibrahima Niang
Université Cheikh Anta Diop, Sénégal
Laboratoire d'Informatique de Dakar (LID)
iniang@ucad.sn

***Abstract*—In the context of Wireless Sensor Networks (WSNs), where sensors have limited energy power, it is necessary to carefully manage this scarce resource by saving communications. Clustering is considered as an effective scheme to increase the scalability and lifetime of wireless sensor networks. Moreover, failures and topological changes are inevitable in sensor networks due to the inhospitable environment, unattended deployment or nodes mobility. Therefore, one of the wanted properties of WSNs is the fault tolerance and adaptivity to topological changes. We propose a fault-tolerant and energy-efficient distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. This protocol is adapted to topological changes, optimizes energy consumption and prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters. Our generic clustering protocol can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads, such as nodes' identity, residual energy or degree. We propose to validate our approach under the different election metrics by evaluating its communication cost in terms of messages, energy consumption and number of clusters. Simulation results show that, in terms of of messages, energy consumption and clusters distribution, it is better to use the Highest-ID metric for electing CHs. Furthermore, after faults occurrence, the re-clustering cost is minimal compared to the clustering cost.**

***Keywords*-Self-stabilizing clustering; Wireless Sensor Networks; Energy-efficient; Fault-tolerant; OMNeT++ simulator.**

I. Introduction

A preliminary version of this paper, entitled "Evaluation Study of Self-Stabilizing Cluster-Head Election Criteria in WSNs", is published in CTRQ'2013 [1]. In this paper, we include fault-tolerance and energy-efficiency mechanisms in the context of heterogeneous Wireless Sensors Networks (WNSs) with energy constraint. To the best of our knowledge, there is no paper in the literature where the solutions are fault-tolerant, energy-aware, self-stabilizing and where the same proposed approach is compared in the case of different CH election methods.

Due to their properties and wide applications, WSNs have been gaining growing interest in the last decades. These networks are used in various domains like: medical, scientific, environmental, military, security, agricultural, smart homes, etc. [2].

In WSN, sensors have very limited energy resources due to their small size. This battery power is consumed by three operations: data sensing, communication, and processing. Communication by messages is the activity that needs the most important quantity of energy, while power required by CPU is minimal. For example, Pottie and Kaiser [3] show that the energy cost of transmitting a 1KB message over a distance of 100 meters is approximately equivalent to the execution of 3 million CPU instructions by a 100 MIPS/W processor. Thus, conserving communication power is more important in WSNs than optimizing processing. Consequently, to extend the sensor network lifetime, it is very important to carefully manage the very scarce battery power of sensors by limiting communications. This can be done through notably efficient routing protocols that optimize energy consumption. Many previous studies (e.g., Yu et *al.* [4] and Younis and Fahmy [5]) proved that clustering is an effective scheme in increasing the scalability and lifetime of wireless sensor networks. Clustering consists in partitioning the network into groups called clusters, thus giving a hierarchical structure [6].

On the other hand, nodes in WSNs are prone to be failure due to energy depletion, hardware failure, communication link errors, malicious attack, and so on. Fault tolerance is one of the critical issues in WSNs as proved in many studies like Liu et *at.* [7], Zhang et *al.* [8] and Hao et *al.* [9]. Fault tolerance is defined as the ability of a system to deliver a desired level of functionality in the presence of faults [10]. Therefore, one of the most wanted properties of WSNs is the fault tolerance and adaptivity to topological changes, which consist of the system's ability to react to faults and perturbations. Self-stabilization is an approach to design fault-tolerant and adaptive to topological changes distributed systems [11].

Several self-stabilization clustering approaches are proposed in the literature and used, for example, in the case of a WSN for routing collected information to a base station. However, most of them are based on state model, so they are not realistic compared to message-passing based clustering ones. Moreover, approaches in the last category are not self-stabilizing and they are generally highly costly in terms of messages; while in the case of WSNs, clustering aims at optimizing communications and energy consumption.

In this paper, we propose a fault-tolerant and energy-efficient distributed self-stabilizing clustering protocol based on message-passing for heterogeneous wireless sensor networks. The proposed algorithm is based only on information from neighboring nodes at distance 1 to build k-hops

clusters. It optimizes energy consumption and then prolongs the network lifetime by minimizing the number of messages involved in the construction of clusters. Our clustering protocol offers an optimized structure for routing. It can be easily used for constructing clusters according to multiple criteria in the election of cluster-heads such as: nodes' identity, residual energy, degree or a combination of these criteria. We propose to validate our approach by evaluating its communication cost in terms of messages, energy consumption and percentage of formed clusters. Thus, on one hand, we compare its performance in the case of using different cluster-heads election methods under the same clustering approach and testing framework. On the other hand, we evaluate the fault-tolerance mechanism of proposed approach. Moreover, we compare our algorithm with some of the most referenced self-stabilizing solutions.

The remainder of the paper is organized as follows. Section II illustrates the related work on clustering approaches. Section III describes the proposed clustering approach, clusterhead election methods and the fault-tolerant mechanism. Theoretical validation is discussed in Section IV, where we compare our algorithm with some of most referenced self-stabilizing solutions. Section V presents the validation of the proposed approach through simulation. Finally, Section VI concludes this paper and presents our working perspectives.

II. Related Work

Several proposals of self-stabilizing clustering have been done in the literature [12], [13], [14], [15], [16], [17], [18]. However, self-stabilizing algorithms presented in [14], [15], [16], [17] are 1-hop clusters solutions.

A metric called *density* is used by Mitton *et al.* in [17], in order to minimize the reconstruction of structures for low topology change. Each node calculates its density and broadcasts it to its neighbors located at 1-hop. For the maintenance of clusters, each node periodically calculates its mobility and density.

Flauzac *et al.* [14] have proposed a self-stabilizing clustering algorithm, which is based on the identity of its neighborhood to build clusters. This construction is done using the identities of each node that are assumed unique. The advantage of this algorithm is to combine in the same phase the neighbors discovering and the clusters establishing. Moreover, this deterministic algorithm constructs disjoint clusters, i.e., a node is always in only one cluster.

In [15], Johnen *et al.* have proposed a self-stabilizing protocol designed for the state model to build 1-hop clusters having a bounded size. This algorithm guarantees that the network nodes are partitioned into clusters where each one has at most $SizeBound$ nodes. The clusterheads are chosen according to their $weight$ value. In this case, the node with the highest weight becomes clusterhead. In [16], Johnen *et al.* have extended this proposal from [15]. They have proposed a robust self-stabilizing weight-based clustering algorithm. The robustness property guarantees that, starting from an arbitrary configuration, after one asynchronous round, the network is partitioned into clusters. After that, the network stays partitioned during the convergence phase toward a legitimate configuration where clusters verify the ad hoc clustering properties. These approaches [15], [16], based on state model, are not realistic in the context of wireless sensor networks.

Self-stabilizing algorithms proposed in [12], [13], [18] are k-hops clustering solutions.

In [18], Mitton *et al.* applied self-stabilization principles over a clustering protocol proposed in [17] and they presented properties of robustness. Each node computes its *k-density* value based on its view ($\{k+1\}$-neighborhood) and locally broadcasts it to all its neighbors at distance k. Thus, each node is able to decide by itself whether it wins in its *1-neighborhood* (as usual, the smallest ID will be used to decide between joint winners). Once a clusterhead is elected, the clusterhead ID and its density are locally broadcasted by all nodes that have joined this cluster. A cluster can then extend itself until it reaches a cluster frontier of another clusterhead. The approach proposed in [17], [18] generates a lot of messages. The main reason is due to the fact that each node must know $\{k+1\}$-*neighboring*, computes its *k-density* value and locally broadcasts it to all its *k-neighbors*. This is very expensive in terms of messages and causes an important energy consumption.

In [13], using the criterion of minimal identity, Datta *et al.* have proposed a self-stabilizing distributed algorithm called *MINIMAL*. This approach is designed for the *state model* (also called *shared memory model*) and uses an unfair daemon. Authors consider an arbitrary network G of processes with unique IDs and no designated leader. Each process can read its own registers and those of its neighbors at distance k, but can write only to its own registers. They compute a subset D, a minimal *k-dominating* set of graph G. D is defined as a *k-dominating* set if every process that is not in D is at distance at most k from a member of D. *MINIMAL* converges in $O(n)$ rounds. Using D as the set of *clusterheads*, a partition of G into clusters, each of radius k follows. Authors show that $O(n^2)$ steps are sufficient for the phase clock to stabilize. And after stabilization, *MINIMAL* requires $O(n^2)$ steps to execute n actions. Thus, the system converges to a terminal configuration in $O(n^2)$ steps starting from any configuration and requires $O(\log(n))$ memory space per process, where n is the size of the network.

Caron *et al.* [12], using as metric a unique ID for each process and weighted edges, have proposed a self-stabilizing *k-clustering* algorithm based on a state model. Note that *k-clustering* of a graph consists in partitioning network nodes into disjoints clusters, in which every node is at a distance of at most k from the clusterhead. This solution is partially inspired by Amis *et al.* [19] and finds a *k-dominating* set in a network of processes. It is a combination of several self-stabilizing algorithms and it uses an unfair daemon. Each process can read its own registers and those of its neighbors at distance $k+1$, but can write only to its own registers. This algorithm executes in $O(nk)$ rounds and requires $O(log(n) + log(k))$ memory space per process, where n is the network size.

III. Proposed Clustering approach

A. Basic idea

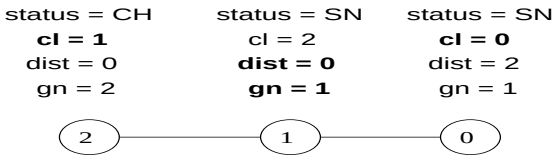
To simplify the description of our approach, we consider the case where the selection criterion to become clusterhead is the node's identity. We will present later the proposed approach using others CHs election criteria.

Our proposed algorithm is self-stabilizing and does not require any initialization. Starting from any arbitrary configuration, with only one type of exchanged message, nodes are structured in non-overlapping clusters in a finite number of steps. This message is called *hello message* and it is periodically exchanged between each neighbor nodes. It contains the following four information: node identity, cluster identity, node status and the distance to cluster-head. Note that cluster identity is also the identity of the cluster-head. Thus, the hello message structure is $hello(id_u, cl_u, status_u, dist_{(u,CH_u)})$. Furthermore, each node maintains a neighbor table $StateNeigh_u$ that contains the set of its neighboring nodes states. Whence, $StateNeigh_u[v]$ contains the states of nodes v neighbor of u.

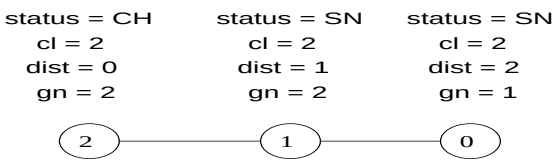
The solution that we propose proceeds as follows:

As soon as a node u receives a hello message, it executes three steps consecutively (see Algorithm 1). The first step is to update neighborhood. The next step is to manage the coherence and the last step is to build the clusters. During the last step, each node u chosen as cluster-head the node that optimizes the criterion and located at most a distance k. At the end of this three steps, u sends a hello message to its neighbors. The details of Algorithm 1 and mathematical proof are describe in Ba et *al.* [20]. Note that we have illustrated this algorithm with the ID criterion. Nevertheless, for the Degree and Energy criteria, we have the same design.

After updating the neighborhood, nodes check their coherency. For example, as a cluster-head has the highest identity, if a node u has CH status, its cluster identity must be equal to its identity. In Fig. 1(a), node 2 is cluster-head. Its identity is 2 and its cluster identity is 1, so node 2 is not a coherent node. Similarly for nodes 1 and 0. Each node detects its incoherence and corrects it during the coherence management step. Fig. 1(b) shows nodes that are coherent.



(a) Incoherent nodes



(b) Coherent nodes

Figure 1. Coherent and incoherent nodes

Algorithm 1: Fault-Tolerant and Energy-Efficient Generic Clustering algorithm for WSNs.

/* Upon receiving message from a neighbor */

Predicates

$P_1(u) \equiv (status_u = CH)$
$P_2(u) \equiv (status_u = SN)$
$P_3(u) \equiv (status_u = GN)$
$P_{10}(u) \equiv (cl_u \neq id_u) \vee (dist_{(u,CH_u)} \neq 0) \vee (gn_u \neq id_u)$
$P_{20}(u) \equiv (cl_u = id_u) \vee (dist_{(u,CH_u)} = 0) \vee (gn_u = id_u)$
$P_{40}(u) \equiv$
$\forall v \in N_u, (id_u > id_v) \wedge (id_u \geq cl_v) \wedge (dist_{(u,v)} \leq k)$
$P_{41}(u) \equiv \exists v \in N_u, (status_v = CH) \wedge (cl_v > cl_u)$
$P_{42}(u) \equiv \exists v \in N_u, (cl_v > cl_u) \wedge (dist_{(v,CH_v)} < k)$
$P_{43}(u) \equiv \forall v \in N_u / (cl_v > cl_u), (dist_{(v,CH_v)} = k)$
$P_{44}(u) \equiv \exists v \in N_u, (cl_v \neq cl_u) \wedge \{(dist_{(u,CH_u)} = k) \vee (dist_{(v,CH_v)} = k)\}$

Rules

/* Update neighborhood */
$StateNeigh_u[v] := (id_v, cl_v, status_v, dist_{(v,CH_v)})$;

/* Cluster-1: Coherent management */
$R_{10}(u) :: P_1(u) \wedge P_{10}(u)$
$\longrightarrow cl_u := id_u; gn_u = id_u; dist_{(u,CH_u)} = 0;$
$R_{20}(u) :: \{P_2(u) \vee P_3(u)\} \wedge P_{20}(u) \longrightarrow$
$status_u := CH; cl_u := id_u; gn_u = id_u; dist_{(u,CH_u)} = 0;$

/* Cluster-2: Clustering */
$R_{11}(u) :: \neg P_1(u) \wedge P_{40}(u) \longrightarrow status_u := CH; cl_u := id_v; dist_{(u,CH_u)} := 0; gn_u := id_u;$
$R_{12}(u) :: \neg P_1(u) \wedge P_{41}(u) \longrightarrow status_u := SN; cl_u := id_v; dist_{(u,v)} := 1; gn_u := NeighCH_u;$
$R_{13}(u) :: \neg P_1(u) \wedge P_{42}(u) \longrightarrow$
$status_u := SN; cl_u := cl_v; dist_{(u,CH_u)} := dist_{(v,CH_v)} + 1; gn_u := NeighMax_u;$
$R_{14}(u) :: \neg P_1(u) \wedge P_{43}(u) \longrightarrow status_u := CH; cl_u := id_v; dist_{(u,CH_u)} := 0; gn_u := id_u;$
$R_{15}(u) :: P_2(u) \wedge P_{44}(u) \longrightarrow status_u := GN;$
$R_{16}(u) :: P_1(u) \wedge P_{41}(u) \longrightarrow status_u := SN; cl_v := id_v; dist_{(u,v)} := 1; gn_u := NeighCH_u;$
$R_{17}(u) :: P_1(u) \wedge P_{42}(u) \longrightarrow$
$status_u := SN; cl_u := cl_v; dist_{(u,CH_u)} := dist_{(v,CH_v)} + 1; gn_u := NeighMax_u;$

/* Sending hello message */
$R_0(u) :: hello(id_u, cl_u, status_u, dist_{(u,CH_u)})$;

B. Cluster-heads election

Existing clustering approaches use one or more criteria for electing cluster-heads, for example: nodes' ID, degree, density, mobility, distance between nodes, service time as a CH, security, information features or a combination of multiple criteria. However, to the best of our knowledge, there is no

paper in the literature where the same proposed approach is compared in the case of different CH election methods. It is important to study the influence of each criterion under the same test conditions and, ideally, under the same clustering approach. To this end, we propose a generic distributed self-stabilizing clustering approach that can be used with any CH election criterion. Then, we compare costs and performance of the proposed solution in the case where several election criteria (Highest-ID, Highest-degree and residual energy of nodes) are used.

1) Highest ID:

Lowest-Identifier based clustering was originally proposed by Baker et *al.* [21]. It has proven that, clustering based on ID criterion is one of the most performant approaches in ad hoc networks [22], [23], [24], [25].

In our approach, each node compares its identity with those of its neighbors a distance 1. A node u elects itself as a clusterhead if it has the highest identity among all nodes of its cluster (in Fig. 2, example of node 9 in cluster V_9). If a node u discovers a neighbor v with a highest identity then it becomes a node of the same cluster as v with SN status (in Fig. 2, example of nodes $1, 3, 4$ and 7 in cluster V_9). If u receives again a hello message from another neighbor which is into another cluster than v, the node u becomes gateway node with GN status (in Fig. 2, example of nodes 5 and 8 in cluster V_{10} and node 2 in cluster V_9). As the hello message contains the distance between each node u and its clusterhead, u knows if the diameter of cluster is reached. So it can choose another cluster.

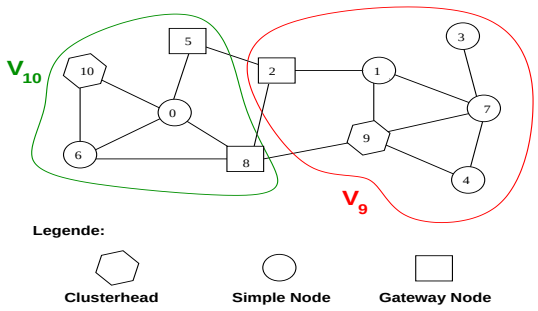


Figure 2. Clusters organization ($k = 2$)

2) Highest or Ideal Degree: In this approach, we determine how well suited a node is for becoming CH according to its degree D (i.e., the number of neighbors). There are two categories of approaches based on nodes' degrees. Some of them propose to limit communications by electing the node having the highest degree as CH. This is an original proposal of Gerla and Tsai [26]. However, each CH can ideally support only ρ (a pre-defined threshold) nodes to ensure an efficient functioning regarding delay and energy consumption. Indeed, at each step of the routing process, when a node has many neighbors it receives as many messages as its degree. This leads to a rapid draining of sensors' battery power. To ensure that a CH handles upto a certain number of nodes in its cluster, some approaches [24], [27], [28] propose to elect as CH the node having the nearest degree to an ideal value ρ. Thus, the best candidate is the one minimizing its distance to this ideal degree $\triangle_d = |D - \rho|$.

For the two cases described above, when more than one node has the maximum (respectively ideal) degree and is candidate to become a CH, the election is done according to a secondary criterion which is the highest ID. As each node of the network has a unique ID, this criterion is discriminating.

3) Residual Energy: In this approach, decision-making concerning the most suitable node to become CH is done according to the residual energy (i.e., remaining battery power level) of each sensor. Indeed, CHs are generally much more solicited during the routing process. So, in order to preserve their energy and to avoid the frequently reconstruction of the clusters, CHs need more important battery levels compared to the others normal nodes.

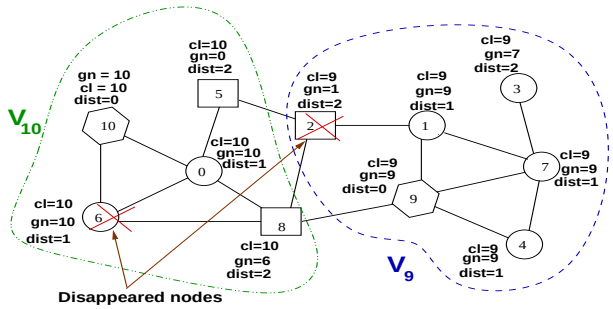
During the clustering procedure, network nodes progressively consume their energy due to the messages exchanges. Thus, after some rounds a node i with initially the maximum battery power level and candidate to become a CH can have later less energy than an another neighbor node j. This can lead to more iterations aiming at electing the other node j with the maximum residual energy. In order to limit the frequently changes of CH candidates for a negligible energy difference, we propose to use an energy gain threshold E_T. Thus, while $\triangle_e = |E_i - E_j|$ is less than E_T, the node i preserves its leadership position. This guarantees more stability of the clustering process and extends the network lifetime by minimizing the energy consumption involved in the clustering procedure.

C. Fault-tolerance mechanism

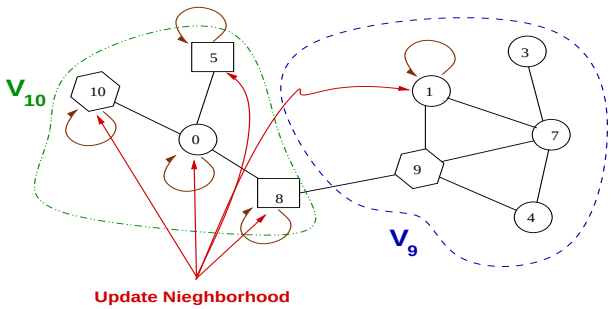
In this section, we study the fault-tolerance mechanism of proposed approach. Our algorithm is fault-tolerant and adapted to topological changes. To the best of our knowledge, there is no paper in the literature where the solutions are fault-tolerant, energy-aware, self-stabilizing and where the same proposed approach is compared in the case of different CH election methods.

A system failure occurs when the delivered service deviates from the specified service [10]. Hardware and software faults affect the system state and the operational behavior, such as memory or register content, program control flow, and communication links, etc. Communication faults can be caused due to hardware failure or energy depletion. Communication can be disrupted due to environmental conditions like wind or rain. Hardware faults can also disrupt radio communication, ending all the communication.

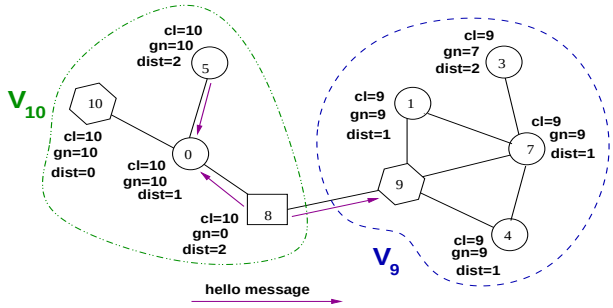
In the following, we consider that after the occurrence of a fault, the concerned node disappears from the network and the graph remains connected. We also assume that faults can occur after stabilization (i.e., after clusters formation). As soon as a node detects the disappearance of a neighbor, it considers this as an occurred fault. Thus, it triggers the fault-tolerance mechanism called *re-clustering*. Let u the disappeared node. According the status of node u, two cases are possible:



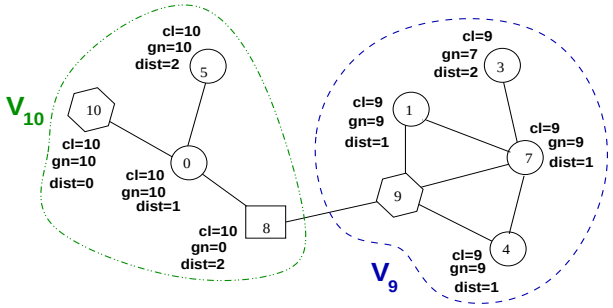
(a) Clusters structure before the disappearance of node's status SN or GN. In this example, node 6 in cluster of node 10 with SN status and node 2 in cluster of node 9 with GN status will disappear from the network.



(b) After the disappearance of node's status SN or GN, nodes 10, 0 and node 8 member of cluster of node 10 that detect the disappearance of node 6 update its neighborhood table. Similarly, node 5 and 1 update their neighborhood tables after the disappearance of node 2 with GN status and member of cluster of node 9.



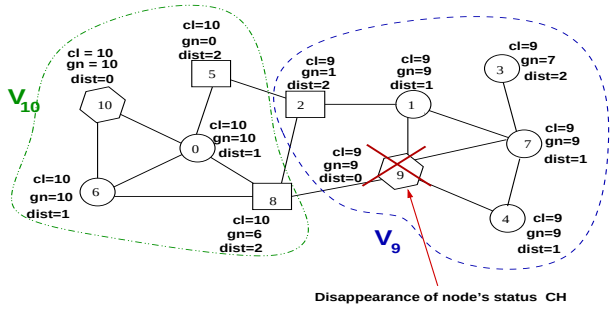
(c) After updating the neighborhood, nodes impacted by the disappearance of a neighbor send a hello message to its neighbors. In this example, node 8 is impacted by the disappearance of node 6. In fact, as shows in Fig. 3(a), the gateway of node 8 to reach its CH (node 10) was node 6. Thus, node 8 selects node 0 as the new gateway to reach its CH. It sends a hello message to its all neighbors at distance 1 to notify of its gateway change. Similarly to node 5, it becomes simple node with SN status. In fact, node 2 was the only one to be a member of another cluster in the neighborhood of node 5.



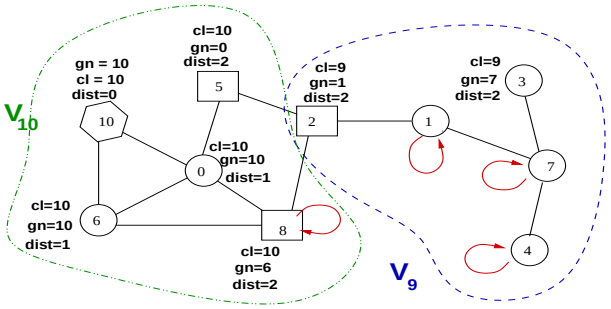
(d) Clusters structure at the end of re-clustering process caused by the disappearance of node's status SN or GN. Thus, the disappearance of node's status SN or GW does not lead to clusters change but only one updating neighborhood table.

Figure 3. Disappearance of node's status SN or GW (in this example $k = 2$)

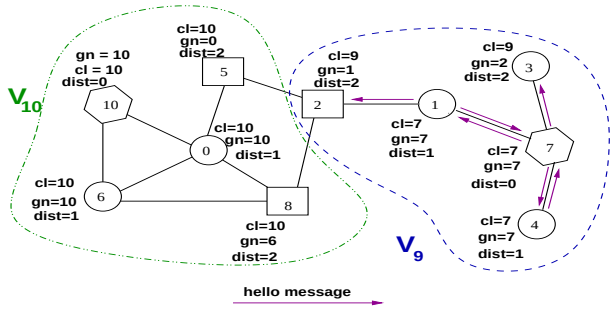
- **Case 1**, $status_u \in \{SN, GN\}$: the disappeared node is a simple or gateway node. In this case, all node v that detects the disappearance of node u removes all information about u from its neighborhood table. As illustrated in Fig. 3, the disappearance of node's status SN or GW does not lead to clusters change but only one updating neighborhood table. However, if the node u has been chosen by a node v as gateway (i.e., $gn_v = id_u$) through which it can reach its CH, then v chooses another node w in its neighborhood table as new gateway to reach its CH. Furthermore, if node u was the only one to be a member of another cluster in the neighborhood of node v, thus v becomes simple node with status SN. After updating the neighborhood table, all node v that is impacted by the disappearance of node u sends a hello message to its all neighbors distance 1.
- **Case 2**, $status_u = CH$: if a node u with CH status disappears from de the network, the fault-tolerance mechanism proceeds as follows (example of the disappearance of node 9 as illustrated in Fig. 4):
  1) For all node v at distance 1 of u that is member of another cluster (i.e., $(cl_v \neq id_u) \wedge (dist_{(v,CH_v)} = k)$), the only requirement action is to remove all information about this CH by updating its neighborhood table. This is the case case of node 8 as illustrated in Fig. 4(a) and Fig. 4(b)
  2) For all node v at distance 1 of u that is member of cluster of node u (i.e., $v \in N_u$ such that $status_v \in \{SN, GN\} \wedge (cl_v = id_u)$) as illustrated in Fig. 4(b) and Fig. 4(c), v executes three actions. Firstly, it removes all information about this CH by updating its neighborhood table. Secondly, it triggers the re-clustering process in order to choose another cluster-head. Thirdly, after having chosen another cluster-head, each node v sends to its neighbors at distance 1 a hello message in order to inform their cluster-head change. Therefore, all node w at distance 2 to u such that $w \in N_v \backslash \{v\} \wedge cl_w = id_u$ receives information about the disappearance of node u.
  3) Thus, in our process of re-clustering, we have the following induction assumption: each node at distance i of the disappeared CH, executes process re-



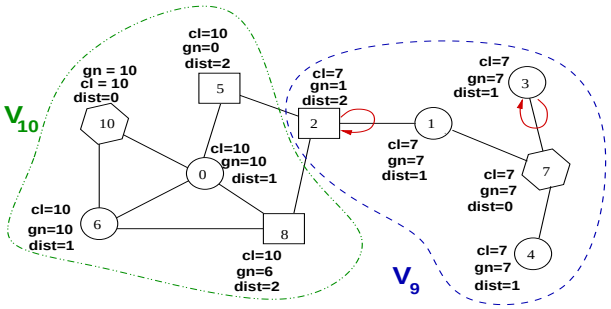
(a) Structure of clusters before the disappearance of node 9 with CH status.



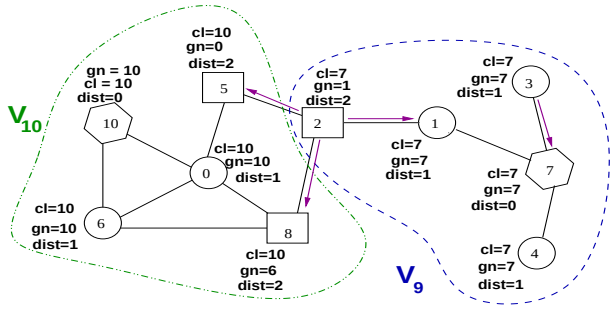
(b) After the disappearance of node 9 with CH status, node 8 removes all information about node 9 in its neighborhood table. In fact, node 8 is member of cluster of node 10 and it is at distance 2 of node 10. Thus, it is not affected by the disappearance of node 9. Nevertheless, nodes 1, 4 and 7 have as cluster-head node 9. Thus, they will trigger the process of re-clustering after updating their neighborhood table.



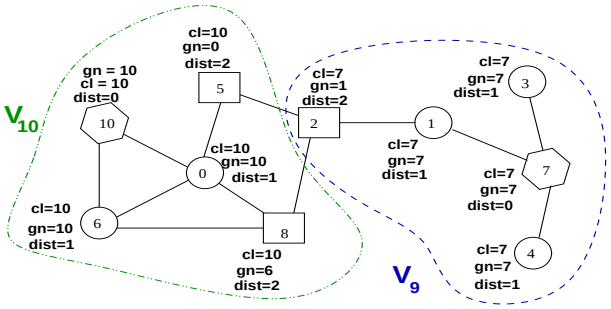
(c) After updating neighborhood, each node impacted by the disappearance of node 9 chooses another cluster-head. To do this, each node select from its neighborhood table the node with highest ID. Node 7, as it have the highest ID in its neighborhood, it becomes cluster-head. Nodes 1 and 4 choose node 7 as cluster-head. In fact, node 7 represents node with highest ID at distance 1 in the neighborhood of nodes 1 and 4. Therefore, each node with state change sends a hello message to its neighbors at distance 1.



(d) Upon receiving a hello message from node 1 notifying its cluster-head change, node 2 knows that its cluster-head (node 9 as illustrated in Fig. 4(a)) has disappeared. Thus, it triggers the process of re-clustering after updating its neighborhood table. As node node 7 is the node with the highest ID in at most at distance 2 (node 10 is at distance 3 and in this example $k = 2$), it is selected as a cluster-head by node 2. Similarly, node 3 applies the same principle and becomes member of cluster of node 7.



(e) Nodes 2 and 3 send a hello message to its neighbors at distance 1 in order to inform about its state change. All nodes that receive message from node 2 and node 3 update information about these nodes in its neighborhood table. As $k = 2$ in this example, the re-clustering process ends at this step. All clusters become stable as showed in Fig. 4(f).



(f) Structure of clusters after the disappearance of node 9 with CH status

Figure 4. Disappearance of node with CH status (in this example $k = 2$)

clustering and informs its neighbors at distance $i+1$. So on until the whole network becomes stable again. As illustrated in Fig. 4(f), nodes 2 and 3 apply this induction assumption to correct the disappearance of its CH.

IV. Theoretical Validation

In [20], we have provided a formal proof of our clustering approach. Table I illustrates a comparison of stabilizing time and memory space between our proposal algorithm and other approach designed for the state model. We note that our stabilization time does not depend on the parameter k contrary to approach proposed by Caron et *al.* [12]. We have a unique

TABLE I
THEORETICAL COMPARISON OF STABILIZING TIME AND MEMORY SPACE

	Stabilization Time	Memory space per node	neighborhood
Our approach	$n+2$	$\log(2n+k+3)$	1 hop
Datta et al. [13]	$O(n)$, $O(n^2)$	$O(\log(n))$	k hops
Caron et al. [12]	$O(n*k)$	$O(log(n)+log(k))$	k+1 hops

phase to discover the neighborhood and build *k-hops* clusters and an unique stabilization time contrary to approach describes in [13]. Furthermore, we consider a 1-hop neighborhood at opposed to Datta et *al.* [13] and Caron et *al.* [12].

Furthermore, in Ba et *al.* [29], we have compared our proposed algorithm with one of most referenced papers on self-stabilizing solutions based on message-passing model [18]. This shows that we reduce communication cost and energy consumption by a factor of at least 2.

V. VALIDATION FRAMEWORK

In this section, we present the evaluation study that we carried out using *ONMeT++* [30] simulator to compare the performance of the previously described clustering approach when utilizing different CH election methods. For generating random graphs, we have used the SNAP [31] library. All simulations were carried out using *Grid'5000* [32] platform.

A. Models

In order to implement our clustering approach in a realistic way, we use standard models for representing both the energy consumption and the network structure.

1) Energetic model: To model the energy consumption for a node when it sends/receives a message, we use the first order radio model proposed by Heinzelman et *al.* [33] and used in many other studies [4], [34], [35]. A sensor node consumes E_{Tx} amount of energy to transmit one l-bits message over a distance d (in meters). As shown in equation (1), when the distance is higher than a certain threshold d_0, a node consumes more energy according to a different energetic consumption model.

$$E_{Tx}(l,d) = \begin{cases} l * E_{elec} + l * \varepsilon_{fs} * d^2, & \text{if d} < d_0; \\ l * E_{elec} + l * \varepsilon_{mp} * d^4, & \text{if d} \geq d_0. \end{cases} \quad (1)$$

Each sensor node will consume E_{Rx} amount energy when receiving a message, as shown in equation (2).

$$E_{Rx}(l) = l * E_{elec} \quad (2)$$

Parameters values used in equations (1) and (2) to model energy are summarized in Table II.

TABLE II
RADIO MODELING PARAMETERS

Parameter	definition	Value
E_{elec}	Energy dissipation rate to run radio	$50nJ/bit$
ε_{fs}	Free space model of transmitter amplifier	$10pJ/bit/m^2$
ε_{mp}	Multi-path model of transmitter amplifier	$0.0013pJ/bit/m^4$
d_0	Distance threshold	$\sqrt{\varepsilon_{fs}/\varepsilon_{mp}}$

2) Network model: In our experimental studies, we consider networks represented by an arbitrary random graph based on a Poisson process with $\lambda > 1$ for all network sizes. In fact, random graphs based on a Poisson process provide a better representation for WSNs. It is used in many studies like [18], [36], [37], [38], [39]. Nodes in the network are distributed uniformly at random as per a homogeneous spatial Poisson process of intensity λ in two-dimensional plane. We model our network by an undirected graph $G = (V, E)$ following standard models for distributed systems given in [40], [41]. $V = n$ is the set of network nodes and E represents all existing connections between nodes. Each node u of the network has a unique identifier noted id_u such that $0 \leq id_u \leq n-1$. An edge exists if and only if the distance between two nodes is less or equal than a fixed radius $r \leq d_0$. This r represents the radio transmission range, which depends on wireless channel characteristics including transmission power. Accordingly, the neighborhood of a node u is defined by the set of nodes that are inside a circle with center at u and radius r and it is denoted by $N_r(u) = N_u = \{\forall v \in V \setminus \{u\} \mid d_{(u,v)} \leq r\}$. The degree of a node u in G is the number of edges that are connected to u, and it is equal to $deg(u) = |N_r(u)|$.

B. Testbed

The parameters used in our simulations are summarized in Table III. In all simulations, a 99% confidence interval I_c is computed for each average value represented in the curves. These intervals are plotted as error bars and computed according to this equation: $I_c = [\overline{x} - t_\alpha \frac{\delta}{\sqrt{n}}; \overline{x} + t_\alpha \frac{\delta}{\sqrt{n}}]$, where n is the population length, $\overline{x}$ is the average value, δ is the standard deviation, and finally, t_α has a fixed value of 2.58 in the case of 99% interval.

TABLE III
SIMULATION PARAMETERS

Parameter	**Value**
Message size	2000 bits
distance between 2 nodes	100 meters
Initial Energy $\mathcal{E}_i^{init}$	{1,2,3} Joules
Ideal degree	{6,10,12,20}
Energy threshold	{0.1,0.05} %
Number of nodes	[100,1000]
Random graph model	Poisson process
λ parameter	[2,11]
k parameter	[1,10]
Number of simulations for each network size	100

C. Simulation results: evaluation of cluster-head election criteria

In this section, we present a performance evaluation of cluster-head election criteria. For each cluster-head election criterion, the following performance parameters are assessed.

- Total exchanged messages ($\mathcal{M}_{total}$): It is defined as the total number of exchanged messages in the whole network until the formation of stable clusters.

$$\mathcal{M}_{total} = \sum_{i=0}^{n-1} \mathcal{M}_i^{Send}$$

  Where $\mathcal{M}_i^{Send}$ is the total number of messages send by sensor node i and n represents the network size.
- Total energy consumption ($\mathcal{E}_{total}$) : It is defined as the energy consumption necessary to the clusters formation.

$$\mathcal{E}_{total} = \sum_{i=0}^{n-1} (\mathcal{E}_i^{init} - \mathcal{E}_i^{av})$$

  Where $\mathcal{E}_i^{init}$ is the initial energy of sensor node i and $\mathcal{E}_i^{av}$ is the available energy of node i at the end of clustering.
- Number of clusters: It is defined as the percentage of formed clusters according to the network size.

Theses performances are evaluated according λ and k parameters.

1) Communication cost (messages): We start the evaluation of our protocol by measuring the necessary communication cost in terms of exchanged messages to achieve the clustering procedure.

In the set of experiments described in Fig. 7, we calculate the communication cost according λ (Fig. 7(a), Fig. 7(c) and Fig. 7(e)) and k (Fig. 7(b), Fig. 7(d) and Fig. 7(f)) parameters for each cluster-head election criterion. These simulations are based on the same network topology for each value of λ and k parameters.

As illustrated in $3D$ curves show in Fig. 7(a), Fig. 7(b), Fig. 7(c), Fig. 7(d), Fig. 7(e) and Fig. 7(f), we observe that, for each cluster-head election criterion, the total number of exchanged messages increases linearly together with the number of nodes in the network. Indeed, the increase in network size entails more communications. However, Fig. 7 shows that our protocol is scalable. Furthermore, λ and k parameters do not affect the amount of generated messages by our protocol. The main reason is that our algorithm is based only on information from neighboring nodes at distance 1 to build k-hops clusters.

Experiments in Fig. 7 show that the clustering based on the criterion of ID generates less messages. Fig. 5 shows the gain of the ID criterion compared to Degree and Energy criteria according λ parameter and $k = 2$. The criterion of ID reduces the communication cost between 7.5% and 10.2% compared to Degree criterion and between 22.6% and 32.1% compared to Energy criterion. The main reason is that the ID criterion brings greater stability during the clustering phase. In addition, the ID criterion is simpler and deterministic

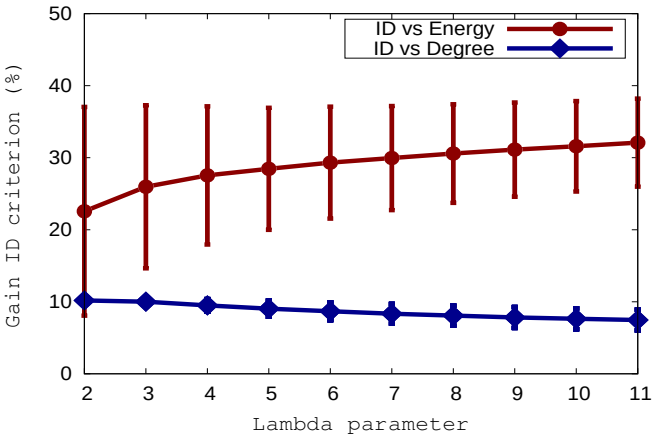


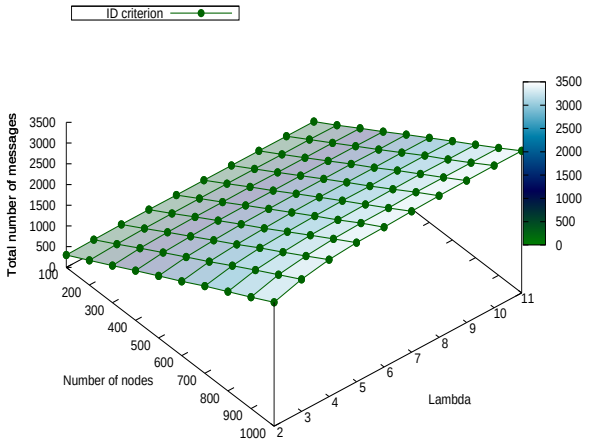
Figure 5. Communication cost reduction of ID criterion (%)

compared to the criteria of degree or energy. Indeed, for the Degree criterion, it is necessary for nodes to receive a message from their neighbors to calculate their degree. Then, the degree is sent by broadcast and after that, clustering phase begins. This is expensive in terms of messages. Also, the residual energy criterion generates more messages compared to the ID and Degree criteria. As energy level is a parameter which decreases during the clustering phase, it provides less stability and requires more messages to reach a stable state in the entire network. Note that we observe the same gain in terms of energy consumption of the ID criterion compared to Degree and Energy criteria.

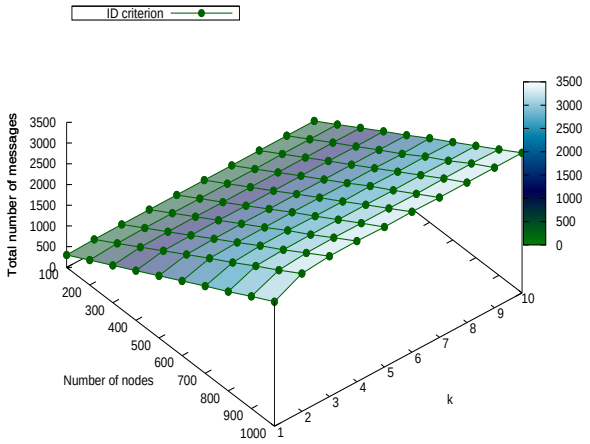
2) Energy consumption: In the second set of experiments shown in Fig. 8, we have measured the energy consumption required for building clusters in the entire network according network size and λ or k parameters.

As illustrated in $3D$ curves described in Fig. 8(a), Fig. 8(b), Fig. 8(c), Fig. 8(d), Fig. 8(e) and Fig. 8(f), we note that for each cluster-head election criterion, the energy consumption increases linearly together with the number of nodes in the network. The main reason is that the energy consumption is a linear function following the communication cost. However, λ and k parameters do not affect the amount the energy consumption required for building clusters. In fact, as illustrated in experiments show in Fig. 7, the communication cost does not depending on λ and k parameters.

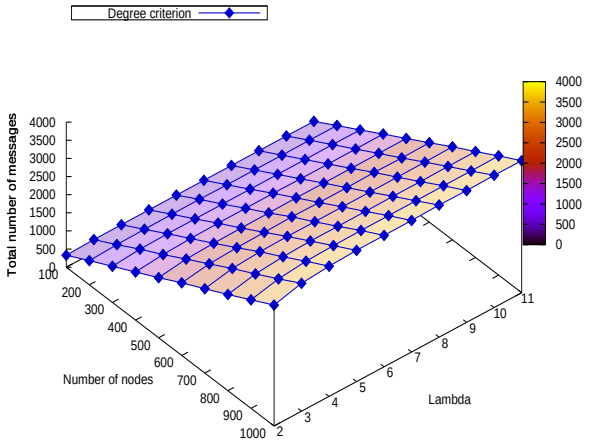
Experiments illustrated in Fig. 8 show that the clustering based on the ID criterion requires less energy consumption during the clustering phase. Indeed, results illustrated in Fig. 7 show that both Degree and Energy criteria generate more messages than ID criterion during the clusters formation. However, communications are the major source of energy consumption in WSNs. Moreover, ID criterion reduces energy consumption required to the clusters formation. Fig. 6 shows the gain of ID criterion compared to Degree and Energy criteria according k parameter and $\lambda = 6$. The ID criterion reduces the energy consumption between 7.3% and 9.7% compared to Degree criterion and between 18.1% and 35.1% compared to Energy



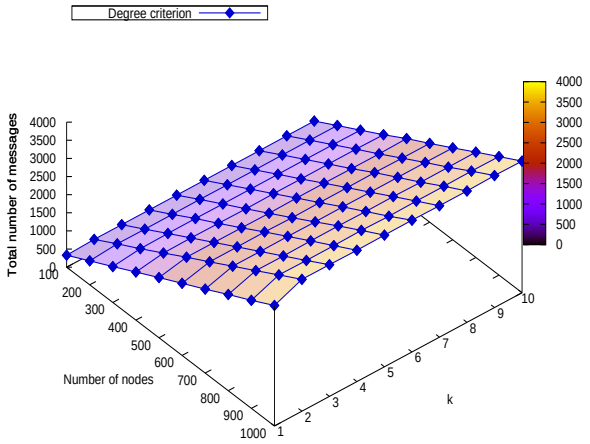
(a) ID criterion according λ parameter



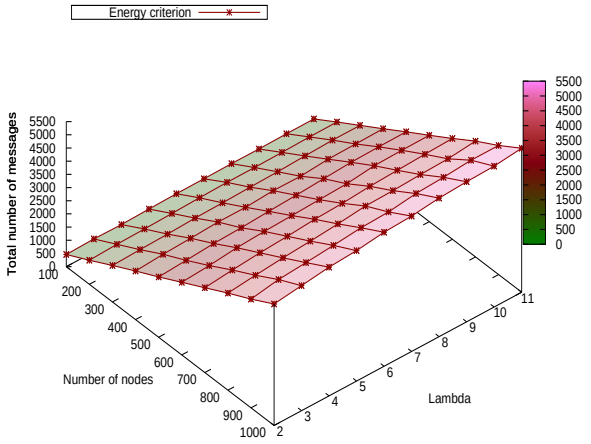
(b) ID criterion according k parameter



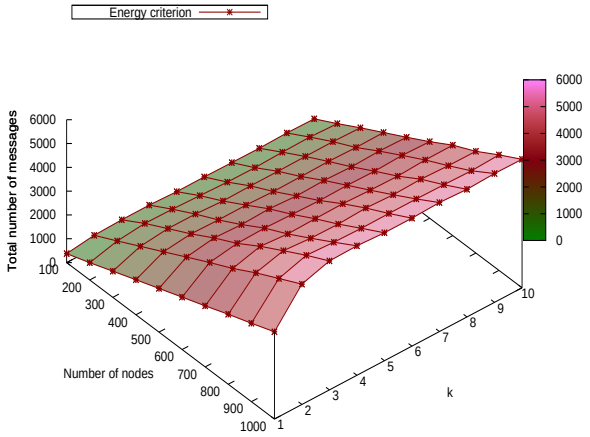
(c) Degree criterion according λ parameter



(d) Degree criterion according k parameter

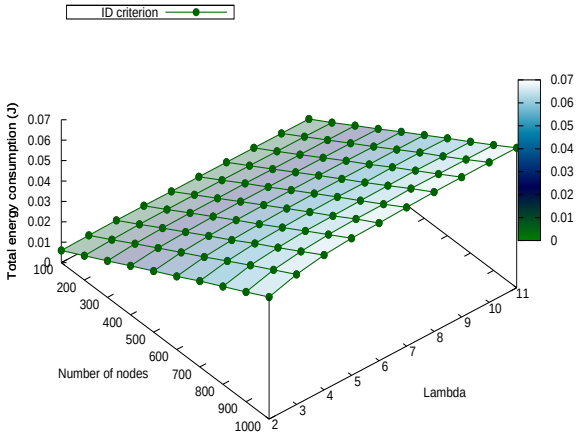


(e) Energy criterion according λ parameter

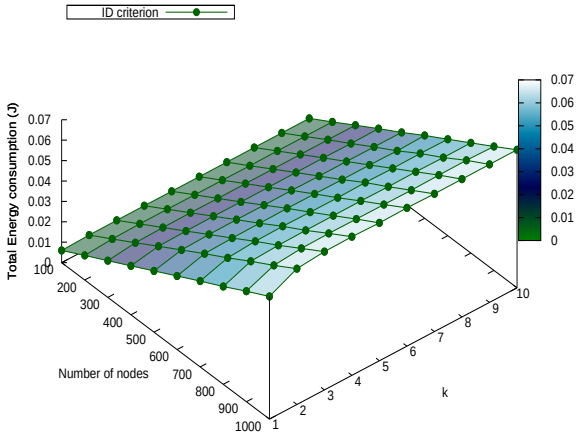


(f) Energy criterion according k parameter

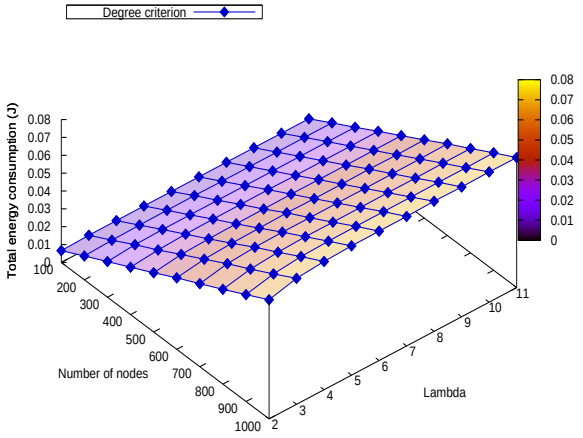
Figure 7. Total exchanged messages according λ and k parameters



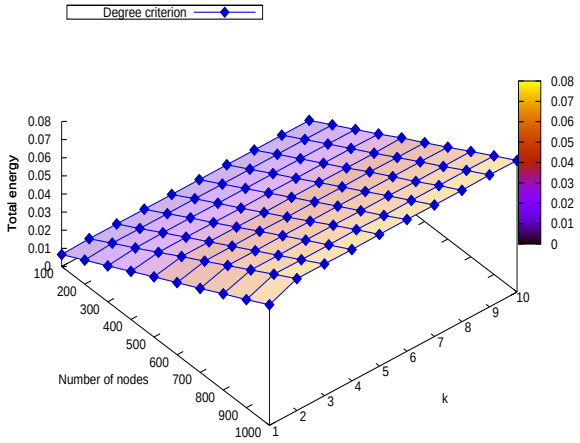
(a) ID criterion according λ parameter



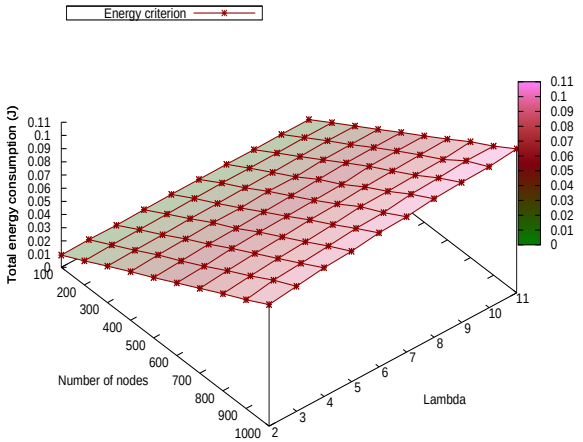
(b) ID criterion according k parameter



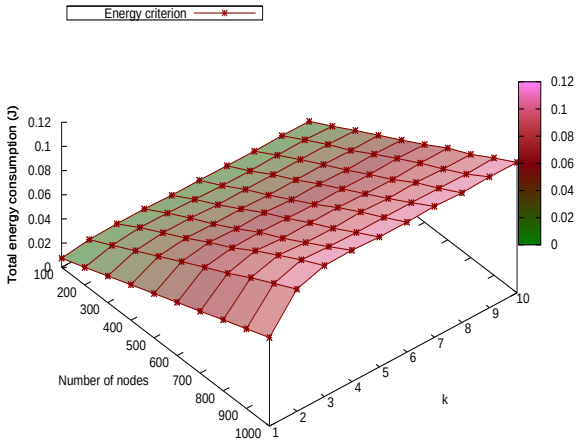
(c) Degree criterion according λ parameter



(d) Degree criterion according k parameter

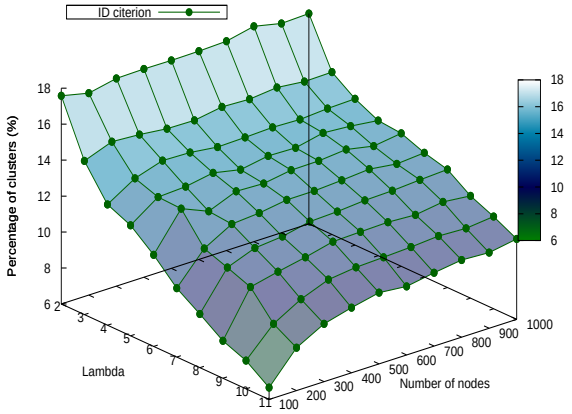


(e) Energy criterion according λ parameter

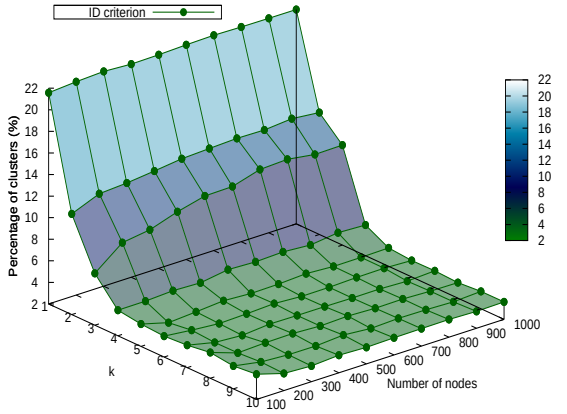


(f) Energy criterion according k parameter

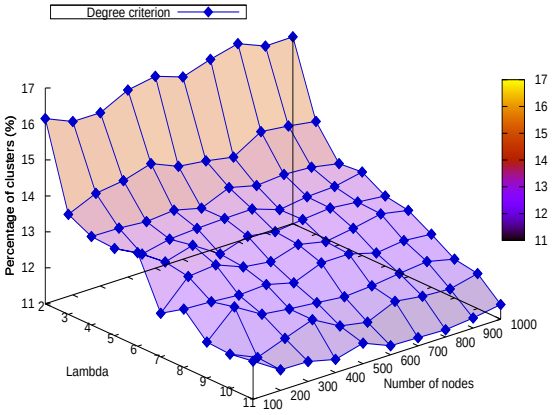
Figure 8. Total energy consumption according λ and k parameters



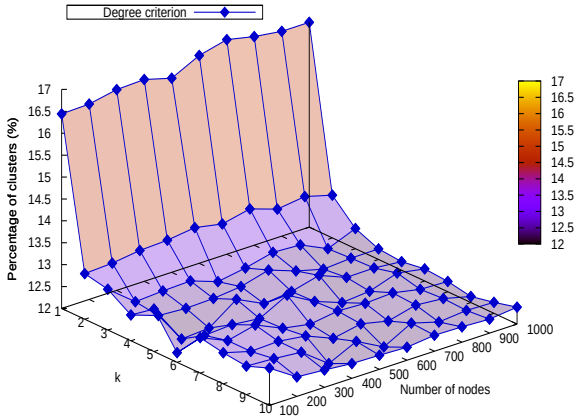
(a) ID criterion according λ parameter



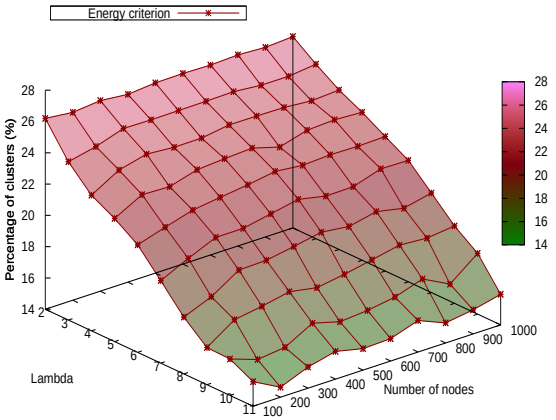
(b) ID criterion according k parameter



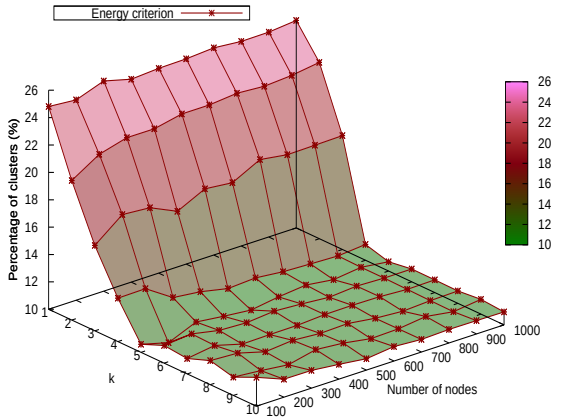
(c) Degree criterion according λ parameter



(d) Degree criterion according k parameter



(e) Energy criterion according λ parameter



(f) Energy criterion according k parameter

Figure 9. Percentage of number of clusters according λ and k parameters

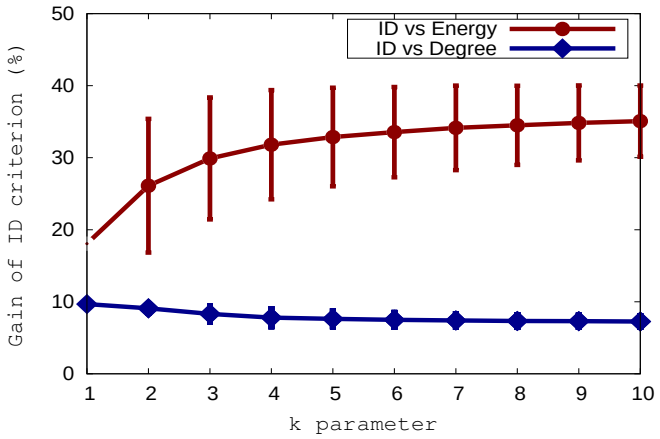


Figure 6. Energy consumption reduction of ID criterion (%)

criterion. Note that we observe the same gain in terms of communication cost for the ID criterion compared to Degree and Energy criteria.

3) Number of clusters : The number of clusters build by our protocol for each cluster-head election criterion is illustrated by the set of experiments described in Fig. 9. These $3D$ curves reflect the percentage of formed clusters according network and λ or k parameters.

In Fig. 9(a), Fig. 9(c) and Fig. 9(e), we set $k = 2$ and we vary arbitrary the value of λ parameter between 2 and 11. Firstly, we observe that for each cluster-head election criterion, the percentage of clusters build does not significantly vary according the network size for each fixed value of λ parameter. Therefore, our approach is scalable in term number of clusters. Secondly, for each fixed network size, the percentage of clusters decreases as the value of λ parameter increases. In fact, the λ parameter represents the average number of neighbors. Thus, network density increases as the λ parameter increases. Therefore, clusters size increases, implying a reduction of the number of clusters. Note that the ID criterion provides a better distribution of clusters (between 6% and 18%) compared to Degree and Energy criteria. The main reason is that ID criterion provides more stability.

In Fig. 9(b), Fig. 9(d) and Fig. 9(f), we set $\lambda = 6$ and we arbitrary vary the value of k parameter between 1 and 10. Firstly, we observe that for each cluster-head election criterion, the percentage of clusters build does not significantly vary according the network size for each fixed value of k parameter. Therefore, our approach is scalable in term number of clusters. Secondly, for each fixed network size, the percentage of clusters decreases significantly as the value of k increases. In fact, if the k parameter increases, clusters of larger diameter are constructed. This implies that clusters size is larger. Thus, a decrease in the percentage of clusters built. Note that, values of k parameter that provide the better distribution of clusters are comprised between 2 and 4. Beyond, we obtain large clusters that will not be easy to manage by the cluster-head.

4) Impact of highest and Ideal degree: To evaluate the impact of highest and Ideal degree as studied in Section III-B2, we arbitrary fix $\triangle_d$ to 6, 10, 12, and 20 and then we evaluate energy consumption. Note that in the set of experiments shows in Fig. 10, we fix $k = 2$ and $\lambda = 6$. We observe a slight decrease in the energy consumption for ideal degree fixed to 6 compared to highest degree as illustrated in Fig. 10. In fact, the Ideal degree fixed is equal to the λ parameter. As the λ parameter represents the average number of neighbors in the whole network, a Ideal degree equal to the λ parameter reduces communications required during the clusters formation implying slight decrease in energy consumption.

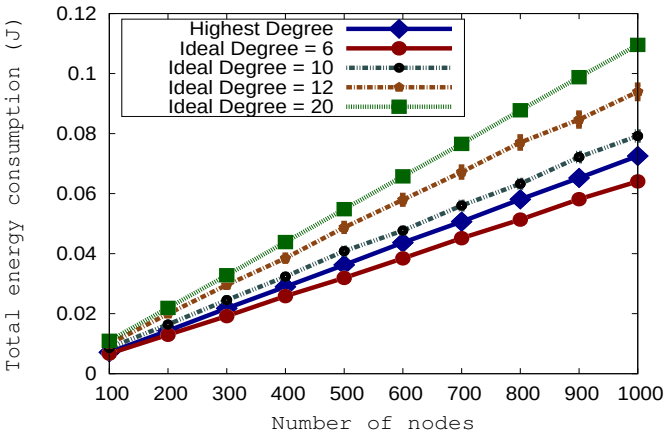


Figure 10. Energy consumption under highest and ideal degrees

On the other hand, we observe an increase in the energy consumption for ideal degree fixed at 12, 15 and 20. The main reason is that nodes attempt to join the cluster-head that is the node minimizing its distance to this ideal degree ρ ($\triangle_d = |D - \rho|$). This leads an increase of communications required during the clusters formation, implying at the same time an increase of energy consumption. The major advantage of this method is to allow the setting of the number nodes managed by cluster-head.

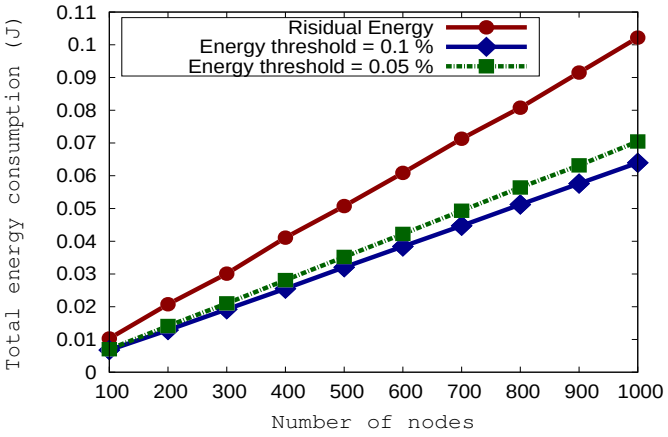
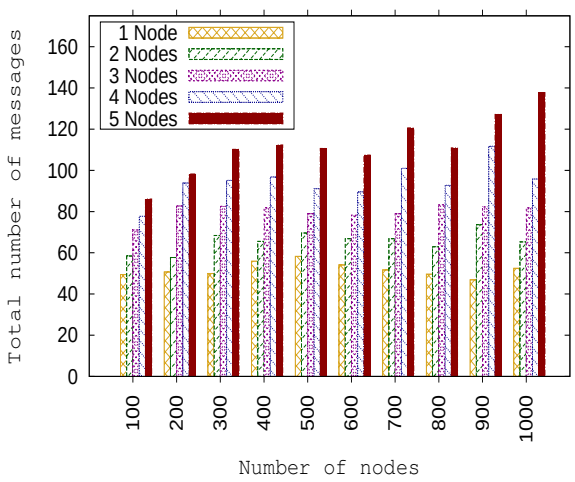
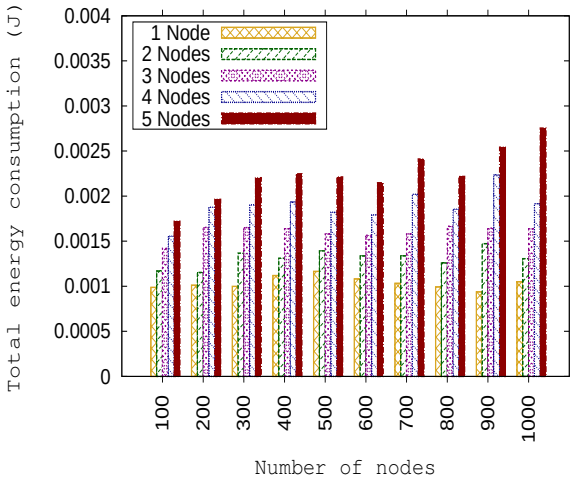


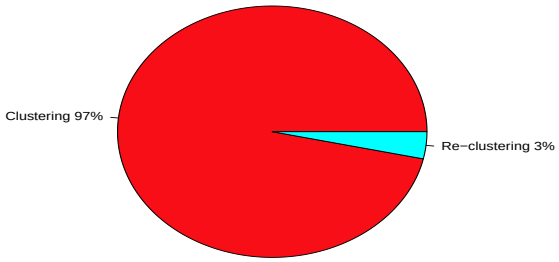
Figure 11. Residual energy vs Energy threshold



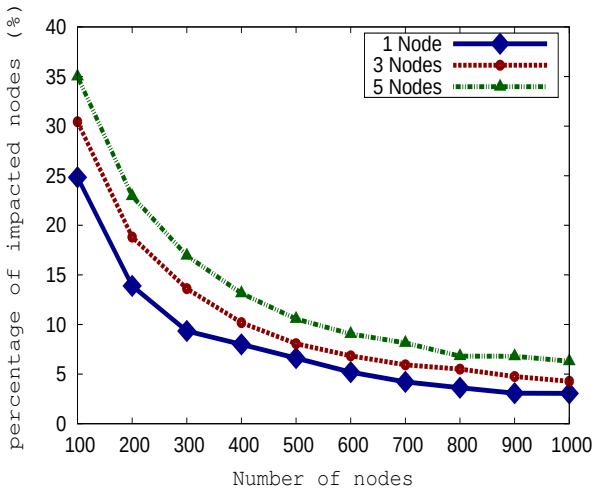
(a) Supplementary communication cost



(b) Supplementary energy consumption



(c) Impact of 5 nodes: percentage of supplementary cost



(d) Percentage of impacted nodes

Figure 12. Fault-tolerant in the case of 1, 2, 3, 4 and 5 disappeared nodes

5) Impact of residual energy or energy threshold: As the main problem with the criterion of energy is its volatility, we fix energy threshold to limit abrupt changes of nodes when their energy CHs decreases substantially. We fixed the energy threshold to 0.1% and 0.05% and we evaluate both energy consumption. Fig. 11 shows that energy threshold reduces energy consumption during the clustering phase. Indeed, nodes no longer change after a slight decrease of their energy CHs. This entails less messages exchanged and less energy consumption.

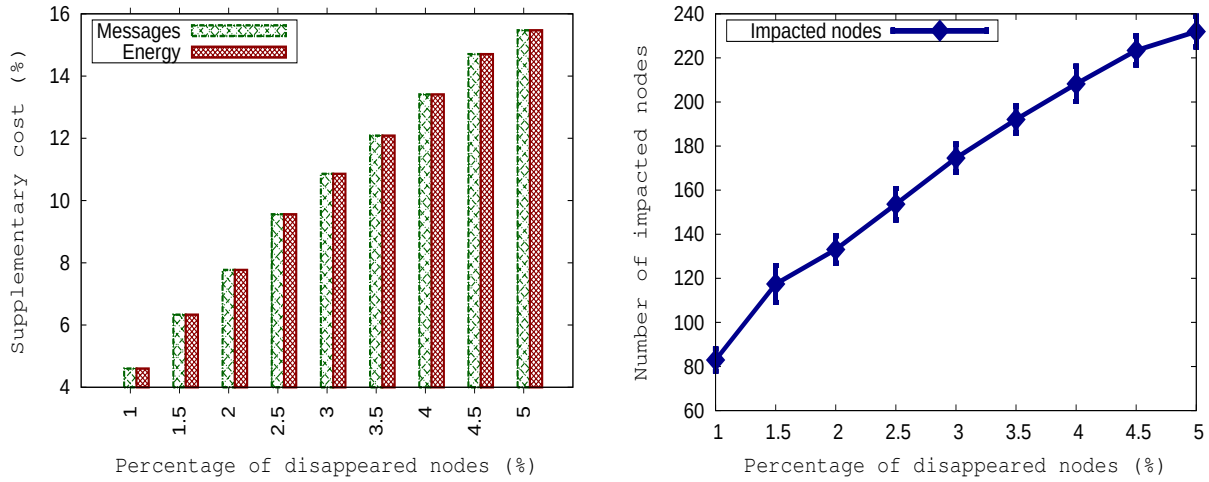
D. Simulation results: fault-tolerant evaluation

In this section, we study by simulation the robustness or our approach again nodes failure. To do this, we consider only the ID criteria of our protocol.

Firstly, we vary the network size between 100 and 1000 nodes. For each network size, after stabilization (i.e., formation of stable clusters in whole network), we randomly disappear 1, 2, 3, 4 and 5 nodes. Thus, the fault-tolerance mechanism is triggered by starting the re-clustering process. At the end of the re-clustering process, we evaluate the supplementary communication cost, energy consumption and percentage of impacted nodes. For each network size, we compute for each metric the average as the average of all values corresponding to 100 simulations results with 99% fixed as confidence interval.

Fig. 12(a) shows the supplementary communication cost at the end of the re-clustering process according the network size. We observe that, the disappearance of 1 until 5 nodes and according network size, generates on average between 50 and 150 supplementary messages in whole network. Fig. 12(b), shows a supplementary energy consumption between 1 mJ and 4 mJ. We remark that the energy consumption follows the same pattern as the communication cost. The main reason is that the energy consumption is a linear function following the communication cost.

In order to evaluate the impact of re-clustering, in Fig. 12(c) we calculate the percentage of re-clustering cost compared to



(a) Supplementary cost: messages and energy

(b) Number of impacted nodes in a network with 1000 nodes

Figure 13. Fault-tolerant according the percentage of disappeared nodes between 1% and 5%

clustering cost. We note that, the re-clustering cost (in terms of communication cost and energy consumption) represents 3% of resource consumption compared to the clustering cost. In fact, with our fault-tolerance mechanism, as illustrated in examples shown in Fig. 3 and Fig. 4 in the Section III-C, the occurrence of fault impacts generally the cluster where the fault has occurred and eventually adjacent clusters. This result is consolidated through Fig. 12(d), where we have estimated the percentage of impacted nodes compared to the network size. We say that a node u is impacted if only if, one of its local variables (cl_u, $statut_u$ or $dist_{(u,CH_u)}$) undergoes a modification caused by the disappearance of a node v. Fig. 12(d) shows that the disappearance of 5 node in the network size 1000 impacts around 6% of nodes.

To better observe the impact of re-clustering as illustrated in Fig. 13, we set a network size at 1000 nodes and we randomly disappear between 1% and 5% of nodes in the network. At the end of re-clustering process, we evaluate the supplementary communication cost, the energy consumption and the percentage of impacted nodes. Fig. 13(a) shows the supplementary re-clustering cost in terms of exchanged messages and energy consumption compared to clustering cost. We observe that the disappearance until 5% of nodes leads an additional cost in terms of exchanged messages and energy consumption of 15.5%. The main reason is due to the fact that the re-clustering caused by disappearance of 5% nodes does not impact the entire network. In fact, as illustrated in Fig. 13(b), disappearance of 5% nodes impacts around $1/4$ of total number of nodes in the network.

VI. Conclusion and Perspectives

In this paper, we proposed a self-stabilizing distributed energy-efficient and fault-tolerant clustering protocol for heterogeneous wireless sensor networks. This protocol prolongs the network lifetime by minimizing the energy consumption involved in the exchanged of messages. It can be used under different CHs election methods like those investigated in this work. Moreover, our proposed protocol is fault tolerance and adapted to topological changes. We have also compared our algorithm with some of most referenced self-stabilizing solutions.

Simulation results show that in terms of number of messages, energy consumption and clusters distribution, it is better to use the Highest-ID metric for electing CHs. Furthermore, after the occurrence of faults, the re-clustering cost is minimal compared to the clustering cost and faults do not affect the entire network.

As future work, we plan to propose a routing process based on our clustering approach.

Acknowledgment

This research was supported by Regional Council of Champagne-Ardenne and European Regional Development Fund through the CPER CapSec ROFICA project. Simulations are executed on Grid'5000 experimental testbed hosted at the ROMEO HPC Center. We express our thanks to Regional Council of Champagne-Ardenne, European Regional Development Fund, Crid'5000 and ROMEO HPC Center. And finally, we wish to thank the reviewers and editors for their valuable suggestions and expert comments that help improve the contents of this paper.

References

[1] M. Ba, O. Flauzac, R. Makhloufi, F. Nolot, and I. Niang, "Evaluation Study of Self-Stabilizing Cluster-Head Election Criteria in WSNs," in *Proceedings of the 6th International Conference on Communication Theory, Reliability, and Quality of Service*, CTRQ'13, pp. 64–69, 2013.

[2] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 38, no. 4, pp. 393 – 422, 2002.

[3] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[4] J. Yu, Y. Qi, G. Wang, and X. Gu, "A cluster-based routing protocol for wireless sensor networks with nonuniform node distribution," *AEU - International Journal of Electronics and Communications*, vol. 66, no. 1, pp. 54 – 61, 2012.

[5] O. Younis and S. Fahmy, "HEED: a Hybrid, Energy-Efficient, Distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[6] C. Johnen and L. Nguyen, "Self-stabilizing weight-based clustering algorithm for ad hoc sensor networks," in *Proceedings of the 2nd International Conference on Algorithmic Aspects of Wireless Sensor Networks*, ALGOSENSORS'06, pp. 83–94, 2006.

[7] H. Liu, P.-J. Wan, and X. Jia, "Fault-tolerant relay node placement in wireless sensor networks," in *Computing and Combinatorics*, Lecture Notes in Computer Science, vol. 3595, pp. 230–239, 2005.

[8] W. Zhang, G. Xue, and S. Misra, "Fault-tolerant relay node placement in wireless sensor networks: Problems and algorithms," in *Proceedings of the 26th IEEE International Conference on Computer Communications*, INFOCOM'07, pp. 1649–1657, 2007.

[9] B. Hao, J. Tang, and G. Xue, "Fault-tolerant relay node placement in wireless sensor networks: formulation and approximation," in *Proceedings of the Workshop on High Performance Switching and Routing*, HPSR'04, pp. 246–250, 2004.

[10] H. Liu, A. Nayak, and I. Stojmenovi, "Fault-tolerant algorithms/protocols in wireless sensor networks," in *Guide to Wireless Sensor Networks*, Computer Communications and Networks, pp. 261–291, 2009.

[11] C. Johnen and F. Mekhaldi, "Self-stabilization versus robust self-stabilization for clustering in ad-hoc network," in *Proceedings of the 17th international conference on Parallel processing*, Euro-Par'11, pp. 117–129, 2011.

[12] E. Caron, A. K. Datta, B. Depardon, and L. L. Larmore, "A self-stabilizing k-clustering algorithm for weighted graphs," *Journal of Parallel and Distributed Computing*, vol. 70, no. 11, pp. 1159–1173, 2010.

[13] A. K. Datta, S. Devismes, and L. L. Larmore, "A Self-Stabilizing O(n)-Round k-Clustering Algorithm," in *Proceedings of the 28th IEEE International Symposium on Reliable Distributed Systems*, SRDS'09, pp. 147–155, 2009.

[14] O. Flauzac, B. S. Haggar, and F. Nolot, "Self-stabilizing clustering algorithm for ad hoc networks," in *Proceedings of the 5th International Conference on Wireless and Mobile Communications*, ICWMC '09, pp. 24–29, 2009.

[15] C. Johnen and L. Nguyen, "Self-stabilizing construction of bounded size clusters," *International Symposium on Parallel and Distributed Processing with Applications*, pp. 43–50, 2008.

[16] C. Johnen and L. H. Nguyen, "Robust self-stabilizing weight-based clustering algorithm," *Theoretical Computer Science*, vol. 410, no. 6-7, pp. 581 – 594, 2009.

[17] N. Mitton, A. Busson, and E. Fleury, "Self-organization in large scale ad hoc networks," in *Proceedings of the 3rd Annual Workshop Mediterranean Ad Hoc Networking*, MED-HOC-NET, 2004.

[18] N. Mitton, E. Fleury, I. Guerin Lassous, and S. Tixeuil, "Self-stabilization in self-organized multihop wireless networks," in *Proceedings of the 2nd International Workshop on Wireless Ad Hoc Networking*, ICDCSW '05, pp. 909–915, 2005.

[19] A. Amis, R. Prakash, T. Vuong, and D. Huynh, "Max-min d-cluster formation in wireless ad hoc networks," in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, INFOCOM'00, pp. 32–41, 2000.

[20] M. Ba, O. Flauzac, B. S. Haggar, F. Nolot, and I. Niang, "Self-Stabilizing k-hops Clustering Algorithm for Wireless Ad Hoc Networks," in *Proceedings of the 7th ACM International Conference on Ubiquitous Information Management and Communication*, IMCOM'13, pp. 38:1–38:10, 2013.

[21] D. J. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," *IEEE Transactions on Communications*, vol. 29, no. 11, pp. 1694–1701, 1981.

[22] Y.-F. Wen, T. A. F. Anderson, and D. M. W. Powers, "On energy-efficient aggregation routing and scheduling in IEEE 802.15.4-based wireless sensor networks," *Wireless Communications and Mobile Computing*, 2012.

[23] I. G. Shayeb, A. H. Hussein, and A. B. Nasoura, "A survey of clustering schemes for mobile ad-hoc network (MANET)," *American Journal of Scientific Research*, pp. 135–151, 2011.

[24] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," *Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[25] C.-C. Chiang, M. Gerla, and L. Zhang, "Forwarding Group Multicast Protocol (FGMP) for multihop, mobile wireless networks," *Cluster Computing*, vol. 1, no. 2, pp. 187–196, 1998.

[26] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," *Wireless Networks*, vol. 1, no. 3, pp. 255–265, 1995.

[27] W. Choi and M. Woo, "A Distributed Weighted Clustering Algorithm for Mobile Ad Hoc Network',' in *Porceedings of the International Conference on Internet and Web Applications and Services/Advanced*, AICT-ICIW '06, 2006.

[28] M. R. Brust, A. Andronache, and S. Rothkugel, "WACA: A hierarchical weighted clustering algorithm optimized for mobile hybrid networks," in *Proceedings of the 3rd International Conference on Wireless and Mobile Communications*, ICWMC'07, pp. 27–37, 2007.

[29] M. Ba, O. Flauzac, R. Makhloufi, F. Nolot, and I. Niang, "Comparison between self-stabilizing clustering algorithms in message-passing model," in *Proceedings of the 9th International Conference on Autonomic and Autonomous Systems*, ICAS'13, pp. 27–32, 2013.

[30] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems*, Simutools '08, pp. 60:1–60:10, 2008.

[31] SNAP: Stanford Network Analysis Platform. [Online]. Available: http://snap.stanford.edu, 2013.

[32] F. Cappello and et *al.*, "Grid'5000: A large scale and highly reconfigurable experimental grid testbed," *International Journal of High Performance Computing Applications*, vol. 20, no. 4, pp. 481–494, 2006.

[33] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000.

[34] J. Wang, J.-U. Kim, L. Shu, Y. Niu, and S. Lee, "A distance-based energy aware routing algorithm for wireless sensor networks," *Sensors*, vol. 10, no. 10, 2010.

[35] J. Wang, J. Cho, S. Lee, K. C. Chen, and Y. K. Lee, "Hop-based energy aware routing algorithm for wireless sensor networks," *IEICE Transactions on Communications*, vol. 93-B, no. 2, pp. 305–316, 2010.

[36] J. C. Hou, N. Li, and I. Stojmenovic, *Topology construction and maintenance in wireless sensor networks*, pp. 311–341, 2005.

[37] S. Bandyopadhyay and E. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proceedings of the 32nd Annual Joint Conference of the IEEE Computer and Communications*, INFOCOM'03, pp. 1713–1723, 2003.

[38] M. Nasim, S. Qaisar, and S. Lee, "An energy efficient cooperative hierarchical mimo clustering scheme for wireless sensor networks," *Sensors*, vol. 12, no. 1, pp. 92–114, 2011.

[39] J. Chen, C.-S. Kim, and F. Song, "A distributed clustering algorithm for voronoi cell-based large scale wireless sensor network," in *Proceedings of the 2010 International Conference on Communications and Mobile Computing*, CMC '10, pp. 209–213, 2010.

[40] H. Attiya and J. Welch, *Distributed computing: fundamentals, simulations, and advanced topics*, Wiley series on parallel and distributed computing, 2004.

[41] G. Tel, *Introduction to Distributed Algorithms*. Cambridge University Press, 2000.

Efficient and Accurate Label Propagation on Dynamic Graphs and Label Sets

Michele Covell and Shumeet Baluja
Google Research
Google Inc., Mountain View CA, USA
covell@google.com shumeet@google.com

Abstract—**Many web-based application areas must infer label distributions starting from a small set of sparse, noisy labels. Previous work has shown that graph-based propagation can be very effective at finding the best label distribution across nodes, starting from partial information and a weighted-connection graph. In their work on video recommendations, Baluja et al. showed high-quality results using *Adsorption*, a normalized propagation process. An important step in the original formulation of Adsorption was re-normalization of the label vectors associated with each node, between every propagation step. That interleaved normalization forced computation of all label distributions, in synchrony, in order to allow the normalization to be correctly determined. Interleaved normalization also prevented use of standard linear-algebra methods, like stabilized bi-conjugate gradient descent (*BiCGStab*) and Gaussian elimination. We show how to replace the interleaved normalization with a single pre-normalization, done once before the main propagation process starts, allowing use of selective label computation *(label slicing)* as well as large-matrix-solution methods. As a result, much larger graphs and label sets can be handled than in the original formulation and more accurate solutions can be found in fewer propagation steps. We further extend that work to handle graphs that change and expand over time. We report results from using pre-normalized Adsorption in topic labeling for web domains, using label slicing and BiCGStab. We also report results from using incremental updates on changing co-author network data. Finally, we discuss two options for handling mixed-sign (positive and negative) graphs and labels.**

Keywords - graph propagation, large-scale labeling, incremental connection-graph changes, stabilized bi-conjugate gradient descent, Gaussian elimination, topic discovery, web domains.

I. INTRODUCTION

Many different approaches have recently been proposed to label propagation across weighted graphs of nodes [1]-[7]. Applications include searching for, recommending, and advertising against image, audio, and video content. These labeling problems must handle millions of interconnected entities (users, domains, content segments) and thousands of competing labels (interests, tags, recommendations, topics). These applications share the characteristics of having a limited amount of label data, often of uneven quality, associated with a large graph of weighted connections between many nodes, some unlabeled and some partially labeled.

We build on the work done by Zhu and Ghahramani [3], Baluja et al. [2] and Covell and Baluja [1]. The Baluja paper [2] described *Adsorption*, a graph-based approach to estimating label distributions, which was applied to providing YouTube video recommendations. The resulting top-pick recommendation was more accurate than the next-best alternative algorithm for all users who had watched 3 or more previous videos, with accuracy improvements of up to 100% for the most frequent watchers. In Adsorption [1],[2], each node (e.g., each video for which we are building a recommendation list) has a limited capacity for labels (e.g., the proposed recommendations for that video). Baluja et al. [2] enforce this constraint by interleaving a normalization step at each node, in between every propagation step. Without this normalization, the solution is not guaranteed to converge.

The interleaved normalization step is needed for convergence but prevents label slicing: under the original formulation [2], we cannot find the estimated distribution of a subset of labels without solving for the full set of labels first. Furthermore, the interleaved normalization prevents the use of most standard linear-algebra techniques, such as Gaussian elimination of nodes that are not of direct interest (though they still are needed for their effect on the remainder of the graph). Additionally, methods for rapid convergence to the final solution, such as stabilized bi-conjugate gradient descent (BiCGStab), cannot be used in the original formulation.

We start the paper with a recap of the original Adsorption application and mathematical description [2], in Section II. This paper then reviews and expands on the work presented by Covell and Baluja [1] for pre-normalizing the Adsorption graph and label weights, such that there is no need for interleaved normalization (Section III). With this, we can use BiCGStab and Gaussian elimination. Our graph size contains more than 10 million nodes and 4 billion inter-connections (i.e., more than 10 million rows and more than 4 billion non-zero entries in the corresponding matrix), which is more than we can reasonably handle in straightforward implementations of these techniques. Instead, we use implementations of BiCGStab and Gaussian elimination in the MapReduce framework. We describe these implementations briefly, in Sections IV and V. In Section VI, we present our results on topic labeling of web domains,

using a graph based on shared keywords between pages across the domains.

In Section VII, we extend the pre-normalized framework [1] to handle fast updates for graphs with newly added nodes and changing connection weights between existing nodes. In Section VIII, we demonstrate this incremental-update approach on a co-author network, as seen originally seen in 2003 and then updated in 2005. Finally, in Section IX, we discuss two alternative approaches to handling negative associations.

II. ADSORPTION (WITH INTERLEAVED NORMALIZATION)

The original formulation of Adsorption [2] can be described as an iteration using two systems of equations:

$$\tilde{\underline{\underline{X}}}_{n+1} = \sigma \underline{\underline{X}}_n + \beta \underline{\underline{W}}\,\underline{\underline{X}}_n + \begin{bmatrix} \gamma \underline{\underline{L}} & \delta \underline{1} \end{bmatrix} \quad (1)$$

$$\left\{\underline{\underline{X}}_{n+1}\right\}_{i*} = \left\{\tilde{\underline{\underline{X}}}_{n+1}\right\}_{i*} \Big/ \left\| \left\{\tilde{\underline{\underline{X}}}_{n+1}\right\}_{i*} \right\|_1 \quad (2)$$

where double underlining indicates a matrix of values, a single underline is a vector, not-underlined values are scalars, and the tilde indicates a not-normalized set of values. The matrix $\underline{\underline{W}}$ holds the connection weights with row i giving the incoming connections into the i'th node. This matrix often is symmetric, to start with, but this property is not required and will be given up later to allow for pre-normalization. The matrix $\underline{\underline{L}}$ holds the weights of the *injection label* information. These are often noisy or incomplete label sets based on some prior information, with the graph propagation as a way to improve and expand these label sets. In $\underline{\underline{L}}$, each label is associated with a column and the weights for the injection labels for the i'th node of the graph are in the i'th row of the matrix. In addition to the true labels, in $\underline{\underline{L}}$, Baluja et al. [2] add an *abandonment label*, represented in Equation (1) by the appended column $\delta \underline{1}$. The scalar δ can be thought of in many different ways: as the loss in certainty about any of the labels that are propagated for one hop in the graph; as the number of random walks through the graph that end with "abandonment", giving no final label set; as the regularization margin in the system of equations. The other scalars (σ, β, and γ) allow graph-wide balancing of the previous (same-node) labels, of the propagated neighbors' labels, and of the injection labels. Finally, the matrix $\underline{\underline{X}}_{\underline{\underline{n}}}$ is the label distribution estimate, with the i'th row containing the estimated labels for the i'th node, including as the last column the abandonment label. In this context, the node's abandonment weight provides a measure, at that node, of the label uncertainty.

Equation (1) creates a new *un-normalized* estimate of the steady-state label distribution across all the nodes using a weighted combination of the previous normalized estimate for the distribution ($\underline{\underline{X}}_{\underline{\underline{n}}}$), of a graph-weighted propagated version of that same distribution ($\underline{\underline{W}}\,\underline{\underline{X}}_{\underline{\underline{n}}}$), of injection labels ($\underline{\underline{L}}$), and of the abandonment label (δ). Equation (2) provides a normalized estimate of the label distribution, by dividing each row of the estimate from Equation (1) by the L_1 norm of the full label set, including the abandonment label.

Iterating over Equations (1) and (2) together is guaranteed to converge to a stable steady-state solution, as long as δ is greater than 0. Baluja et al. [2] used this algorithm to successfully provide video recommendations that, using a top-pick-accuracy measure, outperformed alternative approaches. Our goal is to provide a formulation for the same Adsorption algorithm that does not require per-propagation-step normalization, allowing us to use label slicing and standard linear-algebra tools.

III. PRE-NORMALIZED ADSORPTION

We achieve our goal of pre-normalized Adsorption by first assuming that all associations in our graph and in our label injection are non-negative. Specifically: $sign\left(\left\{\underline{\underline{X}}_n\right\}_{ij}\right) \geq 0$, $sign\left(\left\{\underline{\underline{W}}\right\}_{ij}\right) \geq 0$, and $sign\left(\left\{\underline{\underline{L}}\right\}_{ij}\right) \geq 0$.

This non-negative assumption works well with the partial-information applications that are the most common ones in large-graph labeling formulations: for example, in video recommendation, we can say that two videos are often watched together, within a single viewing session, but it is much more difficult to say that two videos are negatively associated (that watching one means you are significantly less likely to watch the other), since we seldom have enough training data to make such an assertion with any confidence.

For those applications where we do have confidence in negative label-to-node associations (negative values in $\underline{\underline{L}}$), we can handle these by introducing a negated label column and using positive associations with the negated label where we would have otherwise used negative associations with the positive label. Handling negative node-to-node connections (negative values in $\underline{\underline{W}}$) is also possible. We go over all of these cases in more detail in Section IX.

Assuming we have non-negative values in our component matrices, we can consider the denominator of Equation (2) in more detail:

$$\left\| \left\{\tilde{\underline{\underline{X}}}_{n+1}\right\}_{i*} \right\|_1 = \left\| \left\{ \left(\sigma \underline{\underline{I}} + \beta \underline{\underline{W}}\right) \underline{\underline{X}}_n + \begin{bmatrix} \gamma \underline{\underline{L}} & \delta \underline{1} \end{bmatrix} \right\}_{i*} \right\|_1 \quad (3)$$

$$= \sum_j \left(\sum_k \left\{\sigma \underline{\underline{I}} + \beta \underline{\underline{W}}\right\}_{ik} \left\{\underline{\underline{X}}_n\right\}_{kj} \right) + \sum_j \left\{ \begin{bmatrix} \gamma \underline{\underline{L}} & \delta \underline{1} \end{bmatrix} \right\}_{ij} \quad (4)$$

$$= \sum_k \left\{\sigma \underline{\underline{I}} + \beta \underline{\underline{W}}\right\}_{ik} \sum_j \left\{\underline{\underline{X}}_n\right\}_{kj} + \gamma \sum_j \left\{\underline{\underline{L}}\right\}_{ij} + \delta \quad (5)$$

$$= \sum_k \left\{\sigma \underline{\underline{I}} + \beta \underline{\underline{W}}\right\}_{ik} + \gamma \left\| \left\{\underline{\underline{L}}\right\}_{i*} \right\|_1 + \delta \quad (6)$$

$$= \sigma + \beta \left\| \left\{\underline{\underline{W}}\right\}_{i*} \right\|_1 + \gamma \left\| \left\{\underline{\underline{L}}\right\}_{i*} \right\|_1 + \delta \quad (7)$$

Equation (3) simply provides the expansion of the L_1 row norm using the propagation Equation (1). Equation (4) makes use of the non-negativity conditions that we are requiring, in order to remove the absolute values implied by the L_1 norm and expands the norm summation, as well as the summation implicit in the $\underline{\underline{W}}\,\underline{\underline{X}}_{\underline{\underline{n}}}$ matrix multiply. Equation (5) swaps the order of summation, allowing us to make use of the unit L_1 row norm for $\underline{\underline{X}}_{\underline{\underline{n}}}$ in Equation (6). Simplifying

the summations and noting the use of the row-norm definitions for $\underline{\underline{L}}$ and $\underline{\underline{W}}$ finally results in Equation (7).

The useful property of Equation (7) is that $\left\| \left\{ \underline{\underline{\tilde{X}}}_{n+1} \right\}_{i^*} \right\|_1$ depends only the initial combination weights and the row norms of $\underline{\underline{L}}$ and $\underline{\underline{W}}$. We can use this property to pre-normalize by first defining

$$\lambda_i = \sigma + \beta \left\| \left\{ \underline{\underline{W}} \right\}_{i^*} \right\|_1 + \gamma \left\| \left\{ \underline{\underline{L}} \right\}_{i^*} \right\|_1 + \delta \quad (8)$$

$$\hat{\sigma}_i = \sigma / \lambda_i \qquad \underline{\underline{\hat{\sigma}}} = diag(\hat{\sigma}_i) \quad (9)$$

$$\hat{\beta}_i = \beta \left\| \left\{ \underline{\underline{W}} \right\}_{i^*} \right\|_1 / \lambda_i \qquad \underline{\underline{\hat{\beta}}} = diag(\hat{\beta}_i) \quad (10)$$

$$\hat{\gamma}_i = \gamma \left\| \left\{ \underline{\underline{L}} \right\}_{i^*} \right\|_1 / \lambda_i \qquad \underline{\underline{\hat{\gamma}}} = diag(\hat{\gamma}_i) \quad (11)$$

$$\hat{\delta}_i = \delta / \lambda_i \qquad \underline{\hat{\delta}} = vec(\hat{\delta}_i) \quad (12)$$

and then using these new quantities in a pre-normalized Adsorption algorithm.

$$\underline{\underline{X}}_{n+1} = \underline{\underline{\hat{\sigma}}}\,\underline{\underline{X}}_n + \underline{\underline{\hat{\beta}}}\,\underline{\underline{W}}\,\underline{\underline{X}}_n + \left[\begin{array}{cc} \underline{\underline{\hat{\gamma}}}\,\underline{\underline{L}} & \underline{\hat{\delta}} \end{array} \right] \quad (13)$$

Note that direct use of Equation (13) is exactly the power-iteration approach to finding the solution (used in [2]) and will give the same solutions at every iteration as the combination of Equations (1) and (2): the pre-normalization has the exact same effect, even though it is only done once, as the interleaved normalizations. Equation (13), therefore, also is guaranteed to converge to a stable solution, just as the original Adsorption algorithm is guaranteed. The advantage is that we do not need to normalize at each step and, as a result, we can compute an incomplete set of labels, while still deriving the benefits of the full label set to limit belief within the set of labels that are interested in. This slicing directly reduces the computational costs by the same percentage as the percentage of dropped labels. Furthermore, with the use of Equation (13) as the system of equations for which we want a solution, we can use standard linear-algebra tools, like BiCGStab (for faster convergence) and Gaussian elimination (for shrinking our graph matrix). We discuss these algorithms and their large-graph implementations next.

IV. MAP-REDUCE FORMULATION OF STABILIZED BI-CONJUGATE GRADIENT DESCENT (BICGSTAB)

In [2], Baluja et al. implicitly use power iteration to solve their system of constraints. For symmetric systems of constraints, gradient-descent methods can find solutions in fewer iterations, for any given level of accuracy (as measured by the average residual error). However, due to the pre-normalization of Adsorption, we no longer have a symmetric matrix, and must move to bi-conjugate gradient approaches. Since the most direct generalization (biconjugate gradient descent) is not numerically stable, we focus on stabilized biconjugate gradient descent [8], which has been shown to converge more uniformly than power iteration, without the numerical issues of (not-stabilized) bi-conjugate gradient descent. We ran several simulations using power iteration and BiCGStab, based on random graph matrices with the same level of regularization as we expect to see through the abandonment variable in our true graphs. In these tests, when the graph matrix and the beginning label estimates were non-sparse, on average, BiCGStab converged to the correct solution 12 times faster than the power-iteration method (e.g., BiCGStab would converge in two iterations, requiring only 5 graph-matrix multiplies, while power iteration would require 60 iterations, needing 60 graph-matrix multiplies to converge to the same level of accuracy).

When the graph matrix and the beginning label estimates were sparse, there were similar differences in the rate of convergence, away from the "wavefront boundary". We use the term *wavefront* to emphasize that (for both power iteration and BiCGStab), updates are done in such a way that non-zero values propagate through the graph according to the neighborhood connections. When the labels are sparsely injected, non-zero values move in a "wave", outward from non-zero areas into areas that were zero (due to sparseness). Both power iteration and BiCGStab rely on the graph matrix to determine the label-estimate update, so both have their non-zero wavefronts progress in the same way.

Due to the size of the graph over which we will be operating, we implemented BiCGStab using three MapReduce [9] stages per iteration. Using the notation from the Wikipedia article on BiCGStab [10], we have a distinct set of vectors for each of the labels on which we want to estimate the final distribution. We arrive at the BiCGStab components $\underline{\underline{A}}$ and $\underline{b}$ (at least conceptually) by separating $\underline{\underline{\hat{\gamma}}}\,\underline{\underline{L}}$ into columns corresponding to $\underline{b}$, by separating $\underline{\underline{X}}_n$ into columns corresponding to $\underline{x}_n$ and by using

$$\underline{\underline{A}} = \underline{\underline{I}} - \underline{\underline{\hat{\sigma}}} - \underline{\underline{\hat{\beta}}}\,\underline{\underline{W}} \quad (14)$$

We select an initial *shadow direction* $\underline{\hat{r}}_0$ for each column aligned with its first-pass residual vector, $\underline{r}_0$. Note that computing the first-pass residual vector takes one MapReduce to compute $\underline{r}_0 = b - \underline{\underline{A}}\,\underline{x}_0$. (For our applications, $\underline{b}$ itself is often a good initial estimate for $\underline{x}$.) It is this separate estimation of each column (where each column corresponds to a single label) that makes label slicing so simple and powerful in combination with BiCGStab.

Unlike [10], we mark all our auxiliary variables with the iteration on which they were computed, since this makes our Reduce processing more uniform and reliable: therefore, we use α_n, $\underline{s}_n$ and $\underline{t}_n$ here (instead of their un-versioned form from [10]). To allow the remaining framework to operate smoothly, starting from the initialization (the 0'th pass), we also use the settings for our auxiliary variables that are suggested in [10], namely: $\rho_0 = \alpha = \omega_0 = 1$ and $\underline{v}_0 = \underline{p}_0 = \underline{0}$

For all iterations after this initialization, there are 3 MapReduce stages: (*A*) updating the search direction and its projection through $\underline{\underline{A}}$; (*B*) updating the shadow direction and its projection through $\underline{\underline{A}}$; and (*C*) combining the computed components to give a new state estimate and residual.

For all three MapReduce stages, the reduce processing is the same: from the set of inputs computed in the Map stage,

as well as the inputs passed directly through to the Reducer from previous stages or iterations, keep and combine the results for each variable (auxiliary variables, residual, and state estimate) that is marked with the highest iteration number observed for that variable, and throw away earlier versions.

A. Updating the search direction and its projection

1) Map (shared) context:

a. From initial selection: $\hat{\underline{r}}_0$

b. From previous iteration:

ρ_{n-1}, α_{n-1}, ω_{i-1}, $\underline{r}_{n-1}$, $\underline{v}_{n-1}$, $\underline{p}_{n-1}$

c. From pre-map computation:

$\rho_n = \langle \hat{\underline{r}}_0, \underline{r}_{n-1} \rangle$

$\underline{p}_n = \underline{r}_{n-1} + \left(\frac{\rho_n}{\rho_{n-1}}\right)\left(\frac{\alpha_{n-1}}{\omega_{n-1}}\right)\left(\underline{p}_{n-1} - \omega_{n-1}\underline{\eta}_{n-1}\right)$

2) Map computation:

For each row in $\underline{\underline{A}}$, *compute* $\left\{\underline{\eta}_n\right\}_i = \left\{\underline{\underline{A}}\right\}_{i*} \underline{p}_n$

B. Updating the shadow direction and its projection

1) Map (shared) context:

a. From initial selection: $\hat{\underline{r}}_0$

b. From previous iteration: $\underline{r}_{n-1}$

c. From previous stage of current iteration:

ρ_n, $\underline{\eta}_n$

d. From pre-map computation:

$\alpha_n = \rho_n \big/ \langle \hat{\underline{r}}_0, \underline{\eta}_n \rangle$

$\underline{s}_n = \underline{r}_{n-1} - \alpha_n \underline{\eta}_n$

2) Map computation:

For each row in $\underline{\underline{A}}$, *compute* $\left\{\underline{t}_n\right\}_i = \left\{\underline{\underline{A}}\right\}_{i*} \underline{s}_n$

C. Combining components for residual and state estimates

1) Map (shared) context:

a. From previous iteration: $\underline{x}_{n-1}$

b. From previous stages of current iteration:

α_n, $\underline{s}_n$, $\underline{t}_n$, $\underline{p}_n$

2) Map computation: For each label, compute

$\omega_n = \langle \underline{s}_n, \underline{t}_n \rangle \big/ \langle \underline{t}_n, \underline{t}_n \rangle$

$\underline{x}_n = \underline{x}_{n-1} + \alpha_n \underline{p}_n + \omega_n \underline{s}_n$

$\underline{r}_n = \underline{s}_n - \omega_n \underline{t}_n$

V. MAPREDUCE FORMULATION OF GAUSSIAN ELIMINATION

Label slicing allows us to compute our distributions on the subset of labels that are of most interest, while still benefiting from the constraints effectively imposed by the full label set. In a similar way, Gaussian elimination allows us to compute our distribution on a subset of nodes (domains), while still benefiting from the indirect interconnections that are formed through the nodes that we do not want to explicitly include in our calculation. The computational savings provided by Gaussian elimination is linear with the percentage reduction in the number of graph connections. In addition, Gaussian elimination can speed up convergence, by effectively increasing the wavefront-propagation speed through those parts of the graph that were originally connected via the eliminated nodes.

Gaussian elimination is much simpler to implement in the MapReduce framework than BiCGStab, requiring only a single stage and capable of handling elimination of multiple nodes per run. The Reduce processing in the MapReduce is a straight pass-through of the outputs from the map stage.

To make the description more concise, define

$$\underline{\underline{A}}_{keep} = \left\{\underline{\underline{A}}\right\}_{i*} \quad \underline{\underline{L}}_{\gamma keep} = \left\{\hat{\gamma}\underline{\underline{L}}\right\}_{i*} \quad i \in \left\{ \begin{array}{c} nodes \\ to\ be\ kept \end{array} \right\}$$

$$\underline{\underline{A}}_{remove} = \left\{\underline{\underline{A}}\right\}_{j*} \quad \underline{\underline{L}}_{\gamma remove} = \left\{\hat{\gamma}\underline{\underline{L}}\right\}_{j*} \quad j \in \left\{ \begin{array}{c} nodes\ to\ be \\ eliminated \end{array} \right\}$$

Using this notation, the map processing is

1) Map (shared) context:

From stored representation:

$\underline{\underline{A}}_{remove}$, $\underline{\underline{L}}_{\gamma remove}$

2) Map computation: For each row, i, in $\underline{\underline{A}}_{keep}$ *and* $\underline{\underline{L}}_{\gamma keep}$

a) Initialize

$$\tilde{\underline{\underline{A}}}_{keep} = \underline{\underline{A}}_{keep}, \quad \tilde{\underline{\underline{A}}}_{remove} = \underline{\underline{A}}_{remove}$$

$$\tilde{\underline{\underline{L}}}_{\gamma keep} = \underline{\underline{L}}_{\gamma keep}, \quad \tilde{\underline{\underline{L}}}_{\gamma remove} = \underline{\underline{L}}_{\gamma remove}$$

b) Compute the pivot strength, π_{ij}, for each $j \in \{nodes\ to\ be\ eliminated\}$:

$$\pi_{ij} = \left\{\tilde{\underline{\underline{A}}}_{keep}\right\}_{ij} \Big/ \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{jj}$$

and select the elimination node, $\tilde{j}$, with the smallest amplitude $\left|\pi_{ij}\right|$

c) Eliminate all non-zero entries in the $\tilde{j}$'th column in $\left\{\tilde{\underline{\underline{A}}}_{keep}\right\}_{i*}$ and $\tilde{\underline{\underline{A}}}_{remove}$, with matched operations on $\left\{\tilde{\underline{\underline{L}}}_{\gamma keep}\right\}_{i*}$ and $\tilde{\underline{\underline{L}}}_{\gamma remove}$:

$$\left\{\tilde{\underline{\underline{A}}}_{keep}\right\}_{ik} \leftarrow \left\{\tilde{\underline{\underline{A}}}_{keep}\right\}_{ik} - \pi_{i\tilde{j}} \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{\tilde{j}k}$$

$$\left\{\tilde{\underline{\underline{L}}}_{\gamma keep}\right\}_{ik} \leftarrow \left\{\tilde{\underline{\underline{L}}}_{\gamma keep}\right\}_{ik} - \pi_{i\tilde{j}} \left\{\tilde{\underline{\underline{L}}}_{\gamma remove}\right\}_{\tilde{j}k}$$

$$\left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{nk} \leftarrow \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{nk} - \tilde{\pi}_{n\tilde{j}} \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{\tilde{j}k} \quad \forall n \neq \tilde{j}$$

$$\left\{\tilde{\underline{\underline{L}}}_{\gamma remove}\right\}_{nk} \leftarrow \left\{\tilde{\underline{\underline{L}}}_{\gamma remove}\right\}_{nk} - \tilde{\pi}_{n\tilde{j}} \left\{\tilde{\underline{\underline{L}}}_{\gamma remove}\right\}_{\tilde{j}k} \quad \forall n \neq \tilde{j}$$

$$\text{with } \tilde{\pi}_{nj} = \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{nj} \Big/ \left\{\tilde{\underline{\underline{A}}}_{remove}\right\}_{jj}$$

d) Remove row $\tilde{j}$ from $\tilde{\underline{\underline{A}}}_{remove}$, $\tilde{\underline{\underline{L}}}_{\gamma remove}$

e) Repeat (b), (c), and (d), until there are no more rows (nodes) to be removed.

f) Output $\left\{\tilde{\underline{\underline{A}}}_{keep}\right\}_{i*}$ and $\left\{\tilde{\underline{\underline{L}}}_{\gamma keep}\right\}_{i*}$

VI. LARGE-SCALE DOMAIN-LEVEL TOPIC LABELING

Baluja et al. [2] already showed the usefulness of the Adsorption approach in video recommendations. The pre-normalized Adsorption algorithm [1] provides identical results at a fraction of the computational cost using the new formulation with label slicing, Gaussian elimination, and BiCGStab. The final computational cost is reduced by the product of the savings of all three approaches (label slicing, BiCGStab and Gaussian elimination).

In our previous paper [1], we explored using pre-normalized Adsorption for topic labeling on web domains, for search and advertising. Many pages URLs, and even whole domains, are poorly classified by standard topic-analysis approaches, due to having little in the way of machine-understandable content to classify. A standard example of this problem are domains that primarily host images or video – while the page URL can be examined for clues to the topic, as well as the linked-to URLs, the results are impoverished and noisy. If we can improve the topic labeling, we could more accurately index these pages for search and for content-matched advertisement.

Specifically, we created a graph with domains as nodes and a measure of shared searches for cross-domain pairs of URLs as the weighted connections between nodes. Our measure looked at, for each search term, the click rates for each URL served in the results and set the strength of the URL-URL-term triple to the lower of the click rates between the paired URLs. The connection weight between pairs of URLs is the sum over all triples that terminate at those two URLs. To aggregate from URL-pair connections, up to domain-pair connections, we sum across those URL-pair connections where the first of the pair of URLs is from the first domain and the second is from the second domain. Similarly, our injection labeling is based on combining topic analysis of the URLs within the domain, dropping those topics that were based on keywords that showed too much within-domain variance in their strength. We aggregate the link and topic-label strength up to the domain level to improve coverage and reliability of our graph connections. Even with this aggregation of URLs to domain-level nodes and filtering of keyword labels to within-domain-stable sets, our initial data provides a graph of about 13 million domains (nodes), with about 4 billion node-to-node connections based on analysis of more than 253 million search terms. Our topic analysis provides more than 4,500 general topics, using traditional text-based classification.

From this set of 4,500 topics, we focused on 71 commercial topics (see Figure 1 for examples). The computational savings (over the original Adsorption approach) for the label slicing alone was a factor of 63 times. We do not include this savings in the remainder of this discussion, since it is available to both power iteration and BiCGStab, as long as we are using the pre-normalized Adsorption formulation. That said, it is the most significant source of computational savings, compared to the original work [2].

We ran this set of 71 labels through two iterations of BiCGStab (5 graph-matrix multiplies) and through 70 iterations of the power method, both starting from the same initial estimate. Figure 2 shows the size of the per-node residual for BiCGStab on these labels (using an L_1 norm). As with our small-scale simulations, at the end of our second iteration, the not-insignificant residuals occurred at the 3% of the nodes that were at the "wavefront boundary" of one or more of the topic labels. This level of convergence, with just 5 matrix multiplies, is not seen in the power-iteration solution until the 62^{th} iteration (an additional savings of nearly 12.5 times).

Since the goal of our label propagation is to increase the richness and extent of the topic labeling on poorly labeled (or unlabeled) domains without over-extending into domains that are not related to our commercial subset, it is helpful to look at the statistics summarized in Figures 3 through 5.

Figure 3 gives a measure of the richness of our labels on commercial domains and how that richness increases as a function of iteration. The plot shows the percentages of domains by how many commercial-topic labels are seen on that domain. If a domain is commercial, the more commercial labels that are associated with the domain, the richer the topic description. As shown by the plots, our

- Clothing
  - Women's, Men's, Children's
  - Athletic, Casual, Formal, Outerwear, Sleepwear
  - Shoes, Boots
- Accessories
  - Jewelry, Watches, Purses
- Toys
  - Building Toys, Dolls, Stuffed Animals, Ride-on Toys
- Gifts
  - Flowers, Cards, Party Items, Holiday Items
- Discounts
  - Coupons, Loyalty Cards

Figure 1. Examples from selected 71 commercial topics.

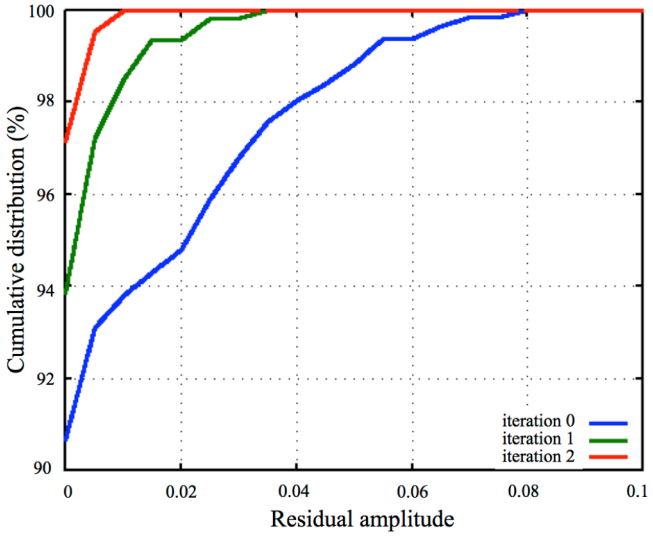


Figure 2. Cumulative residual distribution (by iteration). [1]

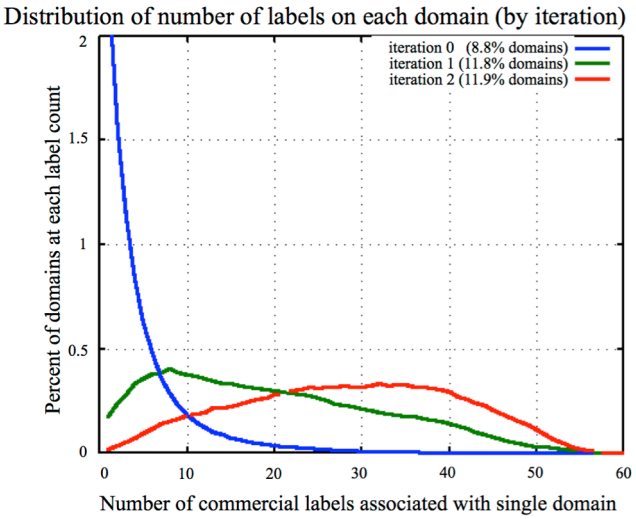


Figure 3. Node-level coherence of commercial labels. [1]

injection labels (those given by topic analysis) within each domain provides sparse topic labels, with the largest percentage of commercial domains having only one label. Since our 71 commercial topics are actually a hierarchical set, this sparseness is unlikely to be correct for most domains. By the end of the second iteration, the mode of that distribution has moved to around 30 topic labels per commercial domain.

Also, the legend in Figure 3 gives us the information needed to check that we are not just expanding the support of our commercial-topic labels indiscriminately across the full domain graph. The first iteration extends the support of the commercial labels by a third, from just under 9% of all domains to just under 12%, suggesting the addition of a subset of the unlabeled domains within the graph. After the first iteration, the support of the commercial-label set is effectively unchanged. This can be traced back to the effect of pre-normalizing on the full set of topic labels. Even though the non-commercial topics are not being explicitly computed in our iterations, they still have an effect, keeping the commercial labels from spreading onto distant (in the graph-connection sense) domains, as they otherwise would as the commercial wavefront progressed. This highlights both one of the main advantages of the original Adsorption as well as the most compelling advantage of the pre-normalized Adsorption. With the original Adsorption, each node has a limited capacity for supporting labels, thereby limiting propagation – but enforcing that limited capacity forced computation of all label distributions, not just the labels of interest. With pre-normalized Adsorption, there is still the per-node limited capacity for supporting labels, but we achieve that capacity limit by pre-normalizing, freeing us to compute only at that subset of labels that we are interested in, without having those labels spread unchecked.

Up to now, our analysis of our results has focused on the richness and extent of our commercial labels but not on the likely quality of the mix of labels that we are introducing onto commercial nodes. Since our topics are structured into a hierarchical framework, intuitively what we would like is to have each commercial site labeled mostly by closely related subsets of the available topics. We can use *dendrite distances* between the labels to capture this sense of closeness among the sets of labels associated with each domain node. As with standard dendrite measures, for each pair of labels on a domain, we count the number of hierarchical topic links that we have to go across in order to travel from one topic label to the other. We lengthen that distance by one for each generation that *both* labels have to travel back through, in order to penalize siblings more than grandparent-grandchild relations. As an example, if we need to calculate the distance between women's jewelry and men's clothing and we have the two tree branches "Jewelry → Women's Accessories → Apparel" and "Men's Clothing → Apparel", our dendrite distance measure would be 4: two (for "Women's Jewelry" to "Apparel") plus one (for "Men's Clothing" to "Apparel") plus one (for the one generation removal from direct descendent connection).

As a way to evaluate our label distributions on domains with 2 to 6 labels, we computed all pairwise dendrite distances within each domain and averaged them (again, on a per-domain basis). Due to the use of the topic hierarchy in our dendrite-distance measure, smaller distances amongst the labels on a single domain correspond to more believable topic mixes. Figure 4 shows our results, as function of iteration. When the initial topic labeling provides more than one label, it includes many dissimilar labels, with the mode of the dendrite average distance being up between 6 and 7. Our propagation reduces that average distance, filling in parent and children nodes, to give a mode that is just above one. While parents could always be filled in by knowing the hierarchical structure of our topic labels, the propagation graph is doing this without that knowledge – it is finding these associations purely through propagation of neighbor

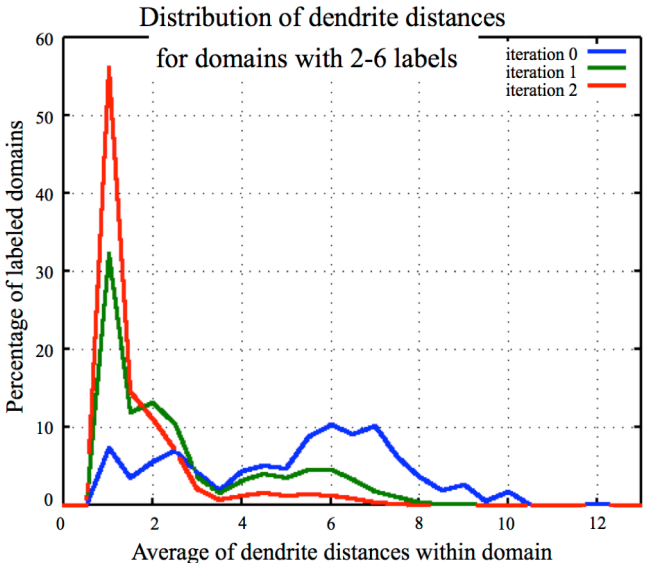


Figure 4. Dendrite topic-label distance on domains with 2-6 labels (by iteration). [1]

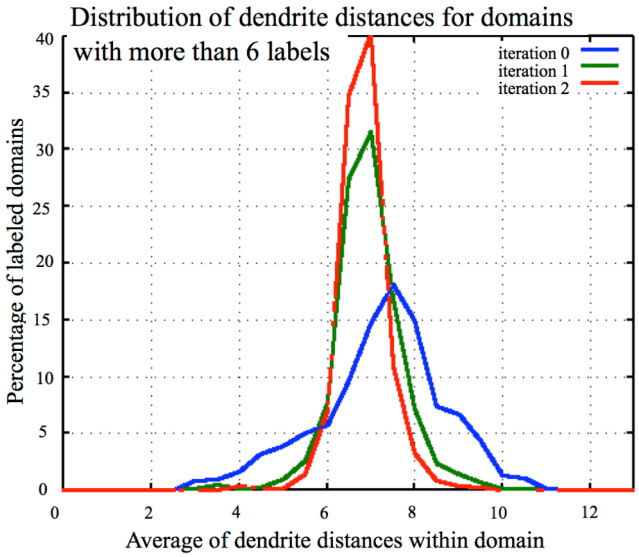


Figure 5. Dendrite topic-label distance on domains with more than 6 labels (by iteration). [1]

labels. (Furthermore, we could not use the tree-structure meta-information to fill in the correct children labels – if we blindly used the tree structure, we would get numerous nearby but irrelevant labels.) For this set of nodes, we are enriching the topic description without introducing unrelated labels. This measure of quality is a stringent one, since at no point do we use the dendrite structure to limit our propagation.

Figure 5 shows a similar measure, for domains with more than 6 labels, again averaging the dendrite distances within each node. We did this separation between Figure 4, for domains with 2-6 commercial labels, and Figure 5, for domains with more than 6 commercial labels, since the dendrite distances across larger sets of labels, taken from the same hierarchy will have a larger minimum-average distance than will smaller sets of labels. For small sets, you can often find 2-6 labels, with all parent-child or sibling relationships with one another but, for large sets of labels, this is not possible and first and second cousin relationships become a major part of even the most compact set of labels. Same as with Figure 4, Figure 5 shows that the average dendrite distance decreases with each iteration, even on nodes with more than 6 labels. Since closely related sets of topic labels are more likely to be a full and accurate description of the domain topic, our topic labeling seems to be improved by our graph propagation work.

All of the measurements conducted on the propagation of web labels on this large set of domains indicate an improvement in search indexing and content-matched advertising. In the future, we will expand these experiments in two directions. First, we will run live trials, with full user-facing experiments, to determine the quality improvement in the user experience. Second, we will increase our graph size and specificity by including individual URLs, for those sites that have enough textual information to support that level of analysis.

VII. Efficient Updating on Dynamic Graphs

In nearly every application of graph-based label propagation, the graph changes over time: in video recommendation, new videos are added and old videos fade in popularity; in social networks, new users join, new friendships are made, and old friendships are ended; and, in topic labeling, the connection strengths between domains change as their content is updated. These changes occur gradually and most of the label distributions within the newly changed graph are only mildly perturbed from those labels that were computed for the original matrix, making it more efficient to do incremental updates than to restart the labeling process from scratch. The largest changes are associated with newly added nodes and labels and with the nodes that connect to either those sets. We focus on the changes to the graph and the labels to create an efficient update process.

Since we are now considering a change in the graph and label distribution, which will necessitate breaking the matrices into pieces, we first define a more compact notation for our pre-normalized adsorption state equation. Instead of using Equation (13), we will use

$$\underline{\underline{X}} = \underline{\underline{\hat{W}}}\,\underline{\underline{X}} + \underline{\underline{\hat{L}}} \tag{15}$$

where $\underline{\underline{\hat{W}}} = \underline{\underline{\hat{\sigma}}} + \underline{\underline{\hat{\beta}}}\,\underline{\underline{W}}$ and $\underline{\underline{\hat{L}}} = \left[\begin{array}{cc} \underline{\underline{\hat{\gamma}}}\,\underline{\underline{L}} & \underline{\underline{\hat{\delta}}} \end{array} \right]$. Equation (15) is identical to Equation (13), with the exception of the symbols that we use to describe it. This notation hides the iteration subscript that we previously associated with $\underline{\underline{X}}$, so that we will be able to use the subscript location for identifying sub-matrices.

To refer to the two related but distinct sets of graph and label weights, we will use a superscript of "-" or "+" to distinguish the pre-change and post-change versions of the graph, respectively. So:

$$\underline{\underline{X}}^{-} \approx \underline{\underline{\hat{W}}}^{-}\,\underline{\underline{X}}^{-} + \underline{\underline{\hat{L}}}^{-} \tag{16}$$

is the pre-change version of the graph state equations, with $\underline{\underline{X}}^{-}$ as the inferred label distributions that we have already computed for the pre-change graph, and

$$\underline{\underline{X}}^{+} = \underline{\underline{\hat{W}}}^{+}\,\underline{\underline{X}}^{+} + \underline{\underline{\hat{L}}}^{+} \tag{17}$$

is the post-change version of the graph state equations, with $\underline{\underline{X}}^{+}$ as the inferred label distributions that we need to compute for the post-change graph. We also define difference matrices:

$$\Delta\underline{\underline{Y}} = \underline{\underline{Y}}^{+} - \underline{\underline{Y}}^{-} \tag{18}$$

where $\underline{\underline{Y}}$ can be any of $\underline{\underline{X}}$, $\underline{\underline{\hat{W}}}$, or $\underline{\underline{\hat{L}}}$.

In order to allow us to use matrix operations across these two graph descriptions, we assume that we have added all-zero rows and columns as needed to the pre-change graph, to allow for the newly added nodes and labels that we need for the post-change description and that we have added all-zero rows and columns as needed to the post-change graph, to allow for the newly removed nodes and labels that were

present in the pre-change description. We rearrange the rows and columns, so that we group these to all-zero rows and columns and use the notation

$$\underline{\underline{Y}} = \begin{bmatrix} \underline{\underline{Y}}_{\|\|} & \underline{\underline{Y}}_{\|-} & \underline{\underline{Y}}_{\|+} \\ \underline{\underline{Y}}_{-\|} & \underline{\underline{Y}}_{--} & \underline{\underline{0}} \\ \underline{\underline{Y}}_{+\|} & \underline{\underline{0}} & \underline{\underline{Y}}_{++} \end{bmatrix} \quad (19)$$

where $\underline{\underline{Y}}$ can be any of $\underline{\underline{X}}$, $\underline{\underline{\hat{W}}}$, or $\underline{\underline{\hat{L}}}$ and $\underline{\underline{Y}}^{-}$ refers to the pre-change versions of these matrices and $\underline{\underline{Y}}^{+}$ refers to the post-change versions. The subscripts "|", "-", and "+" distinguish between nodes and labels according to their need to be part of the pre- and post-change graphs. The first of the subscript pair refers to the row characteristics and the second refers to the column characteristics. The "-" subscript is for rows or columns that are only needed for the pre-change matrices (i.e., they are identically zero for the post-change matrices). The "+" subscript is for rows or columns that are only needed for the post-change matrices (i.e., they are identically zero for the pre-change matrices). The "|" subscript is for rows or columns that are needed for both pre- and post-change matrices. Notice that we can assert that the sub-matrices that would have represented interactions between "-" and "+" rows and columns are known to be identically zero, since these two sets of nodes and labels do not occur (non-trivially) in the same matrices. Also, using this grouping, we know that $\underline{\underline{Y}}^{-}_{|+} = \underline{\underline{Y}}^{-}_{++} = \underline{\underline{Y}}^{-}_{+|} \equiv \underline{\underline{0}}$ and $\underline{\underline{Y}}^{+}_{|-} = \underline{\underline{Y}}^{+}_{--} = \underline{\underline{Y}}^{+}_{-|} \equiv \underline{\underline{0}}$. Finally, we will use the same matrix partitioning for the difference matrices, $\Delta\underline{\underline{Y}}$, as we have described above for the pre- and post-change matrices.

Our goal is to find a good estimate of $\underline{\underline{X}}^{+}$ from $\underline{\underline{X}}^{-}$, with as few computations as possible. Starting from the post-change equation and recasting it in terms of the pre-change matrices and the difference matrices:

$$\underline{\underline{X}}^{+} = \underline{\underline{\hat{W}}}^{+}\underline{\underline{X}}^{+} + \underline{\underline{\hat{L}}}^{+} \quad (20)$$

$$\underline{\underline{X}}^{-} + \Delta\underline{\underline{X}} = \underline{\underline{\hat{W}}}^{+}\left(\underline{\underline{X}}^{-} + \Delta\underline{\underline{X}}\right) + \underline{\underline{\hat{L}}}^{-} + \Delta\underline{\underline{\hat{L}}} \quad (21)$$

$$\underline{\underline{X}}^{-} + \Delta\underline{\underline{X}} = \left(\underline{\underline{\hat{W}}}^{-} + \Delta\underline{\underline{\hat{W}}}\right)\underline{\underline{X}}^{-} + \underline{\underline{\hat{W}}}^{+}\Delta\underline{\underline{X}} + \underline{\underline{\hat{L}}}^{-} + \Delta\underline{\underline{\hat{L}}} \quad (22)$$

Rearranging Equation (22):

$$\Delta\underline{\underline{X}} = \underline{\underline{\hat{W}}}^{+}\Delta\underline{\underline{X}} + \Delta\underline{\underline{\hat{L}}} + \Delta\underline{\underline{\hat{W}}}\,\underline{\underline{X}}^{-} + \left(\underline{\underline{\hat{W}}}\,\underline{\underline{X}}^{-} + \underline{\underline{\hat{L}}}^{-} - \underline{\underline{X}}^{-}\right) \quad (23)$$

Since we have a good estimate of $\underline{\underline{X}}^{-}$ from the pre-change description, we use $\underline{\underline{X}}^{-} \approx \underline{\underline{\hat{W}}}^{-}\underline{\underline{X}}^{-} + \underline{\underline{\hat{L}}}^{-}$ to remove the final term from Equation (23), giving

$$\Delta\underline{\underline{X}} = \underline{\underline{\hat{W}}}^{+}\Delta\underline{\underline{X}} + \left(\Delta\underline{\underline{\hat{L}}} + \Delta\underline{\underline{\hat{W}}}\,\underline{\underline{X}}^{-}\right) \quad (24)$$

Finally, we define

$$\Delta\underline{\underline{X}}^{0} = \Delta\underline{\underline{\hat{L}}} + \Delta\underline{\underline{\hat{W}}}\,\underline{\underline{X}}^{-} \quad (25)$$

to get:

$$\Delta\underline{\underline{X}} = \underline{\underline{\hat{W}}}^{+}\Delta\underline{\underline{X}} + \Delta\underline{\underline{X}}^{0} \quad (26)$$

There are several things of note about Equations (25) and (26). From Equation (25), $\Delta\underline{\underline{X}}^{0}$ is exactly the estimate for $\Delta\underline{\underline{X}}$, if our previous estimate was $\underline{\underline{0}}$. It is also used in later iterations, as a persistent input, so explicitly saving it reduces the computation needed on later iterations. Finally, $\Delta\underline{\underline{X}}^{0}$ is much sparser than $\underline{\underline{\hat{L}}}$, which will be useful in our discussion of Equation (26).

Equation (26) is an update equation, similar to Equation (15). The reason that Equation (26) is preferred over Equation (15) is (as just noted) $\Delta\underline{\underline{X}}^{0}$ is much sparser than $\underline{\underline{\hat{L}}}$ and that $\Delta\underline{\underline{X}}$ is much sparser than $\underline{\underline{\hat{X}}}$, even after several iterations. This sparseness reduces the amount of computation needed per iteration.

We can further improve the efficiency and compactness of our computation by not computing values for the nodes that are not needed for the post-change description. We can now use our matrix partitioning to remove these extra entries. We can also use our knowledge that $\underline{\underline{Y}}^{-}_{|+} = \underline{\underline{Y}}^{-}_{++} = \underline{\underline{Y}}^{-}_{+|} \equiv \underline{\underline{0}}$ and $\underline{\underline{Y}}^{+}_{|-} = \underline{\underline{Y}}^{+}_{--} = \underline{\underline{Y}}^{+}_{-|} \equiv \underline{\underline{0}}$ to simplify the formula. When we use those zero identities along with the sub-matrix notation in Equation (26), we get:

$$\begin{bmatrix} \Delta\underline{\underline{X}}_{\|\|} & -\underline{\underline{X}}^{-}_{\|-} & \underline{\underline{X}}^{+}_{\|+} \\ -\underline{\underline{X}}^{-}_{-\|} & -\underline{\underline{X}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{X}}^{+}_{\|+} & \underline{\underline{0}} & \underline{\underline{X}}^{+}_{++} \end{bmatrix} = \begin{bmatrix} \Delta\underline{\underline{\hat{W}}}_{\|\|} & -\underline{\underline{\hat{W}}}^{-}_{\|-} & \underline{\underline{\hat{W}}}^{+}_{\|+} \\ -\underline{\underline{\hat{W}}}^{-}_{-\|} & -\underline{\underline{\hat{W}}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{\hat{W}}}^{+}_{\|+} & \underline{\underline{0}} & \underline{\underline{\hat{W}}}^{+}_{++} \end{bmatrix} \begin{bmatrix} \Delta\underline{\underline{X}}_{\|\|} & -\underline{\underline{X}}^{-}_{\|-} & \underline{\underline{X}}^{+}_{\|+} \\ -\underline{\underline{X}}^{-}_{-\|} & -\underline{\underline{X}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{X}}^{+}_{\|+} & \underline{\underline{0}} & \underline{\underline{X}}^{+}_{++} \end{bmatrix}$$
$$+ \begin{bmatrix} \Delta\underline{\underline{\hat{L}}}_{\|\|} & -\underline{\underline{\hat{L}}}^{-}_{\|-} & \underline{\underline{\hat{L}}}^{+}_{\|+} \\ -\underline{\underline{\hat{L}}}^{-}_{-\|} & -\underline{\underline{\hat{L}}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{\hat{L}}}^{+}_{\|+} & \underline{\underline{0}} & \underline{\underline{\hat{L}}}^{+}_{++} \end{bmatrix} + \begin{bmatrix} \Delta\underline{\underline{\hat{W}}}_{\|\|} & -\underline{\underline{\hat{W}}}^{-}_{\|-} & \underline{\underline{\hat{W}}}^{+}_{\|+} \\ -\underline{\underline{\hat{W}}}^{-}_{-\|} & -\underline{\underline{\hat{W}}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{\hat{W}}}^{+}_{\|+} & \underline{\underline{0}} & \underline{\underline{\hat{W}}}^{+}_{++} \end{bmatrix} \begin{bmatrix} \underline{\underline{X}}^{-}_{\|\|} & \underline{\underline{X}}^{-}_{\|-} & \underline{\underline{0}} \\ \underline{\underline{X}}^{-}_{-\|} & \underline{\underline{X}}^{-}_{--} & \underline{\underline{0}} \\ \underline{\underline{0}} & \underline{\underline{0}} & \underline{\underline{0}} \end{bmatrix} \quad (27)$$

We can reduce Equation (27) down to the four submatrix update equations that constrain label distributions in the post-change network ($\Delta\underline{\underline{X}}_{\shortparallel}$, $\underline{\underline{X}}^{+}_{\shortparallel+}$, $\underline{\underline{X}}^{+}_{+\shortparallel}$, and $\underline{\underline{X}}^{+}_{++}$). Focusing on those four update equations:

$$\Delta\underline{\underline{X}}_{\shortparallel} = \underline{\underline{\hat{W}}}^{+}_{\shortparallel} \Delta\underline{\underline{X}}_{\shortparallel} + \underline{\underline{\hat{W}}}^{+}_{\shortparallel+} \underline{\underline{X}}^{+}_{+\shortparallel} + \Delta\underline{\underline{\hat{L}}}_{\shortparallel} + \Delta\underline{\underline{\hat{W}}}_{\shortparallel} \underline{\underline{X}}^{-}_{\shortparallel} - \underline{\underline{\hat{W}}}^{-}_{\shortparallel-} \underline{\underline{X}}^{-}_{-\shortparallel} \quad (28)$$

$$\underline{\underline{X}}^{+}_{\shortparallel+} = \underline{\underline{\hat{W}}}^{+}_{\shortparallel} \underline{\underline{X}}^{+}_{\shortparallel+} + \underline{\underline{\hat{W}}}^{+}_{\shortparallel+} \underline{\underline{X}}^{+}_{++} + \underline{\underline{\hat{L}}}^{+}_{\shortparallel+} \quad (29)$$

$$\underline{\underline{X}}^{+}_{+\shortparallel} = \underline{\underline{\hat{W}}}^{+}_{+\shortparallel} \Delta\underline{\underline{X}}_{\shortparallel} + \underline{\underline{\hat{W}}}^{+}_{++} \underline{\underline{X}}^{+}_{+\shortparallel} + \underline{\underline{\hat{L}}}^{+}_{+\shortparallel} + \underline{\underline{\hat{W}}}^{-}_{+\shortparallel} \underline{\underline{X}}^{-}_{\shortparallel} \quad (30)$$

$$\underline{\underline{X}}^{+}_{++} = \underline{\underline{\hat{W}}}^{+}_{+\shortparallel} \underline{\underline{X}}^{+}_{\shortparallel+} + \underline{\underline{\hat{W}}}^{+}_{++} \underline{\underline{X}}^{+}_{++} + \underline{\underline{\hat{L}}}^{+}_{++} \quad (31)$$

Reformatting Equations (28) through (31) back into a single partitioned matrix gives:

$$\begin{bmatrix} \Delta\underline{\underline{X}}_{\shortparallel} & \underline{\underline{X}}^{+}_{\shortparallel+} \\ \underline{\underline{X}}^{+}_{+\shortparallel} & \underline{\underline{X}}^{+}_{++} \end{bmatrix} = \begin{bmatrix} \underline{\underline{\hat{W}}}^{+}_{\shortparallel} & \underline{\underline{\hat{W}}}^{+}_{\shortparallel+} \\ \underline{\underline{\hat{W}}}^{+}_{+\shortparallel} & \underline{\underline{\hat{W}}}^{+}_{++} \end{bmatrix} \begin{bmatrix} \Delta\underline{\underline{X}}_{\shortparallel} & \underline{\underline{X}}^{+}_{\shortparallel+} \\ \underline{\underline{X}}^{+}_{+\shortparallel} & \underline{\underline{X}}^{+}_{++} \end{bmatrix} + \begin{bmatrix} \Delta\underline{\underline{X}}^{0}_{\shortparallel} & \underline{\underline{\hat{L}}}^{+}_{\shortparallel+} \\ \Delta\underline{\underline{X}}^{0}_{+\shortparallel} & \underline{\underline{\hat{L}}}^{+}_{++} \end{bmatrix} \quad (32)$$

where

$$\Delta\underline{\underline{X}}^{0}_{\shortparallel} = \Delta\underline{\underline{\hat{L}}}_{\shortparallel} + \Delta\underline{\underline{\hat{W}}}_{\shortparallel} \underline{\underline{X}}^{-}_{\shortparallel} - \underline{\underline{\hat{W}}}^{-}_{\shortparallel-} \underline{\underline{X}}^{-}_{-\shortparallel} \quad (33)$$

$$\Delta\underline{\underline{X}}^{0}_{+\shortparallel} = \underline{\underline{\hat{L}}}^{+}_{+\shortparallel} + \underline{\underline{\hat{W}}}^{-}_{+\shortparallel} \underline{\underline{X}}^{-}_{\shortparallel} \quad (34)$$

Using Equations (32) through (34) allows us to find an update to the inferred label matrices, starting from the inferred labels for the pre-change graph, even when there are nodes and labels that have been newly added or completely deleted. This approach saves computation on each iteration, since the inferred-label change matrix will be non-zero on a much smaller number of nodes and labels than the full inferred-label matrices are. Further savings can be had by computing and caching the initial update matrices described in Equations (33) and (34) for use in later iterations. We stop iterating on the inferred-label change matrix when the per-entry residuals are similar in size to residuals that we ignored in using $\underline{\underline{X}}^{-} \approx \underline{\underline{\hat{W}}}^{-} \underline{\underline{X}}^{-} + \underline{\underline{\hat{L}}}^{-}$ or when the inferred-label change matrix is no more sparse than the full post-change inferred-label matrix. At that point, the post-change inferred label matrix should be reconstructed, using the values of $\underline{\underline{X}}^{+}_{\shortparallel+}$, $\underline{\underline{X}}^{+}_{++}$, and $\underline{\underline{X}}^{+}_{+\shortparallel}$ as given by Equation (32) and using $\underline{\underline{X}}^{+}_{\shortparallel} = \underline{\underline{X}}^{-}_{\shortparallel} + \Delta\underline{\underline{X}}_{\shortparallel}$ for the last non-trivial submatrix of $\underline{\underline{X}}^{+}$.

The convergence of Equation (32) is guaranteed only indirectly. Equation (32) is formed as the difference of two state equations (one for the pre-change graph and one for the post-change graph). Both of those two state equations, having eigenvalues that are strictly inside the unit circle, are guaranteed to converge. The difference between them will therefore converge.

VIII. Incremental Updating of Co-Author Network Inferences

To demonstrate the use of the incremental update of label distributions on changing graphs, we used condensed-matter collaboration data, posted at [11] from work done by Newman [12]. This co-author network data was first collected from physics pre-print publications for 1995 through 1999 but was twice updated, first to contain co-author connections from 1995 to 2003 and later to extend that time frame to 2005. The 1995-2003 network contains 31,163 authors (nodes) while the 1995-2005 network has 40,421 authors. Based on exact matching of names, all except 57 of the authors from the 2003 network could be uniquely matched to the authors in the 2005 network. These 57 authors had ambiguous matches (e.g., there were 3 "PARK, S" author nodes in both the 2003 and 2005 networks). This original pair of networks had 30% new authors added (and 0.2% dropped, due to ambiguity) between 2003 and 2005, in addition to having changes in the connection weights between the authors that were in both networks. To create the co-author graph, Newman [12] scanned the Los Alamos e-Print Archive on condensed-matter physics for the years in question. For each paper in that database that had n authors, for $n > 1$, he added (or strengthened) a connection between each pair of co-authors by a weight of $1/(n-1)$. In this way, the connections made from each co-author to other researchers is increased by one, for each paper that is a collaborative effort. Newman's research [12] describes the core characteristics of this network: the mean (collaborative) papers-per-author in this field is 3.87 (with a standard deviation of 5); the mean number of authors per paper is 2.66 (with a standard deviation of 1); and the mean number of collaborators for each author is 5.86 (with a standard deviation of 9).

This type of co-author graph can be used to recommend new collaborations to each author in the network, based on propagating the names of potential collaborators. To this end, we use parts of these co-author weighted graphs as our un-normalization node-to-node matrix, $\underline{\underline{W}}$ as used in Equation (1). We created an un-normalized label matrix, $\underline{\underline{L}}$ as used in Equation (1), from the author names, using as an (un-normalized) injection weight the number of papers on which that author collaborated. Using the number of papers as this label weight will result in the prolific authors' names being recommended as potential collaborators more widely and strongly than less prolific authors. To complete the un-normalized constraint equation, we somewhat arbitrarily set $\beta = \gamma = 1$ and $\sigma = \delta = 2$ in Equation (1).

The addition of 30% new authors between 1995-2003 and 1995-2005 is large enough that the incremental update approach would provide little, if any, computational

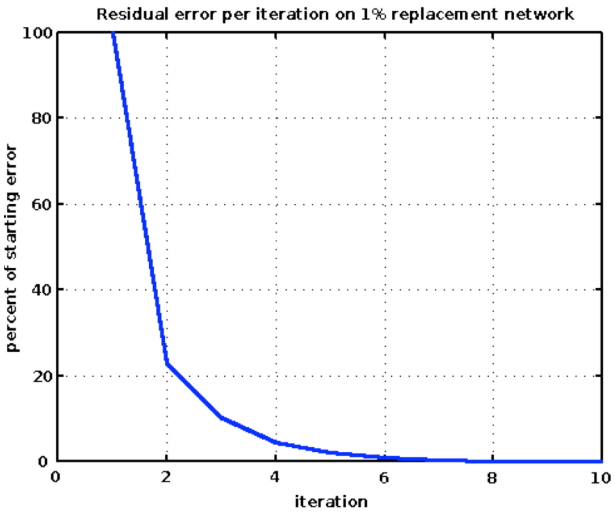


Figure 6. Residual error on 1% replacement network, starting from the label distribution from the pre-change graph, as a function of iteration (using the power method of solution)

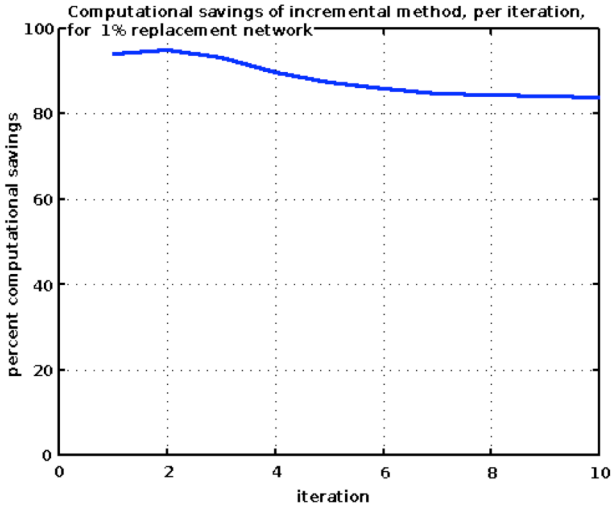


Figure 7. Computational savings on 1% replacement network using incremental updating compared to using the same starting estimate with the full post-change network.

savings: with this much change, $\Delta\underline{\underline{X}}^0$ is almost as dense as $\underline{\underline{\hat{L}}}$ and, due to the authorship fan-out becomes nearly as dense as the full label matrix, $\underline{\underline{\hat{X}}}$, by the second iteration. To concretely demonstrate the potential for large computational savings in incrementally changing networks, we employed a subset of the 1995-2003 and 1995-2005 data. We first reduced the size of both of the 2003 and the 2005 networks down to the same 27,519 authors, ones with unambiguous matches who occurred with a reasonable weight and connectivity in both data sets. From this shared set, we picked equal numbers of distinct nodes to drop from each of the reduced-2003 and reduced-2005 networks, so that both reduced networks retained equal numbers of nodes (authors) by picking the least well-connected nodes and dropping them from one of the two networks. For our "1% replacement experiment", we did this with 1% of the nodes, so that both the pre- and the post-change graphs had 27,244 nodes with 275 of the nodes that are in the reduced-2003 graph being dropped and replaced with a distinct set of 275 nodes for the reduced-2005 graph. The connection weights between the 99% of the nodes (26,969 nodes) that appeared in both of these reduced graphs changed in whatever way that was indicated by the original 2003 or 2005 data from [11]. In a similar manner, we created our "17% replacement experiment", removing two distinct sets of 3,997 nodes from the reduced-2003 and -2005 networks to create two graphs with 23,522 nodes each, 17% of which are present in only one of the two graphs. As before, the connection weights for the nodes that were shared between the graphs was allowed to change, according to the original 2003 or 2005 data from [11].

Once these two pairs of un-normalized networks (pre- and post-change networks for the 1% and 17% replacement experiments) were formed, we separately normalized the matrices for each of the four networks, as described by Equations (8) to (13). These normalizations are based on the entries that are actually in each network: there is no leakage from pre-change networks into the normalization of the post-change network (nor vice-versa) and there is no leakage from anything done in the 1% replacement networks to the 17% replacement networks (nor vice-versa).

The node-to-node connection occupancies were about 0.04% on both the pre- and post-change graphs ($\underline{\underline{\hat{W}}}^-$ and $\underline{\underline{\hat{W}}}^+$) for both the 1% and 17% replacement experiments. This is nearly twice the collaboration rate reported by Newman [12], in part due to the longer time period covered (8 and 10 years, in contrast with 5 years) and in part due to the selection bias for how we created the reduced 2003 and 2005 networks. In contrast to the occupancy of $\underline{\underline{\hat{W}}}^-$ and $\underline{\underline{\hat{W}}}^+$, the occupancy of $\Delta\underline{\underline{\hat{W}}}$ was about 50% of that level (so, 0.02% occupancy) for the 17% replacement experiment and about 5% of that level (so, 0.002% occupancy) for the 1% replacement. These occupancy levels are higher than expected by a factor of 3-5 times, due to the changes in the weights between $\underline{\underline{\hat{W}}}_{\|}^-$ and $\underline{\underline{\hat{W}}}_{\|}^+$. These weights change, even though neither of the connected nodes are added or removed, since the connection (un-normalized) weights are taken from the 2003 and the 2005 co-author data, respectively, as well as indirect effects from changing normalization on rows that do connect with new or removed nodes.

In both experiments, we start from the pre-change label distributions, $\underline{\underline{X}}^-$. We computed these label distributions using power-method iterations for 20 iterations. This brought the per-entry residual errors on the label

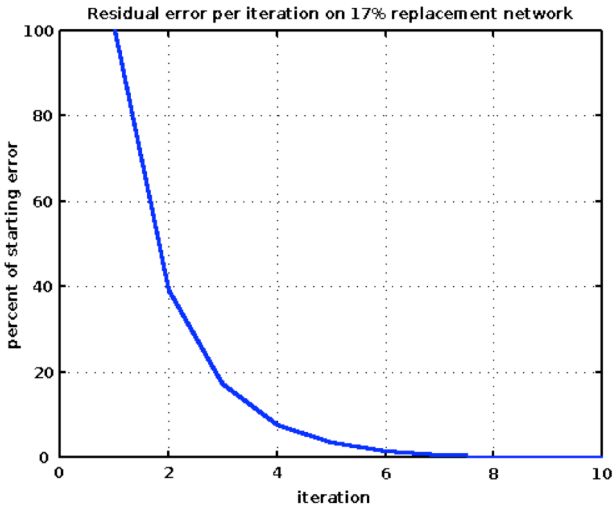


Figure 8. Residual error on 17% replacement network, starting from the label distribution from the pre-change graph, as a function of iteration (using the power method of solution)

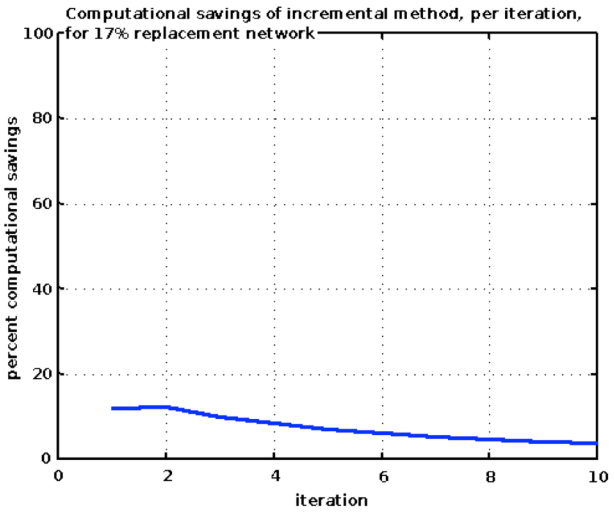


Figure 9. Computational savings on 17% replacement network using incremental updating compared to using the same starting estimate with the full post-change network.

distributions (as measured by $\left|\underline{\underline{X}}_{n+1} - \underline{\underline{X}}_{n}\right|_{ij}$) down to below 10^{-8} for all entries. The result had non-trivial entries on about 70% of the pre-change label distribution matrix, for both 1% and 17% replacement experiments.

Figures 6 and 7 show our results for our 1% replacement experiment. Figures 8 and 9 show our results for our 17% replacement experiment. In both Figures 6 and 8, we show the reduction in the residual label error, starting from the label distribution computed for the pre-change graph (the reduced-2003 graph), when applied to the post-change graph. For these graphs, we use the power-iteration method (instead of bi-conjugate gradient descent). Using power iteration and starting from a well-converged solution for the pre-change graph, the residual errors are exactly the same, whether we use the incremental or non-incremental update methods. We can also see that, for these graph pairs, the residual error is rapidly reduced, falling to well under 1% of the initial error within 8 iterations. The rate at which that reduction happens is slower for the 17% replacement experiment than for the 1% replacement but only by a factor of less than two, in terms of the remaining residual (relative to the starting residual).

Figures 7 and 9 show the computational savings, derived from using the incremental update equations. For the incremental-update method, we use Equation (32), starting with $\Delta\underline{\underline{X}} = \underline{\underline{0}}$ and counting the number of multiplies needed to determine $\Delta\underline{\underline{X}}^0$ as part of that first iteration. For the full-update method, we use Equation (20), using $\underline{\underline{X}}^-$ as the starting estimate for $\underline{\underline{X}}^+$. The computational savings of using the incremental-update approach is very large for the 1% replacement experiment, with as much as 95% of the multiplies needed for Equation (20) avoided by Equation (32), with the same error rates. Even after 10 iterations, the savings is above 84% of the multiplies that would be needed for Equation (20). The savings of the incremental approach is much more modest when 17% of the nodes have been replaced (as well as other edges changing weight). Under those conditions, the savings are 12-13%, for early iterations but falls to only 4% savings (per iteration) by the 10'th iteration. The difference between 1% and 17% replacement, in terms of the levels of savings from incremental updating, can be traced back to the non-zero occupancy rates in $\Delta\underline{\underline{X}}^0$. For the 1% replacement experiment, the occupancy of $\Delta\underline{\underline{X}}^0$ was 2%; for the 17% replacement experiment, it was 25%. Both are less dense than the 70% occupancy of the pre-change label distribution ($\underline{\underline{X}}^-$) but the initial cost is higher than this difference in sparseness would suggest. Part of that reduction in savings is due to the computation of $\Delta\underline{\underline{X}}^0$ itself. Furthermore, for the 17% replacement experiment, the initial difference in sparseness is greatly reduced by even the third iteration, due to the co-authorship fan out in $\underline{\underline{\hat{W}}}^+$.

The specific savings and convergence rates will depend on the specific network configurations and changes that are being used. However, these co-author network examples are somewhat representative of the types of networks that exist for many problems, in that there are only sparse interconnections (having a fan-out level of only 0.04% of what is possible) and having multiple cliques. Fortunately, in large-scale real-world usage scenarios (e.g., YouTube and social networks like G+) updates are computed daily or even more frequently. As a result, the amount of change that will be encountered in the network between update cycles, compared to the overall size of the network, is small. The real-world computational savings of this procedure will

be enormous in practice, allowing responsive targeting to occur using these propagation approaches.

IX. Negative Associations Across Nodes And Between Nodes And Labels

To this point, our derivation has relied on having only non-negative values in our node-to-node connections and in our injection label weights. By restricting our matrices to non-negative values and by pre-normalizing as described in Equations (8) through (12), we are guaranteed to maintain the same unit L_1 norm for all iterations, for all rows of $\underline{\underline{X}}_n$.

In most large graph-based problems, there is no difficulty with restricting the description in this way. In the earlier mentioned example of video-to-video recommendation, we can say that two videos are often watched together, within a single viewing session, but it is much more difficult to say that two videos are negatively associated (that watching one means you are significantly less likely to watch the other); rarely is there enough training data to confidently ascertain disassociations.

Nonetheless, there are several cases in which negative information can be useful. First consider the use of social networks in movie and music recommendations and reviews. Often, users express dislike for particular songs or movies. In this case, though the node-to-node connections (user-to-user) remain positive, the label (the movie/song/artist) may now have a negative connection weight to some nodes. We need to be able to handle and propagate these negative reviews in the same way as positive reviews, as both can shape public opinion. The simplest way to do this is to separately represent positive-bias and negative-bias labels, even when they refer to the same underlying label: in our example we would double the number of labels that we used, with one for "likes" and one for "dislikes" for each movie/song. Representing this using the notation for Equation (15) we would have:

$$\left[\begin{array}{cc} \underline{\underline{X}}^p & \underline{\underline{X}}^n \end{array}\right] = \underline{\underline{\hat{W}}}\left[\begin{array}{cc} \underline{\underline{X}}^p & \underline{\underline{X}}^n \end{array}\right] + \left[\begin{array}{cc} \underline{\underline{\hat{L}}}^p & \underline{\underline{\hat{L}}}^n \end{array}\right] \quad (35)$$

where $\underline{\underline{\hat{L}}}^p = \max\left(\underline{\underline{\hat{L}}}, \underline{\underline{0}}\right)$ are positive injection labels (declared "likes") and $\underline{\underline{\hat{L}}}^n = \max\left(-\underline{\underline{\hat{L}}}, \underline{\underline{0}}\right)$ are negative injection labels (declared "dislikes"). For Equation (35), we are still requiring that all entries in $\underline{\underline{\hat{W}}}$ be non-negative. The result is that the inferred label weights are split, with positive-association weights in $\underline{\underline{X}}^p$ and negative-association weights in $\underline{\underline{X}}^n$.

As long as $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$ are represented separately, the L_1 row norm of $\left[\begin{array}{cc} \underline{\underline{X}}^p & \underline{\underline{X}}^n \end{array}\right]$ will remain one, throughout our iterative estimation, without renormalizing. This approach will have the effect that the amount of 'attention' paid to controversial movies/songs will be higher, in terms of the portion of the available L_1 unit length, than it would be if we did not split positive and negative labels into separate entries. For some applications, this separation of opposite extremes may be the right thing to do. For example, it may be that separately listing strong positive and negative recommendations for a controversial movie, and thereby having that movie feature prominently in terms amount of attention it is given, is better than having the recommendations for that movie "wash out" to neutral, by not exposing the controversy in opinions.

An alternative example would be to use a social-network of voters in conjunction with political-opinion labels to help guide a politician's stance on legislation. The social network could capture regional differences within the politician's constituency. Since the politician needs to be seen as serving all represented regions equally, understating the key issues of a region that has a divided stand on some controversial issue does not serve the politician well. The politician is probably better served by emphasizing the issues with political consensus, for those regions, rather than effectively ignoring all the opinions of the region by understating what they do agree upon. As such, this situation may be better handled by the re-combining and re-normalizing approach we outline later in this section.

In the previous examples, we explored the use of negative values associated with a node's labels. Now, we consider the case where there are negative connection weights between nodes. Consider the case of financial-fund analysis. If we are trying to find closely related (as well as nearly opposite) investment opportunities, we could create a graph with one node for each fund under study and with the node-to-node connection being set by the statistically significant market-adjusted price-change correlations. The labels would then be the fund symbols themselves, optionally with long and short positions represented in $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$, respectively. In this framework, there are many combinations of funds or instruments that would show negative connection weights between them. A fairly simple example would be the connection from either a purchase-to-open put contract or a sell-to-open call contract to the underlying security. Both option contracts are clearly distinct in their valuation from the underlying security (with the time-value changes having the largest exogenous impact) but both have price changes that are strongly (negatively) correlated with the security's price changes. A less direct example of negative connections between funds would by an ultra-short fund on a market sector and any of the largest firms in that sector (for example, QID and AAPL).

To handle negative node-to-node connections, we use a similar slicing-and-doubling approach as we used for negative label weights. Specifically, we can say

$$\begin{bmatrix} \underline{\underline{X}}^p \\ \underline{\underline{X}}^n \end{bmatrix} = \begin{bmatrix} \underline{\underline{\hat{W}}}^p & \underline{\underline{\hat{W}}}^n \\ \underline{\underline{\hat{W}}}^n & \underline{\underline{\hat{W}}}^p \end{bmatrix} \begin{bmatrix} \underline{\underline{X}}^p \\ \underline{\underline{X}}^n \end{bmatrix} + \begin{bmatrix} \underline{\underline{\hat{L}}}^p \\ \underline{\underline{\hat{L}}}^n \end{bmatrix} \quad (36)$$

where $\underline{\underline{\hat{W}}}^p = \max\left(\underline{\underline{\hat{W}}}, \underline{\underline{0}}\right)$ and $\underline{\underline{\hat{W}}}^n = \max\left(-\underline{\underline{\hat{W}}}, \underline{\underline{0}}\right)$. Note that Equation (36) has two "partial" rows per node, one (with positive associations) in $\underline{\underline{X}}^p$ and the other (with negative associations) in $\underline{\underline{X}}^n$. It is the concatenation of these two parts, $\begin{bmatrix} \underline{\underline{X}}^p & \underline{\underline{X}}^n \end{bmatrix}$, that will maintain a unit L_1 row norm, while the energy split between $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$ in each row can vary widely, from one iteration to another. Also note that Equation (35) is a re-arranged and simplified version of Equation (36), with $\underline{\underline{\hat{W}}}^n = \underline{\underline{0}}$.

In the movie/music recommendation example, we suggested that keeping track of diverse opinions about the same movie/song made sense (since controversial movies/music are fundamentally different than ones that do not evoke strong opinions either way). In contrast, in this financial example, keeping positive and negative label entries separately (corresponding to holding long and short positions of the same security) may not be the correct approach. Similarly, in the politician's example, it may be better to refocus the rhetoric on non-controversial issues, instead of tracking the degree of controversy.

Therefore, we investigate the effects of recombining and reweighting positive and negative inferred labels, $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$. When we recombine $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$, by taking $\underline{\underline{X}} = \underline{\underline{X}}^p - \underline{\underline{X}}^n$, we will end up with a row norm less than one, in all rows where one or more non-zero entries of $\underline{\underline{X}}^p$ and $\underline{\underline{X}}^n$ overlap. The total *reduction* in the i^{th} row norm will be

$$L_i^{loss} = 2\sum_j \min\left(\underline{\underline{X}}_{ij}^p, \underline{\underline{X}}_{ij}^n\right) \quad (37)$$

To avoid under-emphasizing the combined information from rows with some conflicting labels, we then need to renormalize, to bring the row norm back up to one. We can do this selectively on only those rows where L_i^{loss} exceeds some pre-defined threshold. When that happens, the label vector for that i^{th} row should be replaced with

$$\underline{\underline{X}}_{ij}^p = \max\left(0, \frac{\underline{\underline{X}}_{ij}^p - \underline{\underline{X}}_{ij}^n}{1 - L_i^{loss}}\right) \quad (38)$$

and

$$\underline{\underline{X}}_{ij}^n = \max\left(0, \frac{\underline{\underline{X}}_{ij}^n - \underline{\underline{X}}_{ij}^p}{1 - L_i^{loss}}\right) \quad (39)$$

This need for re-normalization, unfortunately, means that we can no longer use label slicing, since we need to be able to track the row norm for this normalization process. It also complicates the use of stabilized bi-conjugate gradient descent, since that approach introduces interdependences between update iterations that are not easily adjusted for changing scale. When we use this re-normalizing method, we need to be aware of the increases in the computational costs that result. However, this method does allow us to handle situations that respond best to "fair and equal" representations of each node within the network, throughout the computation, even in the presence of conflicting labeling. The cost/benefit trade-off must be carefully considered in the context of each application.

X. CONCLUSIONS

This paper improves the computational efficiency of Adsorption, a graph-based labeling approach that has already been shown to be highly effective. We do so by replacing propagation-interleaved normalization with pre-normalization, without changing the results provided by Adsorption. Specifically, if the power-method approach to finding a solution is used, as it was with Adsorption, the answers at every iteration will be exactly the same using either the original or the pre-normalized Adsorption. The advantage of the pre-normalized Adsorption is computational efficiency in determining the label distribution. With the pre-normalized version, we can use label slicing, to compute only those labels that are of direct interest, without losing the beneficial belief-limiting characteristics of the full label set. Label slicing reduces the computational cost linearly with the percentage of dropped labels. Similarly, we can use Gaussian elimination, to compute the labels only on those nodes that are of direct interest, without losing the effects of the connections that occur indirectly through currently not-of-interest nodes. Finally, we can speed up convergence to the steady-state solution by a factor of 12 (in numbers of graph matrix multiples), by using stabilized biconjugate gradient descent, instead of power iteration. We tested the pre-normalized Adsorption in a new, large-scale application area, topic labeling on web domains, with promising results.

Additionally, we explored extensions to address two real-world scenarios, in which network propagation will play an important role: (1) networks with both positive and negative connections as well as positive and negative associations with labels and (2) gradually changing networks in which nodes are added and removed (as well as having weight changes between existing nodes). Examples of changing networks include searching for, recommending, and advertising against image, audio, and video content. These labeling problems must handle millions of interconnected entities (users, domains, content segments) and thousands of competing labels (interests, tags, recommendations, topics). By using a label update matrix (instead of the full label distribution matrix) in our update equations, we were able to converge to the correct label distribution on the changed network in the same number of iterations as we would need for the full network but with only one tenth of the computation (for a network that changed by 1% node

replacement). This savings drops rapidly as the percentage replacement increased but was still significant, even at 17% node replacement. We demonstrated the incremental update using co-author networks. We note that, in real-world cases in which rapid and continual updated of large (10^7 – 10^9 node) networks is required, the methods proposed in this paper will make propagation methods feasible.

REFERENCES

[1] M. Covell and S. Baluja, "Efficient and Accurate Label Propagation on Large Graphs and Label Sets," *Proceedings of the International Conferences on Advances in Multimedia*, IARIA, April 2013, pp. 12-18.

[2] S. Baluja, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, and M. Aly, "Video suggestion and discovery for YouTube: taking random walks through the view graph," *Proceedings of the International Conference on World Wide Web*, ACM, April 2008, pp. 895-904.

[3] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," CMU technical report, CMU-CALD-02-107, 2002.

[4] P.P. Talukdar, J. Reisinger, M. Pasca, D. Ravichandran, R. Bhagat, and F. Pereira, "Weakly-supervised acquisition of labeled class instances using graph random walks," *Proceedings of the Conference on Empirical Methods in Natural Language Processing,* Association of Computational Linguistics, October 2008, pp. 582-590.

[5] Y. Jing and S. Baluja, "Visual Rank: applying Page Rank to large-scale image search," *Transactions on Pattern Analysis and Machine Intelligence,* IEEE, vol. 30, November 2008, pp. 1877-1890.

[6] J. Liu, W. Lai, X S. Hua, Y. Huang, and S. Li, "Video search re-ranking via multi-graph propagation," *Proceedings of the International Conference on Multimedia,* ACM, September 2007, pp. 208-217.

[7] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter polarity classification with label propagation over lexical links and the follower graph," *Proceedings of the Workshop on Unsupervised Learning in Natural Language Processing,* Association of Computational Linguistics, July 2011, pp. 53-63.

[8] H. A. Van der Vorst, "Bi-CGSTAB: A Fast and Smoothly Converging Variant of BiCG for the Solution of Nonsymmetric Linear Systems," *Journal on Scientive and Statistical Computing,* SIAM, vol. 13, March 1992, pp. 631-644.

[9] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," *Proceeds of the Symposium on Operating Systems Design and Implementation*, USENIX, December 2004, pp. 137-150.

[10] Wikipedia, "Biconjugate Gradient Stablized Method," http://en.wikipedia.org/wiki/Biconjugate_gradient_stabilized_method [retrieved February, 2013].

[11] M. E. J. Newman, "Condensed Matter Collaborations 2003" and "Condensed Matter Collaborations 2005," http://www-personal.umich.edu/~mejn/netdata [retrieved September, 2013].

[12] M. E. J. Newman, "Structure of Scientific Collaboration Networks," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 98, January 2001, pp. 404-409.

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